Game Theoretic Models in the Social Cloud

Ph.D. Thesis

By

Pramod C. Mane

DISCIPLINE OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

DECEMBER 2019

Game Theoretic Models in the Social Cloud

A THESIS

submitted to the

INDIAN INSTITUTE OF TECHNOLOGY INDORE

in partial fulfillment of the requirements for

the award of the degree

of

DOCTOR OF PHILOSOPHY

By

Pramod C. Mane

DISCIPLINE OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

DECEMBER 2019

INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **Game Theoretic Models in the Social Cloud** in the partial fulfillment of the requirements for the award of the degree of **Doctor of Philosophy** and submitted in the **Discipline of Computer Science and Engineering, Indian Institute of Technology Indore,** is an authentic record of my own work carried out during the time period from July 2012 to December 2019 under the supervision of Dr. Kapil Ahuja (Associate Professor, Indian Institute of Technology Indore, Indore, India) and Dr. Nagarajan Krishnamurthy (Associate Professor, Indian Institute of Management Indore, Indore, India).

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institute.

28/05/2020

Signature of the Student with Date

(Pramod C. Mane)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

28/5/2020

Signature of Thesis Supervisor #1 with date

(Dr. Kapil Ahuja)

28/05/2020

Signature of Thesis Supervisor #2 with date

(Dr. Nagarajan Krishnamurthy)

Pramod C. Mane has successfully given his Ph.D. Oral Examination held on 28/5/2020

Signature of Chairperson, OEB

Date: 28/5/2020

28/05/2020

Signature of External Examiner

Date

Signature of Thesis Supervisors

Date: 28/5/2020

Signature of PSPC Member #1

Date: 28/5/2020

Signature of PSPC Member #2

Date 28/05/2020

Signature of Convener, DPGC

Date: 28/05/2020

Signature of Head of Discipline

Date: 28/05/2020

ACKNOWLEDGEMENTS

I take this solemn opportunity to express my deep sense of gratitude and indebtedness to my supervisors, Dr. Kapil Ahuja of IIT Indore and Dr. Nagarajan Krishnamurthy of IIM Indore. They inspired me with noble feelings for successful completion of this work. But, these two lines are not enough to acknowledge them.

I would like to express my sincere gratitude to Dr. Kapil Ahuja for his continuous support during my PhD study, his patience, motivation, and immense knowledge. He gently corrected me at every stage, coming with relevant ideas, implement them efficiently, presenting them with less ambiguity, and writing concisely. I am especially thankful to him for taking me under his wings and making administrative formalities easier. I could not have imagined having a better adviser and mentor for my Ph.D study.

I owe my gratitude to Dr. Nagarajan Krishanamurthy for his guidance, encouragement, and emotional as well as intellectual support through all these years. He never pressurized me about anything; he was always calm and composed. He taught me quite a lot about how to use and interpret game-theoretic models. He always gave me a lot of ideas - the big picture. I consider myself lucky to have had an adviser like him and shall always be indebted to him.

I am thankful to Dr. Amod Umarkikar and Dr. Abhishek Shrivastava, my research progress committee members for taking out some valuable time to evaluate my progress all these years. Their valuable comments and suggestions helped me to improve my work at various stages.

I would like to express my deep sense of gratitude to Adjunct Prof. T. Parthasarthy of Chennai Mathematical Institute, India. His optimism, words of encouragement and intellectual support have always been a strong pillar of strength.

I wish to express my sincere gratitude and deep appreciation to Prof. K. S. Mallikarjuna Rao of IIT Bombay, India and Dr. Rajesh Sharma of University of Tartu, Estonia for their invaluable comments and remarks in the informal discussion. I am immensely benefited by their scholarly advice, scientific approach and meticulous scrutiny. I am also thankful to Dr. Sajal Mukhopadhyay who introduced me in the field of research by his proper guidelines during my masters.

Every researcher is deeply influenced by his peers around him and I am no different. I have been fortunate to have several friends in my stay here at IIT Indore, who have been source of constant encouragement and inspiration. It has been a pleasure working with Dheeraj, Rohit, Shruti, Navneet, Rajendra, Aditya, Rohit, Aniket, Altaf, and Nitesh. I am also thankful to Manish and Sonakshi colleagues from IIM Indore for their moral support.

I would like to recognize the help I received from Mr. Shailendra Varma, who kept my computer always workable and Mr. Satish Bisen, who make available some important research articles and books whenever I request for the same.

I sincerely thank Mrs. Mira Nagarajan and Mrs. Richa Ahuja for their patience during the technical discussions I had with supervisors and providing me with hot cup of tea/ coffee along with delicious snacks.

I cannot express in words how grateful I am to **my Parents** who have been great over the years and never raised an eyebrow when I claimed my thesis would be finished ‘in the next two months' for nearly one year. A special thanks to my siblings, uncles, in-laws, thank you for your support and unwavering belief in me.

I would like to thank my friends who have kept me on the track, added value to my thinking, taken up the slack I left in the wake of thesis anxiety and stress, buoyed me, and survived years of unfinished business with me. I am particularly indebted to Amol, Shalini, Ganesh, Ajay, Suresh, Vishal, and Dnyaneshwar.

In the end, some special words of gratefulness go to the most important person in my life – my wife **Shital**. She has been a major source of support, strength and inspiration when things during the past a bit discouraging. I am grateful for her love and all of the sacrifices that she has made during these years.

Pramod C. Mane

To my Parents

and

memories of my Father in-Law

Abstract

Purpose: *The social cloud is at the centre of this study. A social cloud can be seen as a resource sharing framework where agents share (or trade) their computing resources (like storage space, processing power, workflows, and so on) with others who are socially connected with them. This study deals with two aspects of social cloud; first, endogenous social cloud formation, and second, properties of the social cloud. In particular, this study analyses the stability, efficiency and contentedness of social cloud in a setting where agents decide with whom they want to form resource sharing connections and with whom they do not. It also aims to examine how a link formation between two agents' impact their as well as others' resource availability.*

Methodology/ approach: *This study is at the intersection of computer science and economics. There is a long tradition in the field of computer science to make use of tools from economics to deal with issues like resource allocation in distributed systems. This study makes use of strategic network formation (from economics) as a tool for investigating social cloud formation in a strategic setting.*

Findings: *This study presents three models of social cloud formation. First is the social storage network model in which agents perform resource sharing with those who have direct connections with them. The utility of agents is a combination of the cost that they pay and the benefit they receive, as a function of the resource sharing network in place. In this model, network formation always leads to a stable network, which need not be efficient. That is, there is a tension between stability and efficiency. Further in a stable network, if the number of agents is an even number, then each agent in the network has the same number of direct connections. Otherwise, there exists an agent who has one less direct connection than the remaining agents. Second is a social storage cloud model in which agents perform closeness-based resource sharing with direct and indirect connections. For the symmetric form of this model, agents form a stable and efficient network, and therefore, the price of anarchy and stability is one. Here, a stable network is always disconnected.*

The third is a social cloud compute model, that mainly focuses on local and global resource availability. Global resource availability is examined in terms of externalities via an empirical approach. Here, the number of agents who experience negative externalities is always greater than the number of agents who experience positive externalities. Further, this study adopts an inverse approach to derive the stability of social compute cloud structures encoded in the standard network structures, namely, star, complete, wheel, and bipartite network.

Limitations: *Although the social cloud models of network formation focus on agents who are heterogeneous concerning the benefit and the cost of link formation, this study mainly investigates social cloud formation with homogeneous agents.*

Value: *The existing social cloud literature only focuses on exogenous social connections to research on and develop of social cloud systems. Different from the existing trend, this study looks at the more practical and intuitive endogenous social cloud formation, which is the first move in this direction.*

Implications: *The theoretical insights would help the real-world social cloud systems in designing efficient workload balancing, resource sharing and incentive policies, and also recommender systems in this context. This study also enhances our knowledge regarding the neighbourhood size, which is a crucial issue in the social cloud context. Finally, this study introduces the social cloud as an application of strategic network formation to economists.*

Keywords: Social Cloud, Social Storage, Sharing Economy Network, Resource Sharing Network, Social Network, Strategic Network Formation, Pairwise Stability, Bilateral Stability, Externalities

List of Publications

Journal Papers

1. **P. C. Mane**, K. Ahuja, and N. Krishnamurthy, "Stability, Efficiency, and Contentedness of Social Storage Networks", *Annals of Operations Research, 287 (2), 811–842, (2020). DOI: 10.1007/s10479-019-03309-9*.

2. **P. C. Mane**, N. Krishnamurthy, and K. Ahuja. "Formation of Stable and Efficient Social Storage Cloud", *Games, MDPI*, 10(4): 44, (2019). *DOI: 10.3390/g10040044*.

3. **P. C. Mane**, K. Ahuja, and N. Krishnamurthy, "Externalities in Socially-Based Resource Sharing Network", *Applied Economics Letter, Taylor and Francis, (2019). DOI: 10.1080/13504851.2019.1683507*.

Conference Proceedings and Presentations

4. **P. C. Mane**, K. Ahuja, and N. Krishnamurthy, "Stability, Efficiency, and Contentedness of Social Storage Networks", (presented by P. C. Mane at) 14^{th} *Inter-Research-Institute Student Seminar in Computer Science (IRISS'20)*, 13-14 February, 2020, Indian Institute of Technology, Gandhinagar, India.

5. **P. C. Mane**, N. Krishnamurthy, and K. Ahuja, "Formation of Stable and Efficient Social Cloud", (presented by P. C. Mane at) *International Conference on Game Theory and Networks (GamesNet'19)*, 6-9 September, 2019, Dibrugarh, India (**Received Second Prize for Poster**).

6. H. Jain, S. Teja, **P. C. Mane**, K. Ahuja and N. Krishnamurthy, "Data Backup Network Formation with Heterogeneous Agents", (presented by H. Jain and S. Teja at) *Proceedings of the 10th International Conference on Communication Systems & Networks (COMSNETS'18)*, pp. 418-420, 3-7 January, 2018, Bangalore, India. IEEE.

7. **P. C. Mane**, K. Ahuja and N. Krishnamurthy, "Unique Stability Point in Social Storage", (presented by N. Krishnamurthy at) *International Conclave on Foundations of Decision and Game Theory*, 14-19 March, 2016, Indian Institute of Technology Bombay and Indira Gandhi Institute of Development Research (IGIDR), Mumbai, India.

   And

   (presented by N. Krishnamurthy at) *The 5th World Congress of the Game Theory Society (GAMES'16)*, 24-28 July, 2016, Maastricht University, The Netherlands.

8. **P. C. Mane**, N. Krishnamurthy, and K. Ahuja, "Externalities and Stability in Social Cloud", in the *Proceedings of the International Conference on Game Theory for Networks (GameNets'14)*, pp. 1-6, 25-27 November, 2014, Beijing, China. IEEE.

Contents

List of Figures

List of Tables

LIST of Abbreviations and Acronyms

CRB Cloud Resource Bartering

CeRSC Collaborative eResearch Social Cloud

F2F Friend-to-Friend

HHSS Husky Healthcare Social Cloud

MO-Framework Multi-Objective Framework

P2P Peer-to-Peer

PeRSC Public eResearch Social Cloud

SCC Social Compute Cloud

SSC Social Storage Cloud

SSN Social Storage Network

SSCC Symmetric Social Compute Cloud

SSSC Symmetric Social Storage Cloud

SSCN Symmetric Social Storage Network

SO-Framework Single Objective Framework

SRN Symmetric Resource Network

SVN Symmetric Value Network

SV-SRN Symmetric Value-Symmetric Resource Network

VO Virtual Organization

NOTATIONS

$\mathfrak{g}$ network (or social cloud a resource sharing network)

$\mathscr{A}$ set of agents (or vertices)

N the number of elements in the set $\mathscr{A}$, which is the number of agents in $\mathfrak{g}$

$\mathscr{G}(N)$ the set of all networks on N agents

$\mathscr{L}$ set of links (or edges)

$\langle ij \rangle$ link between agents i and j

$\eta_i(\mathfrak{g})$ neighborhood size of agent i in $\mathfrak{g}$. Also denotes the set of neighbors of i

$\mathscr{P}_{a_1 a_n}(\mathfrak{g})$ a path from agent a_1 to a_n in $\mathfrak{g}$ such that $\langle a_1, a_2 \rangle, \langle a_2, a_3 \rangle, \cdots, \langle a_{n-1}, a_n \rangle \in \mathscr{L}$

$d_{ij}(\mathfrak{g})$ the length of the shortest path connecting agents i and j in $\mathfrak{g}$

$\mathscr{D}_{\mathfrak{g}}$ the longest of all the shortest paths in the network $\mathfrak{g}$

$\mathfrak{g}(\kappa_i)$ a component of network $\mathfrak{g}$, where κ_i is the set of agents in that component

$\mathfrak{g}^{\mathfrak{c}}$ complement of network $\mathfrak{g}$

$\mathfrak{g} + \langle ij \rangle$ new link $\langle ij \rangle$ is added to $\mathfrak{g}$

$\mathfrak{g} - \langle ij \rangle$ existing link $\langle ij \rangle$ is deleted from $\mathfrak{g}$

$u_i(\mathfrak{g})$ utility of agent i in $\mathfrak{g}$

ς cost incurred by each agent to maintain a link

b is the amount of budget available with an agent

s is the amount of storage space available with an agent

d is the amount of data an agent wants to backup

β worth (or value) that each agent has for its data

λ probability that an agent's disk fail

δ probability that an agent loses its data

$\Phi_i(\mathfrak{g})$ closeness of agent *i* in $\mathfrak{g}$

$\alpha_{ij}(\mathfrak{g})$ probability that agent *i* obtains storage space from agent *j* in $\mathfrak{g}$

$\gamma_i(\mathfrak{g})$ probability that agent *i* obtains storage space from at least one agent in $\mathfrak{g}$

Chapter 1

Introduction

The concept of social cloud [2] has emerged as a significant area of research at the intersection of computer science and economics. Social storage [3], a special case of the social cloud, has also received a lot of attention in recent years. Although there are many views on the concept of social cloud, the definition of social cloud stated in [4] is widely accepted in the literature. According to Chard et al. [4], "a social cloud is a resource and service sharing framework utilizing relationships established between members of a social network". In particular, social clouds allow agents (members of the network) to share or trade their under-utilized computing resources (for example, disk space, processing power, shareable software, and so on) with other agents, by making use of *social connections* to facilitate resource sharing.

Computer scientists believe that, by taking advantage of social connections, it is possible to build a trustworthy distributed computing paradigm that deals with various limitations associated with many other distributed computing frameworks (like volunteer computing [5], peer-to-peer (P2P) computing [6], grid computing [7], utility computing [8] and cloud computing [9]). For economists, social cloud is a case of sharing economy ("the peer-to-peer based activity of obtaining, giving, or sharing access to goods and services" [10]).

Social connections, in this context, are either extracted from an online social network or constructed in an application context [3]. We call extracted social connections as *exogenous social connections*, and social connections that evolve in an application context as *endogenous social connections*. In academic discourse, numerous architectural prototypes have been proposed that make use of exogenous social connections to build social cloud systems. For instance, Social Storage Cloud [4] and FriendBox [11] integrate Facebook social graph, and Friendstore [12] and BlockParty [13] use Orkut and Venus social graphs. The other class of social cloud systems, for example, BuddyBackup[1], and CrashPlan[2] allow agents to share their disk space with other agents who select their storage sharing partners explicitly, resulting in endogenously evolving social connections.

Recent surveys on social cloud [2] and socially-aware peer-to-peer systems [3] have pointed out the numerous advantages as well as disadvantages of making use of social connections for constructing these systems. However, these discussions are restricted to exogenous social connections. In fact, the approach of using exogenous social connections in building social cloud systems is dominant in the literature and the aspect of endogenous social connections is, notably, lacking. This thesis bridges this gap by focusing on endogenous social connections.

1.1 Motivation

Existing research on social cloud is moving in two directions. One direction [12, 14, 11, 13, 15, 16] focuses on various technical approaches such as development of social cloud applications and their implementation, and various other aspects associated with social clouds, such as trust [17], incentives [18, 19], and resource management [20, 21]. The other direction [22, 23, 24] focuses on Quality of Service (QoS) related issues that include data availability, reliability, storage availability, and designing data maintenance and data placement policies. The latter body of research provides several important insights, especially regarding the correlation between the social connection pattern and QoS, trust, resource allocation and so on. For example, some studies [23, 24] observe that a small friend set is the major cause of poor QoS, imbalanced workload (the number of resource requests) and low resource utilization. However, it should be highlighted that these understandings are achieved in the context of exogenous social connections.

[1] http://www.buddybackup.com (accessed on 21 June 2019)
[2] https://support.crashplan.com (accessed on 21 June 2019)

On the one hand, focusing on exogenous social connections has several advantages. For example, some level of trust between socially connected agents can mitigate issues related to security (for example, Denial-of-Service attack[3]) and agents' malicious behaviour. Further, homophily[4] in exogenous social connections helps to design efficient service search algorithms [3]. Then, exogenous social connections provide incentives for cooperation that effectively deal with issues like random arrival and departure of agents in a system, and free riding[5].

On the other hand, focusing *only* on exogenous social connections has several limitations. For example, Fitzpatrick and Recordon[6], and Iskold[7] have discussed various issues associated with utilizing social graph information for making use of them for application development. These issues include: 1) agent identity (agent may be a participant of multiple networks), 2) types of relationship (a link between two agent represents a different kind of relationship between them), privacy (a social graph contains private as well as public information), 3) social networking inter-op protocol, 4) node communication, and 5) overlapping social graphs.

In addition to the above, this approach fails to answer several questions. First, it does not provide an understanding of agents' preferential attachment in the social cloud system and the underlying social graph (on which the social storage is built). For example, an agent may not want to perform resource sharing with one of its neighbours in the underlying social graph. Therefore, the understanding of the correlation between friend-set (in the underlying social graph) and QoS is in question. Second, a limitation of this approach is that it does not explain why each agent has a particular set of neighbours. Third, it does not take agents' choices of selecting their storage partners into consideration.

Online social cloud systems like BuddyBackup impel us to focus on endogenous social connections. In fact, the approach of looking at the endogenous social connections (or network formation) helps answer the aforementioned questions. Besides, this approach would

[3]A denial-of-service (DoS) attack refers to the situation in which legitimate agents of the system are not able to access resources/ services due to the act of malicious agents.

[4]Homophily refers to the tendency of agents to have social relationships with those agents who mostly meet their own type (e.g., religion, demographic propinquity, age, etc.).

[5]Free riding refers to the tendency of agents to offers less to a system but consumes more.

[6]https://bradfitz.com/social-graph-problem/

[7]https://readwrite.com/2007/09/11/social graph concepts and issues/

help to model agents' incentives or cost-benefit trade-off in these systems. By this, it is easy to analyse agents' behavior in the process of building social resource sharing connections and the outcome of their interactions.

Surprisingly, the social cloud literature has not treated endogenous social connections in much detail. In a nutshell, agents' behavior in building resource sharing connections, the cost-benefit trade off, and the outcome of their interactions are still poorly understood. With this background, we now state our objectives.

1.2 Objectives

This study does not engage with either of the two existing research directions discussed in the previous section. The discussion of exogenous social connections is beyond the scope of this study. Specifically, this study does not deal with the technical facets as mentioned earlier, such as application development, designing the policies related data and resource management, trust or incentives aspects.

This thesis aims to gain theoretical insights into endogenous social cloud formation. The approach of looking at endogenous social connections, in particular, endogenous social cloud formation introduces a rich set of questions. The following questions set the objective of this study.

1: How to model endogenous social cloud formation? There are numerous cases of social cloud systems that allow agents to select their resource sharing partners explicitly which eventually leads to resource sharing network formation. To examine the formation of endogenous social cloud, there is a need for a model that provides a better understanding of resource sharing network formation and insights into its basic principles. Further, it must capture agents' behaviour as well as their freedom to decide what should be done in a particular situation while building their resource sharing connections.

2: Which resource sharing networks are formed by agents? It is widely known that the shape of social connections has an effect on the agent as well as aggregate behaviour. This leads to the question, which types of social connections (network structures) are likely to emerge when agents are decision-makers. This question is interesting when agents are self-interested and choose resource sharing connections to maximize their well-being.

3: Are the resource sharing networks that form stable (where no agent wants to change the prevailing network structure)? If so, under what conditions? What structures (shapes) do they have?

4: Are the networks that form efficient or inefficient? That is, does the network formed by self-interested agents lead to an outcome (network structure) that is also preferable (best) from a societal viewpoint? It might be possible that self-interested agents build their resource sharing connections that are beneficial from the point of view of respective agents, but that this formation leads to a 'bad' outcome from a societal viewpoint. In other words, network formation may lead to an inefficient outcome.

5: Do externalities exist? How do the agents' choice of selecting their resource sharing partners impact other agents' behaviour as well as their possibilities of obtaining resources in the resource sharing network?

1.3 The Approach

As mentioned earlier, there is a need of a model to look into social cloud formation (as resource sharing network formation). The two types of network formation models are widely discussed, namely, *non-strategic* and *strategic* models of network formation. The non-strategic models of network formation are based on the theory of random graph models [25, 26]. The random graph models are the easiest way to model network formation, where link formation between two nodes takes place with some probability **p**. Although the random graph models are a simple approach, they do not explore the link formation (network formation) process on the agent level. That is, these models do not consider the choices of agents in building their social connections. Further, these models fail to examine the stability and efficiency of network formation. Another class of network formation models are strategic network formation models [27, 28, 29, 30], stem from economics, make use of game-theoretic approach to model network formation. The strategic network formation models assume agents are rational and discretionary who build their social connection to obtain payoffs that depend on the shape of the network take place. Unlike the random graph models, in strategic network formation models, link formation takes place between agents based on the cost-benefit trade-off. Therefore, these models allow studying network stability (in equilibrium) and efficiency.

This study follows the approach of strategic network formation modelling to examine the social cloud formation and to deal with the above objectives. To model network formation there is a foremost requirement of defining the utility function, which captures the cost and benefit of the agents that they receive as a function of the established network. This study examines strategic social cloud formation (in three different resource sharing setups) by proposing three different utility models and gain insights into the various objectives stated earlier.

1.4 Thesis Contribution

This study contributes to the research in the field of social cloud and strategic network formation. First, it contributes to the field of social cloud in various ways. To our knowledge, in the context of social cloud, no study has analysed social cloud formation in a strategic setting. This study tried to fill the research gap in the field of social cloud by touching the approach of endogenous social cloud formation. We believe this study provides considerable insight into social cloud formation, their stability and efficiency. Further, this study is the opening move towards enhancing our understanding of a particular friend set size that agents have in social clouds, which the exogenous social connection approach fails to answer. This study proposes three utility functions, which are the first of their kinds in the context of social cloud. Second, this study offers social cloud as an application to the research field of strategic network formation. This study offers numerous challenges, which researchers in the field of strategic network formation could be interested in.

Now, we overview our contributions in brief.

Contribution I: Chapter 3 and 4

The first part of the thesis focuses on social storage systems, which are emerging as a good alternative to existing data backup systems of local, centralized, and P2P backup. In this research, we, first, model the social storage system as a strategic network formation game. We define the utility of each agent in the network under two different frameworks, one where the cost to add and maintain links is considered in the utility function and the other where budget constraints are considered.

Second, we propose the solution concept termed *bilateral stability* which refines the *pairwise stability* solution concept defined by Jackson and Wolinsky [31], by requiring mutual consent for both addition and deletion of links, as compared to mutual consent just for link addition. Mutual consent for link deletion is especially important in the social storage setting. The notion of bilateral stability subsumes the bilateral equilibrium definition of Goyal and Vega-Redondo [32]. We discuss the above in Chapter 3. Third, we prove necessary and the sufficient conditions for bilateral stability of social storage networks. For symmetric social storage networks, we prove that there exists a unique neighbourhood size, independent of the number of agents (for all non-trivial cases), where no pair of agents has any incentive to increase or decrease their neighbourhood size. We call this neighbourhood size as the stability point. Fourth, given the number of agents and other parameters, we discuss which bilaterally stable networks would evolve and also discuss which of these stable networks are efficient, that is, stable networks with maximum sum of utilities of all agents. We also discuss ways to build contented networks, where each agent achieves the maximum possible utility.

We discuss these results in Chapter 4, which also appeared in: *P. C. Mane, K. Ahuja, N. Krishnamurthy "Stability, efficiency, and contentedness of social storage networks". Annals of Operations Research, 287 (2), 811–842, (2020).*

Contribution II: Chapter 5

This research presents social storage cloud model (as described in Chapter 3), where agents are involved in a closeness-based conditional storage sharing and build their storage sharing network themselves. In the previous research, we analysed social storage network formation by proposing a degree-based utility function. In this research, we propose a degree-distance-based utility model, which is a combination of benefit and cost functions.

The benefit function of an agent captures the expected benefit that the agent obtains by placing its data on others' storage devices, given the prevailing data loss rate in the network. The cost function of an agent captures the cost that the agent incurs to maintain links in the network. With this utility function, we analyse what network is likely to evolve when agents themselves decide with whom they want to form links and with whom they do not. Further, we analyse which networks are pairwise stable and efficient. We show that for the proposed utility function, there always exists a pairwise stable network, which is also efficient. We show that all pairwise stable networks are efficient, and hence, the price of anarchy is the best that is possible. We also study the effect of link addition and deletion between a pair of agents on their, and others', closeness and storage availability.

We study externalities in the social storage network, that is, the effect of link formation between a pair of agents on the utility of the other agents. We find, in a connected social storage cloud, agents experience either positive or negative externalities and there is no case such as ”no” externalities. However, agents observe ”no” externalities if and only if the network is disconnected and consists of more than two components (sub-networks). Second, if there is a no change in agents' (who are not involved in link formation) closeness due to newly added links then agent experiences negative externalities. We conjecture, and later experimentally support (discussed in Chapter 6), that for an agent to experience positive externalities, an increase in its closeness is necessary. The condition is not sufficient though. We provide a necessary and sufficient condition under which agents experience positive and negative externalities.

We discuss these results in Chapter 5, which also appeared in: *P. C. Mane, N. Krishnamurthy, K. Ahuja “Formation of stable and efficient social storage cloud”. Games, 10, article no. 44, (2019).*

Contribution III: Chapter 6 and 7

The objectives of this research are: first, to investigate the impact of agents' decision of link addition and deletion on their local resource availability. Second, to extend our knowledge of externalities (spillover). In particular, this research studies the role of agents' closeness, and the network size and density in determining what type of externalities these agents experience in the network. Third, to model the choices of agents that suggest with whom agents want to add links in the social cloud. Fourth, to perform pairwise stability analysis of social compute cloud formation.

The findings include; first, agents' decisions of link addition (deletion) increases (decreases) their local resource availability. However, these observations do not hold in the case of global resource availability. That is, the action of link addition (deletion) of an agent may decrease (increase) its global resource availability. We, then, study externalities (in terms of global resource availability) experimentally, show that for populated ring networks, one or more agents experience positive externalities due to an increase in the closeness of agents. We show that, in a two-diameter network agents always experience negative externalities. Further, the initial distance between agents forming a link has a direct bearing on the number of beneficiaries, and the number of beneficiaries is always less than that of non-beneficiaries. Next, Further, by focusing on the parameters such as closeness and the shortest distances, we provide conditions under which agents choose with whom they will form a link to maximize their utility. This research adopts a reverse approach of understanding stability of social cloud formation.

We discuss the above observations in Chapter 6. The results regarding an impact of network size and density on externalities are also appeared in: *P. C. Mane, K. Ahuja, N. Krishnamurthy "Externalities in endogenous sharing economy networks". Applied Economics Letter, (2019). DOI: 10.1080/13504851.2019.1683507*

We state our initial observations on pairwise stability of social compute cloud (see Chapter 7. In particular, we provide conditions a two-diameter network and other standard topologies (e.g., the star, the complete, the complete bipartite network) are pairwise stable.

1.5 Organization of the Thesis

The rest of the thesis is organized into the following chapters, with the summary of each chapter as provided below:

Chapter 2 (Background and Preliminaries)

This chapter provides background information related to social cloud and network formation games. It presents a general framework of the social cloud, which includes the concept of social storage also. We discuss the different views, applications and the current status of social cloud in Appendix A. The concept of social storage, its characteristics and taxonomy are discussed in Appendix B in detail. Further, this chapter introduces the framework of strategic network formation model, which acts as a tool to analyse social cloud formation. It discusses several aspects (such as solution concept, efficiency and inefficiency, and externalities) related to the strategic network formation model.

Chapter 3 (Social Storage Models)

This chapter introduces our models of social cloud formation, namely, social storage network and social storage cloud. Specifically, it presents two types of utility functions: degree-based in social storage network context and degree-distance-based, in the social storage cloud context. Then, it discusses network stability notions, namely, pairwise stability and its refinement bilateral stability, introduced by us.

Chapter 4 (Social Storage Networks: Stability, Efficiency and Contentedness)

This chapter examines bilateral stability, efficiency and contentedness of social storage networks.

Chapter 5 (Social Storage Cloud: Stability and Efficiency)

This chapter analyses pairwise stability, efficiency and inefficiency of social storage cloud. It also examines externalities in the the social storage cloud.

Chapter 6 (Social Storage Cloud: Resource Availability)

The focus of this chapter is on the analysis of resource availability. It examines local resource availability and global resource availability in the context of social storage cloud. It also discusses preference modelling, which suggests to whom agents would prefer to form resource sharing connections.

Chapter 7 (Social Compute Cloud: Pairwise Stability)

This chapter discusses our observations on pairwise stability of symmetric social compute cloud.

Chapter 8 (Conclusion and Future Work)

This chapter summarizes the contribution of this thesis and states possible future directions of our work.

Chapter 2

Background and Preliminaries

This chapter provides an overview of social cloud, social storage and network formation game. In Section 2.1, we discuss a brief literature review of social cloud and social storage, so as to lay foundation for discussion in subsequent chapters. Further details are discussed in Appendices A and B. In Section 2.2, we provide a glimpse on the concept of networks, network formation games. Appendix C contains more details.

2.1 Social Cloud and Social Storage

Although there is no general agreement on what exactly social cloud is, it has been widely accepted that the concept of social cloud integrates social knowledge (in terms of social network or connections) and the idea of sharing computing resources between socially connected users. Social cloud is a framework, which allows users to share or trade computing resource available at their end. Note that, shareable entities in Social cloud could be processing power, storage space, workflows, information, and shareable software that are available with users. Till date, social cloud has appeared in various forms of distributed computing (community cloud, grid computing and volunteer computing) through various applications. We provide a detailed survey of social cloud in Appendix A.

Social storage is another concept (inspired by P2P data backup) and follows the same notion of a combination of social network and storage resource sharing between users. Unlike social cloud, social storage is specific about the shareable entity, which is storage space.

We will discuss the idea of social storage in more details in Appendix B (that includes social storage functionalities, characteristics and its taxonomy).

One can view social storage as a subset or a special case of social cloud. In this section, we present the view of computer scientists and economists on social cloud and social storage. Then, we provide a general framework of social cloud, which subsumes social storage.

2.1.1 Social Cloud and Social Storage: Computer Science Outlook

Computing paradigm has seen a massive shift in last few years. This gradual shift in computing paradigm is essentially to address the different needs of the hour. Distributed computing concept emerged with an aim to connect agents and computing resources in a transparent, reliable and scalable way to achieve huge computing power. Subsequently, cluster computing, a form of distributed computing enables utilization of computational power of standalone computers by integrating them. Cluster computing logically provides a single unified computational resource. This follows a number of resource sharing computing paradigms namely Grid computing [7], Peer-to-Peer (P2P) computing [6], Volunteer computing [5], Utility computing [8] and recently Cloud computing [9]. These computing paradigm share common characteristics like resource sharing or utilization of computational power of standalone resources. However, these are distinguishable with respect to their resource provision mechanisms, domains of applicability and associated principal stakeholders. For example, Grid and Volunteer computing operates in trust less framework wherein risk is associated with the notion of 'sharing' resources or services [33]. Specifically, in volunteer computing, the volunteered hosts are unreliable and insecure, hence, it is likely that incorrect result can be yielded by malicious volunteers [34]. The more recent cloud computing is very much business centric and may not fit well for those who are concerned about cost or do not bother about strong performance guarantees [35].

In the past few years, researchers have shown an increased interest in social networks to develop the idea of *social cloud*. One group of computer scientists [36, 4, 2, 20, 37, 38, 39] have been focusing on social cloud. The increasing trends of donating computing resources by users (at the edge of the internet) to scientific projects such as SETI@home[8], Folding@home[9] inspires these researchers to think about social cloud. This community aims to take advantage of social connections to deal with the limitations associated with the above computing paradigms. Although there are many views on social cloud, the definition stated in [4] is widely accepted in the literature. According to Chard et al. [4], a social cloud is "a resource and service sharing framework utilizing relationships established between members of a social network". Researchers believe that social connections will assure interpersonal trust and help incentivizing and regulating resource sharing among the participants. In social cloud, shareable entities could be computing capacity (storage space, computing power, etc.), people, software, or information [4].

Another group of computer scientists [12, 16, 11, 13, 14] have been focusing on *social storage* or *Friend-to-Friend* data backup systems. This community believes that exploiting social knowledge can help to deal with various issues such as quality of service (include resource availability, data availability, reliability, and its security), resource management, and trust that are associated with peer-to-peer data backup systems [40, 41, 42, 43, 44].

2.1.2 Social Cloud and Social Storage: Economics Outlook

Many forms of distributed computing (Grid, Cloud, P2P computing) have been adopted in commercial settings with various business models. The studies suggest that these computing paradigms have made a positive impact on business and organizations [45, 46, 47, 48], and therefore, on the economy. For example, cloud computing offers various computing services (infrastructure-, software-, platform- as a service) through different deployment plans (e.g., public, private and community) suitable to business needs.

From the point of view of economists, traditional cloud services like Amazon' s AWS S3[10] and IBM Cloud[11] are examples of agents in a *horizontal market* where different cloud providers compete for customers' requests for resources. On the other hand, cloud services

[8]https://setiathome.berkeley.edu/
[9]https://foldingathome.org/
[10]https://aws.amazon.com/s3
[11]https://www.ibm.com/cloud

like DropBox[12] and Mega Cloud[13] are examples of agents in a *vertical market* where a broker resells the resources of a cloud provider.

Social cloud and social storage are examples of *the sharing economy network* [49, 50, 51], where two individual agents (embedded in a network) share (or trade) resources directly with each other, without an intermediary third-party (cloud provider or broker). For example, CuckuBackup and BackupCow are online data backup systems, which allow their users to select their data backup partners who share their underutilized disk space and perform data backup on the shared space. These kinds of social storage systems have made a positive impact on the economy. The recent report of kbv research[14] expects that the data backup and recovery market will grow at a Compound Annual Growth Rate (CAGR) of 10.2 of during the forecast period and reach 12.9 billion by 2023.

On the one side, economy is benefited by these computing paradigms. On the other side, these computing paradigms are also benefited by various economic approaches to deal with resource allocations in these computing frameworks [52, 53, 54]. Many studies have focused on strategic resource sharing settings and make use of game-theoretic models to deal with resource allocation in such settings [55, 56, 57]. In the context of social cloud and social storage, researchers have been studying resource allocation on existing networks.

Now, we present a general purpose framework of the social cloud (as shown in Figure 2.1) that subsumes social storage. This framework ignores the technical aspects of both social cloud and social storage paradigms.

2.1.3 Social Cloud: A General Framework

A general framework of the social cloud consists of two layers. The underlying layer is social knowledge in the form of social connections or network. The top layer is an application layer.

[12]https://www.dropbox.com

[13]https://mega.nz/

[14]7Report ID: 978-1-68038; On-line available at: https://kbvresearch.com/data-backup-and-recoverymarket/

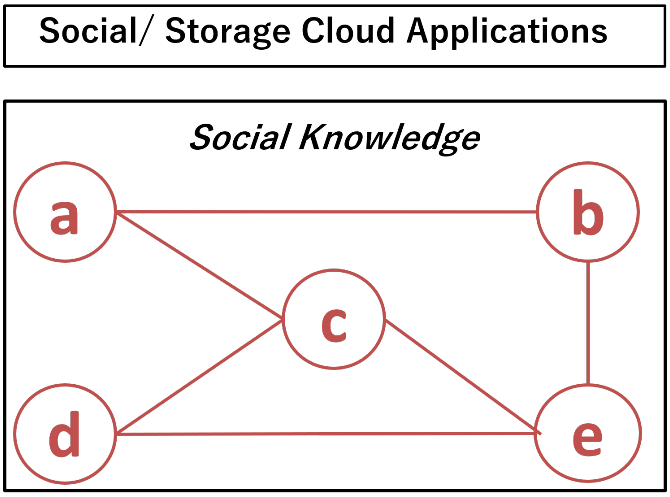


Figure 2.1: A general framework of Social Cloud.

2.1.3.1 Application Layer

An application makes use of social knowledge to facilitate resource sharing between users.

2.1.3.2 Social Knowledge

As stated earlier, a social network is at the core of the social cloud. We define these social networks (connections) based on how they are obtained for the development of social cloud systems.

2.1.3.2.1 Exogenous social networks: An exogenous social network is a social graph that is extracted from an online social network. For example, SSS (a social cloud system) extracts a social graph from Facebook and allows Facebook users to share their storage resources with each other.

2.1.3.2.2 Endogenous social networks: An endogenous social network is one which evolves through the actions of agents in the social cloud application context. In other words, agents select their resource sharing partners in the context of the application. For example, BuddyBackup (an online social storage system) allows agents to select their partners with whom they are socially connected in the real world.

2.1.3.2.3 Exo-Endo-genous social networks An exo-endo-genous social network is a combination of exogenous and endogenous social cloud in which, first, a social graph is obtained from an online social network and, next, agents select their resource sharing partners from the extracted social graph. For example, Cucku-backup[15] (an online social storage system) obtains a social graph by extracting Skype[16] and then allow agents to select their partners with whom they are connected in Skype.

The survey [2] distinguishes numerous social cloud systems based on how social knowledge (or social network) is obtained. We extend this by incorporating many other social storage systems and also the type of social connections. Table 2.1 (an extension of Table A in [2]), shows the comparison of various social storage (cloud) systems along with resource and social network type.

2.2 Network Formation Game (NFG)

This study adopts the approach of strategic network formation modelling to model endogenous social storage cloud formation. This approach stems from economics, where agents build their relationships based on their payoffs as a function of the network. This section discusses the network formation game (NFG) model with its elements. Then, it discusses the aspects of NFG such as the solution concepts, network efficiency, price of anarchy and stability. We begin the discussion with a mathematical presentation of networks.

2.2.1 Network: Mathematical Foundation

Conceptually, a network is simply a collection of nodes and links connecting these nodes. In fact, a network represents a system, which is made up of entities (represented by

[15]https://cucku-backup.apps112.com/
[16]ttps://www.skype.com

System	Resource	Social Network
SSC†	Storage	Exogenous
SCC†	Compute	Exogenous
F2Box(FriendBox)‡	Storage	Exogenous
Friendstore‡	Storage	Exogenous
CRB − Model†	Compute	Exogenous
SocialCloud†	Compute	Exogenous
BuddyBackup‡	Storage	Endogenous
Crashplan‡	Storage	Endogenous
Multi-community-cloud collaboration	Compute	Exogenous
CuckuBackup‡	Storage	Exo-endo-genous
BackupCow‡	Storage	Exogenous
Community clouds and Community Networks†	Compute	Exogenous
HSSC†	Information	Exo-endo-genous
Blockparty‡	Storage	Exo-endo-genous

Table 2.1: Comparison of existing *social cloud*† and *social storage*‡ systems along with resource and social network type.

the nodes of the network) and the interactions (or relations) between these entities (captured by the links of the network). For example, in a friendship network, persons are represented by nodes and the friendship between pairs of people are captured by links between them.

Table 2.2 lists a few real-world systems and corresponding network representations that consist of nodes (the entities involved in the system) and links (the relation type between these entities). For example, a scientific collaboration (**system**) can be represented by a co-authorship **network** where **nodes** are scientists and a **link** between a pair of scientists exist if they share a research article.

System	**Network**	**Node**	**Link**
Online communication	Online social network	User profile	Sharing
World Wide Web	World Wide Web	Web page	Hyperlink
Air transport	Air transport network	Airport	Flight
Movie collaborations	Movie collaborations	Actor	Movie
Scientific Collaboration	Co-authorship network	Scientist	Scientific paper
Trading	Trade network	Trader	Bilateral trade

Table 2.2: Examples of real-world systems with network representation

Formally, a network is defined as follows.

Definition 2.1. *A network $\mathfrak{g} = (\mathscr{A}, \mathscr{L})$ is a pair of sets $\mathscr{A}$, a non-empty set of agents that correspond to nodes, and $\mathscr{L}$, a set of unordered pairs of agents that correspond to links.*

The set $\mathscr{A}$ of N agents is indexed by $i, j, \cdots$, unless otherwise specified. The number of agents N (that is, $|\mathscr{A}| = N$) is the size of the network $\mathfrak{g}$. A link $\langle ij \rangle \in \mathscr{L}$ is the unordered pair of agents $\{i, j\}$, which suggests that $\langle ij \rangle$ joins (or links) agents i and j. That is, links are non-directed, and hence, the network $\mathfrak{g}$ is undirected. The number of links (elements) in the set $\mathscr{L}$ is denoted by ℓ, that is, $|\mathscr{L}| = \ell$. If $\langle ij \rangle \in \mathscr{L}$, we call the agents i and j as neighbors (or adjacent) in the network $\mathfrak{g}$. The set of agents with whom agent i has direct links in $\mathfrak{g}$ is represented by $\eta_i(\mathfrak{g})$. We also use $\eta_i(\mathfrak{g})$ to represent the neighborhood size of agent i in $\mathfrak{g}$, which will be clear from the context. The set, $\mathscr{G}(N)$, consists of all possible networks on N agents.

Given distinct agents $a_1, a_2, \cdots, a_n \in \mathscr{A}$, if $\langle a_1, a_2 \rangle, \langle a_2, a_3 \rangle, \cdots, \langle a_{n-1}, a_n \rangle \in \mathscr{L}$, then there is a *path* $\mathscr{P}_{a_1 a_n}(\mathfrak{g})$, from a_1 to a_n, of length $n-1$. The distance $d_{ij}(\mathfrak{g}) (= d_{ji}(\mathfrak{g}))$ between a pair of agents i and j is the length of the shortest path connecting them in $\mathfrak{g}$. The diameter $\mathscr{D}_{\mathfrak{g}}$ of network $\mathfrak{g}$ is the maximum distance between any pair of agents. A path of length ≥ 2 between a pair of agents is an indirect communication channel between them.

A network $\mathfrak{g}$ may be connected or may consist of two or more connected components. We say a network $\mathfrak{g}$ is connected if there exists at least one path between any pair of agents i and $j \in \mathbf{A}$, or else the network $\mathfrak{g}$ is disconnected. A disconnected network $\mathfrak{g}$ can be partitioned into disjoint sub-networks $\mathfrak{g}(\kappa_1), \mathfrak{g}(\kappa_2), ..., \mathfrak{g}(\kappa_n)$, where $\kappa_1 \cup \kappa_2 \cup \ldots \cup \kappa_n =$

$\mathbf{A}$, $\kappa_r \cap \kappa_s = \phi$ for all $r, s \in \{1, 2, \ldots, n\}, r \neq s$, such that any pair of agents i and j are connected if and only if i and j are elements of the same set κ_r. Such sub-networks are called as *components* of the network $\mathfrak{g}$.

The notation we use for mathematical representation of a network are used in the subsequent chapters. Appendix C discusses more details and defines standard network structures (topologies), for example, complete, star network, etc.

2.2.2 NFG: Model

We define a non-cooperative game with agents, actions, rules, outcome, and the utility or payoff of agents (refer Appendix C.3). We can apply these elements to a network formation game. Table 2.3 maps non-cooperative games to network formation games.

Elements of Non-cooperative Game	**Elements (applied to) Network Formation Game**
Agents	Agents who correspond to points
Actions	The link formation actions available to each agent
Rules	Rules that define how agents may form links
Outcomes	Network(s) that is(are) formed
Payoff (utility) function	Function that assigns a payoff (utility) to each agent given the formed network

Table 2.3: Description of network formation game [1]

A network formation game can be described in terms of the following elements.

1. Network with its Agents: We are given a network (graph) $\mathfrak{g} \in \mathscr{G}(\mathscr{N})$ on N agents, where $\mathfrak{g} = (\mathscr{A}, \mathscr{L})$, $\mathscr{A} = \{1, 2, \cdots N\}$ being the set of rational agents (players), who are the decision makers and $\mathscr{L}$ being the set of ℓ undirected links connecting these agents. Recall that a link that connects agents i and j in $\mathfrak{g}$ is denoted by $\langle ij \rangle$.

2. Actions: It specifies a set of link *update* choices (e.g., add and delete links, or accept and reject links, etc.) available to each agent. For instance, agents can perform the actions, such as, link addition and deletion, a replacement or pass of a link in the local- and global-Nash networks model [1]. In our study, we consider only link addition and link deletion actions that are available to agents.

3. Rules: We consider the set of rules that define how agents may *update* links, that is, unilateral link formation and (or) unilateral link deletion, or bilateral link formation and (or) bilateral link deletion, and so on. For instance, an agent needs mutual consent of another agent with whom they want to form a link [31], or an agent forms link without mutual consent of another agent [58].

4. Outcome: An *outcome* is a network $\mathfrak{g}$ that forms as a result of the agents' choices (link addition or deletion).

5. Utility: A *utility (payoff) function* assigns a payoff to each agent as a function of the network. That is, given a network $\mathfrak{g}$, the utility of each agent i is given by $u_i : \mathscr{G}(N) \to \mathbb{R}^+$. The profile of utility functions $(u_1, u_2, \cdots, u_n)$ is a vector of utilities for all agents that is represented by $\mathbf{u}$. Therefore, $\mathbf{u} : \mathscr{G}(N) \to \mathbb{R}^N$. The value $\upsilon(\mathfrak{g})$ of network $\mathfrak{g}$, is the total of all agents' utilities in the network $\mathfrak{g}$, that is, $\upsilon(\mathfrak{g}) = \sum_{i=1}^{N} u_i(\mathfrak{g})$.

In general, a utility function can be categorized based upon agents from whom each receives benefits in the network — from only directly connected agents or from indirectly connected agents too, and also whether it depends on the number of connections or the distance between agents or both. There are three types of utility functions, which are widely discussed in the strategic network formation literature, namely, *degree-based utility*, *distance-based utility* and *degree-distance-based utility*.

Degree-based utility: A utility function is degree-based if agents benefit only from direct neighbors, and the benefit decreases with an increase in the number of neighbors of each neighbour [59]. In degree based utility functions, one is only concerned about the effects of its local neighbourhood. Although indirect connections are not considered explicitly here, they do affect an agent's utility either positively or negatively. For example, in the co-author model [31], where agents are involved in a collaborative project, an agent's utility goes down if its neighbours are tightly connected (or agents are densely connected in the network). In the job contact network [60], for an agent, the probability of getting job information increases as its neighbourhood size increases. However, the chance of obtaining job information also depends on how the agents are connected (tightly or loosely) and unemployment in the network.

Distance-based utility: A utility function is distance-based, if agents obtain benefit from direct as well as indirect connections; but utility decreases as the distance between agents increases [61]. The connection model [31], the network creation game [62], the locality game [63], are some examples where a distance based utility function is used. In general, a distance based modelling is suitable for those settings where agents are aiming to minimize the cost of communication.

Degree-distance-based utility: A utility function is degree-distance-based if agents obtain benefits from direct and indirect links, but the benefit decreases with an increase in the number of direct and indirect neighbors [64].

2.2.3 Modeling Network Formation

An exhaustive survey of network formation games and games on networks has been done in the following works: [28, 27, 1, 65, 30]. Few of these models broadly cover strategic network formation modelling. This includes, the cooperative game theory model, the unilateral connection model, the link investments model, and the bilateral connection model ([1]). Next, we briefly discuss these models and also relate them to our model.

Aumann and Myerson [66] have proposed an extensive network formation game, where agents form links sequentially (one after another) using some exogenous rules. Agents propose with whom they want to form links, and later this proposal is either accepted or rejected by others. But, once a link is formed between a pair of agents, it cannot be withdrawn. This is the essence of the cooperative game theory model. In the unilateral connection model, agents form links without consent and links are directional ([58]). In the link investments model ([67]) and its variant ([68]), agents propose investments for their every direct link. These investments are either positive or negative. Linking between a pair of agents takes place if and only if total investment on that link is positive.

Myerson [69] has proposed the link-announcement game. In this, agents proposing to form links announce the name of the agents with whom they want to form these links. This announcement is done simultaneously. A link between two agents takes place if and only if both agents announce each others name. Inspired by the link-announcement model, Jackson and Wolinsky [31] proposed the pairwise connection model. Here, link formation takes place with the mutual consent of the involved agents, however, link deletion takes place without consent. Our social storage network formation game is inspired by the pairwise connection model. We differ in the link deletion scenario, where both the agents must agree to delete the link between them.

2.2.4 NFG: Solution Concepts

Which network is likely to emerge when agents strategically decide with whom they want to form links and with whom they do not? This is the central question in the field of strategic network formation. In particular, researchers are interested in a *strategically stable or equilibrium network*. We say, a network is strategically stable or in equilibrium if no agent has incentives to alter the network structure in place, either by adding or deleting links.

In Subsection 2.2.4.1, we discuss *the pairwise stability* solution concept (due to Jackson and Wolinsky [31]), a widely discussed and accepted solution concept in the literature of strategic network formation. We propose the concept of *bilateral stability*, a variant of pairwise stability and discuss the same in Chapter 3. In Subsection 2.2.4.2, we discuss the *bilateral equilibrium* solution concept (due to Goyal et al. [32]) and list other solution concepts, some of them being refinements of pairwise stability.

2.2.4.1 Pairwise Stability

Jackson and Wolinsky [31] observe that Nash equilibrium as a solution concept is not useful in the network formation context for two reasons. First, due to the existence of multiple Nash equilibria (for instance, the null network is always a Nash equilibrium irrespective of the utility function), and second, it fails to capture the requirement of mutual consent of agents in link formation. Hence, they propose the *pairwise stability solution concept*, which is defined below.

Definition 2.2. *[31] A network* $\mathfrak{g}$ *is pairwise stable if*

1. *for all* $\langle ij \rangle \in \mathfrak{g}$, $u_i(\mathfrak{g}) \geq u_i(\mathfrak{g} - \langle ij \rangle)$ *and* $u_j(\mathfrak{g}) \geq u_j(\mathfrak{g} - \langle ij \rangle)$, *and*

2. *for all* $\langle ij \rangle \notin \mathfrak{g}$, *if* $u_i(\mathfrak{g} + \langle ij \rangle) > u_i(\mathfrak{g})$, *then* $u_j(\mathfrak{g} + \langle ij \rangle) < u_j(\mathfrak{g})$.

That is, a network $\mathfrak{g}$ is pairwise stable with respect to utility function u, if (1) no agent benefits by deleting an existing link and (2) no two agents benefit by adding a new link between them.

The main shortcoming of pairwise stability is that it only considers a simple single link deviation. Despite this limitation, it is a suitable solution concept to characterize networks in contexts where self interested agents decide with whom they want to form social connections and with whom they do not [70]. In fact, one can view pairwise stability as a necessary condition (although not sufficient), to understand the true stability of a network.

2.2.4.2 Other Solution Concepts

Bilateral equilibrium [32] is another refinement of *pairwise stability* [31]. Goyal et al. [32] define strategies of agents as sets of links they would want to add, and define *bilateral equilibrium* as a strategy profile that is a Nash equilibrium (that is, no agent benefits by unilaterally deviating) and is pairwise stable (where both addition and deletion require mutual consent). The set of all bilaterally stable strategies as defined by us (see Definition 3.3) is a super-set of the set of all bilateral equilibrium strategies [32].

Other network formation game solution concepts include strong and coalition-proof Nash equilibria [71], strong pairwise stability [72], pairwise stable Nash equilibrium [73], farsighted equilibrium [74], pairwise farsightedly stable [75], Nash-Cournot equilibrium [76], and monadic stability [77].

2.2.5 NFG: Efficiency

One of the central questions in the study of strategic network formation is whether the network that is evolved when rational agents take decisions of link formation and deletion themselves is efficient (value-maximizing) or not? Under what condition(s) do rational agents form a network that is 'best' from the society's perspective. Identifying the network(s) that maximize the overall benefit of the society means finding the efficient network(s).

Definition 2.3. *A social storage network* $\mathfrak{g}$ *is efficient with respect to utility profile* $(u_1,...,u_N)$ *if* $\sum_i u_i(\mathfrak{g}) \geq \sum_i u_i(\mathfrak{g}')$*, for all* $\mathfrak{g}' \in \mathscr{G}(N)$.

The above notion of efficiency is *utilitarian efficiency*, which says that a network is *utilitarian efficient* with respect to the utility function if it maximizes the total of utilities of agents of the society. However, there are many studies [78, 79] which consider Pareto efficiency as a network efficiency measure. A network is Pareto efficient if it is not possible to make any agent better off without making at least one other agent worse off. In this study, we do not focus on the notion of Pareto efficiency.

Jackson and Wolinsky [31] observe the tension between pairwise stability and efficiency. They identify that a pairwise stable network may not be efficient and an efficient network may not be pairwise stable. If a network is pairwise stable but not efficient then agents may improve the overall benefit of the society by altering the network structure. However, no pairs of agents want to add a new link or no agent wants to delete an existing link in the network that is in place.

In general, distance-based utility models display the tension between pairwise stability and efficiency. For example, in the connection model [31] all pairwise networks are not efficient. The spatial connections model [78] is consistent with the insights derived in [31] regarding the conflict between pairwise stable and efficient networks.

However, this is not always the case. For example, in the symmetric connection model, for some cost-benefit range there are a few networks which are pairwise stable and efficient also. Similarly, the centrality[17] based network formation model proposed by Buechel [80] observes unique pairwise stable networks which are also efficient. We have consistent observations with the centrality model for the case of the social storage cloud formation model (discussed in Chapter 5).

2.2.6 NFG: Inefficiency

When rational agents take decisions of link formation and deletion themselves, the network formation may lead to inefficiency. The lack of coordination between self-interested agents (who may have competing interests) results in network inefficiency. The question that arises here is how inefficient the evolving or evolved network(s) is (are)? It is worth noting that the measures of efficiency are inadequate to assess *how inefficient* the emerging networks are? The question of network inefficiency is answered by computing the gap of the welfare (i.e., overall benefit) between the set of stable networks and the efficient network. The notion of Price of Anarchy (PoA) quantifies this gap. Formally, it is defined as below.

Definition 2.4. *The price of anarchy (PoA) is the ratio of the worst sum of the utility value of an equilibrium network and the optimal sum of the utility value in any network.*

So, in the context of pairwise stability, PoA is the ratio of the lowest welfare (overall benefit) generated by any pairwise stable network to the value of efficient network.

Just as Price of Anarchy looks at the worst pairwise stable networks, the Price of Stability (PoS) looks at the best possible pairwise stable networks.

[17] Network centrality (a widely discussed phenomenon in the literature of social network analysis) measures the importance of agents (nodes) in a network. We discuss network centrality in more details in Appendix C.2.

Definition 2.5. *The price of stability (PoS) is the ratio of the best sum of the utility value of an equilibrium network and the optimal sum of the utility value in any network.*

Therefore, in the context of pairwise stability, PoS is the ratio of the largest welfare (overall benefit) generated by any pairwise stable network to the value of efficient network.

Remark 2.1. *If PoA equals to 1 then a pairwise stable network is also efficient. If PoS equals to 1 then then it means there is an efficient network which is also pairwise stable [31]*

2.2.7 NFG: Externalities

A network formation may lead to an inefficient outcome. To be specific, the questions here are as follows. First, why does network formation lead to inefficiency? And second, why is there conflict between pairwise stability and efficiency? Externalities are considered as the main source of the tension between pairwise stability and efficiency. Externalities (spillover) are viewed as an impact on the agent's utility due to the actions of link formation by other agents. Formally, externalities are defined as follows.

Definition 2.6. *[61] Consider a network, $\mathfrak{g}$, with agents $i, j \in \mathfrak{g}$ such that $i \neq j$ and $\langle ij \rangle \notin \mathfrak{g}$. Suppose agents i and j form a direct link $\langle ij \rangle$. Then, agent $k \in \mathfrak{g} \setminus \{i, j\}$ experiences*

1. *Positive externalities if $u_k(\mathfrak{g} + \langle ij \rangle) > u_k(\mathfrak{g})$;*
2. *Negative externalities if $u_k(\mathfrak{g} + \langle ij \rangle) < u_k(\mathfrak{g})$;*
3. *No externalities if $u_k(\mathfrak{g} + \langle ij \rangle) = u_k(\mathfrak{g})$.*

In general, a network formation model with partition value function is without externalities, since, the utility function in such networks assign payoffs to the partitions. Normally, a degree based utility model displays negative externalities while a distance-based utility model displays positive externalities. However, these observations are obtained in the context of information flow networks.

The literature has several network formation models, which exhibit positive, negative, positive-negative externalities, no externalities. For example, the *connection model* [31], the *provision of a pure public good* [73], the *market sharing agreement* [81] models are few examples that exhibit positive externalities. The models such as *Co-author* [31], *the free trade agreements* [82], and the *patent races* [73] exhibit negative externalities. Several authors [83, 64, 84] have studied the network formation in the presence of both positive and negative externalities.

The analysis of externalities is crucial for a third-party observer/ planner who wants to convert the network formation to an efficient outcome. In order to lead network formation to a socially preferred outcome, the planner may perturb the network formation by subsidizing agents for link formation.

2.3 Chapter Summary

In this chapter, we discussed the concept of Social Cloud, a socially-aware resource sharing framework that allows users to share computing resources (e.g., storage, computing capacity, work-flow, etc.) available at their ends with other users in the social network context. Although there is no consensus on the view of Social Cloud, the numerous and different social cloud systems show the potential of Social Cloud to act as a complementary to various other distributed computing frameworks (like Grid, Cloud, Volunteer computing). At present, researchers are dealing with various issues associated with Social Cloud, such as trust, incentives, resource management, computing infrastructure and security.

Further, the concept of Social Storage, a twin phenomenon of Social Cloud. Social Storage is a special case of Social Cloud, where users share their storage resources with each other for data backup purpose, in a social network context.

In this chapter, we presented a general framework of the social cloud that includes social storage as well. This framework consists of two layers, namely, application and social knowledge. The social knowledge layer can be viewed in the form of social connections, such as, exogenous, endogenous and exo-endogenous.

As stated in the Introduction, the research carried out is in order to analyse social cloud (informally, a socially-aware resource sharing network) formation in a strategic setting.

We present the fundamentals of how network formation games are represented. Further, we discuss the solution concepts of network formation with specific focus on pairwise stability. Then, we discuss various aspects such as network efficiency, inefficiency, and externalities (or spillover), which are at the centre of the study of network formation games.

Chapter 3

Social Cloud Models

This study presents three models of strategic social cloud formation, namely, social storage network model, social storage cloud model and social compute cloud model. In the social storage model, agents share their storage space with only their immediate neighbours. This model is inspired by studies like [12, 13] where exogenous social connections are considered. The second model is a social storage cloud model where agents perform storage resource sharing with agents who are directly as well as indirectly connected. This study follows the idea of closeness based resource sharing due to [4]. In fact, studies [23, 24] have suggested incorporating indirect links in such storage sharing. The third model is social compute cloud model, which is a variant of the above two models. In this model, agents share computing resources (not necessarily storage) with other agents who are directly or indirectly connected with them. This model is inspired by the social cloud system presented in [85] where agents want to complete a computational task and, hence, strive for resources in the social cloud. Note that, in the social cloud model [85], agents outsource computational tasks to their immediate neighbours. In our social compute cloud model, agents perform their computational task themselves.

Modelling a utility function (the payoff that each agent receives in a network) is the foremost requirement to study network formation in a strategic setting [61]. This aspect has not been given much attention by researchers working in the social storage domain. In the strategic network formation literature, specifically endogenous network formation games, different kinds of utility functions have been proposed and successfully validated, such as degree-based etc., as summarized in the previous chapter. Utility modelling is more crucial in the social storage context, where decision makers are human agents who aim to optimise their own goals. This is in comparison to the P2P storage context (our closest cousin), where nodes (computer systems) are decision makers. In this study, we define three utility functions, namely, a degree-based, a degree-distance based and a variant of the latter, one for each model as discussed above. We describe these utility functions in the respective models.

Table 3.1 summarizes all notations used in this thesis.

$\mathfrak{g}$	social storage network or social storage cloud or social compute cloud.
$\mathscr{A}$	set of agents (or vertices).
N	number of agents in $\mathfrak{g}$ (that is, N is the number of elements in the set $\mathscr{A}$).
$\mathscr{L}$	set of links (or edges).
$\langle ij \rangle$	link between agents i and j.
a_{ij}	indicator for data backup partnership between agents i and j.
ς	cost incurred by an agent to maintain a link.
β_i	worth (or value) that agent i has for its data.
$\mathbf{s}_i$	amount of storage available with agent i that it can contribute to other agents.
$\mathbf{d}_i$	amount of data that agent i wants to backup.
$\mathbf{b}_i$	budget allocated by agent i towards backup partnerships.
λ	probability of failure of a disk.
$\eta_i(\mathfrak{g})$	neighbourhood size of agent i in $\mathfrak{g}$. (Also denotes the set of neighbours of i).
$\mathfrak{g}+\langle ij \rangle$	new link $\langle ij \rangle$ is added to $\mathfrak{g}$.
$\mathfrak{g}-\langle ij \rangle$	existing link $\langle ij \rangle$ is deleted from $\mathfrak{g}$.
$\mathscr{G}(N)$	the set of all networks on N agents.
$\mathfrak{g}(\kappa_i)$	a component of network $\mathfrak{g}$, where κ_i is the set of agents in that component.
$\mathfrak{g}^c$	complement of network $\mathfrak{g}$.

Table 3.1: Notation summary

3.1 Social Storage Network Model

Definition 3.1. *A social storage network* $\mathfrak{g} = (\mathscr{A}, \mathscr{L})$ *consists of a set of agents,* $\mathscr{A}$*, and a set of links connecting these agents,* $\mathscr{L}$*, where a link between two agents represents a data backup partnership between them.*

Given a social storage network $\mathfrak{g} = (\mathscr{A}, \mathscr{L})$, the link $\langle ij \rangle \in \mathscr{L}$ represents the fact that agents i and j are neighbours of each other, and are involved in a data backup partnership. This partnership indicates that both the agents commit to share their storage resources with each other so that they can backup their data on each other' s shared storage space. At any given point in time, each agent plays a dual role: that of a data owner who wants to back up its data, and that of storage provider who provides storage space for each of its backup partners. Storage resource sharing and data backup activity are bidirectional and occur with the mutual consent of i and j. This implies, the link $\langle ij \rangle$ and the link $\langle ji \rangle$ are identical. We also refer to i and j as backup partners. The set of agents with whom agent i has links is represented by $\eta_i(\mathfrak{g})$. In other words, $\eta_i(\mathfrak{g})$ is the neighbourhood of agent i. We also use $\eta_i(\mathfrak{g})$ to represent the neighbourhood size of i, which will be clear from the context.

3.1.1 Link Addition and Deletion Rule

Pairs of agents may add a new link (or continue to maintain the existing link) or delete the existing link (or continue to remain without a link). In the context of social storage, mutual consent is necessary for adding as well as for deleting links. That is, *an agent does not add a new link without the consent of the agent with whom it wants to add the new link and does not delete an existing link without the consent of the agent from whom it wants to delete the existing link.*

3.1.2 Network Formation

The structure of the network, $\mathfrak{g}$, is determined by actions of the agents. Firstly, the network is updated when two agents i and j add a new link $\langle ij \rangle$, and we denote this by $\mathfrak{g} + \langle ij \rangle$. Secondly, the network is updated when a pair agents i and j delete an existing link $\langle ij \rangle$, and we denote this by $\mathfrak{g} - \langle ij \rangle$. As agents themselves decide with whom they want to perform backup partnerships and with whom they do not, this is a process of endogenous network formation (or partner selection In this chapter, we do not explicitly consider trust between pairs of agents. We assume that *all agents* trust each other, and thus, anyone can form links with anyone.

3.1.3 Utility of an Agent in a Social Storage Network

Data stored on local hard disk is in danger of getting lost or damaged due to local disk failure. Hence, to keep data safe, each data owner wants to backup its data. Social storage systems use two types of techniques to backup data. The first is erasure coding, and the second is replication[18] [22]. Erasure coding is the data redundancy technique in which a data object is divided into x blocks and recoded into y blocks ($y > x$). Then the main data block can be recovered from any subset of y. Replication is the data redundancy technique in which an agent maintains a single data copy on each partner' s storage device. In this chapter, we consider the replication technique.

As hard disks are prone to failure, there is a chance that a data owner's backup partner's hard disk also fails. It is likely that each backup partner's hard disk fails, so each data owner's interest lies in recovering at least one copy of its data so that the value of the data is intact. It is not hard to observe that each agent's chance of data recovery, given a particular disk failure rate, depends on its neighbourhood size. The more the number of neighbours, the higher the chance of data recovery.

In the absence of costs to add and maintain links, the aim of each agent in a social storage network is to maximize the chance of data recovery, given that the local copy of data has been damaged or lost. However, every agent incurs a cost for each of its links. Keeping this in mind, we define the utility of each agent in the network under two frameworks. The utility of agent i in the network $\mathfrak{g}$ is represented by a function $u_i : \mathscr{G}(\mathscr{A}) \to \mathbb{R}$, where $\mathscr{G}$ is the set of all networks, ($\mathfrak{g}$ is an element of $\mathscr{G}$). The profile of utility functions $(u_1, ..., u_n)$ is a vector of utilities for all agents. We first define the parameters required to define the utility function. $\lambda \in (0,1)$ is the average disk failure rate in the network. That is, at any point in time, the probability of failure of agent i' s disk is λ. For data owner (agent) i, the value of the local data that is to be backed up, is β_i. Each agent incurs a cost ς to maintain a link. That is, the total cost of adding and/ or maintaining a link is 2ς, and we assume that the agents connected by the link equally share this cost. This cost can be interpreted as the cost required for infrastructure, bandwidth, time, and so on. There is no *additional* cost to add a new link. Each agent i also has allocated budget $\mathbf{b}_i$ for maintaining its links. Further, each

[18][86] perform quantitative comparisons between these two techniques.

agent i has a certain amount of local data $\mathbf{d}_i$ that the agent wants to store on storage devices of backup partners. Also, each agent i has a certain amount of storage space $\mathbf{s}_i$ available for sharing with other agents in the network. Using these parameters, we now define the utility of an agent in the following two frameworks.

3.1.3.1 Multi-Objective Framework (MO-Framework)

In the first framework, there are two objective functions that each agent i tries to optimize. Firstly, each agent i wants to minimize the total cost associated with maintaining the links, i.e., $\varsigma\eta_i(\mathfrak{g})$. Secondly, each agent wants to maximize the expected value of backup data. Since the disk failure rate is λ, and i has $\eta_i(\mathfrak{g})$ neighbours, the expected value of i' s backup data is $\beta_i(1-\lambda^{\eta_i(\mathfrak{g})})$. Note that, as each agent is interested in "how many links to maintain", we look at the expected value of an agent' s backup data *given* that the local copy of the agent's data has been damaged or lost. For each agent i, these two objective functions can be written as a single objective function as follows:

$$[\alpha(\beta_i(1-\lambda^{\eta_i(\mathfrak{g})}))]-[(1-\alpha)(\varsigma\eta_i(\mathfrak{g}))], \quad \text{where} \quad \alpha \in (0,1). \tag{3.1}$$

For elegance of results on stability, we let $\alpha = 1/2$. We drop the factor of $1/2$ from (3.1), for all $i \in \mathscr{A}$, and just consider the following utility function $u_i(\mathfrak{g})$, for all $i \in \mathscr{A}$, for the given network $\mathfrak{g}$:

$$u_i(\mathfrak{g}) = \beta_i(1-\lambda^{\eta_i(\mathfrak{g})}) - \varsigma\eta_i(\mathfrak{g}). \tag{3.2}$$

As evident above, this is no longer a MO-optimization problem. We have done this conversion because (a) this is one of the easiest way to solve a MO-problem, and (b) our focus is on the network formation game, stability, efficiency, and contentedness of the network. Solving the MO-optimization problem without this conversion is part of future work, and we discuss that in Section 7. We also still call this a MO-framework a nomenclature (to differentiate with Single Objective (SO)-framework discussed below).

Each agent i wants to maximize $u_i(\mathfrak{g})$ over all possible values of $\eta_i(\mathfrak{g})$. The social optimization problem can be formulated as

$$max \sum_{i \in \mathbf{A}} (u_i(\mathfrak{g}))$$

such that

$$\eta_i(\mathfrak{g}) = \sum_{i,j \in \mathfrak{g}} a_{ij} \text{ and}$$
$$\mathbf{s}_i \geq \sum_{j \in \eta_i(\mathfrak{g})} \mathbf{d}_j a_{ij},$$

where,

$$a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ have a backup agreement,} \\ 0 & \text{otherwise.} \end{cases}$$

3.1.3.2 Single Objective Framework (SO-Framework)

In this framework, each agent i has only one objective (as compared to two in the previous framework). Each agent tries to maximize the expected value of backup data. The cost, $\varsigma\eta_i(\mathfrak{g})$, incurred by agent i to maintain links (which was the second objective function in the MO-Framework), appears in constraints here. This is because,

Remark 3.1. *The utility function in the SO-Framework may be reduced to the Constant Absolute Risk Aversion (CARA [87])*[19] *function. In the context of social storage, agents are* risk averse *as they do not want to "risk" losing their data, which is what the above utility function captures. This function may also be viewed as the Cumulative Distribution Function of an Exponential distribution, given that the disk failure rate is Poisson.*

Remark 3.2. *We explicitly write the formulation of the social optimization problems in the two different frameworks, as above, primarily to highlight that the cost is moved from the utility function in the MO-framework to budget constraints in the SO-framework. Our goal is not to solve these problems but rather analyse the corresponding network properties, for example, the efficiency of the resulting networks.*

[19] We refer the readers to a survey by [88] on functional forms for the utility functions of agents, based on their risk taking abilities.

3.1.4 Bilateral Stability, Efficiency and Contentedness

Recall, the *pairwise stability* solution concept introduced in [31] (see Definition 3.2) is an appropriate solution concept when agents require mutual consent while adding a link, but any agent can delete any of its existing links without consent.

Definition 3.2. *[31] A network* $\mathfrak{g}$ *is pairwise stable if and only if*

1. *for all* $\langle ij \rangle \in \mathfrak{g}$, $u_i(\mathfrak{g}) \geq u_i(\mathfrak{g} - \langle ij \rangle)$ *and* $u_j(\mathfrak{g}) \geq u_j(\mathfrak{g} - \langle ij \rangle)$*, and*

2. *for all* $\langle ij \rangle \notin \mathfrak{g}$*, if* $u_i(\mathfrak{g} + \langle ij \rangle) > u_i(\mathfrak{g})$*, then* $u_j(\mathfrak{g} + \langle ij \rangle) < u_j(\mathfrak{g})$.

However, the social storage system discussed earlier, impels us to focus on the requirement of bilateral consent while deleting a link as well. For instance, let agents i and j be backup partners. That is, i provides its storage space to j for the purpose of storing j's data, and vice versa. Now, let us assume that breaking a backup partnership without mutual consent is allowed. If agent i breaks the partnership without consent of j, then there is a threat that j will lose its data which is stored on i's storage space. Hence, backup partnerships in social storage networks have to be viewed as mutual contracts which cannot be broken unilaterally. We call this as *bilateral stability*.

We modify the pairwise stability concept introduced by [31] so as to ensure that deletion of links also happens with mutual consent. We call this modified pairwise stability as *bilateral stability*.

Bilateral equilibrium [32] is another refinement of *pairwise stability* [31][20]. [32] define strategies of agents as sets of links they would want to add, and define *bilateral equilibrium* as a strategy profile that is a Nash equilibrium (that is, no agent benefits by unilaterally deviating) and is bilateral stable (that is, no pair of agents can deviate bilaterally and benefit from the deviation and at least one of them strictly).

The set of all bilaterally stable strategies (see Definition 3.3) is a superset of the set of all bilateral equilibrium strategies [32], as discussed earlier.

The modified definition of pairwise stability we use for social storage is given below.

[20]Note that, Bilateral equilibrium refines pairwise stability by allowing pairs of agents to add and delete links simultaneously.

Definition 3.3. *A social storage network* $\mathfrak{g}$ *is bilaterally stable if and only if*

1. *for all* $\langle ij \rangle \in \mathfrak{g}$*, if* $u_i(\mathfrak{g} - \langle ij \rangle) > u_i(\mathfrak{g})$*, then* $u_j(\mathfrak{g} - \langle ij \rangle) < u_j(\mathfrak{g})$*, and*
2. *for all* $\langle ij \rangle \notin \mathfrak{g}$*, if* $u_i(\mathfrak{g} + \langle ij \rangle) > u_i(\mathfrak{g})$*, then* $u_j(\mathfrak{g} + \langle ij \rangle) < u_j(\mathfrak{g})$.

Definition 3.3 is a network stability concept, whose first part states that no pair of agents with a link between them, wants to delete the link, and the second part states that no pair of agents has an incentive to add a new link. Note that neither link formation (addition) nor link deletion can happen without mutual consent. Our further discussions about social storage stability stands on Definition 3.3.

Remark 3.3. *Definition 3.3 can be rewritten using only conditions on the addition of links by rewriting the first condition (that is, the deletion condition) as a condition for addition in* $\mathfrak{g}^{\mathfrak{c}}$*. Similarly, we can also rewrite Definition 3.3 using only deletion conditions.*

Now, we generalize Definition 3.3 so that it is suitable as a solution concept for the two frameworks discussed in the previous section.

For this, we first define *remaining storage* available with agent i in a network $\mathfrak{g}$ as

$$RS_i = \mathbf{s}_i - \sum_{j \in \eta_i(\mathfrak{g})} \mathbf{d}_j a_{ij}, \tag{3.3}$$

and *remaining budget* of agent i in $\mathfrak{g}$ as

$$RB_i = \mathbf{b}_i - \sum_{j \in \eta_i(\mathfrak{g})} \varsigma a_{ij}, \tag{3.4}$$

where

$$a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ have a backup agreement,} \\ 0 & \text{otherwise.} \end{cases}$$

For the MO-Framework, where we have storage constraints, the following modification of Definition 3.3 is appropriate.

Definition 3.4. *A social storage network* $\mathfrak{g}$ *with storage constraints is bilaterally stable if and only if*

1. *for all* $\langle ij \rangle \in \mathfrak{g}$, *if* $u_i(\mathfrak{g} - \langle ij \rangle) > u_i(\mathfrak{g})$, *then* $u_j(\mathfrak{g} - \langle ij \rangle) < u_j(\mathfrak{g})$, *and*

2. *for all* $\langle ij \rangle \notin \mathfrak{g}$, *if* $[u_i(\mathfrak{g} + \langle ij \rangle) > u_i(\mathfrak{g})$ *and* $RS_j \geq \mathbf{d}_i]$, *then*

$$[u_j(\mathfrak{g} + \langle ij \rangle) < u_j(\mathfrak{g}) \text{ or } RS_i < \mathbf{d}_j].$$

In the above definition, there is no change in the link deletion condition of Definition 3.3. However, while adding a link, an agent has to ensure that the other agent has sufficient storage to store its data (besides ensuring increase in its utility). We assume that the agents are rational and self-centred (and hence, it is up to agent j to check whether agent i has sufficient storage for agent j' s data or not).

Next, we adapt Definition 3.3 for the SO-Framework, where we have storage and budget constraints.

Definition 3.5. *A social storage network* $\mathfrak{g}$ *with storage and budget constraints is bilaterally stable if and only if*

1. *for all* $\langle ij \rangle \in \mathfrak{g}$, *if* $u_i(\mathfrak{g} - \langle ij \rangle) > u_i(\mathfrak{g})$, *then* $u_j(\mathfrak{g} - \langle ij \rangle) < u_j(\mathfrak{g})$, *and*

2. *for all* $\langle ij \rangle \notin \mathfrak{g}$, *if* $[u_i(\mathfrak{g} + \langle ij \rangle) > u_i(\mathfrak{g})$ *and* $RS_j \geq \mathbf{d}_i$ *and* $RB_i \geq \varsigma)]$, *then*

$$[u_j(\mathfrak{g} + \langle ij \rangle) < u_j(\mathfrak{g}) \text{ or } RS_i < \mathbf{d}_j \text{ or } RB_j < \varsigma].$$

As in the case of MO-Framework, there is no change in the link deletion condition of Definition 3.3. However, while adding a link, an agent has to ensure that the other agent has sufficient storage to store its data and agent itself has sufficient budget to form the link (besides ensuring increase in its utility). This is, again, based on the assumption that the agents are rational and self-centred.

We, now, define efficient and contented social storage networks, with as well as without constraints. Efficient social storage networks are social storage networks where as many agents as possible achieve maximum utility, whereas contented social storage networks are those where all agents achieve maximum utility.

Definition 3.6. *A social storage network* $\mathfrak{g}$ *is efficient with respect to utility profile* $(u_1, ..., u_N)$ *if* $\sum_i u_i(\mathfrak{g}) \geq \sum_i u_i(\mathfrak{g}')$, *for all* $\mathfrak{g}' \in \mathscr{G}(N)$.

Definition 3.7. *A social storage network* $\mathfrak{g}$ *with storage constraints is efficient with respect to utility profile* $(u_1,...,u_N)$ *if* $\sum_i u_i(\mathfrak{g}) \geq \sum_i u_i(\mathfrak{g}')$*, for all* $\mathfrak{g}' \in \mathscr{G}(N)$ *where* $RS_i \geq 0$ *for all* $i \in \mathfrak{g}'$ *.*

Definition 3.8. *A social storage network* $\mathfrak{g}$ *with storage and budget constraints is efficient with respect to utility profile* $(u_1,...,u_N)$ *if* $\sum_i u_i(\mathfrak{g}) \geq \sum_i u_i(\mathfrak{g}')$*, for all* $\mathfrak{g}' \in \mathscr{G}(N)$ *where* $RS_i \geq 0$ *and* $RB_i \geq 0$*, for all* $i \in \mathfrak{g}'$ *.*

Definition 3.9. *A social storage network* $\mathfrak{g}$ *is contented with respect to utility profile* $(u_1,...,u_N)$ *if, for each* $i \in \mathscr{A}$*,* $u_i = \max_{\eta_i(\mathfrak{g})} \{\beta_i(1-\lambda^{\eta_i(\mathfrak{g})}) - c\eta_i(\mathfrak{g})\}$*, under the MO-Framework, and* $u_i = \max_{\eta_i(\mathfrak{g})} \{\beta_i(1-\lambda^{\eta_i(\mathfrak{g})})\}$*, under the SO-Framework.*

Remark 3.4. *If maximum possible utility is not achievable by a one or more agents because of storage or budget constraints, then those agents are not contented, and hence, the social storage network is not contented. Therefore, we do not define contentedness with constraints.*

3.2 Social Storage Cloud Model

In this section, we describe the social storage cloud model through an interaction structure, a storage-sharing framework and cost-benefit analysis of agents. In the context of social storage networks discussed in the previous section, bilateral stability is the appropriate solution concept because neighbours mutually store their data with each other. However, in the social storage cloud model, agents may share their storage space and save their data with any agent connected directly or indirectly. Therefore, mutual consent for deletion is not realistic and hence, pairwise stability is the appropriate solution concept to look at.

3.2.1 Interaction Structure

A social storage cloud $\mathfrak{g} = (\mathscr{A}, \mathscr{L})$ is a storage-sharing and data backup network that consists of a non-empty set $\mathscr{A}$ of N agents who are involved in storage (disk)-space sharing and data backup activity; and a set, $\mathscr{L}$, of links that connect these agents. The set $\mathscr{L}$ acts as a communication infrastructure for agents to share their storage space with others and search for storage space provided by others. A link, $\langle ij \rangle \in \mathscr{L}$, represents a direct communication channel between agents i and j, which is bidirectional (and hence, $\langle ij \rangle = \langle ji \rangle$). If $\langle ij \rangle \in \mathscr{L}$, we call the agents i and j as neighbours in the network $\mathfrak{g}$. The number of neighbours of agent i in $\mathfrak{g}$ is denoted by $\eta_i(\mathfrak{g})$.

Data stored on local storage space is prone to loss due to multiple reasons such as virus infection, software or hardware failure, data corruption, and so on. Therefore, each agent wants to backup its data on remote storage (disk) space. For any agent, data loss is costly. We capture this by assuming that the value each agent associates with its data is quantifiable and given. Every agent (as a data owner) strives for obtaining storage space provided by other agents (as storage providers) in $\mathfrak{g} \in \mathscr{G}(N)$. Agent i wants to backup $\bar{b}_i$ amount of data and shares $\bar{s}_i = \sum_{j \in \mathscr{A} \setminus \{i\}} \bar{b}_j$ amount of storage space. This leads to endogenous social storage cloud formation, where each agent builds its communication channel to seek storage space from direct and indirect communication channels. We assume that each agent has global (complete) information about the network structure.

3.2.2 Link Formation Rules and Network Formation

A network $\mathfrak{g}$ evolves when agents perform two actions, namely, link addition ($\mathfrak{g} + \langle ij \rangle$) and link deletion ($\mathfrak{g} - \langle ij \rangle$). *Mutual consent of a pair of agents is required for addition of a link between them, but any link can be unilaterally deleted.*

3.2.3 Storage Sharing

According to [4], agents could limit storage-sharing with those who are close to them in the social cloud. In order to capture this, we make use of the harmonic closeness[21] (discussed in [89, 90, 91]), defined as follows:

$$\Phi_i(\mathfrak{g}) = \sum_{j \in \mathfrak{g} \setminus \{i\}} \frac{1}{d_{ij}(\mathfrak{g})}, \tag{3.5}$$

[21]We overview a few centrality measures in Appendix C.2.

whereas, $d_{ij}(\mathfrak{g})$ is the shortest distance (length of the shortest path) between agents i and j in $\mathfrak{g}$.

The harmonic closeness (centrality) of an agent is the sum of the inverse of its shortest distance with other agents in the network. We use harmonic centrality as it deals with disconnected networks as well.

In $\mathfrak{g}$, an agent j (as a storage provider) computes a probability distribution on all agents for the purpose of allocating storage space to agent $i \in \mathfrak{g}$ (as a data owner), as below:

$$\alpha_{ij}(\mathfrak{g}) = \frac{\frac{1}{d_{ij}(\mathfrak{g})}}{\sum\limits_{j \in \mathfrak{g} \setminus \{i\}} \frac{1}{d_{ij}(\mathfrak{g})}} = \frac{1}{d_{ij}(\mathfrak{g}) \Phi_i(\mathfrak{g})}, \tag{3.6}$$

where $\alpha_{ij}(\mathfrak{g})$ is the probability that agent i will obtain storage space from agent j in $\mathfrak{g}$.

Remark 3.5. *If $d_{ij}(\mathfrak{g}) = \infty$, then $\alpha_{ij}(\mathfrak{g}) = 0$ (and $\alpha_{ji}(\mathfrak{g}) = 0$). As agents i and j are disconnected in $\mathfrak{g}$, their chances of obtaining storage space from each other is zero.*

The probability that an agent i obtains storage space from at least one agent in $\mathfrak{g}$ is

$$\gamma_i(\mathfrak{g}) = 1 - \prod_{j \in \mathfrak{g} \setminus \{i\}} (1 - \alpha_{ij}(\mathfrak{g})). \tag{3.7}$$

3.2.4 Agent' s Utility and Symmetry

We define the utility of agents in a social storage cloud $\mathfrak{g}$ with the given below.For agent i, β_i is the value of the local data that is to be backed up. An agent i loses its data with probability $\delta_i \in (0,1)$. Agent i obtains storage space provided by others in $\mathfrak{g}$, with probability $\gamma_i(\mathfrak{g})$. These three aspects capture the expected benefit of agent i in $\mathfrak{g}$.

An agent searches for storage by staying connected in the network. Direct as well as indirect links help agents to get storage space. The direct link between agents i and j costs ς_i. This cost can be interpreted as the cost required for maintaining storage space, infrastructure, bandwidth, time, and so on. The cost to maintain an existing link and that for adding (and maintaining) a new link are the same. There is no *additional* cost to add a new link. Thus, agent i incurs a total cost of $\varsigma_i \eta_i(\mathfrak{g})$ in order to obtain an expected benefit of $\beta_i \delta_i \gamma_i(\mathfrak{g})$, in case of data loss. But the network is formed upfront, before the data loss happens. The cost to maintain links is, hence, incurred even in the case of no data loss, where the expected benefit to i is $\beta_i(1-\delta_i)$.

Therefore, given the aforementioned parameters, the expected utility is

$$u_i(\mathfrak{g}) = \beta_i(1-\delta_i)+\beta_i\delta_i\gamma_i(\mathfrak{g}) - \varsigma_i\eta_i(\mathfrak{g}). \tag{3.8}$$

Equation 3.8 is a form of degree-distance based utility, where an agent's benefit decreases with an increase in the number of neighbours of other agents.

Due to similar reasons as discussed earlier, we define a symmetric social storage cloud $\mathfrak{g}$ as follows.

Definition 3.10. *A symmetric social storage cloud (SSSC) $\mathfrak{g}$ is a network where the benefit (value) associated with backed-up data is the same for all agents in the network, that is, $\beta_i = \beta_j$ (say β), $\varsigma_i = \varsigma_j$ (say ς[22]), and $\delta_i = \delta_j$ (say δ[23]) for all $i, j \in \mathscr{A}$, and hence, utility of each agent i in $\mathfrak{g}$ is*

$$u_i(\mathfrak{g}) = \beta(1-\delta)+\beta\delta\gamma_i(\mathfrak{g}) - \varsigma\eta_i(\mathfrak{g}), \tag{3.9}$$

where $\beta, \delta, \varsigma \in (0,1)$.

For further study, we consider the above utility function (Equation (3.9)). Henceforth, whenever we refer to a network, or just $\mathfrak{g}$, we mean an SSSC.

3.3 Social Compute Cloud: The Model

In this section, we describe the social compute cloud model, which is similar to the social storage cloud model. We discuss this model separately as it is one of the extensions of our social storage models to social cloud networks where any computational resource may be shared among agents, directly or indirectly connected.

Definition 3.11. *A social compute cloud (SCC) model $\mathfrak{g} = \{\mathscr{A}, \mathscr{L}\}$ consists of a non-empty set $\mathscr{A}$ of N agents and a set $\mathscr{L}$ of ℓ undirected links, connecting these agents, where a link between two agents represents a communication channel between them.*

[22]We assume, $\varsigma = \frac{\varsigma_i+\varsigma_j}{2}$, that is, a pair of agents involved in a link share the cost ς.
[23]For simplicity, we assume uniform data loss rate δ.

In a social compute cloud $\mathfrak{g}$, the set $\mathscr{L}$ can be viewed as a communication infrastructure that facilitates agents to share their computing resources (such as disk space and computing power, workflow, etc.), with others, and search for resources shared by others. In $\mathfrak{g}$, agents provide resource to other agents who are directly or indirectly connected with them.

3.3.1 Assumptions

This model stands on the following basic assumptions.

Assumption 3.1. *In an SCC $\mathfrak{g}$, each agent shares a single type of resource of one unit.*

Assumption 3.2. *In a prevailing SCC $\mathfrak{g}$, an agent has a resource with probability p and needs to perform a computational task with probability q.*

The prevailing resource sharing situation in $\mathfrak{g}$ is determined by resource provision and requirement rate. We capture the resource availability in $\mathfrak{g}$ with rate p. Another aspect is that agent i wants to accomplish a computational task τ_i, for example, an agent may need to backup its data. We capture the task performing rate with the parameter q.

Assumption 3.3. *A agent plays two roles; a resource provider with the probability $p(1-q)$ and a resource consumer with the probability $q(1-p)$.*

Assumption 3.4. *Each agent has global information, that is, each agent is aware of the network structure $\mathfrak{g}$ and the prevailing resource sharing situation in $\mathfrak{g}$.*

3.3.2 Closeness-Based Resource Sharing

Similar to SSC, here, agents perform closeness based resource sharing. The closeness of agents is captured by harmonic closeness here also.

In an SCC $\mathfrak{g}$, an agent $j \in \mathfrak{g}$ (who acts as a resource provider) computes a probability distribution on all agents for the purpose of allocating storage resource to agent $i \in \mathfrak{g}$ (who acts as a resource consumer), which is as below:

$$\alpha_{ij}(\mathfrak{g}) = p(1-q)\frac{\frac{1}{d_{ij}(\mathfrak{g})}}{\sum\limits_{j\in\mathfrak{g}\setminus\{i\}}\frac{1}{d_{ij}(\mathfrak{g})}} = \frac{p(1-q)}{d_{ij}(\mathfrak{g})\Phi_i(\mathfrak{g})}. \tag{3.10}$$

In other words, $\alpha_{ij}(\mathfrak{g})$ is the probability that agent i will obtain storage space from agent j in $\mathfrak{g}$.

Remark 3.6. *If $d_{ij}(\mathfrak{g}) = \infty$ then $\alpha_{ij}(\mathfrak{g}) = 0$ ($= \alpha_{ji}(\mathfrak{g})$). As agents i and j are disconnected in $\mathfrak{g}$ their chances of obtaining storage space from each other is nil.*

The probability that agent i obtain resource from at least one agent in $\mathfrak{g}$ is given as below:

$$\gamma_i(\mathfrak{g}) = 1 - \prod_{j \in \mathfrak{g} \setminus \{i\}} (1 - \alpha_{ij}(\mathfrak{g})). \tag{3.11}$$

3.3.3 Utility Structure

In an SCC $\mathfrak{g}$, an agent i gains benefit θ_i and ξ_i by accomplishing a computational task τ_i and by providing a resource to others, respectively. An agent $i \in \mathfrak{g}$ gains ξ_i with the probability $p(1-q)$. An agent i's expected benefit θ_i depends on whether the agent has its own resource or depend on resource availability in $\mathfrak{g}$. Note that, with the probability pq the agent is self-reliant, therefore, do not depends on $\mathfrak{g}$. However, the agent depends on others for resource availability and seeks resource from others in $\mathfrak{g}$ with the probability $q(1-p)$.

An agent searches resources in $\mathfrak{g}$ by maintaining direct links. Each agent i pays cost ς_i for each direct link in $\mathfrak{g}$. Thus, i incurs total cost $\eta_i(\mathfrak{g})$ times ς_i in $\mathfrak{g}$. The cost ς_i can be interpreted as the efforts or time that agent i spent to maintain active connection (or link). However, we consider agents i and j share the link formation cost such that, $\varsigma = \frac{\varsigma_i + \varsigma_j}{2}$.

Then for a given resource sharing network $\mathfrak{g}$, the expected payoff $u_i(\mathfrak{g})$ of agent i is given as:

$$u_i(\mathfrak{g}) = \underbrace{p(1-q)}_{\text{provider}}\xi_i + \overbrace{pq}^{\text{self-reliant}}\theta_i + \underbrace{q(1-p)}_{\text{consumer}}\gamma_i(\mathfrak{g})\theta_i - \overbrace{\varsigma\eta_i(\mathfrak{g})}^{\text{total cost}}.$$

$$u_i(\mathfrak{g}) = p(1-q)\xi_i + q[p + (1-p)\gamma_i(\mathfrak{g})]\theta_i - \varsigma\eta_i(\mathfrak{g}). \tag{3.12}$$

Definition 3.12. *A symmetric social compute cloud (SSCC) $\mathfrak{g}$ is a social compute cloud where the link formation cost, and the benefits that associated with accomplishing a computational task and providing resource to others associated are the same for all agents in* *the network, i.e., $\varsigma_i = \varsigma_j$ (say ς), $\theta_i = \theta_j$ (say θ) and $\xi_i = \xi_j$ (say ξ) for all $i, j \in \mathscr{A}$, and hence, utility of each agent i in $\mathfrak{g}$ is*

$$u_i(\mathfrak{g}) = p(1-q)\xi + q[p + (1-p)\gamma_i(\mathfrak{g})]\theta_i - \varsigma\eta_i(\mathfrak{g}). \tag{3.13}$$

3.4 Chapter Summary

In this chapter, we have presented three models of social cloud formation. In the subsequent chapters we will discuss the stability, efficiency and resource availability with respect to these models.

Chapter 4

Social Storage Networks: Stability, Efficiency and Contentedness

This chapter deals with two facets of social storage. *First*, as stated in Chapter 1, major social storage studies (such as [16, 24, 92]) have considered exogenous social networks (an underlying social network, for instance Facebook, Orkut, Venus, and so on) to construct a social storage system and to study QoS related issues.

The approach of considering an exogenous social network to build a social storage system (or to do QoS analysis) stands on the assumption that, an agent in the underlying network is involved in data backup activity with all its .

However, this approach neglects the possibility that agents do not want to perform a data backup activity with their set of existing neighbours (in the underlying network). In other words, this approach does not focus on the preferences of agents over selecting their backup partners. Such preferences may be derived by the cost and benefit that these agent experience in selecting their backup partners. It also not consider a rational (or self-interested) behaviour of agents involved in the data backup activity is not taken into consideration[24].

Therefore, the QoS analysis, which is based upon the neighbourhood size in the underlying network, is no longer valid. Thus, it is important to study when agents want to perform a data backup activity and (or) when they do not. Hence, in this chapter, we model the social storage system as an *endogenous network formation game*.

[24]Although Sharma and others [23] begin discussing about agents' strategic behaviour in a scenario where limited storage is available for the agents, this has just been touched upon and has not been looked at in detail.

Second, social storage systems may not be stable (when agents have no incentive to add new partners or delete existing partners). Even if stable, they may not be efficient (maximizing the sum of utilities of all agents), and even if efficient, they may not be contended (when all agents achieve their maximum utility). There is limited study on stability and efficiency of social storage systems. While proposing the idea of F2F backup systems, [15] argue that social ties between agents act as incentives for them to stay in the system, thereby resulting in a stable social storage system. In their context, a system is unstable when agents arrive and depart the social storage system randomly — lesser this randomness, more the stability of the system. In our case, a social storage system, as above, is *stable* when agents have no incentive to add new partners or delete existing partners. In the following subsections, we motivate this definition of stability in detail.

In this chapter, we consider two frameworks for utility of agents in the social storage network. We modify the pairwise stability definition of [31] to include mutual consent for link deletion too (as required for social storage networks), and also to include storage and budget constraints. After defining bilateral stability as a modification of pairwise stability, we analyse bilateral stability of symmetric social storage networks. Our stability analysis involved restudying conditions of stability under the new definition of pairwise stability (that is, bilateral stability), derivation of a unique stability point (which is a neighbourhood size where no agent has any incentive to add or delete a link), and some necessary and sufficient conditions for symmetric social storage networks to be bilaterally stable. We also show that ideally all agents in a network want to achieve their stability point but a network can be bilaterally stable even when this stability point is not reached for one agent.

Further, we discuss which bilaterally stable networks would evolve. We also discuss why just studying stability is not enough and one has to look at efficiency and contentment of the network. Efficiency is the case when the sum of utilities of all agents is maximized, and contentment is when the individual utility of every agent is maximized. We relate these three properties of the network with one another. We also give conditions on the number of agents and stability point (besides other constraints) to achieve bilaterally stable, efficient, and contented networks.

4.1 Stable Network Characterization and Stability Point

In this section, we study the bilateral stability aspects of social storage networks considering the utilities of agents and the solutions concept as defined in Section 3.1.3.

Free riding is a situation where an agent offers less storage space, but consumes more. To deal with free riding, many backup systems have used the concept of symmetric resource sharing (or equal resource trading). Internet Cooperative Backup System [44], PeerStore [42], Pastiche [93], are a few examples of P2P backup systems, which use symmetric resource trading to mitigate free riding.

We term a social storage system with symmetric resource sharing as a symmetric social storage system. We consider symmetry in the agents' value of their respective data, storage space available, amount of data to be shared, and budget in two different scenarios. These scenarios are discussed next.

Definition 4.1. *A symmetric value network (SVN)* $\mathfrak{g}$ *is a social storage network where the benefit (value) associated with backed-up data is the same for all agents in the network, i.e.,* $\beta_i = \beta_j$ *(say* β*), for all* $i, j \in \mathscr{A}$*, and hence, utility of each agent* i *in the network is*

$$\begin{aligned} u_i(\mathfrak{g}) &= \beta(1-\lambda^{\eta_i(\mathfrak{g})}) - \varsigma\eta_i(\mathfrak{g}) \text{ for MO-Framework 3.1.3.1 and,} \\ u_i(\mathfrak{g}) &= \beta(1-\lambda^{\eta_i(\mathfrak{g})}) \text{ for SO-Framework 3.1.3.2,} \end{aligned} \tag{4.1}$$

$$\text{where } \beta, \lambda, \varsigma \in (0,1).$$

□

Definition 4.2. *A symmetric resource network (SRN)* $\mathfrak{g}$ *is a social storage network where all agents in* $\mathfrak{g}$ *have an equal amount of (limited) storage space available to them, an equal amount of data that they want to backup, and have the same budget. That is, for all* $i, j \in \mathfrak{g}$*,* $\mathbf{s}_i = \mathbf{s}_j$ *(say* $\mathbf{s}$*),* $\mathbf{d}_i = \mathbf{d}_j$ *(say* $\mathbf{d}$*), and* $\mathbf{b}_i = \mathbf{b}_j$ *(say* $\mathbf{b}$*).*

Remark 4.1. *From this symmetric setup, we can move to real life scenarios in many ways. We can have different value of cost and benefit for different agents. Another way to include* *heterogeneity in this model is by using the concept of Social Range Matrix [94], which we have done recently [95]. Here, each agent is concerned about its perceived utility, which is a linear combination of its utility as well as others utilities (depending upon whether the pair are friends, enemies or do not care about each other).*

Now, we work with SVN under the MO-Framework, where each agent in the given network $\mathfrak{g}$ has as much storage as is required for all other agents in $\mathfrak{g}$. That is,

$$\mathbf{s}_i \geq \sum_{\substack{j \in \mathfrak{g}, \\ j \neq i}} \mathbf{d}_j, \quad \text{for all } i \in \mathfrak{g}. \tag{4.2}$$

Note that $\mathbf{s}_i$ may be different from some other $\mathbf{s}_j$. For convenience, we shall call such a network as SVN with sufficient storage. The reason we do this is that it leads to the results of the realistic scenario, that is, SV-SRN under the MO-Framework.

Remark 4.2. *An SV-SRN, $\mathfrak{g}$ under the MO-Framework is a social storage network where the utility of each agent $i \in \mathfrak{g}$ is $u_i(\mathfrak{g}) = \beta_i(1 - \lambda^{\eta_i(\mathfrak{g})}) - \varsigma\eta_i(\mathfrak{g})$, and for all agents $i, j \in \mathfrak{g}$, $\beta_i = \beta_j, \mathbf{s}_i = \mathbf{s}_j$, and $\mathbf{d}_i = \mathbf{d}_j$.*

Next, we work with SVN under the SO-Framework where each agent in the given network $\mathfrak{g}$ has as much storage as is required for all other agents in $\mathfrak{g}$, and each agent in the given network $\mathfrak{g}$ has as much budget as is required to maintain backup-partnerships with every other agent in $\mathfrak{g}$. That is,

$$\mathbf{s}_i \geq \sum_{j \in \mathscr{A}, j \neq i} \mathbf{d}_j, \quad \text{for all } i \in \mathfrak{g}, \quad \text{and,} \quad \mathbf{b}_i \geq \varsigma(N-1), \quad \text{for all } i \in \mathfrak{g} \quad \text{where } N = |\mathscr{A}|. \tag{4.3}$$

As in the SO-Framework, this leads to the scenario of SV-SRN under the SO-Framework. However, for SO-Framework, we present the results for SRN directly rather than SV-SRN. This is because SV-SRN is a subset of SRN and so, what holds for SRN does for SV-SRN as well.

Remark 4.3. *An SV-SRN, $\mathfrak{g}$ under the SO-Framework is a social storage network where the utility of each agent $i \in \mathfrak{g}$ is $u_i(\mathfrak{g}) = \beta_i(1 - \lambda^{\eta_i(\mathfrak{g})})$, and for all agents $i, j \in \mathfrak{g}$, $\beta_i = \beta_j, \mathbf{s}_i = \mathbf{s}_j, \mathbf{d}_i = \mathbf{d}_j$, and $\mathbf{b}_i = \mathbf{b}_j$.*

For ease of exposition, from now onward, whenever we discuss SVN networks, we will always assume sufficiency of every resource — that is, sufficient storage under MO-Framework, and sufficient storage and budget under SO-Framework. Whenever we discuss SRN or SV-SRN networks, we will not make these assumptions of sufficiency. These are summarized in Table 4.1.

Network Type	**Framework**	**Resource Availability**
SVN	MO-Framework	Sufficient Storage.
SV-SRN	MO-Framework	Limited Storage and Limited Budget.
SVN	SO-Framework	Sufficient Storage and Sufficient Budget.
SRN	SO-Framework	Limited Storage and Limited Budget.

Table 4.1: Summary of network study under different frameworks with/ without sufficient resources

In the following subsections, we characterize bilaterally stable symmetric social storage networks, by first deriving the deviation conditions — conditions for an agent to have an incentive to add or delete a link, given the network parameters (that is, disk failure rate λ, value of backup data β, and the cost of maintaining a link ς). This also gives us necessary and sufficient conditions for bilateral stability, in terms of the network parameters (λ, β, and ς). Further, this makes it easier to visualize a bilaterally stable network, and we use these conditions to derive the ideal neighbourhood size for having a bilaterally stable network. We term this ideal neighbourhood size as the *stability point* (see Definition 4.3)[25]).

Definition 4.3. *Given a network* $\mathfrak{g}$*, we define the stability point* $\hat{\eta}$ *of* $\mathfrak{g}$ *as the neighbourhood size (degree) such that no agent in* $\mathfrak{g}$ *has any incentive to increase its neighbourhood size to more than* $\hat{\eta}$ *and to decrease it to less than* $\hat{\eta}$*.*

We, now, characterize SVN and SV-SRN under the MO-Framework, and SVN and SRN under the SO-Framework. Further, we prove uniqueness of the *stability point* of these networks and also show that the stability point is independent of the number of agents for

[25]Note that, Definition 4.3 is a result of Theorems 4.3, 4.4 and 4.7, which we have discussed in the next section.

all cases under the MO-Framework and for all cases but one trivial case under the SO-Framework, the trivial case being SVN with sufficient storage and sufficient budget where it is easy to see that the complete network is the only stable network.

4.1.1 Characterization Under the MO-Framework

In this subsection, we characterize bilaterally stable SVN and SV-SRN under the MO-Framework. We, first, derive conditions under which an agent has an incentive to add a new link or delete an existing link. Then, we derive necessary and sufficient conditions for bilateral stability of SVN and SV-SRN under the MO-Framework, and prove that the stability point of these networks is unique and independent of the number of agents.

Lemma 4.1. *In an SVN $\mathfrak{g}$, under the MO-Framework, for any agent $i \in \mathfrak{g}$, forming a partnership with another agent $j \in \mathfrak{g}$ is beneficial if and only if $\varsigma < \beta\lambda^{\eta_i(\mathfrak{g})}[1-\lambda]$.*

Proof. As $\mathfrak{g}$ is an SVN under the MO-Framework,
$u_i(\mathfrak{g}) = \beta(1-\lambda^{\eta_i(\mathfrak{g})}) - \varsigma\eta_i(\mathfrak{g})$, for all $i \in \mathfrak{g}$.

For $i, j \in \mathfrak{g}$, if $\langle ij\rangle \notin \mathfrak{g}$, then
$u_i(\mathfrak{g}+\langle ij\rangle) = [\beta(1-\lambda^{\eta_i(\mathfrak{g})+1})] - [\varsigma(\eta_i(\mathfrak{g})+1)]$.

Adding a new link or backup partner is beneficial for i if and only if
$u_i(\mathfrak{g}+\langle ij\rangle) > u_i(\mathfrak{g})$, if and only if
$[\beta(1-\lambda^{\eta_i(\mathfrak{g})+1}) - \varsigma(\eta_i(\mathfrak{g})+1)] > [\beta(1-\lambda^{\eta_i(\mathfrak{g})}) - \varsigma(\eta_i(\mathfrak{g}))]$, if and only if
$\varsigma < \beta[\lambda^{\eta_i(\mathfrak{g})} - \lambda^{\eta_i(\mathfrak{g})+1}]$. □

Remark 4.4. *The term on the left-hand side of the inequality in Lemma 4.1 is the cost ς that agent i incurs in order to add a new neighbour j. The term on the right-hand side is the expected benefit that agent i receives by forming a new link with neighbour j.*

Lemma 4.2. *In an SVN $\mathfrak{g}$, under the MO-Framework, for any agent $i \in \mathfrak{g}$, breaking an existing partnership with another agent $j \in \mathfrak{g}$ is beneficial if and only if*
$\varsigma > \beta\lambda^{\eta_i(\mathfrak{g})-1}[1-\lambda]$.

Proof. As $\mathfrak{g}$ is an SVN, under the MO-Framework, $u_i(\mathfrak{g}) = \beta(1-\lambda^{\eta_i(\mathfrak{g})}) - \varsigma\eta_i(\mathfrak{g})$, for all $i \in \mathfrak{g}$.

If $\langle ij \rangle \in \mathfrak{g}$, then $u_i(\mathfrak{g} - \langle ij \rangle) = [\beta(1-\lambda^{\eta_i(\mathfrak{g})-1})] - [\varsigma(\eta_i(\mathfrak{g})-1)]$.

Deleting an existing link is beneficial for any agent i if and only if

$u_i(\mathfrak{g} - \langle ij \rangle) > u_i(\mathfrak{g}))$, if and only if

$[\beta(1-\lambda^{\eta_i(\mathfrak{g})-1}) - \varsigma(\eta_i(\mathfrak{g})-1)] > [\beta(1-\lambda^{\eta_i(\mathfrak{g})}) - \varsigma(\eta_i(\mathfrak{g}))]$, if and only if

$\varsigma > \beta[\lambda^{\eta_i(\mathfrak{g})-1} - \lambda^{\eta_i(\mathfrak{g})}]$. □

Remark 4.5. *We interpret Lemma 4.2 in lines similar to Remark 4.4.*

Theorem 4.1. *An SVN $\mathfrak{g}$, under the MO-Framework, is bilaterally stable if and only if*

1. *for all $\langle ij \rangle \in \mathfrak{g}$, if $\beta\lambda^{\eta_i(\mathfrak{g})-1}[1-\lambda] < \varsigma$, then $\beta\lambda^{\eta_j(\mathfrak{g})-1}[1-\lambda] > \varsigma$, and*

2. *for all $\langle ij \rangle \notin \mathfrak{g}$, if $\beta\lambda^{\eta_i(\mathfrak{g})}[1-\lambda] > \varsigma$, then $\beta\lambda^{\eta_j(\mathfrak{g})}[1-\lambda] < \varsigma$.*

Proof. Follows from Lemma 4.1, Lemma 4.2 and Definition 3.3 of bilateral stability. □

We state and prove the following for SV-SRN, under the MO-Framework.

Lemma 4.3. *Let $\mathfrak{g}$ be an SV-SRN, under the MO-Framework. For any agent $i \in \mathfrak{g}$, adding a new partnership with agent $j \in \mathfrak{g}$ is beneficial if and only if*
$(\varsigma < \beta\lambda^{\eta_i(\mathfrak{g})}[1-\lambda] \quad \text{and} \quad \mathbf{s} - \mathbf{d}\eta_j(\mathfrak{g}) \geq \mathbf{d})$,
and breaking an existing partnership with agent $j \in \mathfrak{g}$ is beneficial if and only if
$\varsigma > \beta\lambda^{\eta_i(\mathfrak{g})-1}[1-\lambda]$.

Proof. If $\langle ij \rangle \notin \mathfrak{g}$, agent i has an incentive to add a link with agent j, if and only if

$\varsigma < \beta[\lambda^{\eta_i(\mathfrak{g})} - \lambda^{\eta_i(\mathfrak{g})+1}]$ (from Lemma 4.1), where $\eta_i(\mathfrak{g})$ = neighbourhood size of i, and the amount of storage available with agent $j \geq$ agent i's data size, if and only if

$\varsigma < \beta[\lambda^{\eta_i(\mathfrak{g})} - \lambda^{\eta_i(\mathfrak{g})+1}]$ and $s_j - \sum_{k \in \eta_j(\mathfrak{g})} d_k \geq d_i$, where $\eta_j(\mathfrak{g})$ is the set of neighbours of j, if and only if

$\varsigma < \beta[\lambda^{\eta_i(\mathfrak{g})} - \lambda^{\eta_i(\mathfrak{g})+1}]$ and $s - d\eta_j(\mathfrak{g}) \geq d$, (as $s_k = s_l, d_k = d_l$, for all $k, l \in \mathfrak{g}$),

where $\eta_j(\mathfrak{g})$ is the neighbourhood size of j.

To delete an existing link, agent i only looks at the cost for maintaining the link, and hence, from Lemma 4.2, agent i has an incentive to delete a link if and only if $\varsigma > \beta[\lambda^{\eta_i(\mathfrak{g})-1} - \lambda^{\eta_i(\mathfrak{g})}]$. □

Theorem 4.2. *An SV-SRN* $\mathfrak{g}$*, under the MO-Framework, is bilaterally stable if and only if*

1. for all $\langle ij \rangle \in \mathfrak{g}$, $\beta\lambda^{\eta_i(\mathfrak{g})-1}[1-\lambda] < \varsigma \Rightarrow \beta\lambda^{\eta_j(\mathfrak{g})-1}[1-\lambda] > \varsigma$, and

2. for all $\langle ij \rangle \notin \mathfrak{g}$, $\beta\lambda^{\eta_i(\mathfrak{g})}[1-\lambda] > \varsigma$ and $s - d\eta_j(\mathfrak{g}) \geq d \Rightarrow$ $\beta\lambda^{\eta_j(\mathfrak{g})}[1-\lambda] < \varsigma$ or $s - d\eta_i(\mathfrak{g}) < d$.

Proof. Follows from Lemma 4.3, and Definition 3.4 of bilateral stability. □

Now, we look at the stability point of SVN and SV-SRN under the MO-Framework.

Theorem 4.3. *Let* $\mathfrak{g}$ *be an SVN under the MO-Framework. Then, the stability point* $\hat{\eta}$ *of* $\mathfrak{g}$ *is unique and is given by* $\hat{\eta} = \left\lceil \frac{|\ln(\frac{\varsigma}{\beta(1-\lambda)})|}{|\ln\lambda|} \right\rceil = \left\lfloor \frac{|(\ln\frac{\varsigma\lambda}{\beta(1-\lambda)})|}{|\ln\lambda|} \right\rfloor$.

Proof. From Lemma 4.1, adding a link for agent i is beneficial if and only if

$\eta_i(\mathfrak{g}) \ln\lambda > \ln(\frac{\varsigma}{\beta(1-\lambda)})$, if and only if

$\eta_i(\mathfrak{g}) < \frac{|\ln(\frac{\varsigma}{\beta(1-\lambda)})|}{|\ln\lambda|}$

Hence, for agent i, increasing neighbourhood size is not beneficial if and only if

$\eta_i(\mathfrak{g}) \geq \frac{|\ln(\frac{\varsigma}{\beta(1-\lambda)})|}{|\ln\lambda|}$.

Similarly, from Lemma 4.2, deleting a link for agent i is beneficial if and only if

$\ln(\frac{\varsigma\lambda}{\beta(1-\lambda)}) > \eta_i(\mathfrak{g}) \ln\lambda$, if and only if

$\frac{|\ln(\frac{\varsigma\lambda}{\beta(1-\lambda)})|}{|\ln\lambda|} < \eta_i(\mathfrak{g})$

So, decreasing neighbourhood size is not beneficial for agent i if and only if $\frac{|\ln(\frac{\varsigma\lambda}{\beta(1-\lambda)})|}{|\ln\lambda|} \geq \eta_i(\mathfrak{g})$.

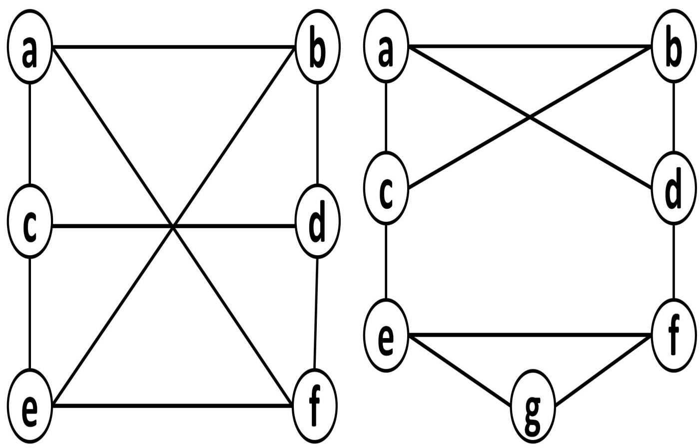
Therefore, $L = \frac{|\ln(\frac{\varsigma}{\beta(1-\lambda)})|}{|\ln\lambda|}$ and $U = \frac{|\ln(\frac{\varsigma\lambda}{\beta(1-\lambda)})|}{|\ln\lambda|}$ are, respectively, the lower and upper bounds of $\hat{\eta}$.

$$\text{U} = \frac{|\ln(\frac{\varsigma}{\beta(1-\lambda)})|}{|\ln\lambda|} + \frac{|\ln\lambda|}{|\ln\lambda|} = L+1.$$

It is easy to see that if L is not an integer (and hence, U is not an integer), the stability point $\hat{\eta}$ is the unique positive integer between L and U. □

Remark 4.6. *For most values of ς, β, and λ, $\frac{|\ln(\frac{\varsigma}{\beta(1-\lambda)})|}{|\ln\lambda|}$, and hence, $\frac{|\ln(\frac{\varsigma\lambda}{\beta(1-\lambda)})|}{|\ln\lambda|}$ are non-integers.*

Example 4.1. *Consider the networks $\mathfrak{g}_1$ and $\mathfrak{g}_2$ (see Figure 4.1). In both the networks, let the cost $\varsigma = 0.0055$, $\beta = 0.6$, and $\lambda = 0.2$. Here, $\left\lceil \frac{|\ln(\frac{\varsigma}{\beta(1-\lambda)})|}{|\ln\lambda|} \right\rceil = \lceil 2.72 \rceil$ and $\left\lfloor \frac{|(\ln\frac{\varsigma\lambda}{\beta(1-\lambda)})|}{|\ln\lambda|} \right\rfloor = \lfloor 3.72 \rfloor$, and hence, $\hat{\eta} = 3$. In network $\mathfrak{g}_1$, all agents have three neighbours each, and hence, $\mathfrak{g}_1$ is bilaterally stable. Despite the fact that agent g in the network $\mathfrak{g}_2$ has an incentive to add one more link, network $\mathfrak{g}_2$ is also bilaterally stable.*



((a)) Network $\mathfrak{g}_1$ ((b)) Network $\mathfrak{g}_2$

Figure 4.1: Stable SVN networks under the MO-Framework with sufficient storage

Now, we derive the stability point for SV-SRN network under the MO-Framework. Here, Definition 3.4 is relevant, and for simplicity, we assume that $\frac{\mathbf{s}}{\mathbf{d}}$ is an integer.

Theorem 4.4. *Let $\mathfrak{g}$ be an SV-SRN, under the MO-Framework.*

Then, $\tilde{n} = min\{\hat{\eta}, \frac{\mathbf{s}}{\mathbf{d}}\}$, is the unique stability point of $\mathfrak{g}$.

Proof. If all agents have sufficient storage, then from Theorem 4.3, $\hat{\eta}$ is the stability point.

Now, let us assume that each agent has a total amount of storage, **s**, available for sharing, **d** amount of data to backup. Then, $\frac{\mathbf{s}}{\mathbf{d}}$ defines the maximum possible neighbourhood size of each agent in the network.

Therefore, $min\{\hat{\eta}, \frac{\mathbf{s}}{\mathbf{d}}\}$ is the stability point, given $\varsigma, \lambda, \beta, \mathbf{s}, \mathbf{d}$.

Alternatively, we may also use the bound $\frac{\mathbf{s}}{\mathbf{d}}$ in Lemmas 4.1, 4.2 and Theorem 4.3. □

Henceforth, for the sake of uniformity, we shall use $\hat{\eta}$ (and not $\tilde{n}$) for the stability point of SV-SRN under the MO-Framework too.

4.1.2 Characterization Under the SO-Framework

In this subsection, we derive necessary and sufficient conditions for bilateral stability of SVN and SRN under the SO-Framework, and then discuss the stability point of these networks.

Lemma 4.4. *In an SVN, $\mathfrak{g}$, under the SO-Framework, for any agent $i \in \mathfrak{g}$, forming a partnership with another agent $j \in \mathfrak{g}$ is always beneficial.*

Proof. As $\mathfrak{g}$ is an SVN, under the SO-Framework, $u_i(\mathfrak{g}) = \beta(1 - \lambda^{\eta_i(\mathfrak{g})})$, for all $i \in \mathfrak{g}$.
For $i, j \in \mathfrak{g}$, if $\langle ij \rangle \notin \mathfrak{g}$, then

$u_i(\mathfrak{g} + \langle ij \rangle) = [\beta(1 - \lambda^{\eta_i(\mathfrak{g})+1})]$.

Adding a new link or backup partner is beneficial for agent i if and only if

$u_i(\mathfrak{g} + \langle ij \rangle) > u_i(\mathfrak{g})$, if and only if

$[\beta(1 - \lambda^{\eta_i(\mathfrak{g})+1})] > [\beta(1 - \lambda^{\eta_i(\mathfrak{g})})]$, if and only if

$\lambda^{\eta_i(\mathfrak{g})+1} < \lambda^{\eta_i(\mathfrak{g})}$, if and only if

$\lambda < 1$, which is always true. □

Corollary 4.1. *In an SVN,* $\mathfrak{g}$*, under the SO-Framework, no agent benefits by deleting any existing partnership.*

Theorem 4.5. *An SVN,* $\mathfrak{g}$*, under the SO-Framework, is bilaterally stable if and only if* $\mathfrak{g}$ *is a complete network.*

Proof. Follows from Lemma 4.4 and Corollary 4.1. □

Now, we state and prove the following for SRN, under the SO-Framework.

Lemma 4.5. *In an SRN,* $\mathfrak{g}$*, under the SO-Framework, for any agent* $i \in \mathfrak{g}$*, forming a partnership with another agent* $j \in \mathfrak{g}$ *is beneficial if and only if* $\mathbf{b} - \varsigma\eta_i(\mathfrak{g}) \geq \varsigma$ *and* $\mathbf{s} - \mathbf{d}\eta_j(\mathfrak{g}) \geq \mathbf{d}$.

Proof. In the SO-Framework, the utility of each agent $i \in \mathfrak{g}$ increases with increase in its neighbourhood size $\eta_i(\mathfrak{g})$.

Therefore, for any agent $i \in \mathfrak{g}$, forming a partnership with another agent $j \in \mathfrak{g}$ is beneficial if and only if agent *i*'s budget allows this link addition and agent *j* has free storage space for agent *i*'s data. (Refer Definition 3.5).

Agent *j* has free storage space for *i*'s data, if and only if $\mathbf{s} - \mathbf{d}\eta_j(\mathfrak{g}) \geq \mathbf{d}$. (Similar to the proof of Lemma 4.3).

Similarly, agent *i*'s budget allows adding a link, if and only if $\mathbf{b} - \varsigma\eta_i(\mathfrak{g}) \geq \varsigma$. □

Corollary 4.2. *In an SRN,* $\mathfrak{g}$*, under the SO-Framework, no agent benefits by deleting any existing partnership.*

Theorem 4.6. *An SRN* $\mathfrak{g}$ *under the SO-Framework is bilaterally stable if and only if* $[\mathbf{b} - \varsigma\eta_i(\mathfrak{g}) \geq \varsigma$ *and* $\mathbf{s} - \mathbf{d}\eta_j(\mathfrak{g}) \geq \mathbf{d}] \Rightarrow [\mathbf{b} - \varsigma\eta_j(\mathfrak{g}) < \varsigma$ *or* $\mathbf{s} - \mathbf{d}\eta_i(\mathfrak{g}) < \mathbf{d}]$*, for all* $\langle ij \rangle \notin \mathfrak{g}$.

Proof. Follows from Lemma 4.5, and Definition 3.5. □

Remark 4.7. *Since in an SRN* $\mathfrak{g}$ *under the SO-Framework, no agent benefits by deleting any existing partnership, link deletion does not appear in the bilateral stability conditions above.*

Now, we look at the stability point of SVN and SRN under the SO-Framework. The following case (Theorem 4.7) is the only case where the stability point depends on the number of agents, N.

Theorem 4.7. *In an SVN $\mathfrak{g}$, under the SO-Framework, $\hat{\eta} = N - 1$, is the unique stability point, where N is the number of agents.*

Proof. Follows from Lemma 4.4. □

Except SVN under the SO-Framework, in all other scenarios (including the following), the stability point is independent of N. In all cases (including the above), the stability point is unique. In the following, for simplicity, we assume that $\frac{\mathbf{s}}{\mathbf{d}}$ and $\frac{\mathbf{b}}{\varsigma}$ are integers.

Theorem 4.8. *In an SRN $\mathfrak{g}$, under the SO-Framework, $\hat{\eta} = min\{\frac{s}{d}, \frac{b}{\varsigma}\}$, for all $i \in \mathfrak{g}$, is the unique stability point, where no agent has incentive to add or delete a link.*

Proof. A constructive proof follows from Lemma 4.5.
Alternatively, it is clear that it is beneficial for each agent to add as many links as possible. The degree of agent i in $\mathfrak{g}$, $\eta_i(\mathfrak{g})$, is limited only by its storage space $\mathbf{s}$ and budget $\mathbf{b}$. That is,

$\mathbf{s} \geq \mathbf{d}\eta_i(\mathfrak{g})$ and $\mathbf{b} \geq \varsigma\eta_i(\mathfrak{g})$.

The theorem follows as the above is true for all $i \in \mathfrak{g}$. □

Example 4.2. *Let us consider the networks $\mathfrak{g}_1$ (see Figure 4.2(a)) and $\mathfrak{g}_2$ (see Figure 4.2(b)), each consisting of six agents, and network $\mathfrak{g}_3$ (see Figure 4.2(c)) consisting of seven agents. Assume that, in $\mathfrak{g}_1$ and $\mathfrak{g}_3$, $s = 60$ TB, $d = 20$ TB, $b = 0.5$, and $\varsigma = 0.1$. Assume that, in network $\mathfrak{g}_2$, $s = 60$ TB, $d = 10$ TB, $b = 0.4$, and $\varsigma = 0.1$.*

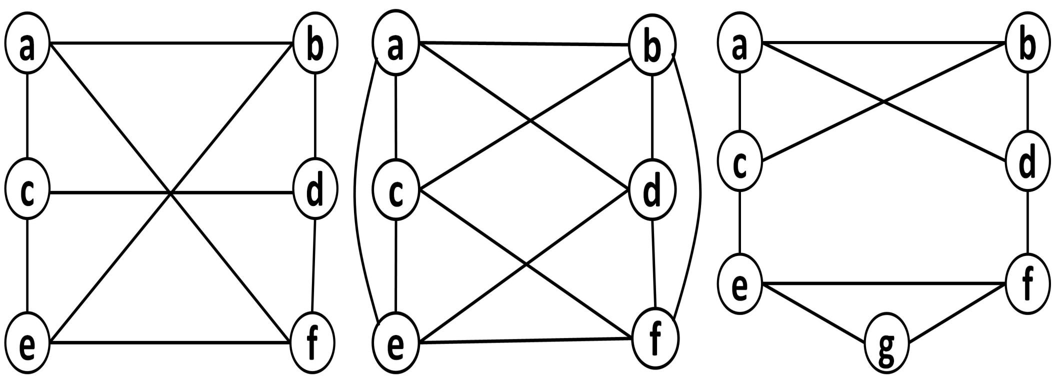
Note that in networks $\mathfrak{g}_1$ and $\mathfrak{g}_3$, although the budget constraints permit agents to maintain five neighbours each, storage limitations do not permit agents to maintain more than three neighbours each, and hence, $\mathfrak{g}_1$ and $\mathfrak{g}_3$ are bilaterally stable networks.

In network $\mathfrak{g}_2$, although storage constraints permit agents to maintain six neighbours each, budget constraints do not allow agents to maintain more than four neighbours each. Hence, $\mathfrak{g}_2$ is bilaterally stable.

We summarize the above results on bilateral stability conditions and stability point in Table 4.2 and Table 4.3, respectively.

Network Type	**Framework**	**Condition(s) for Bilateral Stability**
SVN	MO-Framework	1. For all $\langle ij\rangle \in \mathfrak{g}$, $\beta\lambda^{\eta_i(\mathfrak{g})-1}[1-\lambda] < \varsigma \Rightarrow \beta\lambda^{\eta_j(\mathfrak{g})-1}[1-\lambda] > \varsigma$, and 2. For all $\langle ij\rangle \notin \mathfrak{g}$, $\beta\lambda^{\eta_i(\mathfrak{g})}[1-\lambda] > \varsigma \Rightarrow \beta\lambda^{\eta_j(\mathfrak{g})}[1-\lambda] < \varsigma$.
SV-SRN	MO-Framework	1. For all $\langle ij\rangle \in \mathfrak{g}$, $\beta\lambda^{\eta_i(\mathfrak{g})-1}[1-\lambda] < \varsigma \Rightarrow \beta\lambda^{\eta_j(\mathfrak{g})-1}[1-\lambda] > \varsigma$, and 2. For all $\langle ij\rangle \notin \mathfrak{g}$, $\beta\lambda^{\eta_i(\mathfrak{g})}[1-\lambda] > \varsigma$ and $s-d\eta_j(\mathfrak{g}) \geq d$ $\Rightarrow \beta\lambda^{\eta_j(\mathfrak{g})}[1-\lambda] < \varsigma$ or $s-d\eta_i(\mathfrak{g}) < d$.
SVN	SO-Framework	Each agent $i \in \mathfrak{g}$ has backup partnerships with all agents $j \in \mathfrak{g}$ with $j \neq i$.
SRN	SO-Framework	For all $\langle ij\rangle \notin \mathfrak{g}$, $[b-\varsigma\eta_i(\mathfrak{g}) \geq \varsigma$ and $s-d\eta_j(\mathfrak{g}) \geq d]$ $\Rightarrow [b-\varsigma\eta_j(\mathfrak{g}) < \varsigma$ or $s-d\eta_i(\mathfrak{g}) < d]$.

Table 4.2: Summary of stability condition for different network



((a)) Network $\mathfrak{g}_1$ ((b)) Network $\mathfrak{g}_2$ ((c)) Network $\mathfrak{g}_3$

Figure 4.2: Stable SRN networks under the SO-Framework

Network Type	Framework	Unique Stability Point
SVN	MO-Framework	$\hat{\eta} = \left\lceil \frac{\lvert \ln(\frac{\varsigma}{\beta(1-\lambda)}) \rvert}{\lvert \ln \lambda \rvert} \right\rceil = \left\lfloor \frac{\lvert (\ln \frac{\varsigma \lambda}{\beta(1-\lambda)}) \rvert}{\lvert \ln \lambda \rvert} \right\rfloor.$
SV-SRN	MO-Framework	$\tilde{n} = min\{\hat{\eta}, \frac{s}{d}\}$
SVN	SO-Framework	$\hat{\eta} = N - 1$
SRN	SO-Framework	$\hat{\eta} = min\{\frac{s}{d}, \frac{b}{\varsigma}\}$

Table 4.3: Summary of stability point for different network types under MO- and SO-frameworks

4.2 Stable, Efficient and Contented Networks

We first discuss conditions on N and $\hat{\eta}$ for connected networks to be bilaterally stable in Section 4.2.1.1. We, then, look at networks that are comprised of multiple connected components, and discuss conditions on N, $\hat{\eta}$ as well as number of agents in individual components that lead to a bilaterally stable network in Section 4.2.1.2. Finally, we discuss conditions that lead to unique bilaterally stable networks in Section 4.2.1.3.

Henceforth, whenever we say $\mathfrak{g}$ is a symmetric social storage network, $\mathfrak{g}$ may be any of the networks SVN, SRN or SV-SRN with N agents, under the MO- or SO- Framework, with the unique stability point $\hat{\eta}$ corresponding to that network type and framework.

4.2.1 Stable Networks

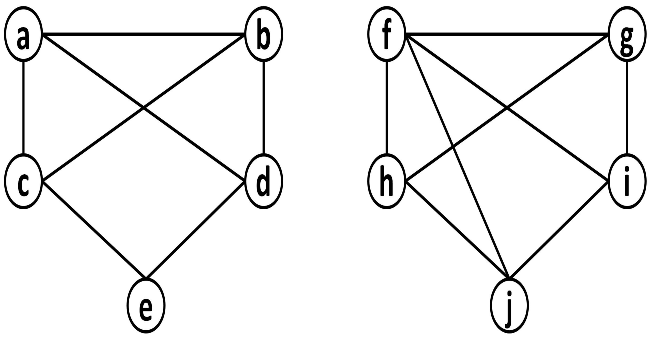
Up to this point, we have not explicitly discussed the process of network formation. This is because all our results above are independent of any process or protocol for network formation. However, the following results depend on the where we start the network formation from (refer [96] for different network configurations). We consider networks that evolve either from a null network (where all agents are initially disconnected) or from a complete network (where all agents are initially connected). When a network evolves from the null network, every agent starts contacting other agents to form links, in no particular order. This happens until there is no pair of agents who would consent to form a link. Similarly, when a network evolves from the complete network, pairs of agents consider deleting links if beneficial.

4.2.1.1 Connected Stable Networks

We start our discussion with the following remark.

Remark 4.8. *Each agent* aims *to achieve neighbourhood size* $\hat{\eta}$.

Though agents want to achieve neighbourhood size $\hat{\eta}$, this may not always be possible. This process of link addition or deletion by agents to reach $\hat{\eta}$ leads to a bilaterally stable network. The following example demonstrates how stable networks may evolve when all agents are isolated (Figure 4.3(a)) or all connected (Figure 4.3(b)), initially.



((a)) Network $\mathfrak{g}_1$ ((b)) Network $\mathfrak{g}_2$

Figure 4.3: Stable networks $\mathfrak{g}_1$ evolved from the null network, and $\mathfrak{g}_2$ from the complete network

Example 4.3. *We have two networks* $\mathfrak{g}_1$ *(see Figure 4.3(a)) and* $\mathfrak{g}_2$ *(see Figure 4.3(b)). Let us assume* $\hat{\eta} = 3$ *(indicates that no agent benefits by adding or more than three links or deleting less than three links).*

Let us consider, at the beginning, all agents are isolated in network $\mathfrak{g}_1$ *(thus, it evolves from the empty network) and all agents are connected with each other in network* $\mathfrak{g}_2$ *(thus, it evolves from the complete network). Note that, networks* $\mathfrak{g}_1$ *and* $\mathfrak{g}_2$ *both are bilaterally stable.*

In $\mathfrak{g}_1$*, although agent e has an incentive to add another link, no other agent having no link with e would consent to add a new link with e as they have already reached their* *stability point $\hat{\eta}$ (that is, their neighbourhood size is $\hat{\eta} = 3$) and hence, have no incentive to add or delete any link.*

In $\mathfrak{g}_2$, although agent f has an incentive to delete a link, no other agent (having a link with f) would consent to delete their link with f as they (that is, agents g, h, i and j) have already reached their stability point $\hat{\eta}$ (that is, their neighbourhood size is $\hat{\eta} = 3$) and hence, have no incentive to add or delete any link. □

In Proposition 4.1 and 4.2 below, we provide results that would be useful for an independent observer in checking for a bilateral stable symmetric social storage network, how many agents have maximised their utility. Thus, as discussed earlier, such an observer (say, an administrator or regulator) can externally perturb the system so that all agents achieve maximum utility.

Proposition 4.1. *Let N and $\hat{\eta}$ be (positive) odd integers, with $\hat{\eta} < N$. Then:*

1. *Any symmetric social storage network $\mathfrak{g}$ with N agents and stability point $\hat{\eta}$ consists of at least one agent who has an incentive to either add or delete a link.*

2. *There exists a connected, bilaterally stable, symmetric social storage network with exactly $N-1$ agents who have no incentive to add or delete any link.*

Proof. Let $\mathfrak{g}$ be bilaterally stable, and let ℓ be the number of links in $\mathfrak{g}$.

$\hat{\eta} < N$ ensures that ℓ does not exceed the maximum number of links $\mathfrak{g}$ can possibly have, that is, $\frac{N\times(N-1)}{2}$.

As the utility of each agent is maximum when its neighbourhood size is $\hat{\eta}$, total number of links $\tilde{\ell} = \frac{N\times\hat{\eta}}{2}$ will be attained if possible. However, $\tilde{\ell}$ is not an integer, as both N and $\hat{\eta}$ are odd.

This implies that, not all N agents have a neighbourhood size of $\hat{\eta}$ at stability. This proves (1).

Now, $N-1$ agents having $\hat{\eta}$ neighbours and the N^{th} agent having $\hat{\eta}-1$ or $\hat{\eta}+1$ neighbours are, however, possible. Let g be such a network with exactly $N-1$ agents who have no incentive to add or delete any link. These $N-1$ agents have neighbourhood size $\hat{\eta}$. None of these $N-1$ agents will consent to add or delete any link (among themselves, or with the

N^{th} agent). Thus, the symmetric social storage network $\mathfrak{g}$ is bilaterally stable. If $\mathfrak{g}$ is connected, we are done. Otherwise, all non-trivial components (that is, components with 2 or more agents in each) of $\mathfrak{g}$ can be connected as follows, without changing the neighbourhood sizes of any of the agents. Let $\langle i_1 j_1 \rangle$ and $\langle i_2 j_2 \rangle$ be links in two different (non-trivial) components, say $\mathfrak{g}(\kappa_1)$ and $\mathfrak{g}(\kappa_2)$ of $\mathfrak{g}$. Deleting both these links, and replacing them with $\langle i_1 j_2 \rangle$ and $\langle i_2 j_1 \rangle$ connects $\mathfrak{g}(\kappa_1)$ and $\mathfrak{g}(\kappa_2)$, without changing the neighbourhood sizes of any of the agents. As neighbourhood sizes of all the agents remain the same, the resulting graph is bilaterally stable too. Now, if i is an isolated agent and $\langle jk \rangle$ is a link in $\mathfrak{g}$, delete $\langle jk \rangle$, and add $\langle ij \rangle$ and $\langle ik \rangle$ instead. In this case, clearly, i continues to be the only agent with an incentive to either add or delete a link. This proves (2). □

Remark 4.9. *In the proof of Proposition 4.1, on the one hand, when the network evolves from the null network,* $\hat{\eta} - 1$ *neighbours for the* N^{th} *agent is as beneficial as possible, and the total number of links will, hence, be* $\ell = \frac{[(N-1)\hat{\eta}+(\hat{\eta}-1)]}{2}$.

On the other hand, when the network evolves from the complete network, $\hat{\eta} + 1$ *neighbours for the* N^{th} *agent is as beneficial as possible, and the total number of links will, hence, be* $\ell = \frac{[(N-1)\hat{\eta}+(\hat{\eta}+1)]}{2}$. *This number also does not exceed the maximum possible number of links, as* $\hat{\eta} \leq N-2$ *(because* $\hat{\eta} < N$*, and both* $\hat{\eta}$ *and* N *are odd).*

Proposition 4.2. *Let at least one of* N *and* $\hat{\eta}$ *be even, and let* $\hat{\eta} < N$. *Then, there exists a connected bilaterally stable symmetric social storage network* $\mathfrak{g}$ *where no agent has incentives to add or delete any link.*

Proof. Existence of the $\hat{\eta}$-regular network on N agents, $\mathfrak{g}$, follows trivially from the Erdős–Gallai theorem. Clearly $\mathfrak{g}$ is a bilaterally stable. □

4.2.1.2 Stable Network with Multiple Connected Components

We, now, discuss results on stability of symmetric storage networks with two or more components. Examples of scenarios where this might be useful include companies under the same umbrella group, where the social storage networks of each of these companies may be viewed as a component of a larger network, which may be monitored or analysed by an independent observer (as discussed in the previous section).

Claim 4.1. *Suppose $\mathfrak{g}$ is a symmetric social storage network with two or more components. If $\mathfrak{g}$ is bilaterally stable, then there is at most one component with less than or equal to $\hat{\eta}$ agents.*

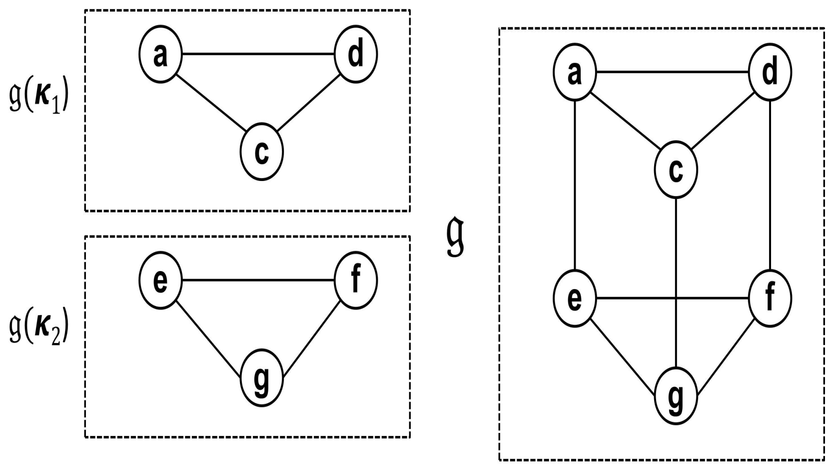
Proof. Suppose, $\mathfrak{g}(\kappa_i)$ and $\mathfrak{g}(\kappa_j)$ are two different (non-empty) components with less than or equal to $\hat{\eta}$ agents. It is easy to see that all agents in $\mathfrak{g}(\kappa_i)$ as well as in $\mathfrak{g}(\kappa_j)$ have less than $\hat{\eta}$ neighbours. Consider agents $i \in \mathfrak{g}(\kappa_i)$ and $j \in \mathfrak{g}(\kappa_j)$. Clearly, $\langle ij \rangle \notin \mathfrak{g}$ but both i and j have incentives to form (at least) one link each, implying $\mathfrak{g}$ is not bilaterally stable. □

Proposition 4.3. *Let $\mathfrak{g}$ be a symmetric social storage network which has evolved from the null network. Let $\hat{\eta}$ be odd. Suppose $\mathfrak{g}$ consists of κ connected components, $\kappa \geq 2$. Suppose at least two of the components, say $\mathfrak{g}(\kappa_1)$ and $\mathfrak{g}(\kappa_2)$, each have either $\leq \hat{\eta}$ agents or an odd number of agents more than $\hat{\eta}$. Then $\mathfrak{g}$ is not bilaterally stable.*

Proof. Follows from Proposition 4.1 and Claim 4.1. □

One agent in $\mathfrak{g}(\kappa_1)$ (say p_{κ_1}) and another agent in $\mathfrak{g}(\kappa_2)$ (say p_{κ_2}) each have neighbourhood size less than $\hat{\eta}$. Hence, both p_{κ_1} and p_{κ_2} have an incentive to add a link, and gain by forming $\langle p_{\kappa_1} p_{\kappa_2} \rangle \notin \mathfrak{g}$. Thus, the network $\mathfrak{g}$ is not bilaterally stable.

Example 4.4.



((a)) Two components, $\mathfrak{g}(\kappa_1)$ and $\mathfrak{g}(\kappa_2)$, of $\mathfrak{g}$

((b)) Bilaterally stable network $\mathfrak{g}'$

Figure 4.4: Two components $\mathfrak{g}(\kappa_1)$ and $\mathfrak{g}(\kappa_2)$, though complete, are bilaterally unstable when $\hat{\eta} = 3$ and form a bilaterally stable network $\mathfrak{g}'$. However, the network $\mathfrak{g}$ consisting of $\mathfrak{g}(\kappa_1)$ and $\mathfrak{g}(\kappa_2)$ as two components is bilaterally stable when $\hat{\eta} = 2$

Consider network $\mathfrak{g}$ (see Figure 4.4(a)), with two components $\mathfrak{g}(\kappa_1)$ and $\mathfrak{g}(\kappa_2)$.

If $\hat{\eta} = 3$, then both components of $\mathfrak{g}$ have $\hat{\eta}$ agents. Every agent has an incentive to add one more link. Thus $\mathfrak{g}$ is bilaterally unstable. Clearly, no agent can add any more links within the same component. The network $\mathfrak{g}'$ (see Figure 4.4(b)) is an example of a bilaterally stable network, which evolves from $\mathfrak{g}$. Now, if $\hat{\eta} = 2$, the network $\mathfrak{g}$ is bilaterally stable. □

Corollary 4.3. *Let $\mathfrak{g}$ be a symmetric social storage network which has evolved from the null network and which consists of κ components, $\kappa \geq 2$. Let $\hat{\eta}$ be odd, and let $N > \hat{\eta}$. If $\mathfrak{g}$ is bilaterally stable, then at least $\kappa - 1$ components must consist of an even number of agents greater than $\hat{\eta}$.*

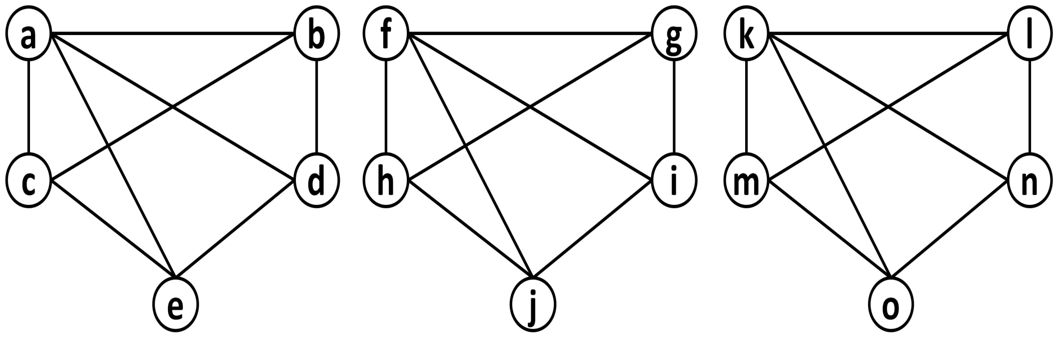
Remark 4.10. *In Proposition 4.3 and Corollary 4.3, if we consider networks which have evolved from the complete network, then Example 4.5 below acts as a counter example. If $\hat{\eta}$ is even, we apply Proposition 4.2 to each component having more than $\hat{\eta}$ agents to see that each of these components is bilaterally stable. Now, there can be at most one component with $\leq \hat{\eta}$ agents (refer Claim 4.1), and if there is such a component, $\mathfrak{g}$ is bilaterally stable if and only if that component is complete.*

The following example shows a bilaterally stable network, which has evolved from the complete network.

Example 4.5. *Let $N = 15$ and $\hat{\eta} = 3$. Consider the network $\mathfrak{g}$ on N agents (see Figure 4.5) that consists of three components, $\mathfrak{g}(\kappa_1)$, $\mathfrak{g}(\kappa_2)$ and $\mathfrak{g}(\kappa_3)$. Though $\mathfrak{g}$ consists of three agents, a, f and k, who have an incentive to delete a link each, $\mathfrak{g}$ is bilaterally stable. This is because the agents, a, f and k, are in three different components, in each of which all other agents have neighbourhood size $\hat{\eta}$.*

Claim 4.2. *Suppose $\mathfrak{g}$ is a symmetric social storage network. If $\hat{\eta} = 1$, and if $\mathfrak{g}$ has evolved from the null network, then $\mathfrak{g}$ is bilaterally stable if and only if $\mathfrak{g}$ consists of a set of $\frac{N-1}{2}$ connected pairs of agents plus one isolated agent if N is odd, and a set of $\frac{N}{2}$ connected pairs of agents if N is even.*

Proof. As $\mathfrak{g}$ has evolved from the null network and as $\hat{\eta} = 1$, no agent has two or more neighbours. Hence, if N is even, $\mathfrak{g}$ consists of $\frac{N}{2}$ connected pairs of agents. Similarly, if N is odd, $\mathfrak{g}$ consists of one isolated agent and the remaining $N - 1$ agents connect in pairs. □



((a)) $\mathfrak{g}(\kappa_1)$ ((b)) $\mathfrak{g}(\kappa_2)$ ((c)) $\mathfrak{g}(\kappa_3)$

Figure 4.5: Stable Network $\mathfrak{g}$ on 15 agents, consisting of 3 components $\mathfrak{g}(\kappa_1)$, $\mathfrak{g}(\kappa_2)$ and $\mathfrak{g}(\kappa_3)$

Remark 4.11. *In Claim 4.2, if* $\mathfrak{g}$ *has evolved from the complete network (by mutual deletion of links), then networks consisting of star components are also bilaterally stable as per Definitions 3.3, 3.4, and 3.5.*

Remark 4.12. *In Claim 4.2, if* $\mathfrak{g}$ *is a given network, then in addition to the star components as discussed in Remark 4.11,* $\mathfrak{g}$ *may also consist of (at most) one isolated agent and continue to be bilaterally stable.*

It is interesting to note that in any star network, given that $\hat{\eta} = 1$, though the universal agent has incentive to delete a link (or links), no other (pendant) agent will consent to deletion. However, if we start from the null network, we have the following observation.

Claim 4.3. *Suppose* $\mathfrak{g}$ *has evolved from the null network. Then, if* $\mathfrak{g}$ *is bilaterally stable,* $\mathfrak{g}$ *can never contain a star network as component.*

Proof. If $\hat{\eta} = 1$, the result follows from Claim 4.2.

Suppose $\hat{\eta} > 1$. If possible, let $\mathfrak{g}$ be a star network. It is easy to see that all pendant agents have incentives to add (at least) one more link implying that $\mathfrak{g}$ is not bilaterally stable, a contradiction. □

4.2.1.3 Unique Stable Networks

In the previous subsections, we have seen results on the existence of a bilaterally stable social storage network. In this subsection, we look at conditions under which a unique bilaterally stable social storage network exists. Whenever a unique bilaterally stable network exists, the agents themselves endogenously form this network. Any independent observer or regulator knows precisely which network would form (or has formed).

Claim 4.4. *If $N = \hat{\eta} + 1$ or $\hat{\eta} \geq N$, then there exists a unique symmetric social storage network $\mathfrak{g}$ that is bilaterally stable, namely the complete network on N agents.*

Proof. In both cases (that is, $N = \hat{\eta} + 1$ or $\hat{\eta} \geq N$), the complete network is the one which maximises the utility of each agent. That is, no agent has an incentive to delete any existing link and, clearly, no agent can add any more links. □

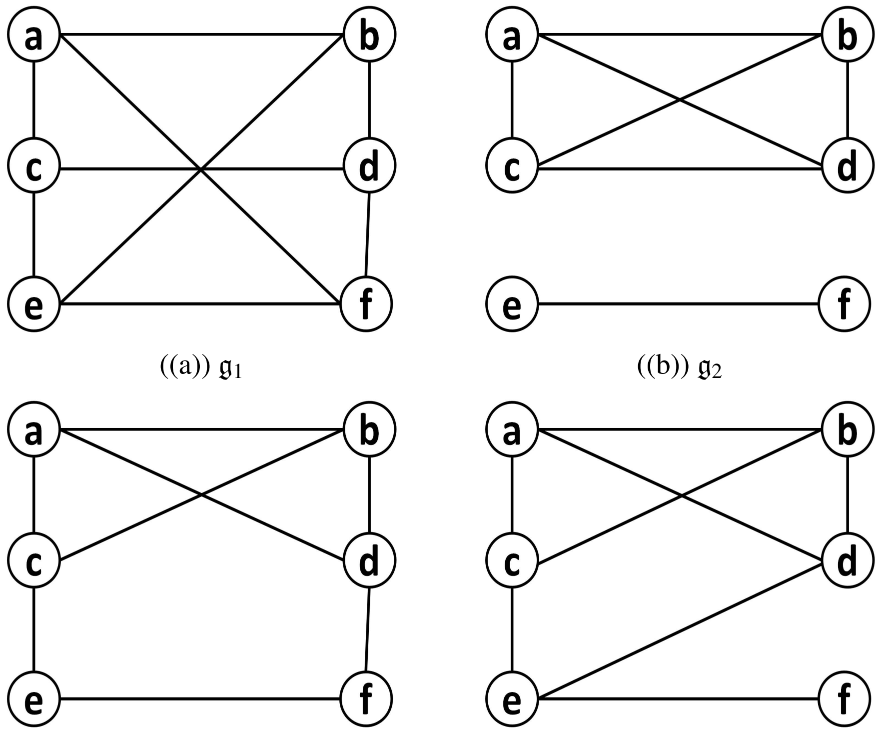
Claim 4.5. *If $N > \hat{\eta} + 1$, then there are always two or more different (with respect to degree sequence[26]) bilaterally stable networks.*

Proof. If $N = \hat{\eta} + 2$, the following stable networks are possible, which are different with respect to degree sequence. The first, where $N-1$ agents form a clique and the other agent is isolated. The second network is as follows. If $\hat{\eta}$ and, hence, N are even, the connected regular network with N agents, where each agent has a neighbourhood size of $\hat{\eta}$ is bilaterally stable. If $\hat{\eta}$ and, hence, N are odd, the connected network with N agents, where $N-1$ agents have a neighbourhood size of $\hat{\eta}$ and the other agent has a neighbourhood size of $\hat{\eta} - 1$, is bilaterally stable. (If $\hat{\eta}$ and N are odd, the connected network where $N-1$ agents have a neighbourhood size of $\hat{\eta}$ and the other agent has a neighbourhood size of $\hat{\eta} + 1$, is a third bilaterally stable network). □

Example 4.6. *Let $\hat{\eta} = 3$ and $N = 6$. Then, there are four networks $\mathfrak{g}_1$ (see Figure 4.6(a)), $\mathfrak{g}_2$ (see Figure 4.6(b)), $\mathfrak{g}_3$ (see Figure 4.6(c)) and $\mathfrak{g}_4$ (see Figure 4.6(d)) which are bilaterally stable.*

---

[26] Two networks are *different with respect to degree sequence* if the sorted sequence of degrees (neighbourhood sizes) in one is different from that of the other. Note that, both sequences are sorted in the ascending order (or both in the descending order).



((a)) g_1 ((b)) g_2

((c)) g_3 ((d)) g_4

Figure 4.6: Bilaterally stable networks with $N = 6$ agents, $\hat{\eta} = 3$.

If we look at specific protocols of network formation, then we get further uniqueness results. For example, in Claim 4.2, starting from the null network (or any network where no agent has more than 1 neighbour), the resulting bilaterally stable network is unique up to isomorphism.

4.2.2 Efficient and Contented Social Storage Networks

In this subsection, we look at efficient social storage networks and contented social storage networks. As discussed earlier, an observer who observes or monitors or regulates the network may externally perturb the system so as to reach an efficient or a contented network. We have seen in Section 4.1 that there exists a unique stability point, $\hat{\eta}$, (for each network type, under the given framework) such that, no agent gains by adding more neighbours than $\hat{\eta}$, and severing existing relationships resulting in a neighbourhood size of less than $\hat{\eta}$. An efficient social storage network is, hence, one in which maximum possible number of agents have $\hat{\eta}$ neighbours.

Remark 4.13. *An efficient social storage network is bilaterally stable.*

We, now, discuss an example to highlight the fact that not all stable networks are efficient.

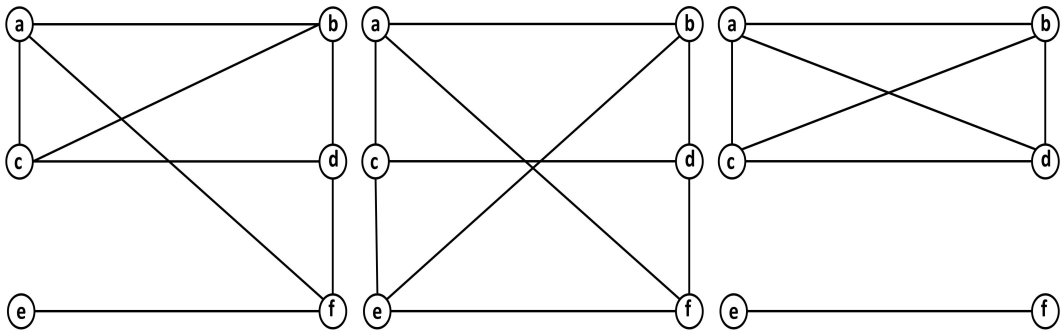
Example 4.7. *Suppose there are six agents, a,b,c,d,e and f, in a social storage network with stability point $\hat{\eta} = 3$. Assume that, starting from the null network, these agents add links (that is, build mutual data backup partnerships). Different network structures may emerge, for example Figure 4.7(a), Figure 4.7(b), and Figure 4.7(c)).*

In network $\mathfrak{g}_1$ (see Figure 4.7(a)), agent e's expected value of data backup is less than that of the rest of the agents. In $\mathfrak{g}_2$ (see Figure 4.7(b)), all agents achieve the same (and maximum) expected value of data backup, and in $\mathfrak{g}_3$ (see Figure 4.7(c)), agents a,b,c, and d achieve higher expected value of data backup than agents e and f. $\mathfrak{g}_2$ is efficient, whereas $\mathfrak{g}_1$ and $\mathfrak{g}_3$ are not (though they are bilaterally stable).

We, now, discuss contented networks.

Remark 4.14. *A contented social storage network is bilaterally stable.*

It is easy to see that not all stable networks are contented. In Example 4.3, though both $\mathfrak{g}$ and $\mathfrak{s}$ are stable, neither of these networks are contented. Consider $\mathfrak{g}$. An independent observer could just add a storage device, p, to the network, which leads to a contented



((a)) $\mathfrak{g}_1$ ((b)) $\mathfrak{g}_2$ ((c)) $\mathfrak{g}_3$

Figure 4.7: Network structure and social welfare

network as explained below. This storage device acts as a dummy agent, not trying to maximize its utility, and always agreeing to add or delete any link with any agent. Hence, for contented networks, we do not consider any dummy agent as a part of the network. In $\mathfrak{g}$, agent e has not achieved the maximum possible utility (as $\hat{\eta} = 3$, but e has 2 neighbours) while all other agents have. By allowing e to store a copy of its data on storage device p, e also obtains the maximum possible utility. This network is, now, a contented network (where the utility of the dummy agent p is not considered). This is, in fact, a hybrid model — hybrid between a centralized storage system and a decentralized one.

Next, we relate contented networks and efficient networks.

Proposition 4.4. *Let $\mathfrak{g}$ be a symmetric social storage network with N agents and stability point $\hat{\eta}$. Then:*

1. *If at least one of N and $\hat{\eta}$ is (are) even, then, $\mathfrak{g}$ is efficient if and only if $\mathfrak{g}$ is contented.*

2. *Suppose N and $\hat{\eta}$ are odd. Then, an efficient network does exist but there does not exist any contented network.*

Proof. *1* follows from Proposition 4.2, since, if at least one of N and $\hat{\eta}$ is/ are even, then, $\mathfrak{g}$ is efficient if and only if $\mathfrak{g}$ is $\hat{\eta}$-regular.

2 follows from Proposition 4.1. □

Remark 4.15. *Not all efficient networks are contented.*

In Example 4.3, neither $\mathfrak{g}$ nor $\mathfrak{s}$ are contented. However, (at least) one of them is efficient. The following Propositions help identify which of them is/ are efficient, under the MO- as well as SO-Frameworks, for SVN, SRN and SV-SRN networks, as the case may be (Refer Table 4.1). Note that any stable network in which maximum possible number of agents have $\hat{\eta}$ neighbours is not necessarily efficient, as per Definitions 3.6, 3.7, and 3.8.

Proposition 4.5. *Let $\mathfrak{g}$ be an SVN or SV-SRN under the MO-Framework, with N agents and stability point $\hat{\eta}$. Suppose both N and $\hat{\eta}$ are odd. Then $\mathfrak{g}$ is efficient if and only if $\mathfrak{g}$ has $N-1$ agents with neighbourhood size $\hat{\eta}$ and one of the following holds:*

1. $c < \frac{\beta\lambda^{\hat{\eta}}}{2}(\frac{1}{\lambda} - \lambda)$ *and* $\mathfrak{g}$ *has one agent with neighbourhood size* $\hat{\eta}+1$.

2. $c > \frac{\beta\lambda^{\hat{\eta}}}{2}(\frac{1}{\lambda} - \lambda)$ *and* $\mathfrak{g}$ *has one agent with neighbourhood size* $\hat{\eta}-1$.

3. $c = \frac{\beta\lambda^{\hat{\eta}}}{2}(\frac{1}{\lambda} - \lambda)$ *and* $\mathfrak{g}$ *has one agent with neighbourhood size either* $\hat{\eta}+1$ *or* $\hat{\eta}-1$.

Proof. For each $i \in \mathbf{A}$, its utility is $u_i(\mathfrak{g}) = \beta_i(1-\lambda^{\eta_i(\mathfrak{g})}) - c\eta_i(\mathfrak{g})$. (Refer Equation 3.2). As the network is SVN or SV-SRN, $\beta_i = \beta$, for all i.

$$\underset{\eta_i(\mathfrak{g})}{max}\, u_i(\mathfrak{g}) = \underset{\eta_i(\mathfrak{g})}{max}\,\{\beta(1-\lambda^{\eta_i(\mathfrak{g})}) - c\eta_i(\mathfrak{g})\} = \beta(1-\lambda^{\hat{\eta}}) - c\hat{\eta}$$

Let $\mathfrak{g}_1$ be the network where $N-1$ agents have $\hat{\eta}$ neighbours and the other agent has $\hat{\eta}-1$ neighbours. Let $\mathfrak{g}_2$ be the network where $N-1$ agents have $\hat{\eta}$ neighbours and the other agent has $\hat{\eta}+1$ neighbours.

$$u_i(\mathfrak{g}_1) = \beta(1-\lambda^{(\hat{\eta}-1)}) - c(\hat{\eta}-1) \text{ and } u_i(\mathfrak{g}_2) = \beta(1-\lambda^{(\hat{\eta}+1)}) - c(\hat{\eta}+1).$$

From Proposition 4.1 and Definition 3.6, it is easy to see that either $\mathfrak{g}_1$ or $\mathfrak{g}_2$ (or both) is (are) efficient. That is, $max\{\sum_i u_i(\mathfrak{g})\}$ is either $u_i(\mathfrak{g}_1)$ or $u_i(\mathfrak{g}_2)$. If $u_i(\mathfrak{g}_1) < u_i(\mathfrak{g}_2)$, we get result 1, if $u_i(\mathfrak{g}_1) > u_i(\mathfrak{g}_2)$, we get result 2 and if $u_i(\mathfrak{g}_1) = u_i(\mathfrak{g}_2)$, both $\mathfrak{g}_1$ and $\mathfrak{g}_2$ are efficient, leading to result 3. □

Proposition 4.6. *An SVN under the SO-Framework is efficient if and only if it is a complete network.*

Proof. Follows from Theorem 4.7. □

Proposition 4.7. *For an SRN under the SO-Framework,* $\mathbf{s}/\mathbf{d}$ *and* $\mathbf{b}/\varsigma$ *act as constraints for the maximum number of links possible.*

Let $\mathfrak{g}$ *be an SRN under the SO-Framework, with N agents and stability point* $\hat{\eta}$*. Then:*

1. *If at least one of N and* $\hat{\eta}$ *is (or are) even, then* $\mathfrak{g}$ *is efficient if and only if* $\mathfrak{g}$ *is* $\hat{\eta}$*-regular.*

2. *If both N and* $\hat{\eta}$ *are odd, then* $\mathfrak{g}$ *is efficient if and only if* $\mathfrak{g}$ *has* $N-1$ *agents with* $\hat{\eta}$ *neighbours and the other agent with* $\hat{\eta}-1$ *neighbours.*

Proof. From Theorem 4.8, $\hat{\eta} = min\{\frac{\mathbf{s}}{\mathbf{d}}, \frac{\mathbf{b}}{\varsigma}\}$.

As the budget **b** and the storage space available **s** act as constraints, no agent can have more than $\hat{\eta}$ neighbours. Therefore, part 1 follows from Proposition 4.1. (We do not have the possibility of one agent having $\hat{\eta}+1$ neighbours as we had in Proposition 4.5).

Part 2 follows from Proposition 4.2. □

Chapter Summary

This chapter examined social storage network formation and its stability, efficiency, and contentedness in a strategic setting. Studying social storage systems as an endogenous network formation game, and then analysing its stability may, on first glance, seem contradictory — since one cannot do anything from outside the (endogenous) system if the agents' themselves do not form a stable network, efficient or contended network.

This study observed the tension between stability and efficiency, that is, in the social storage network, agents always form a stable network, but the network may not be *efficient*. That is, the sum of the utilities of all agents may not be the maximum possible. In our case, as many agents as possible may not be *contented* as well, i.e. all agents achieve maximum possible utility. Contented networks are also efficient.

Looking at both endogenous network formation and efficiency and contentment is useful because, though the social storage system is built endogenously, an independent observer (say, an administrator or a regulator) can check whether the system is efficient and contented, and if not, can externally do a small perturbation to the network. In some scenarios, we looked at, the independent administrator may achieve efficiency and (or) contentment by just introducing a small number of dummy agents.

Chapter 5

Social Storage Cloud: Stability and Efficiency

In the social storage network model discussed in the previous chapter, agents perform data backup activity with their neighbours only, that is, they share their disk space with only those who are directly connected with them. In the social storage network model, disk failure is assumed as a cause of data loss. In this chapter, we present the social storage cloud model with the assumption that the reasons for data loss could be hardware, software, human error, or so on. Further, the recent studies [24, 23] in social storage suggest incorporating indirect connections to improve QoS in these systems. The social storage cloud model presented in this chapter focuses on indirect connections as well. In other words, in the social storage cloud model, agents perform storage resource (disk space) sharing with directly and indirectly connected agents, however, the storage resource allocation is conditional and based on the closeness. This chapter proposes the degree-distance-based utility in the social storage cloud context, where the benefit of agents depends on the chance of obtaining storage in the network.

Recall that the expected utility of agent i in a social storage cloud is

$$u_i(\mathfrak{g}) = \beta_i(1-\delta_i)+\beta_i\delta_i\gamma_i(\mathfrak{g}) - \varsigma_i\eta_i(\mathfrak{g}). \tag{5.1}$$

As discussed in the previous chapter, in order to deal with free riding, many P2P storage systems (for example, Internet Cooperative Backup System [44], PeerStore [42], Pastiche [93]) follow a symmetric storage-sharing mechanism, where agents share the same amount of storage space.

We define a symmetric social storage cloud $\mathfrak{g}$ as follows.

Definition 5.1. *A symmetric social storage cloud (SSSC) $\mathfrak{g}$ is a network where the benefit (value) associated with backed-up data is the same for all agents in the network, that is, $\beta_i = \beta_j$ (say β), $\varsigma_i = \varsigma_j$ (say ς[27]), and $\delta_i = \delta_j$ (say δ[28]) for all $i, j \in \mathscr{A}$, and hence, utility of each agent i in $\mathfrak{g}$ is*

$$u_i(\mathfrak{g}) = \beta(1-\delta)+\beta\delta\gamma_i(\mathfrak{g}) - \varsigma\eta_i(\mathfrak{g}), \tag{5.2}$$

where $\delta, \beta, \varsigma \in (0,1)$.

For further study, we consider the above utility function (Equation (5.2)). Henceforth, whenever we refer to a network, or just $\mathfrak{g}$, we mean an SSSC. In order to characterize endogenously built social storage cloud, we adopt *pairwise stability* [31] as a solution concept.

5.1 Closeness and Storage Availability

One of the objectives of this study is to understand the impact of link addition and deletion on storage availability for those agents who are involved in the link addition (or deletion) as well as those who are not. The storage availability is determined by the distances between them and their closeness (from Equation (3.6)). Therefore, first we study how addition and deletion of a link impacts the shortest distances between pairs of agents and, therefore, their closeness. This analysis provides a base for understanding the effect of link-addition (or deletion) on agents' storage availability in $\mathfrak{g}$.

5.1.1 Closeness

In this section, we examine how the action of link addition (deletion) between a pair of agents impacts their as well as others' closeness.

[27]We assume, $\varsigma = \frac{\varsigma_i+\varsigma_j}{2}$, that is, a pair of agents involved in a link share the cost ς.
[28]For simplicity, we assume uniform data loss rate δ.

5.1.1.1 Closeness: A Pair of Agents Involved in the Actions

First, we discuss our results regarding the impact of link addition and deletion on the closeness of those who are involved in the actions.

Lemma 5.1. *Suppose* $\langle ij \rangle \notin \mathfrak{g}$*. Then,* $\Phi_i(\mathfrak{g} + \langle ij \rangle) > \Phi_i(\mathfrak{g})$.

Proof. Clearly, $d_{ij}(\mathfrak{g} + \langle ij \rangle) < d_{ij}(\mathfrak{g})$. As $\langle ij \rangle \notin \mathfrak{g}$, we have, $d_{ij}(\mathfrak{g}) \geq 2$. Also, $d_{ij}(\mathfrak{g} + \langle ij \rangle) = 1$. Thus, $\Phi_i(\mathfrak{g})$ and $\Phi_j(\mathfrak{g})$ increase by at least $\frac{d_{ij}(\mathfrak{g})-1}{d_{ij}(\mathfrak{g})}$ in $\mathfrak{g} + \langle ij \rangle$. □

Lemma 5.2. *Suppose* $\langle ij \rangle \in \mathfrak{g}$*. Then,* $\Phi_i(\mathfrak{g} - \langle ij \rangle) < \Phi_i(\mathfrak{g})$.

Proof.

1. Let us assume there is no path between i and j in $\mathfrak{g} - \langle ij \rangle$, then $d_{ij}(\mathfrak{g} - \langle ij \rangle) = \infty$, thus, $\Phi_i(\mathfrak{g})$ and $\Phi_j(\mathfrak{g})$ decrease by 1 in $\mathfrak{g} - \langle ij \rangle$.

2. Now, let us assume there exists a path $\mathscr{P}_{ij}(\mathfrak{g} - \langle ij \rangle)$ between i and j in $\mathfrak{g} - \langle ij \rangle$, the distance between i and j in $\mathfrak{g} - \langle ij \rangle$ being at least 1 more than that in $\mathfrak{g}$. Thus, $\Phi_i(\mathfrak{g})$ and $\Phi_j(\mathfrak{g})$ decrease by at least $\frac{d_{ij}(\mathfrak{g} - \langle ij \rangle)-1}{d_{ij}(\mathfrak{g} - \langle ij \rangle)}$ in $\mathfrak{g} - \langle ij \rangle$. □

Lemmas 5.1 and 5.2 show that, with respect to closeness, every link benefits agents on either side of the link. An action of link addition or deletion between a pair of agents not only impacts their closeness, but also that of other agents.

Now, we study the impact of link addition or deletion between a pair of agents (say, i and j) on the closeness of the other agents $k \in \mathfrak{g} \setminus \{i, j\}$.

Lemma 5.3. *Suppose* $\langle ij \rangle \notin \mathfrak{g}$ *and* $k \in \mathfrak{g} \setminus \{i, j\}$*. Then,* $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g} + \langle ij \rangle)$ *if and only if* $d_{kl}(\mathfrak{g}) = d_{kl}(\mathfrak{g} + \langle ij \rangle)$ *for all* $l \in \mathfrak{g}$.

Proof. If $d_{kl}(\mathfrak{g}) = d_{kl}(\mathfrak{g} + \langle ij \rangle)$ for all $l \in \mathfrak{g}$, then by Equation (3.5), $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g} + \langle ij \rangle)$.

Conversely, suppose $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g} + \langle ij \rangle)$.

It is easy to see that, if for some $l \in \mathfrak{g}$, if $d_{kl}(\mathfrak{g} + \langle ij \rangle) \neq d_{kl}(\mathfrak{g})$, then $d_{kl}(\mathfrak{g} + \langle ij \rangle) < d_{kl}(\mathfrak{g})$. (Paths in $\mathfrak{g}$ exist in $\mathfrak{g} + \langle ij \rangle$ too).

We have $d_{kl}(\mathfrak{g} + \langle ij \rangle) \leq d_{kl}(\mathfrak{g})$ for all $l \in \mathfrak{g}$ and, if there exists x such that $d_{kx}(\mathfrak{g} + \langle ij \rangle) < d_{kx}(\mathfrak{g})$, then $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g} + \langle ij \rangle)$, a contradiction. □

Lemma 5.4. *Suppose* $\langle ij\rangle \in \mathfrak{g}$ *and* $k \in \mathfrak{g}\setminus\{i,j\}$. *Then,* $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g}-\langle ij\rangle)$ *if and only if* $d_{kl}(\mathfrak{g}) = d_{kl}(\mathfrak{g}-\langle ij\rangle)$ *for all* $l \in \mathfrak{g}$.

Proof. As $d_{kl}(\mathfrak{g}-\langle ij\rangle) \geq d_{kl}(\mathfrak{g})$ for all l, the proof follows in lines similar to that of Lemma 5.3. □

5.1.1.2 Closeness: Agents not Involved in the Actions

We now show necessary and sufficient conditions for increase or decrease in the closeness of agents who are not involved in link addition or deletion.

Theorem 5.1. *Suppose* $\langle ij\rangle \notin \mathfrak{g}$, *and let* k *be an agent distinct from* i *and* j. *Then,* $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$ *if and only if there exists at least one agent* $l \in \mathfrak{g}$ *such that* $d_{kl}(\mathfrak{g}) \geq 3$ *and all shortest paths* $\mathscr{P}_{kl}(\mathfrak{g}+\langle ij\rangle)$ *from* k *to* l *in* $\mathfrak{g}+\langle ij\rangle$ *contain* $\langle ij\rangle$.

Proof. Let $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$. Then, by Lemma 5.3, there must be at least one agent, say l, such that $d_{kl}(\mathfrak{g}) > d_{kl}(\mathfrak{g}+\langle ij\rangle)$.

Suppose i,k, and l are all distinct. Note that j may be the same as l.

If possible, let $d_{kl}(\mathfrak{g}) < d_{ki}(\mathfrak{g})+d_{ij}(\mathfrak{g})+d_{jl}(\mathfrak{g})$ for all $l \in \mathfrak{g}$. Then, $d_{kl}(\mathfrak{g}) = d_{kl}(\mathfrak{g}+\langle ij\rangle)$ for all $l \in \mathfrak{g}$. From Lemma 5.1, $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$, a contradiction. Therefore, there exists an $l \in \mathfrak{g}$ such that $d_{kl}(\mathfrak{g}) = d_{ki}(\mathfrak{g})+d_{ij}(\mathfrak{g})+d_{jl}(\mathfrak{g})$.

As $\langle ij\rangle \notin \mathfrak{g}$, $d_{ij}(\mathfrak{g}) \geq 2$. As $k \neq i$, $d_{ik}(\mathfrak{g}) \geq 1$ and $j = l$. Hence, $d_{kl}(\mathfrak{g}) \geq 3$.

$$\begin{aligned}
\text{Now, } d_{kl}(\mathfrak{g}+\langle ij\rangle) &= d_{ki}(\mathfrak{g}+\langle ij\rangle)+d_{ij}(\mathfrak{g}+\langle ij\rangle)+d_{jl}(\mathfrak{g}+\langle ij\rangle)\\
&= d_{ki}(\mathfrak{g})+d_{ij}(\mathfrak{g}+\langle ij\rangle)+d_{jl}(\mathfrak{g})\\
&< d_{ki}(\mathfrak{g})+d_{ij}(\mathfrak{g})+d_{jl}(\mathfrak{g})\\
&= d_{kl}(\mathfrak{g}).
\end{aligned}$$

It follows that every shortest path between k and l in $\mathfrak{g}+\langle ij\rangle$ contains $\langle ij\rangle$. (Note that if there exists a shortest path from k to l in $\mathfrak{g}+\langle ij\rangle$ that does not contain $\langle ij\rangle$, then this shortest path exists in $\mathfrak{g}$ too).

Conversely, let $l \in \mathfrak{g}$ such that $d_{kl}(\mathfrak{g}) \geq 3$ and all shortest paths $\mathscr{P}_{kl}(\mathfrak{g}+\langle ij\rangle)$ from k to l in $\mathfrak{g}+\langle ij\rangle$ contain $\langle ij\rangle$.

Clearly, $\Phi_k(\mathfrak{g}) \leq \Phi_k(\mathfrak{g}+\langle ij\rangle)$.

If possible, let $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g}+\langle ij\rangle)$. This means for every l in $\mathfrak{g}$ there exists a shortest path from k to l in $\mathfrak{g}+\langle ij\rangle$ that does not contain $\langle ij\rangle$, a contradiction. Therefore, $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$. □

Theorem 5.2. *Suppose $\langle ij\rangle \in \mathfrak{g}$, and let k be an agent distinct from i and j. Then, $\Phi_k(\mathfrak{g}-\langle ij\rangle) < \Phi_k(\mathfrak{g})$ if and only if there exists at least one agent $l \in \mathfrak{g}$ such that $d_{kl}(\mathfrak{g}) \geq 2$ and all shortest paths $\mathscr{P}_{kl}(\mathfrak{g})$ from k to l in $\mathfrak{g}$ contain $\langle ij\rangle$.*

We skip the proof as it is similar to the proof of Theorem 5.1.

5.1.1.3 Effect of Closeness on Distances of Agents not Involved in Link Alteration

In this section, we classify agents whose mutual distances from each other remain the same after link alteration. We use the same to analyse the effect of closeness on distances between agents who are not involved in the link addition or deletion.

Given k such that $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$, we use L_k^+ to denote the set of all $l \in \mathfrak{g}$ such that all shortest paths from k to l in $\mathfrak{g}+\langle ij\rangle$ contain $\langle ij\rangle$. We use l_k^+ to denote an agent in L_k^+.

Proposition 5.1. *Suppose i, j, and k are distinct agents in $\mathfrak{g}$. Suppose l is another agent, distinct from i and k, and suppose $\Phi_k(\mathfrak{g}+\langle ij\rangle) > \Phi_k(\mathfrak{g})$. If $d_{ki}(\mathfrak{g}+\langle ij\rangle) < d_{kj}(\mathfrak{g}+\langle ij\rangle) \leq d_{kl}(\mathfrak{g}+\langle ij\rangle)$, then $d_{ik}(\mathfrak{g}+\langle ij\rangle) = d_{ik}(\mathfrak{g})$.*

Proof. We have $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$. Then, from Theorem 5.1, there exists $l \in \mathfrak{g}$ such that all shortest paths $\mathscr{P}_{kl}(\mathfrak{g}+\langle ij\rangle)$ from k to l in $\mathfrak{g}+\langle ij\rangle$ contain $\langle ij\rangle$.

We consider the two cases $j = l$ and $j \neq l$.

1. Suppose $j = l$. As $d_{ki}(\mathfrak{g}+\langle ij\rangle) < d_{kj}(\mathfrak{g}+\langle ij\rangle)$, k observes i before j on all shortest paths $\mathscr{P}_{kl}(\mathfrak{g}+\langle ij\rangle)$. This implies $d_{ik}(\mathfrak{g}+\langle ij\rangle) = d_{ik}(\mathfrak{g})$.

2. Suppose $j \neq l$. As $d_{ki}(\mathfrak{g}+\langle ij\rangle) < d_{kj}(\mathfrak{g}+\langle ij\rangle) \leq d_{kl}(\mathfrak{g}+\langle ij\rangle)$, k observes i before j, and j before l, on all shortest paths $\mathscr{P}_{kl}(\mathfrak{g}+\langle ij\rangle)$. This implies $d_{ik}(\mathfrak{g}+\langle ij\rangle) = d_{ik}(\mathfrak{g})$. □

Definition 5.2. *Suppose* $\langle ij\rangle \notin \mathfrak{g}$ *and* k *is an agent such that* $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$. *A* $(k,+ij)$*-shortest-path-network,* $\mathfrak{g}_{ij}^{k+}$, *is a subnetwork of* $\mathfrak{g}+\langle ij\rangle$ *that consists of all shortest paths from* k *to* l_k^+ *in* $\mathfrak{g}+\langle ij\rangle$, *which contain* $\langle ij\rangle$, *for all* $l_k^+ \in L_k^+$.

Definition 5.3. *An* $(\text{all } k,+ij)$*-shortest-path-network,* $\mathfrak{g}_{ij}^+$, *is* $\bigcup_{\substack{k\in\mathfrak{g},\\ \Phi_k(\mathfrak{g})<\Phi_k(\mathfrak{g}+\langle ij\rangle)}} \mathfrak{g}_{ij}^{k+}$, *the smallest network consisting of all* $(k,+ij)$*-shortest-path-networks.*

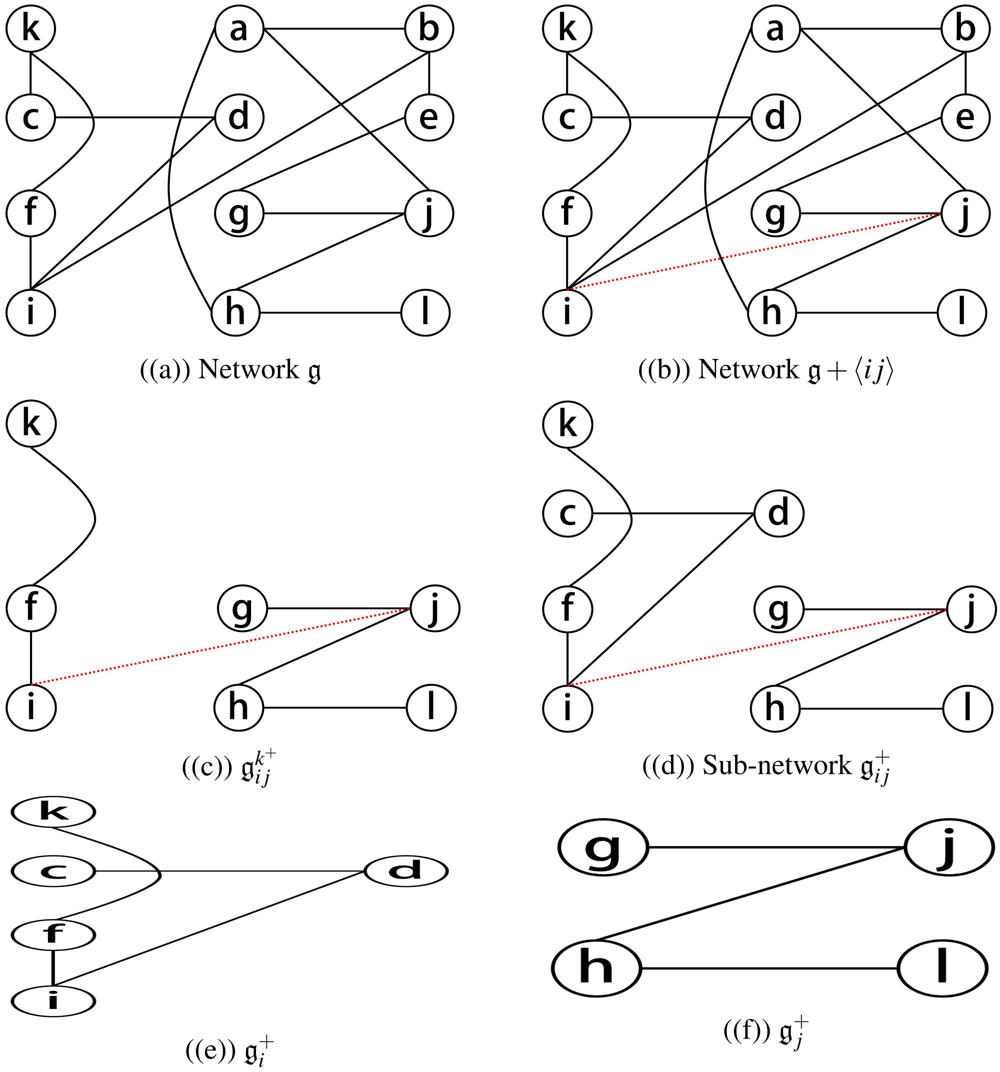
Definition 5.4. *A sub-*$(i,+)$*-network,* $\mathfrak{g}_i^+$ *of* $\mathfrak{g}_{ij}^+$, *is the induced subnetwork of* $\mathfrak{g}_{ij}^+$ *consisting of all agents* $k \in \mathfrak{g}_{ij}^+$ *such that* $d_{ik}(\mathfrak{g}) = d_{ik}(\mathfrak{g}+\langle ij\rangle)$. *Similarly, we define the sub-*$(j,+)$*-network,* $\mathfrak{g}_j^+$ *of* $\mathfrak{g}_{ij}^+$, *as the induced subnetwork of* $\mathfrak{g}_{ij}^+$ *consisting of all agents* $l \in \mathfrak{g}_{ij}^+$ *such that* $d_{jl}(\mathfrak{g}) = d_{jl}(\mathfrak{g}+\langle ij\rangle)$.

The following example illustrates the above.

Example 5.1. *Consider networks* $\mathfrak{g}$ *and* $\mathfrak{g}+\langle ij\rangle$, *as shown in Figures 5.1(a) and 5.1(b), respectively. The newly added link* $\langle ij\rangle$ *in* $\mathfrak{g}$ *increases the closeness of agents* $k,c,d,f,g,j,i,h,$ *and* l *in* $\mathfrak{g}+\langle ij\rangle$. *For instance, from Equation (3.5), we have* $\Phi_k(\mathfrak{g}) = 4.60$ *and* $\Phi_k(\mathfrak{g}+\langle ij\rangle) = 4.87$. *However, there is no change in the closeness of agents* $a,b,$ *and* e. *For instance,* $\Phi_a(\mathfrak{g}) = 6.17 = \Phi_a(\mathfrak{g}+\langle ij\rangle)$. *Figure 5.1(c) shows the shortest-path-network* $\mathfrak{g}_{ij}^{k+}$ *for agent* k, *where the set* L_k^+ *consists of agents* $j,g,h,$ *and* l. *This suggests that the newly added link* $\langle ij\rangle$ *in* $\mathfrak{g}$ *brings agents* $j,g,h,$ *and* l *close to* k *in* $\mathfrak{g}+\langle ij\rangle$, *and therefore,* $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$. *Figure 5.1(d) represents the shortest-path-network* $\mathfrak{g}_{ij}^+$ *that satisfies Definition 5.3. In this case,* $\mathfrak{g}_{ij}^+$ *is the union of* $\mathfrak{g}_{ij}^{k+}, \mathfrak{g}_{ij}^{c+}, \mathfrak{g}_{ij}^{d+}, \mathfrak{g}_{ij}^{f+}, \mathfrak{g}_{ij}^{g+}, \mathfrak{g}_{ij}^{h+}, \mathfrak{g}_{ij}^{i+}, \mathfrak{g}_{ij}^{j+}$, *and* $\mathfrak{g}_{ij}^{l+}$.

Figures 5.1(e) and 5.1(f) show the induced subnetworks $\mathfrak{g}_i^+$ *and* $\mathfrak{g}_j^+$ *of* $\mathfrak{g}_{ij}^+$, *respectively. The induced subnetworks* $\mathfrak{g}_i^+$ *and* $\mathfrak{g}_j^+$ *are as per Definition 5.4. In* $\mathfrak{g}_i^+$, *the distances between agent* i, *and other agents* $k,c,d,$ *and* f *are the same in networks* $\mathfrak{g}$ *and* $\mathfrak{g}+\langle ij\rangle$. *We have* $d_{ik}(\mathfrak{g}) = d_{ik}(\mathfrak{g}+\langle ij\rangle) = 2$, $d_{ic}(\mathfrak{g}) = d_{ic}(\mathfrak{g}+\langle ij\rangle) = 2$, $d_{id}(\mathfrak{g}) = d_{id}(\mathfrak{g}+\langle ij\rangle) = 1$, *and* $d_{if}(\mathfrak{g}) = d_{if}(\mathfrak{g}+\langle ij\rangle) = 1$. *Similarly, in* $\mathfrak{g}_j^+$, *the distances between agent* j, *and other agents* $g,h,$ *and* l *are the same in network* $\mathfrak{g}$ *and* $\mathfrak{g}+\langle ij\rangle$; $d_{jg}(\mathfrak{g}) = d_{jg}(\mathfrak{g}+\langle ij\rangle) = 1$, $d_{jh}(\mathfrak{g}) = d_{jh}(\mathfrak{g}+\langle ij\rangle) = 1$, *and* $d_{jl}(\mathfrak{g}) = d_{jl}(\mathfrak{g}+\langle ij\rangle) = 2$.

Proposition 5.2. *For all* $k,\bar{k} \in \mathfrak{g}_i^+$, $d_{k\bar{k}}(\mathfrak{g}) = d_{k\bar{k}}(\mathfrak{g}+\langle ij\rangle)$.



((a)) Network $\mathfrak{g}$ ((b)) Network $\mathfrak{g} + \langle ij \rangle$

((c)) $\mathfrak{g}_{ij}^{k+}$ ((d)) Sub-network $\mathfrak{g}_{ij}^{+}$

((e)) $\mathfrak{g}_{i}^{+}$ ((f)) $\mathfrak{g}_{j}^{+}$

Figure 5.1: Induced subnetworks of $\mathfrak{g} + \langle ij \rangle$

Proof. If $k, \bar{k} \in \mathfrak{g}_i^+$ then, from Definition 5.4, $d_{ik}(\mathfrak{g}) = d_{ik}(\mathfrak{g} + \langle ij \rangle)$ and $d_{i\bar{k}}(\mathfrak{g}) = d_{i\bar{k}}(\mathfrak{g} + \langle ij \rangle)$. As $k, \bar{k} \in \mathfrak{g}_{ij}^+$ as well, there exists l and $\bar{l}$ such that $d_{kl}(\mathfrak{g}) > d_{kl}(\mathfrak{g} + \langle ij \rangle)$ and $d_{\bar{k}\bar{l}}(\mathfrak{g}) > d_{\bar{k}\bar{l}}(\mathfrak{g} + \langle ij \rangle)$.

It is sufficient to show that, given $\bar{k}$, $\bar{l}$ can never be k.

If possible, let $\bar{l} = k$. Then, from Definition 5.4, $d_{\bar{k}i}(\mathfrak{g}) = d_{\bar{k}i}(\mathfrak{g} + \langle ij \rangle)$ implies $\bar{k}$ observes i first, and subsequently j to reach k, on all shortest paths $\mathscr{P}_{\bar{k}k}(\mathfrak{g} + \langle ij \rangle)$ from $\bar{k}$ to k in $\mathfrak{g} + \langle ij \rangle$. Then, $d_{ik}(\mathfrak{g}) \neq d_{ik}(\mathfrak{g} + \langle ij \rangle)$. This is because, if $k = j$, $d_{ik}(\mathfrak{g}) < d_{ik}(\mathfrak{g} + \langle ij \rangle = 1$. Therefore, $k \notin \mathfrak{g}_i^+$, which is a contradiction. Now, if $k \neq j$, then k must first visit j, and later i, to reach $\bar{k}$ on all shortest paths $\mathscr{P}_{k\bar{k}}(\mathfrak{g} + \langle ij \rangle)$ from $\bar{k}$ to k. This implies $d_{ik}(\mathfrak{g}) \neq d_{ik}(\mathfrak{g} + \langle ij \rangle)$

and hence, $k \notin \mathfrak{g}_i^{+}$, again, a contradiction. Thus, $k \neq \bar{l}$. □

We discuss our results on shortest distances due to link deletion.

5.1.1.4 Effect of Closeness on Distances of Agents not Involved in Link Deletion

Given k such that $\Phi_k(\mathfrak{g}) > \Phi_k(\mathfrak{g} - \langle ij \rangle)$, we use L_k^- to denote the set of all $l \in \mathfrak{g}$ such that all shortest paths from k to l in $\mathfrak{g} - \langle ij \rangle$ contain $\langle ij \rangle$. We use l_k^- to denote an agent in L_k^-.

Proposition 5.3. *Let $i \neq j \neq k$, $l \neq k$, $i \neq l$, and $\Phi_k(\mathfrak{g} - \langle ij \rangle) < \Phi_k(\mathfrak{g})$. If $d_{ik}(\mathfrak{g}) < d_{ij}(\mathfrak{g}) \leq d_{kl}(\mathfrak{g})$, then $d_{ik}(\mathfrak{g} - \langle ij \rangle) = d_{ik}(\mathfrak{g})$.*

We skip the proof as it is similar to the proof of Proposition 5.1.

Definition 5.5. *Suppose $\langle ij \rangle \in \mathfrak{g}$ and k is an agent such that $\Phi_k(\mathfrak{g}) > \Phi_k(\mathfrak{g} - \langle ij \rangle)$. A $(k, -ij)$-shortest-path-network containing $\langle ij \rangle$, $\mathfrak{g}_{ij}^{k-}$, is a subnetwork of $\mathfrak{g}$ that consists of all shortest paths from k to l_k^- in $\mathfrak{g}$, which contain $\langle ij \rangle$, for all $l_k^- \in L_k^-$.*

Definition 5.6. *An $(all\ k, -ij)$-shortest-path-network, $\mathfrak{g}_{ij}^-$ is $\bigcup_{\substack{k \in \mathfrak{g}, \\ \Phi_k(\mathfrak{g}) > \Phi_k(\mathfrak{g} - \langle ij \rangle)}} \mathfrak{g}_{ij}^{k-}$, the smallest network that contains all $(k, -ij)$-shortest-path-networks.*

Definition 5.7. *A sub-$(i, -)$-network, $\mathfrak{g}_i^-$ of $\mathfrak{g}_{ij}^-$, is the induced subnetwork of $\mathfrak{g}_{ij}^-$ consisting of all agents $k \in \mathfrak{g}_{ij}^-$ such that $d_{ik}(\mathfrak{g}) = d_{ik}(\mathfrak{g} - \langle ij \rangle)$. Similarly, we define the sub-$(j, -)$-network, $\mathfrak{g}_j^-$ of $\mathfrak{g}_{ij}^-$, as the induced subnetwork of $\mathfrak{g}_{ij}^-$ consisting of all agents $l \in \mathfrak{g}_{ij}^-$ such that $d_{jl}(\mathfrak{g}) = d_{jl}(\mathfrak{g} - \langle ij \rangle)$.*

Proposition 5.4. *Let $k, \hat{k} \in \mathfrak{g}_i^-$. Then, for all $k, \hat{k} \in \mathfrak{g}_i^-$, $d_{k\hat{k}}(\mathfrak{g}) = d_{k\hat{k}}(\mathfrak{g} - \langle ij \rangle)$.*

We skip the proof as it is similar to the proof of Proposition 5.2.

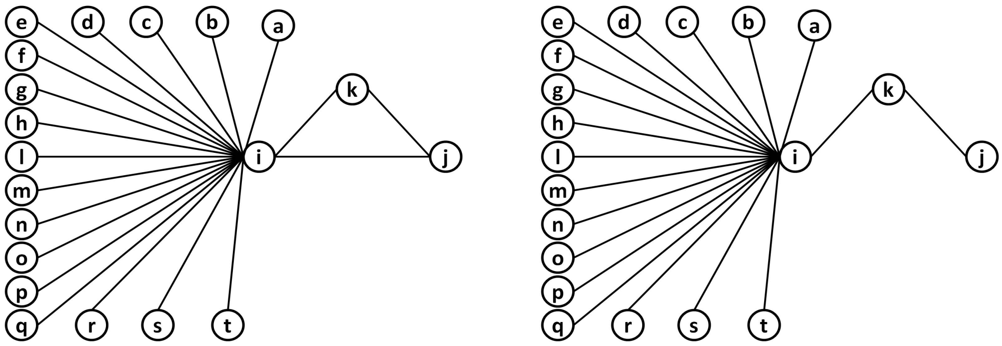
5.1.2 Storage Availability

In this section, we study the effect of link addition and deletion on the storage space availability of agents who are involved in these actions. We expand on our motivation with the following example.

Example 5.2. *We discuss the link deletion case as well as the link addition case, below.*

1. *Consider the network $\mathfrak{g}$ as shown in Figure 5.2(a). If agent i decides to delete the existing link with agent j, we have the network $\mathfrak{g} - \langle ij \rangle$ as shown in Figure 5.2(b). The storage space availability of each agent in these networks, $\mathfrak{g}$ and $\mathfrak{g} - \langle ij \rangle$, are tabulated in Table 5.1 under "Link Deletion". This table shows that agent i benefits by deleting an existing link with j as its storage space availability increases by* 0.00058. *However, agent j's storage space availability decreases by* 0.13142.

2. *To understand that link addition is not a profitable deal for an agent, we reverse the above situation. That is, we have network $\mathfrak{g}'$ as shown in Figure 5.2(b). Now, if agent i decides to add a direct link with agent j, we have network $\mathfrak{g}' + \langle ij \rangle$, as shown in Figure 5.2(a), as a result. The storage space availability of each agent in these networks $\mathfrak{g}$ and $\mathfrak{g} - \langle ij \rangle$ are tabulated in Table 5.1 under "Link Addition". The table shows that agent i does not benefit by adding a direct link with j as its storage space availability decreases by* 0.00058. *However, agent j's storage space availability increases by* 0.13142.



((a)) Network $\mathfrak{g} = \mathfrak{g}' + \langle ij \rangle$ ((b)) Network $\mathfrak{g}' = \mathfrak{g} - \langle ij \rangle$

Figure 5.2: Link additin/ deletion and global resource availability

The above example motivates us to analyse under what conditions agents' chance of obtaining storage space in the network increases or decreases by adding a new link. Then, we present our results in the case of link deletion.

	Link Deletion			Link Addition		
Agent	$\gamma_i(\mathfrak{g})$	$\gamma_i(\mathfrak{g}-\langle ij\rangle)$	$\gamma_i(\mathfrak{g}-\langle ij\rangle)-\gamma_i(\mathfrak{g})$	$\gamma_i(\mathfrak{g}')$	$\gamma_i(\mathfrak{g}'+\langle ij\rangle)$	$\gamma_i(\mathfrak{g}'+\langle ij\rangle)-\gamma_i(\mathfrak{g}')$
a	0.62729	0.62180	0.00549	0.62180	0.62729	-0.00549
. . .	. . .	. . .	. . .	. . .	. . .	. . .
t	0.62729	0.62180	0.00549	0.62180	0.62729	-0.00549
i	0.86406	0.86348	0.00058	0.86348	0.86406	-0.00058
k	0.66479	0.64161	0.02318	0.64161	0.66479	-0.02318
j	0.51019	0.64161	-0.13142	0.64161	0.51019	0.13142

Table 5.1: Link deletion and addition, and storage space availability

5.1.2.1 Effect of Link Addition on Storage Availability

First, we discuss the observations related link addition and its effect on the storage availability of those agents who are involved in the link addition.

Lemma 5.5. *Suppose agents i and j add a direct link in $\mathfrak{g}$ and let $k \notin \mathfrak{g}_{ij}^{+}$. Then, $\alpha_{ik}(\mathfrak{g}) = \alpha_{ik}(\mathfrak{g}+\langle ij\rangle)$ and $\alpha_{jk}(\mathfrak{g}) = \alpha_{jk}(\mathfrak{g}+\langle ij\rangle)$.*

Proof. If agent $k \notin \mathfrak{g}_{ij}^{+}$ then $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g}+\langle ij\rangle)$. Thus, $d_{ki}(\mathfrak{g}) = d_{ki}(\mathfrak{g}+\langle ij\rangle)$. Therefore, from Equation (3.6), $\alpha_{ik}(\mathfrak{g}) = \alpha_{ik}(\mathfrak{g}+\langle ij\rangle)$. A similar proof holds for j too. □

Lemma 5.6. *Suppose agents i, j, k, and l are such that $i \neq j$, $j \neq k$, $i \neq l$, and $k \neq l$. (Agents i and k may be the same, and agents j and l may be the same). Suppose $\langle ij\rangle \notin \mathfrak{g}$, $k \in \mathfrak{g}_i^{+}$, and $l \in \mathfrak{g}_j^{+}$. Then,*

1. *$\alpha_{kl}(\mathfrak{g}) < \alpha_{kl}(\mathfrak{g}+\langle ij\rangle)$, and*
2. *$i \neq k$ implies that $\alpha_{ik}(\mathfrak{g}) > \alpha_{ik}(\mathfrak{g}+\langle ij\rangle)$. Similarly, if $j \neq l$, then $\alpha_{jl}(\mathfrak{g}) > \alpha_{jl}(\mathfrak{g}+\langle ij\rangle)$.*

Proof. Proof of 2 follows from Proposition 5.1.

Proof of 1 is as follows.

As $k \in \mathfrak{g}_i^{+}$ and $l \in \mathfrak{g}_j^{+}$, $d_{kl}(\mathfrak{g}) > d_{kl}(\mathfrak{g}+\langle ij\rangle)$, hence, $\Phi_k(\mathfrak{g}) < \Phi_k(\mathfrak{g}+\langle ij\rangle)$ and $\Phi_l(\mathfrak{g}) < \Phi_l(\mathfrak{g}+\langle ij\rangle)$.

It is easy to see that network $\mathfrak{g}$ in Figure 5.3a is the one where adding link $\langle ij\rangle$ leads to the maximum increment in l' s closeness and the minimum decrements in the distances between l and $k_m, (m = 1,2\ldots,n-4$, where n is the number of agents in $\mathfrak{g}$), the maximum and the minimum being across all network structures.

We consider two cases, $j \neq l$ and $j = l$.

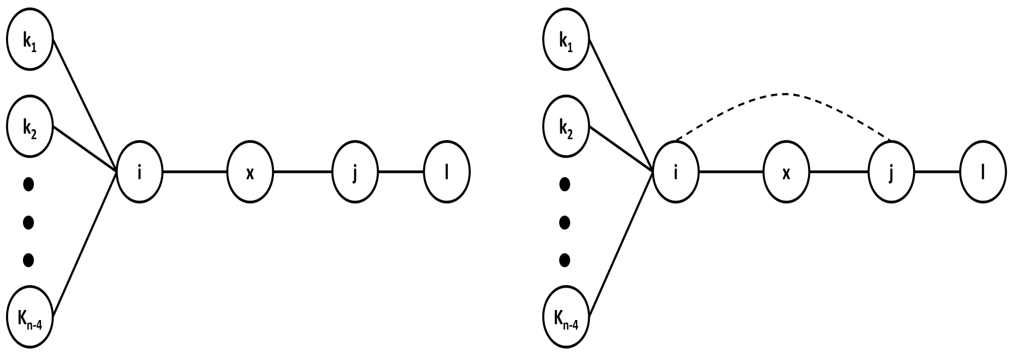
Suppose $j \neq l$. Consider the network $\mathfrak{g}$, as shown in Figure 5.3a.

From Equation (3.5), $\Phi_l(\mathfrak{g}) = \frac{1}{d_{lj}(\mathfrak{g})} + \frac{1}{d_{lx}(\mathfrak{g})} + \frac{1}{d_{li}(\mathfrak{g})} + \sum_{\substack{k\in\mathfrak{g},\\ d_{kl}(\mathfrak{g})=4}} \frac{1}{d_{kl}(\mathfrak{g})}$

$$= 1 + \tfrac{1}{2} + \tfrac{1}{3} + \tfrac{n-4}{4} = \tfrac{3n+10}{12}.$$

Without loss of generality, let $k = k_m$ for some $m \in \{1,2,\ldots,n-4\}$. Then, from Equation (3.6), $\alpha_{kl}(\mathfrak{g}) = \frac{(\frac{1}{4})}{\Phi_l(\mathfrak{g})} = \frac{3}{3n+10}$.

If agents i and j add a direct link in $\mathfrak{g}$, we have network $\mathfrak{g} + \langle ij \rangle$, as shown in Figure 5.3b. Then, from Equations (3.5) and (3.6), we have $\Phi_l(\mathfrak{g} + \langle ij \rangle) = \frac{n+2}{3}$ and $\alpha_{kl}(\mathfrak{g} + \langle ij \rangle) = \frac{1}{n+2}$. From the above, clearly, $\alpha_{kl}(\mathfrak{g}) < \alpha_{kl}(\mathfrak{g} + \langle ij \rangle)$.



((a)) $\mathfrak{g}$ ((b)) $\mathfrak{g} + \langle ij \rangle$

Figure 5.3: Network structure $\mathfrak{g}$ and $\mathfrak{g} + \langle ij \rangle$ with n agents

Now, suppose $l = j$.

From Equations (3.5) and (3.6) applied to Figure 5.3a,b, we have $\Phi_j(\mathfrak{g}) = \frac{2n+7}{6}$, $\alpha_{kj}(\mathfrak{g}) = \frac{2}{2n+7}$, $\Phi_j(\mathfrak{g} + \langle ij \rangle) = \frac{n+2}{2}$, and $\alpha_{kj}(\mathfrak{g} + \langle ij \rangle) = \frac{1}{n+2}$.

It is easy see that $\alpha_{kj}(\mathfrak{g}) < \alpha_{kj}(\mathfrak{g} + \langle ij \rangle)$ in this case as well. This completes the proof of 1. □

Lemma 5.7. *Let k and $\bar{k}$ be agents in $\mathfrak{g}_i^+$. Then, $\alpha_{k\bar{k}}(\mathfrak{g}) = \alpha_{k\bar{k}}(\mathfrak{g} + \langle ij \rangle)$ and $\alpha_{\bar{k}k}(\mathfrak{g}) = \alpha_{\bar{k}k}(\mathfrak{g} + \langle ij \rangle)$.*

Proof. The proof follows from Proposition 5.2. □

Theorem 5.3. *Suppose agents i and j are such $i \neq j$, and $\langle ij \rangle \notin \mathfrak{g}$. Then, $\gamma_i(\mathfrak{g}) < \gamma_i(\mathfrak{g} + \langle ij \rangle)$ if and only if* $\frac{\prod_{k \in \mathfrak{g}_i^+} (1-\alpha_{ik}(\mathfrak{g}+\langle ij \rangle))}{\prod_{l \in \mathfrak{g}_j^+} (1-\alpha_{il}(\mathfrak{g}))} < \frac{\prod_{k \in \mathfrak{g}_i^+} (1-\alpha_{ik}(\mathfrak{g}))}{\prod_{l \in \mathfrak{g}_j^+} (1-\alpha_{il}(\mathfrak{g}+\langle ij \rangle))}$.

Additionally, $\gamma_i(\mathfrak{g}) < \gamma_i(\mathfrak{g} + \langle ij \rangle)$ if and only if $\frac{\prod_{k \in \mathfrak{g}_i^+} (\alpha_{ik}(\mathfrak{g}+\langle ij \rangle))}{\prod_{l \in \mathfrak{g}_j^+} (\alpha_{il}(\mathfrak{g}))} > \frac{\prod_{k \in \mathfrak{g}_i^+} (\alpha_{ik}(\mathfrak{g}))}{\prod_{l \in \mathfrak{g}_j^+} (\alpha_{il}(\mathfrak{g}+\langle ij \rangle))}$.

Proof. The proof follows from Lemmas 5.5, 5.6, and 5.7. □

5.1.2.2 Effect of Link Deletion on Storage Availability

We discuss the effect of link deletion on agents' storage availability who are in the action.

Lemma 5.8. *Suppose $\langle ij \rangle \in \mathfrak{g}$ and $k \notin \mathfrak{g}_{ij}^-$. Then, $\alpha_{ik}(\mathfrak{g}) = \alpha_{ik}(\mathfrak{g} - \langle ij \rangle)$ and $\alpha_{jk}(\mathfrak{g}) = \alpha_{jk}(\mathfrak{g} - \langle ij \rangle)$.*

Proof. The proof follows from Proposition 5.3 and is in similar lines to the proof of Lemma 5.5. □

Lemma 5.9. *Suppose agents i, j, k, and l are such that $i \neq j$, $j \neq k$, $i \neq l$, and $k \neq l$. (Agents i and k may be the same, and agents j and l may be the same). Suppose $\langle ij \rangle \in \mathfrak{g}$, $k \in \mathfrak{g}_i^-$, and $l \in \mathfrak{g}_j^-$. Then,*

1. *$\alpha_{kl}(\mathfrak{g}) > \alpha_{kl}(\mathfrak{g} - \langle ij \rangle)$.*

2. *If $i \neq k$, then $\alpha_{ik}(\mathfrak{g}) < \alpha_{ik}(\mathfrak{g} - \langle ij \rangle)$. Similarly, if $j \neq l$, then $\alpha_{jl}(\mathfrak{g}) < \alpha_{jl}(\mathfrak{g} - \langle ij \rangle)$.*

Proof. The proof of 1 is in similar lines to the proof of 1 of Lemma 5.6. The proof of 2 follows from Proposition 5.4. □

Lemma 5.10. *Let k and $\hat{k}$ be agents in $\mathfrak{g}_i^-$. Then, $\alpha_{k\hat{k}}(\mathfrak{g}) = \alpha_{k\hat{k}}(\mathfrak{g} - \langle ij \rangle)$ and $\alpha_{\hat{k}k}(\mathfrak{g}) = \alpha_{\hat{k}k}(\mathfrak{g} - \langle ij \rangle)$.*

Proof. The proof follows from Lemma 5.4. □

Theorem 5.4. *Suppose agents i, j, k, and l are such that $i \neq j$ and $k \neq l$. Suppose $\langle ij \rangle \in \mathfrak{g}$. Then,* $\gamma_i(\mathfrak{g}) < \gamma_i(\mathfrak{g} - \langle ij \rangle)$ *if and only if* $\frac{\prod_{l \in \mathfrak{g}_j^-} (1-\alpha_{ik}(\mathfrak{g}))}{\prod_{l \in \mathfrak{g}_j^-} (1-\alpha_{il}(\mathfrak{g}-\langle ij \rangle))} < \frac{\prod_{k \in \mathfrak{g}_i^-} (1-\alpha_{ik}(\mathfrak{g}-\langle ij \rangle))}{\prod_{k \in \mathfrak{g}_i^-} (1-\alpha_{il}(\mathfrak{g}))}$.

In addition, $\gamma_i(\mathfrak{g}) < \gamma_i(\mathfrak{g} - \langle ij \rangle)$ *if and only if* $\frac{\prod_{l \in \mathfrak{g}_j^-} (\alpha_{ik}(\mathfrak{g}))}{\prod_{l \in \mathfrak{g}_j^-} (\alpha_{il}(\mathfrak{g}-\langle ij \rangle))} > \frac{\prod_{k \in \mathfrak{g}_i^-} (\alpha_{ik}(\mathfrak{g}-\langle ij \rangle))}{\prod_{k \in \mathfrak{g}_i^-} (\alpha_{il}(\mathfrak{g}))}$.

Proof. The proof follows from Lemmas 5.8, 5.9, and 5.10. □

5.1.3 Externalities

In this section, we study externalities, that is, how a link that is added between a pair of agents affects the utility of others. (Refer to Definition 5.8). The particular form of externalities (positive, negative, or none) is crucial in determining which network is likely to evolve and the conditions under which it will lead to a stable and efficient network.

Definition 5.8. *[61] Consider a network,* $\mathfrak{g}$, *with agents* $i, j \in \mathfrak{g}$ *such that* $i \neq j$ *and* $\langle ij \rangle \notin \mathfrak{g}$. *Suppose agents i and j form a direct link* $\langle ij \rangle$. *Then, agent* $k \in \mathfrak{g} \setminus \{i, j\}$ *experiences*

1. *Positive externalities if* $u_k(\mathfrak{g} + \langle ij \rangle) > u_k(\mathfrak{g})$*;*
2. *Negative externalities if* $u_k(\mathfrak{g} + \langle ij \rangle) < u_k(\mathfrak{g})$*;*
3. *No externalities if* $u_k(\mathfrak{g} + \langle ij \rangle) = u_k(\mathfrak{g})$.

We now show that the type of externalities an agent $k \in \mathfrak{g}$ experiences, can be determined using conditions on the storage availability, independent of the data loss rate and the value that agents associate with their data.

Proposition 5.5. *In an SSSC* $\mathfrak{g}$, *an agent* $k \in \mathfrak{g}$ *experiences*

1. *Positive externalities if* $\gamma_k(\mathfrak{g} + \langle ij \rangle) > \gamma_k(\mathfrak{g})$*;*
2. *Negative externalities if* $\gamma_k(\mathfrak{g} + \langle ij \rangle) < \gamma_k(\mathfrak{g})$*;*
3. *No externalities if* $\gamma_k(\mathfrak{g} + \langle ij \rangle) = \gamma_k(\mathfrak{g})$.

Proof.

1. By Definition 5.8, $u_k(\mathfrak{g}+\langle ij\rangle) > u_k(\mathfrak{g})$

   $\Rightarrow \beta(1-\delta)+\beta\delta\gamma_k(\mathfrak{g}+\langle ij\rangle) - \varsigma\eta_k(\mathfrak{g}+\langle ij\rangle) > \beta(1-\delta)+\beta\delta\gamma_k(\mathfrak{g}) - \varsigma\eta_k(\mathfrak{g})$.

   As agent k does not pay the cost for link $\langle ij\rangle$, we have $\varsigma\eta_k(\mathfrak{g}+\langle ij\rangle) = \varsigma\eta_k(\mathfrak{g})$.

   Thus, $\beta\delta\gamma_k(\mathfrak{g}+\langle ij\rangle) > \beta\delta\gamma_k(\mathfrak{g}) \Rightarrow \gamma_k(\mathfrak{g}+\langle ij\rangle) > \gamma_k(\mathfrak{g})$.

2. For Cases 2 and 3, the proof is similar to that of Case 1. □

Proposition 5.6. *Let i, j, and k be distinct agents in* $\mathfrak{g}$. *Suppose* $k \notin \mathfrak{g}^+_{ij}$. *Then, k experiences only negative externalities.*

Proof. If agents i and j add a direct link in $\mathfrak{g}$, then, from Lemma 5.1, $\Phi_i(\mathfrak{g}) < \Phi_i(\mathfrak{g}+\langle ij\rangle)$. If $k \notin \mathfrak{g}^+_{ij}$, then, from Theorem 5.1, $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g}+\langle ij\rangle)$, thus, $d_{kl}(\mathfrak{g}) = d_{kl}(\mathfrak{g}+\langle ij\rangle)$ for all $l \in \mathfrak{g}$. Therefore, $\alpha_{ki}(\mathfrak{g}+\langle ij\rangle) < \alpha_{ki}(\mathfrak{g})$, by Equation (3.6). Now, for all $l \in \mathfrak{g}$, $\Phi_l(\mathfrak{g}) \leq \Phi_l(\mathfrak{g}+\langle ij\rangle)$. If $\Phi_l(\mathfrak{g}) = \Phi_l(\mathfrak{g}+\langle ij\rangle)$, then $\alpha_{kl}(\mathfrak{g}+\langle ij\rangle) = \alpha_{kl}(\mathfrak{g})$ and, if $\Phi_l(\mathfrak{g}) < \Phi_l(\mathfrak{g}+\langle ij\rangle)$, then $\alpha_{kl}(\mathfrak{g}+\langle ij\rangle) < \alpha_{kl}(\mathfrak{g})$. Thus, $\gamma_k(\mathfrak{g}+\langle ij\rangle) < \gamma_k(\mathfrak{g})$. □

Proposition 5.7. *Let i, j, and k be distinct agents in* $\mathfrak{g}$, *such that* $\langle ij\rangle \notin \mathfrak{g}$, *an agent* $k \in \mathfrak{g}\setminus\{i,j\}$ *experiences negative externalities on formation of the link* $\langle ij\rangle$, *if* $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g}+\langle ij\rangle)$.

Proof. Let agent i and j add a direct link in $\mathfrak{g}$. This new link $\langle ij\rangle$ reduces their distance by at least 1, to at most $d_{ij}(\mathfrak{g})-1$ in $\mathfrak{g}+\langle ij\rangle$. Thus, their closeness increases by at least $\frac{d_{ij}(\mathfrak{g})-1}{d_{ij}(\mathfrak{g})}$ in $\mathfrak{g}+\langle ij\rangle$.

Therefore, $\Phi_i(\mathfrak{g}) < \Phi_i(\mathfrak{g}+\langle ij\rangle)$ and $\Phi_j(\mathfrak{g}) < \Phi_j(\mathfrak{g}+\langle ij\rangle)$.

As $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g}+\langle ij\rangle)$, we have $d_{kl}(\mathfrak{g}) = d_{kl}(\mathfrak{g}+\langle ij\rangle)$, for all $l \in \mathfrak{g}$.

We know, $d_{ik}(\mathfrak{g}) = d_{ik}(\mathfrak{g}+\langle ij\rangle)$ and $\Phi_i(\mathfrak{g}) < \Phi_i(\mathfrak{g}+\langle ij\rangle)$ implies $\alpha_{ki}(\mathfrak{g}) > \alpha_{ki}(\mathfrak{g}+\langle ij\rangle)$.

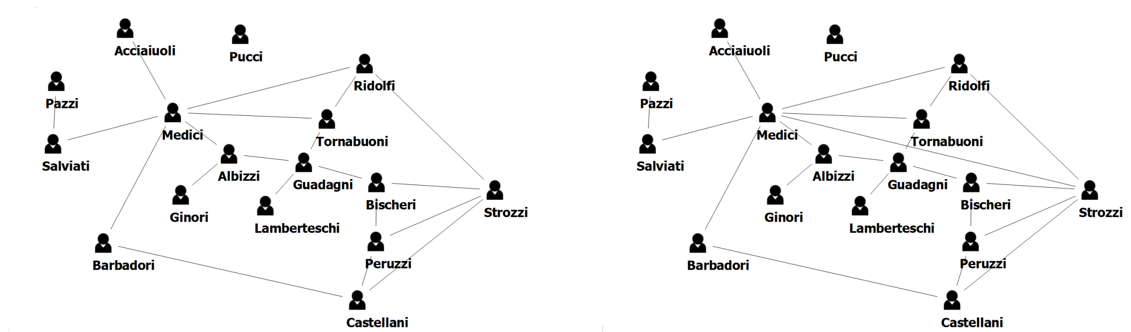
Similarly, we have, $\alpha_{kj}(\mathfrak{g}) > \alpha_{kj}(\mathfrak{g}+\langle ij\rangle)$.

Therefore, $(1-\alpha_{ki}(\mathfrak{g}))(1-\alpha_{ki}(\mathfrak{g})) < (1-\alpha_{ki}(\mathfrak{g}+\langle ij\rangle))(1-\alpha_{ki}(\mathfrak{g}+\langle ij\rangle))$.

This implies $\gamma_k(\mathfrak{g}) > \gamma_k(\mathfrak{g}+\langle ij\rangle)$, as $\Phi_l(\mathfrak{g}) \leq \Phi_l(\mathfrak{g}+\langle ij\rangle)$ for all $l \in \mathfrak{g}$, implying $\alpha_{kl}(\mathfrak{g}) \geq \alpha_{kl}(\mathfrak{g}+\langle ij\rangle)$ for all $l \in \mathfrak{g} \setminus \{i,j\}$. $\square$

Corollary 5.1. *Suppose* $\mathfrak{g}$ *is a two-diameter SSC with distinct agents* i, j, *such that* $\langle ij\rangle \notin \mathfrak{g}$. $\Phi_k(\mathfrak{g}) = \Phi_k(\mathfrak{g}+\langle ij\rangle)$ *for all* $k \in \mathfrak{g} \setminus \{i,j\}$, *and hence, all agents experience only negative externalities if* $\langle ij\rangle)$ *is formed.*

Remark 5.1. *Proposition 5.6 and Proposition 5.7 show that an increase in agents' closeness is necessary for them to experience positive externalities. However, the increase in agents' closeness is not a sufficient condition for positive externalities, as demonstrated by the following example.*



((a)) Network $\mathfrak{g}$ ((b)) Network $\mathfrak{g}+\langle Medici, Strozzi\rangle$

Figure 5.4: Externalities in SCC $\mathfrak{g}$

Example 5.3. *For this example, we consider the Pedgetts's Florentine_Families network [97] (shows business and marital ties of 16 agents) generated by Social Network Visualizer*[29] *tool from its known data set, as shown in Figure 5.4(a). Let us say* $\mathfrak{g}$ *to the generated network as shown in Figure 5.4(a). Suppose Medici and Strozzi add a link in* $\mathfrak{g}$ *resulting in the* $\mathfrak{g}+\langle Medici, Strozzi\rangle$ *(say* $\mathfrak{g}'$*), as shown in Figure 5.4(b). Then from Equation (3.5),* $\Phi_{Albizzi}(\mathfrak{g}) = 7.83$ *and* $\Phi_{Albizzi}(\mathfrak{g}') = 8.00$. *However, from Equation (3.7),* $\gamma_{Albizzi}(\mathfrak{g}) = 0.701$ *and* $\gamma_{Albizzi}(\mathfrak{g}') = 0.696$. *This indicates that, although the closeness of Albizzi is increased*

[29] https://socnetv.org/

$Agent(x)$	$\Phi_x(\mathfrak{g})$	$\Phi_x(\mathfrak{g}+xy)$	$\Phi_x(\mathfrak{g}+xy)-\Phi(\mathfrak{g})$	$\gamma_x(\mathfrak{g})$	$\gamma_x(\mathfrak{g}+xy)$	$\gamma_x(\mathfrak{g}+ij)-\gamma(\mathfrak{g})$
$Acciaiuodi$	5.917	6.250	0.333	0.574	0.580	0.006
$Albizzi$	7.833	8.000	0.167	0.701	0.696	-0.005
$Barbadori$	7.083	7.083	0.000	0.649	0.635	-0.014
$Bischeri$	7.200	7.583	0.383	0.657	0.662	0.005
$Castellani$	6.917	6.917	0.000	0.642	0.625	-0.017
$Ginori$	5.333	5.417	0.083	0.537	0.531	-0.007
$Gaudagni$	8.083	8.083	0.000	0.713	0.702	-0.011
$Lamberteschi$	5.367	5.367	0.000	0.537	0.526	-0.011
$Medici$	9.500	10.333	0.833	0.773	0.788	0.015
$Pazzi$	4.767	4.950	0.183	0.503	0.502	-0.001
$Peruzzi$	6.783	7.167	0.383	0.633	0.639	0.006
$Ridolf$	8.000	8.000	0.000	0.693	0.677	-0.016
$Salviati$	6.583	6.917	0.333	0.641	0.644	0.002
$Strozzi$	7.833	9.000	1.167	0.691	0.729	0.038
$Tornabuoni$	7.833	7.833	0.000	0.685	0.673	-0.012
$Pucci$	0.000	0.000	0.000	0.000	0.000	0.000

Table 5.2: Externalities in SCC $\mathfrak{g}$

in $\mathfrak{g}'$, its storage availability is decreased. The same is true for Ginori and Pazzi. On the contrary, the newly added link between Medici and Strozzi increases not only Bischeri's closeness (i.e.$\Phi_{Bischeri}(\mathfrak{g}) = 7.20$ and $\Phi_{Bischeri}(\mathfrak{g}') = 7.58$), but also storage availability (i.e., $\gamma_{Bischeri}(\mathfrak{g}) = 0.657$ and $\gamma_{Bischeri}(\mathfrak{g}') = 0.662$). We have a similar observation for Acciaiuodi, Peruzzi and Salviati.

——————————————————————————- The following result provides a necessary and sufficient condition under which an agent $k \in \mathfrak{g}$ experiences positive or negative externalities.

Theorem 5.5. *Suppose agents i, j, k, $\bar{k}$, and l are such that $i \neq j$, $i \neq k$, $i \neq l$, $j \neq k$, $k \neq \bar{k}$, and $k \neq l$. (Agents i and $\bar{k}$ may be the same, and agents j and l may be the same). Suppose $\langle ij \rangle \notin \mathfrak{g}$, $\bar{k} \in \mathfrak{g}_i^+$ and $l \in \mathfrak{g}_j^+$. Then, agent k experiences positive externalities if and only if $k \in \mathfrak{g}_{ij}^+$ and $\frac{\prod_{\bar{k}\in\mathfrak{g}_i^+}(1-\alpha_{k\bar{k}}(\mathfrak{g}+\langle ij\rangle))}{\prod_{l\in\mathfrak{g}_l^+}(1-\alpha_{kl}(\mathfrak{g}))} < \frac{\prod_{\bar{k}\in\mathfrak{g}_i^+}(1-\alpha_{k\bar{k}}(\mathfrak{g}))}{\prod_{l\in\mathfrak{g}_j^+}(1-\alpha_{kl}(\mathfrak{g}+\langle ij\rangle))}$, otherwise k experiences negative externalities.*

Proof. From Proposition 5.6, it is required to increment in agent k's closeness. It is straightforward to observe that $\frac{\prod_{\bar{k}\in\mathfrak{g}_i^+}(1-\alpha_{k\bar{k}}(\mathfrak{g}+\langle ij\rangle))}{\prod_{l\in\mathfrak{g}_l^+}(1-\alpha_{kl}(\mathfrak{g}))} < \frac{\prod_{\bar{k}\in\mathfrak{g}_i^+}(1-\alpha_{k\bar{k}}(\mathfrak{g}))}{\prod_{l\in\mathfrak{g}_j^+}(1-\alpha_{kl}(\mathfrak{g}+\langle ij\rangle))}$, then $\gamma_k(\mathfrak{g}+\langle ij\rangle) > \gamma_k(\mathfrak{g})$. Thus, k experiences positive externalities.

Conversely, let $\gamma_k(\mathfrak{g}+\langle ij\rangle) < \gamma_k(\mathfrak{g})$, then either from Proposition 5.6, $d_{ki}(\mathfrak{g}) = d_{ki}(\mathfrak{g}+\langle ij\rangle)$, for all $i \in \mathfrak{g}$ or $\frac{\prod_{\bar{k}\in\mathfrak{g}_i^+}(1-\alpha_{k\bar{k}}(\mathfrak{g}+\langle ij\rangle))}{\prod_{l\in\mathfrak{g}_l^+}(1-\alpha_{kl}(\mathfrak{g}))} < \frac{\prod_{\bar{k}\in\mathfrak{g}_i^+}(1-\alpha_{k\bar{k}}(\mathfrak{g}))}{\prod_{l\in\mathfrak{g}_j^+}(1-\alpha_{kl}(\mathfrak{g}+\langle ij\rangle))}$. □

Although we have provided a necessary and sufficient condition for positive and negative externalities by performing a microscopic analysis of externalities, it is hard to obtain a general characterization of networks where agents experience only positive externalities. This leads us to the following question. At least for specific network structures, can we show positive (or negative) externalities? For instance, we can argue that in a two diameter network, agents never experience positive externalities.

Theorem 5.6. *In any connected SSC $\mathfrak{g}$ with three or more agents, agents experience either positive or negative externalities, and there is no case where any agent experiences no externalities.*

Proof. Consider an SCC $\mathfrak{g}$ with distinct agents i, j and k such that $\langle ij\rangle \notin \mathfrak{g}$. If possible, let k have no externalities when $\langle ij\rangle$ is added. This means $\gamma_k(\mathfrak{g}) = \gamma_k(\mathfrak{g}+\langle ij\rangle)$, by Proposition 5.5.

Therefore, by Equation (3.7), $\alpha_{kl}(\mathfrak{g}) = \alpha_{kl}(\mathfrak{g}+\langle ij\rangle)$, for all $l \in \mathfrak{g}$. (Note that $\alpha_{kl}(\mathfrak{g}) \geq \alpha_{kl}(\mathfrak{g}+\langle ij\rangle)$, for all $l \in \mathfrak{g}$).

This implies that $d_{kl}(\mathfrak{g}) = d_{kl}(\mathfrak{g}+\langle ij\rangle)$ and $\Phi_l(\mathfrak{g}) = \Phi_l(\mathfrak{g}+\langle ij\rangle)$ for all $l \in \mathfrak{g}$, from Equations (3.5) and (3.6).

The link addition between agents i and j in $\mathfrak{g}$ decreases their distance by $d_{ij}(\mathfrak{g})-1$ in $\mathfrak{g}+\langle ij\rangle$, and thus, the closeness of both i and j in $\mathfrak{g}+\langle ij\rangle$ increases by $\frac{1}{d_{ij}(\mathfrak{g})-1}$.

Hence, $\Phi_i(\mathfrak{g}) < \Phi_i(\mathfrak{g}+\langle ij\rangle)$ and $\Phi_j(\mathfrak{g}) < \Phi_j(\mathfrak{g}+\langle ij\rangle)$, a contradiction to our deduction that $\Phi_l(\mathfrak{g}) = \Phi_l(\mathfrak{g} + \langle ij \rangle)$ for all $l \in \mathfrak{g}$.

Therefore, our assumption that k has no externalities is incorrect. In other words, k has either positive or negative externalities. □

Corollary 5.2. *Suppose an SCC $\mathfrak{g}$ is a disconnected network consisting of more than two disjoint components. Let us assume that we have three distinct agents i, j, and k such agent $i \in \mathfrak{g}(\kappa_x)$, $j \in \mathfrak{g}(\kappa_y)$, and $k \in \mathfrak{g}(\kappa_z)$ and agents i and j add a direct link in $\mathfrak{g}$, then $\gamma_k(\mathfrak{g}) = \gamma_k(\mathfrak{g} + \langle ij \rangle)$ for all $k \notin \mathfrak{g}(\kappa_x), \mathfrak{g}(\kappa_y)$.*

5.2 Characterization of Stable and Efficient Networks

One of the central focuses of this study is to analyze what network is likely to emerge when each agent (or pair of agents) decides selfishly which link they want to delete (respectively, whether to add a link or not), when agents build their social connections (links) based on the benefit associated with their data, the cost for link formation, and the prevailing data loss rate.

In the following subsections, we discuss pairwise stable networks, efficient networks, and the measures of efficiency, namely, price of anarchy (PoA) and price of stability (PoS). In our analysis of stable and efficient networks, *we assume that network formation takes place starting with the null network* (where there are no links between any pair of agents).

5.2.1 Stable Networks: Characterization, Existence, and Uniqueness

We now discuss the conditions under which an agent prefers to add a new link or delete an existing link, and use the same to characterize stable networks.

Lemma 5.11. *Let $\langle ij \rangle \notin \mathfrak{g}$. An agent $i \in \mathfrak{g}$ is benefited by adding a direct link with agent $j \in \mathfrak{g}$ if and only if $\beta\delta[\gamma_i(\mathfrak{g} + \langle ij \rangle) - \gamma_i(\mathfrak{g})] > \varsigma$.*

Proof. Agent i has incentive to form a link with agent j if and only if $u_i(\mathfrak{g} + \langle ij \rangle) > u_i(\mathfrak{g})$
$\Rightarrow \beta(1-\delta) + \beta\delta\gamma_i(\mathfrak{g} + \langle ij \rangle) - \varsigma(\eta_i(\mathfrak{g}) + 1) > \beta(1-\delta) + \beta\delta\gamma_i(\mathfrak{g}) - \varsigma\eta_i(\mathfrak{g})$
$\Rightarrow \beta\delta[\gamma_i(\mathfrak{g} + \langle ij \rangle) - \gamma_i(\mathfrak{g})] > \varsigma$. □

Corollary 5.3. *An agent $i \in \mathfrak{g}$ has no incentive to add a link with agent $j \in \mathfrak{g}$ if and only if $\delta[\gamma_i(\mathfrak{g}+\langle ij\rangle)-\gamma_i(\mathfrak{g})] \leq \frac{\varsigma}{\beta}$.*

Lemma 5.12. *Let $\langle ij\rangle \in \mathfrak{g}$. An agent $i \in \mathfrak{g}$ benefits by deleting a link with agent j if and only if $\beta\delta[\gamma_i(\mathfrak{g})-\gamma_i(\mathfrak{g}-\langle ij\rangle)] < \varsigma$.*

Proof. An agent i has incentive to delete a link with agent j if and only if $u_i(\mathfrak{g}-\langle ij\rangle) > u_i(\mathfrak{g})$.
$\Rightarrow \beta(1-\delta)+\beta\delta\gamma_i(\mathfrak{g}-\langle ij\rangle)-\varsigma(\eta_i(\mathfrak{g})-1) > \beta(1-\delta)+\beta\delta\gamma_i(\mathfrak{g})-\varsigma\eta_i(\mathfrak{g})$
$\Rightarrow \varsigma > \beta\delta[\gamma_i(\mathfrak{g})-\gamma_i(\mathfrak{g}-\langle ij\rangle)]$. □

Corollary 5.4. *An agent i has no incentive to delete an existing link with agent j if and only if $\delta[\gamma_i(\mathfrak{g})-\gamma_i(\mathfrak{g}-\langle ij\rangle)] \geq \frac{\varsigma}{\beta}$.*

Proposition 7.1, stated below, provides an easy characterization of a stable network $\mathfrak{g}$.

Proposition 5.8. *A network $\mathfrak{g}$ is pairwise stable if and only if*

1. *for all $i, j \in \mathfrak{g}, \delta[\gamma_i(\mathfrak{g})-\gamma_i(\mathfrak{g}-\langle ij\rangle)] \geq \frac{\varsigma}{\beta}$, and $\delta[\gamma_j(\mathfrak{g})-\gamma_j(\mathfrak{g}-\langle ij\rangle)] \geq \frac{\varsigma}{\beta}$; and*
2. *for all $i, j \in \mathfrak{g}$, if $\delta[\gamma_i(\mathfrak{g}+\langle ij\rangle)-\gamma_i(\mathfrak{g})] > \frac{\varsigma}{\beta}$, then $\delta[\gamma_j(\mathfrak{g}+\langle ij\rangle)-\gamma_j(\mathfrak{g})] < \frac{\varsigma}{\beta}$.*

Proof. The proof follows from Definition 2.2, Corollary 7.2, and Lemma 7.1. □

In the following theorem, we prove existence and uniqueness of pairwise stable networks, given the values of the parameters.

Theorem 5.7. *There always exists a pairwise stable network. Given N, there exists exactly two pairwise stable networks. For each β, ς, and δ,* **the** *pairwise stable network $\mathfrak{g}$ is unique.*

1. *If $\delta \leq \frac{\varsigma}{\beta}$, then $\mathfrak{g}$ is the null network.*
2. *If $\delta > \frac{\varsigma}{\beta}$, then $\mathfrak{g}$ consists of*
   (a) *a set of $\frac{N}{2}$ connected pairs of agents, if N is even; or*
   (b) *a set of $\frac{N-1}{2}$ connected pairs of agents and one isolated agent, if N is odd.*

Proof. Initially, all agents are isolated in $\mathfrak{g}$, hence, for all $i \in \mathfrak{g}$, $\gamma_i(\mathfrak{g}) = 0$.

If agents i and j form a direct link $\langle ij \rangle$, then $\Phi_i(\mathfrak{g} + \langle ij \rangle) = \Phi_j(\mathfrak{g} + \langle ij \rangle) = 1$.

Thus, $\gamma_i(\mathfrak{g} + \langle ij \rangle) = \gamma_j(\mathfrak{g} + \langle ij \rangle) = 1$.

However, from Lemma 7.1, agents i and j benefit by forming a direct link if and only if

$\delta\beta[\gamma_i(\mathfrak{g} + \langle ij \rangle) - \gamma_i(\mathfrak{g})] > \varsigma$ and $\delta\beta[\gamma_j(\mathfrak{g} + \langle ij \rangle) - \gamma_j(\mathfrak{g})] > \varsigma$, respectively.

This implies that a pair of agents have no incentive to add a direct link if and only if $\delta \leq \frac{\varsigma}{\beta}$.

Therefore, $\mathfrak{g}$ is the null network. This completes the proof of 1.

Now, if $\delta > \frac{\varsigma}{\beta}$, then every pair of agents has an incentive to add a direct link. Suppose agents i and j add a direct link, and suppose link $\langle ij \rangle$ is the only link in the network, say $\mathfrak{g}'$. Let k be another agent, different from i and j, in $\mathfrak{g}'$. Then, $\gamma_i(\mathfrak{g}' + \langle ik \rangle) = 1 - (1 - \frac{1}{1.5})^2$.

By Lemma 7.1, agent i benefits by adding the link $\langle ik \rangle$ if and only if $\delta[\gamma_i(\mathfrak{g}' + \langle ik \rangle) - \gamma_i(\mathfrak{g}')] > \frac{\varsigma}{\beta}$. Here, $\delta[\gamma_i(\mathfrak{g}' + \langle ik \rangle) - \gamma_i(\mathfrak{g}')] < 0 \not> \frac{\varsigma}{\beta}$.

This implies that no agent benefits by adding more than one link, proving 2. □

5.2.2 Efficient Network, Price of Anarchy, and Price of Stability

We analyse whether the network formed by self-interested agents is also efficient, that is, socially optimal or in other words, "good" for all the agents put together.

Definition 5.9. *A social storage network* $\mathfrak{g}$ *is efficient with respect to utility profile* $(u_1, \dots, u_N)$ *if*

$$\sum_{i \in N} [\beta(1-\delta) + \beta\delta\gamma_i(\mathfrak{g}) - \varsigma\eta_i(\mathfrak{g})] \geq \sum_{i \in N} [\beta(1-\delta) + \beta\delta\gamma_i(\bar{\mathfrak{g}}) - \varsigma\eta_i(\bar{\mathfrak{g}})], \forall \bar{\mathfrak{g}} \in \mathscr{G}(N).$$

It might be possible that when self-interested agents build their social connections for their own benefit, the resulting network formation will lead to a "bad" outcome from a societal viewpoint. That is, the resulting network may be advantageous for a set of agents, while other agents may not be benefited by the outcome. This results in an inefficient network. In this state of affairs, we would like to measure how far a pairwise stable network is from an efficient network. For this, we make use of the widely discussed measures, namely, price of anarchy (PoA) and price of stability (PoS). We restate these measures as follows.

Definition 5.10. *The price of anarchy (PoA) is the ratio of the worst sum of the utility values of an equilibrium network and the optimal sum of the utility values in any network.*

Definition 5.11. *The price of stability (PoS) is the ratio of the best sum of the utility values of an equilibrium network and the optimal sum of the utility values in any network.*

Theorem 5.8. *Every pairwise stable network is efficient. Therefore, PoA = 1. In addition, every efficient network is pairwise stable. Hence, PoS = 1.*

Proof. The proof follows from Theorem 5.7 and Definition 5.9, the fact that network formation starts with the null network, and the fact ([61]) that PoS = 1 if and only if every efficient network is pairwise stable, and PoA = 1 if and only if all pairwise stable networks are efficient. □

Example 5.4. *Let network* $\mathfrak{g}$ *and* $\bar{\mathfrak{g}}$ *consist of* $N > 2$. *Let in network* $\mathfrak{g}$ *each agent maintains only a single link, whereas in network* $\bar{\mathfrak{g}}$ *each agent maintains relations with the remaining agents. Therefore,* $\sum_{i \in N} \beta\lambda\gamma_i(\mathfrak{g}) - \varsigma\eta_i(\mathfrak{g}) = N(\beta\lambda - \varsigma)$ *and* $\sum_{i \in N} \beta\lambda\gamma_i(\bar{\mathfrak{g}}) - \varsigma\eta_i(\bar{\mathfrak{g}}) = N(\frac{\beta\lambda}{N-1} - \varsigma)$. *It is easy to observe that* $\lambda\beta > \frac{\lambda\beta}{N-1}$.

5.3 Experimental Results

We discuss our experimental results on random stable networks where, for 150 random scenarios, no agent loses its data. That is, even if the disk of an agent fails, in our random experiments, the disk of the agent' s neighbour (from whom it can retrieve its data) remains intact.

We conduct random experiments to answer the following question. Though agents form links and backup their data with adjacent agents, can any agent still lose its data? From Theorem 5.7, the null network is the unique pairwise stable network when the cost to add links is “high”, that is, $\varsigma \geq \frac{\beta}{\delta}$, and pairs of agents with links between them is the unique stable network otherwise. Therefore, as far as formation of networks is concerned, we always obtain one of these two networks, depending on the values of ς, β, and δ. In our experiment, we randomly generate networks of the second type, namely, pairwise links. We generate such networks on 30 agents and consider 150 random scenarios, by generating 10 random networks, 5 different sets of randomly chosen agents whose storage disks fail, for 3 cases, $\delta = 1\%$, 2%, and 4%. Our assumption on the values of δ is based on data from Backblaze[30] on hard drive failure rates. Interestingly, in none of the random cases we generated did agents on the two sides of a link fail at the same time.

Chapter Summary

In this chapter, we have presented the model of social storage cloud network formation, where agents (involved in storage sharing and data backup) wish to form a network strategically. The agents in this network strive for increasing the probability of obtaining storage space by minimizing the distances with others. We have proposed a degree-distance-based utility function and use the same to study network formation. We also studied the impact of the decision of link addition (deletion) between a pair of agents on shortest distances, closeness, and storage availability.

We have studied the deviation conditions under which agents have an incentive to add or delete a link in a given network structure. With these conditions, we analysed pairwise stability and efficiency of social storage cloud. We shown that there always exist a unique pairwise stable network, which is also efficient. Hence, the price of anarchy and the price of stability are, both, one.

[30] `https://www.backblaze.com/blog/backblaze-hard-drive-stats-q1-2019/` (accessed on 04 September 2019)

Chapter 6

Social Storage Cloud: Resource Availability

In the previous chapter, we studied network formation in the SSSC context, where agents form a stable network that is disconnected, where each agent is engaged in storage sharing with a single partner. In other words, in a stable SSSC, each agent maintains a single link and has no incentives to add more than one link, resulting in a disconnected network (for 3 or more agents). But this network structure introduces a crucial issue regarding QoS (low data availability and reliability), imbalanced workload distribution and poor resource utilization. In the context of exogenous social connections, it is observed that a network structure (in terms of agents' neighbourhood size) influences workload (request for resources) distribution and resource utilization. In load placement, agents with high neighbourhood size are frequently requested and those with less neighbours seldom. This leads to imbalanced workload distribution and poor resource utilization.

Let us look at this from the point of view of a system administrator who wants to deal with these issues. Let us assume that the system administrator gives incentives to its agents (who are users) to form at least one more link so that it leads to at least a minimally connected network, for instance, the ring network. Then, the system administrator observes the process of network formation. In this case, it is crucial to understand the impact of link formation between a pair of agents on their chance of obtaining a resource from another agent. It is also important to estimate how network size and density influence externalities, and how many agents benefit due to the newly added link. This chapter examines these aspects. It also studies the preferences of agents in link formation by considering factors like their closeness and their mutual distances.

6.1 Social Storage Cloud: Closeness-Based Resource Sharing

In an SSC $\mathfrak{g}$, agents perform closeness-based resource sharing, which is captured by the *harmonic centrality* measure as given below.

$$\Phi_i(\mathfrak{g}) = \sum_{j \in \mathfrak{g} \setminus \{i\}} \frac{1}{d_{ij}(\mathfrak{g})} \tag{6.1}$$

In an SCC $\mathfrak{g}$, an agent $j \in \mathfrak{g}$ (who acts as a resource provider) computes a probability distribution on all agents for the purpose of allocating storage resource to agent $i \in \mathfrak{g}$ (who acts as a resource consumer) and defined as below:

$$\alpha_{ij}(\mathfrak{g}) = \frac{\frac{1}{d_{ij}(\mathfrak{g})}}{\sum\limits_{j \in \mathfrak{g} \setminus \{i\}} \frac{1}{d_{ij}(\mathfrak{g})}} = \frac{1}{d_{ij}(\mathfrak{g})\Phi_i(\mathfrak{g})}. \tag{6.2}$$

In other words, $\alpha_{ij}(\mathfrak{g})$ is the probability that agent i will obtain storage space from agent j in $\mathfrak{g}$.

Remark 6.1. *If $d_{ij}(\mathfrak{g}) = \infty$ then $\alpha_{ij}(\mathfrak{g}) = 0 (= \alpha_{ji}(\mathfrak{g}))$. As agents i and j are disconnected in $\mathfrak{g}$ their chances of obtaining storage space from each other is nil.*

The probability that agent i obtains resource from at least one agent in $\mathfrak{g}$ is given as bellow:

$$\gamma_i(\mathfrak{g}) = 1 - \prod_{j \in \mathfrak{g} \setminus \{i\}} (1 - \alpha_{ij}(\mathfrak{g})). \tag{6.3}$$

Definition 6.1. *We refer to $\alpha_{ij}(\mathfrak{g})$ as the local resource availability of i from j, and $\gamma_i(\mathfrak{g})$, as the global resource availability of i.*

6.2 Local Resource Availability

This section focuses on local resource availability. Specifically, it studies the impact of link addition and deletion between a pair of agents on their local resource availability. It also studies how link addition (or deletion) between a pair of agents impacts the local resource availability from their neighbours.

An agent's chance of obtaining a resource from another agent is determined by, first, the distance between the agent (who wants a resource) and the other agent (who may provide the resource), and second, the other agent's closeness. Hence, it is important to know, how a newly added link affects the distance between pairs of agents and their closeness.

Remark 6.2. *Suppose* $\mathfrak{g}$ *is an SSC with distinct agents* k, l, *such that* $\langle kl \rangle \notin \mathfrak{g}$.
Then,

either $d_{ij}(\mathfrak{g}) = d_{ij}(\mathfrak{g} + \langle kl \rangle)$, *for all* $i \in \mathfrak{g} \setminus \{k, l\},\ j \in \mathfrak{g}$,

or $d_{ij}(\mathfrak{g}) = d_{ij}(\mathfrak{g} + \langle kl \rangle)$ *for some* $i \in \mathfrak{g} \setminus \{k, l\},\ j \in \mathfrak{g}$,

and $d_{ij}(\mathfrak{g}) > d_{ij}(\mathfrak{g} + \langle kl \rangle)$ *for some* $i \in \mathfrak{g} \setminus \{k, l\},\ j \in \mathfrak{g}$.

Similarly,

either $\Phi_i(\mathfrak{g}) = \Phi_i(\mathfrak{g} + \langle kl \rangle)$, *for all* $i \in \mathfrak{g} \setminus \{k, l\}$,

or $\Phi_i(\mathfrak{g}) = \Phi_i(\mathfrak{g} + \langle kl \rangle)$, *for some* $i \in \mathfrak{g} \setminus \{k, l\}$

and $\Phi_i(\mathfrak{g}) < \Phi_i(\mathfrak{g} + \langle kl \rangle)$, *for some* $i \in \mathfrak{g}$.

Due to this Remark, we study an agent's probability of obtaining a resource in $\mathfrak{g}$ by taking the following cases into consideration.

1. $d_{ij}(\mathfrak{g}) = d_{ij}(\mathfrak{g} + \langle kl \rangle)$ and $\Phi_i(\mathfrak{g}) = \Phi_i(\mathfrak{g} + \langle kl \rangle)$.

2. $d_{ij}(\mathfrak{g}) = d_{ij}(\mathfrak{g} + \langle kl \rangle)$ and $\Phi_i(\mathfrak{g}) < \Phi_i(\mathfrak{g} + \langle kl \rangle)$.

3. $d_{ij}(\mathfrak{g}) > d_{ij}(\mathfrak{g} + \langle kl \rangle)$ and $\Phi_i(\mathfrak{g}) < \Phi_i(\mathfrak{g} + \langle kl \rangle)$.

6.2.0.1 Link Alteration and Local Resource Availability

We first look at the relation between link alteration (in terms of link addition and deletion) and local resource availability.

Lemma 6.1. *Suppose $\mathfrak{g}$ and $\mathfrak{g}'$ are social storage clouds, and suppose $i, j \in \mathfrak{g} \cap \mathfrak{g}'$. Then $\alpha_{ij}(\mathfrak{g}) > \alpha_{ij}(\mathfrak{g}')$ if and only if $d_{ij}(\mathfrak{g}') \sum_{k \in \mathfrak{g}' \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g}')} > d_{ij}(\mathfrak{g}) \sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})}$.*

Proof. $\alpha_{ij}(\mathfrak{g}) > \alpha_{ij}(\mathfrak{g}')$, if and only if

$$\frac{1}{d_{ij}(\mathfrak{g})(\frac{1}{d_{ij}(\mathfrak{g})} + \sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})})} > \frac{1}{d_{ij}(\mathfrak{g}')(\frac{1}{d_{ij}(\mathfrak{g}')} + \sum_{k \in \mathfrak{g}' \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g}')})}, \text{ if and only if}$$

$$[\frac{1}{1 + d_{ij}(\mathfrak{g}) \sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})}}] > [\frac{1}{1 + d_{ij}(\mathfrak{g}') \sum_{k \in \mathfrak{g}' \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g}')}}], \text{ if and only if}$$

$$1 + d_{ij}(\mathfrak{g}') \sum_{k \in \mathfrak{g}' \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g}')} > 1 + d_{ij}(\mathfrak{g}) \sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})}, \text{ if and only if}$$

$$d_{ij}(\mathfrak{g}') \sum_{k \in \mathfrak{g}' \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g}')} > d_{ij}(\mathfrak{g}) \sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})}.$$

□

Proposition 6.1. *Suppose $\mathfrak{g}$ is an SSC, and suppose i and j are distinct agents in $\mathfrak{g}$ such that $\langle ij \rangle \notin \mathfrak{g}$. Then, $\alpha_{ij}(\mathfrak{g} + \langle ij \rangle) > \alpha_{ij}(\mathfrak{g})$.*

Proof. Owing to Lemma 6.1, it suffices to show that

$$d_{ij}(\mathfrak{g}) \sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})} > d_{ij}(\mathfrak{g} + \langle ij \rangle) \sum_{k \in \mathfrak{g} + \langle ij \rangle \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g} + \langle ij \rangle)} \tag{6.4}$$

Note that $d_{ij}(\mathfrak{g} + \langle ij \rangle) = 1$ and $d_{ij}(\mathfrak{g}) \in \{2, 3, \ldots, \}$.

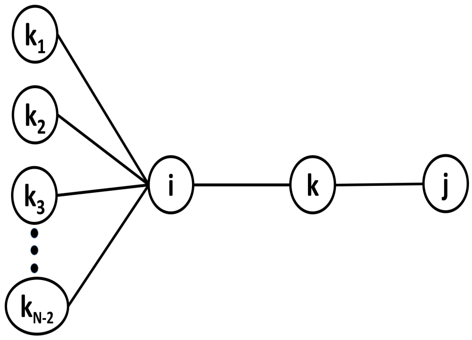
It suffices to check that Inequality (6.4) holds in the following three cases.

1. Let $d_{ij}(\mathfrak{g}) = \infty$. That is, i and j are not connected in $\mathfrak{g}$. Inequality (6.4) clearly holds.

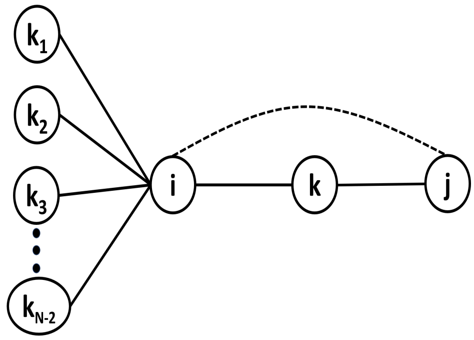
2. Let $\sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})} = \sum_{k \in \mathfrak{g} + \langle ij \rangle \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g} + \langle ij \rangle)}$. That is, addition of link $\langle ij \rangle$ does not change the shortest path between j and any other agent k except i. It is easy to see that Inequality (6.4) holds in this case too.

3. Now, suppose the addition of link $\langle ij \rangle$ changes the shortest path(s) between j and at least one k (besides i). Every shortest path between j and such k in $\mathfrak{g} + \langle ij \rangle$ is, obviously, shorter than that in $\mathfrak{g}$. Hence, $\sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})} < \sum_{k \in \mathfrak{g} + \langle ij \rangle \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g} + \langle ij \rangle)}$.

To show that the Left Hand Side of Inequality (6.4) is greater than the Right Hand Side, we show that it holds for the worst (or tightest) possible case of the Inequality (given N). This happens when $d_{ij}(\mathfrak{g})$ is the minimum possible, that is, 2, and for $\mathfrak{g}$, with N agents, as shown in Figure 6.1(a). Agent i is connected with $N-2$ agents, agent j has a single neighbour k, and k is an intermediary between i and j. In $\mathfrak{g}$, $\Phi_j(\mathfrak{g}) = 1$ and $d_{ij}(\mathfrak{g}) = 2$. Let agent i and j add a direct link in $\mathfrak{g}$, resulting in the network structure $\mathfrak{g} + \langle ij \rangle$ as shown in Figure 6.1(b). Here, $\Phi_j(\mathfrak{g} + \langle ij \rangle) = \frac{N+1}{2}$, which is an upper bound on the closeness of j in any connected network $\mathfrak{g} + \langle ij \rangle$, when $\Phi_j(\mathfrak{g}) = 1$. We have, $\sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})} = \frac{N}{3}$ and $\sum_{k \in \mathfrak{g} + \langle ij \rangle \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g} + \langle ij \rangle)} = \frac{N-1}{2}$, which



((a)) Network $\mathfrak{g}$



((b)) Network $\mathfrak{g} + \langle ij \rangle$

Figure 6.1: Link addition and local resource availability

satisfies Inequality (6.4). □

The above result shows that agents always improve their local resource availability by adding new resource sharing connections.

Proposition 6.2. *Suppose* $\mathfrak{g}$ *is an SSC with distinct agents i and j such that* $\langle ij \rangle \in \mathfrak{g}$*. Then,* $\alpha_{ij}(\mathfrak{g}) > \alpha_{ij}(\mathfrak{g} - \langle ij \rangle)$.

Proof. Owing to Lemma 6.1, it suffices to show that

$$d_{ij}(\mathfrak{g} - \langle ij \rangle) \sum_{k \in \mathfrak{g} - \langle ij \rangle \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g} - \langle ij \rangle)} > d_{ij}(\mathfrak{g}) \sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})} \tag{6.5}$$

We know that $d_{ij}(\mathfrak{g}) = 1$, $d_{ij}(\mathfrak{g} - \langle ij \rangle) \in \{2, 3, \ldots\}$, and $0 \leq \sum_{k \in \mathfrak{g} \setminus \{i,j\}} \frac{1}{d_{jk}(\mathfrak{g})} \leq N-2$, 0 when

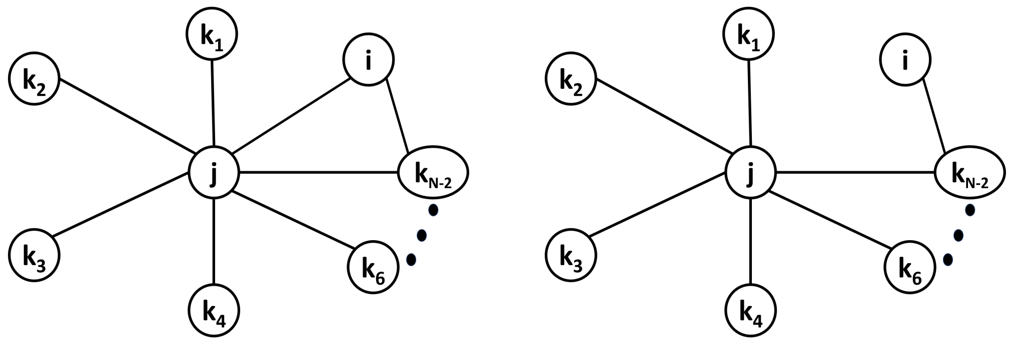
j is isolated and $N-2$ when j is connected to all k.

It suffices to check that Inequality (6.5) holds in the following three cases.

1. Let $d_{ij}(\mathfrak{g}-\langle ij\rangle)=\infty$. That is, $\langle ij\rangle$ is the only path between i and j in $\mathfrak{g}$. Inequality (6.5), clearly, holds in this case.

2. Let $\sum\limits_{k\in\mathfrak{g}\setminus\{i,j\}}\frac{1}{d_{jk}(\mathfrak{g})}=\sum\limits_{k\in\mathfrak{g}-\langle ij\rangle\setminus\{i,j\}}\frac{1}{d_{jk}(\mathfrak{g}-\langle ij\rangle)}$.That is, deletion of link $\langle ij\rangle$ does not change the shortest path between j and any other agent k except i. It is easy to see that Inequality (6.5) holds in this case too.

3. Suppose $\sum\limits_{k\in\mathfrak{g}\setminus\{i,j\}}\frac{1}{d_{jk}(\mathfrak{g})}=N-2$, the least upper bound of the summation. This means j has links with all other agents. Refer Figure 6.2(a).



((a)) Network $\mathfrak{g}$ ((b)) Network $\mathfrak{g}-\langle ij\rangle$

Figure 6.2: Link deletion and local resource availability

Now, suppose $d_{ij}(\mathfrak{g}-\langle ij\rangle)=2$, the greatest lower bound on the distance as $\langle ij\rangle$ has been deleted. In $\mathfrak{g}-\langle ij\rangle$, j has links with all agents except i. An agent i has a link to at least one other agent, for otherwise we have Case 1. Refer Figure 6.2(b). Therefore, $\sum\limits_{k\in\mathfrak{g}-\langle ij\rangle\setminus\{i,j\}}\frac{1}{d_{jk}(\mathfrak{g}-\langle ij\rangle)}=N-2$.

Hence, the Left Hand Side of Inequality (6.5) is $2(N-2)$ and its Right Hand Side is $N-2$, thereby proving that the inequality holds. □

The above result shows that an agent's decision to delete an existing resource sharing connection decreases the local resource availability of the pair of agents from each other.

Theorem 6.1. *Suppose $\mathfrak{g}$ is an SSC with distinct agents i and j. Then, the link $\langle ij \rangle$ is always strictly beneficial to both i and j, with respect to local resource availability from each other.*

Proof. Follows from Propositions 6.1, 6.2 and Lemma 6.1. □

6.2.0.2 Neighbors and Local Storage Availability

In the previous section, we saw that an agent i improves its local resource availability from another agent j by forming link $\langle ij \rangle$. However, this newly added link decreases agent i's local resource availability from its existing neighbours k who are at least three hops away from j. For neighbours k who are less than three hops away from j, agent i's local resource availability from them remains the same. Similarly, while an agent's local resource availability from another agent decreases if their existing link is deleted, the agent's local resource availability from its existing neighbours who are at least three hops away increases, and remains the same for the other neighbours. We prove these results below.

Proposition 6.3. *Suppose $\mathfrak{g}$ is an SSC. Suppose i, j and k are distinct agents in $\mathfrak{g}$ such that $\langle ij \rangle \notin \mathfrak{g}$ and $k \in \eta_i(\mathfrak{g})$. Then, the following hold:*

1. *If $k \in \eta_j(\mathfrak{g})$, then $\alpha_{ik}(\mathfrak{g}) = \alpha_{ik}(\mathfrak{g} + \langle ij \rangle)$.*

2. *If $d_{kj}(\mathfrak{g}) = 2$, then $\alpha_{ik}(\mathfrak{g}) = \alpha_{ik}(\mathfrak{g} + \langle ij \rangle)$.*

3. *If $d_{kj}(\mathfrak{g}) > 2$, then $\alpha_{ik}(\mathfrak{g}) > \alpha_{ik}(\mathfrak{g} + \langle ij \rangle)$.*

Proof. If $d_{kj}(\mathfrak{g}) > 2$, then $\Phi_k(\mathfrak{g} + \langle ij \rangle) > \Phi_k(\mathfrak{g})$ as $d_{kj}(\mathfrak{g} + \langle ij \rangle) = 2 < d_{kj}(\mathfrak{g})$, the new shortest path being the path with the two links $\langle ki \rangle$ and $\langle ij \rangle$.

Therefore, from Equation (6.2), $\alpha_{ik}(\mathfrak{g}) = \frac{1}{\Phi_k(\mathfrak{g})} > \frac{1}{\Phi_k(\mathfrak{g}+\langle ij \rangle)} = \alpha_{ik}(\mathfrak{g} + \langle ij \rangle)$, thereby proving case 3.

If $d_{kj}(\mathfrak{g}) \leq 2$, then $d_{kj}(\mathfrak{g} + \langle ij \rangle) = d_{kj}(\mathfrak{g})$ and, hence, $\Phi_k(\mathfrak{g} + \langle ij \rangle) = \Phi_k(\mathfrak{g})$.

It follows that $\alpha_{ik}(\mathfrak{g}) = \alpha_{ik}(\mathfrak{g} + \langle ij \rangle)$, proving cases 1 as well as 2. □

Similar results hold for agent k's resource availability from agent i too, as stated in the following corollary.

Corollary 6.1. *Suppose i, j and k are distinct agents in an SSC $\mathfrak{g}$ such that $\langle ij \rangle \notin \mathfrak{g}$ and $k \in \eta_i(\mathfrak{g})$. Then, the following hold:*

1. *If $k \in \eta_j(\mathfrak{g})$, then $\alpha_{ki}(\mathfrak{g}) = \alpha_{ki}(\mathfrak{g} + \langle ij \rangle)$.*

2. *If $d_{kj}(\mathfrak{g}) = 2$, then $\alpha_{ki}(\mathfrak{g}) = \alpha_{ki}(\mathfrak{g} + \langle ij \rangle)$.*

3. *If $d_{kj}(\mathfrak{g}) > 2$, then $\alpha_{ki}(\mathfrak{g}) > \alpha_{ki}(\mathfrak{g} + \langle ij \rangle)$.*

We have the following results on the aggregate local resource availability, aggregated over all neighbors of i in $\mathfrak{g}$.

Corollary 6.2. *Suppose $\mathfrak{g}$ is a two-diameter SSC and Suppose $\langle ij \rangle \notin \mathfrak{g}$. Then, $\prod_{k \in \eta_i(\mathfrak{g})} \alpha_{ik}(\mathfrak{g}) = \prod_{k \in \eta_i(\mathfrak{g} + \langle ij \rangle) \setminus \{j\}} \alpha_{ik}(\mathfrak{g} + \langle ij \rangle)$.*

Corollary 6.3. *Suppose i and j are distinct agents in an SSC $\mathfrak{g}$ such that $\langle ij \rangle \notin \mathfrak{g}$. Suppose the radius of the shortest-path-$\{\eta_i, \{j\}\}$-induced-subgraph of $\mathfrak{g}$ is at least 3. Then $\prod_{k \in \eta_i(\mathfrak{g})} \alpha_{ik}(\mathfrak{g}) > \prod_{k \in \eta_i(\mathfrak{g} + \langle ij \rangle) \setminus \{j\}} \alpha_{ik}(\mathfrak{g} + \langle ij \rangle)$.*

We, now, see that an agent's local connections increase its aggregate local resource availability, when the agent deletes a link with one of the neighbours who is at least three hops away from the other neighbours.

Proposition 6.4. *Suppose i, j and k are distinct agents in an SSC $\mathfrak{g}$ such that $\langle ij \rangle \in \mathfrak{g}$ and $k \in \eta_i(\mathfrak{g})$. Then, the following hold:*

1. *If $k \in \eta_j(\mathfrak{g})$, then $\alpha_{ik}(\mathfrak{g}) = \alpha_{ik}(\mathfrak{g} - \langle ij \rangle)$.*

2. *If $d_{kj}(\mathfrak{g}) = 2$, then $\alpha_{ik}(\mathfrak{g}) = \alpha_{ik}(\mathfrak{g} - \langle ij \rangle)$.*

3. *If $d_{kj}(\mathfrak{g}) > 2$, then $\alpha_{ik}(\mathfrak{g}) < \alpha_{ik}(\mathfrak{g} - \langle ij \rangle)$.*

Proof. The result can be proved in lines similar to the proof of Lemma 6.3. □

Corollary 6.4. *Suppose i, j and k are distinct agents in an SSC $\mathfrak{g}$ such that $\langle ij \rangle \notin \mathfrak{g}$ and $k \in \eta_i(\mathfrak{g})$. Then, the following hold:*

1. *If $k \in \eta_j(\mathfrak{g})$, then $\alpha_{ki}(\mathfrak{g}) = \alpha_{ki}(\mathfrak{g} - \langle ij \rangle)$.*

2. *If* $d_{kj}(\mathfrak{g}) = 2$*, then* $\alpha_{ki}(\mathfrak{g}) = \alpha_{ki}(\mathfrak{g} - \langle ij \rangle)$.

3. *If* $d_{kj}(\mathfrak{g}) > 2$*, then* $\alpha_{ki}(\mathfrak{g}) > \alpha_{ki}(\mathfrak{g} - \langle ij \rangle)$.

Corollary 6.5. *Suppose* $\mathfrak{g}$ *is a two-diameter SCC where* i *and* j *are distinct agents and* $\langle ij \rangle \in \mathfrak{g}$*. Then* $\prod_{k \in \eta_i(\mathfrak{g}) \setminus \{j\}} \alpha_{ik}(\mathfrak{g}) = \prod_{k \in \eta_i(\mathfrak{g} - \langle ij \rangle)} \alpha_{ik}(\mathfrak{g} - \langle ij \rangle)$.

Corollary 6.6. *Suppose* i *and* j *are distinct agents in an SSC* $\mathfrak{g}$ *where* $\langle ij \rangle \in \mathfrak{g}$*. Suppose the radius of the shortest-path-*$\{\eta_i(\mathfrak{g}), \{j\}\}$*-induced-subgraph of* $\mathfrak{g}$ *is at least* 3*. Then* $\prod_{k \in \eta_i(\mathfrak{g}) \setminus \{j\}} \alpha_{ik}(\mathfrak{g}) < \prod_{k \in \eta_i(\mathfrak{g} - \langle ij \rangle)} \alpha_{ik}(\mathfrak{g} - \langle ij \rangle)$.

Theorem 6.2. *Suppose* $\mathfrak{g}$ *is an SSC with distinct agents* i*,* j *and* k*, such that* $k \in \eta_i(\mathfrak{g})$*. Then, the link* $\langle ij \rangle$ *is always strictly beneficial to* i *as well as* k*, with respect to local resource availabilities from each other, if and only if* $d_{kj}(\mathfrak{g}) > 2$.

Proof. Follows from Propositions 6.3, 6.4 and Lemma 6.1. □

Now, we examine the impact on agents' local resource availability due to a newly added link (say between j and k) in $\mathfrak{g}$.

Lemma 6.2. *Let* $\alpha_{ij}(\mathfrak{g})$ *and* $\alpha_{ij}(\mathfrak{g} + \langle kl \rangle)$ *are the probabilities that agent* i *will obtain a resource from* j *in* $\mathfrak{g}$ *and* $\mathfrak{g} + \langle kl \rangle$*, respectively.*

1. *If* $d_{ij}(\mathfrak{g}) = d_{ij}(\mathfrak{g} + \langle kl \rangle)$ *and*

   - $\Phi_j(\mathfrak{g}) = \Phi_j(\mathfrak{g} + \langle kl \rangle)$ *then* $\alpha_{ij}(\mathfrak{g}) = \alpha_{ij}(\mathfrak{g} + \langle kl \rangle)$.
   - $\Phi_j(\mathfrak{g}) < \Phi_j(\mathfrak{g} + \langle kl \rangle)$ *then* $\alpha_{ij}(\mathfrak{g}) > \alpha_{ij}(\mathfrak{g} + \langle kl \rangle)$.

2. *If* $d_{ij}(\mathfrak{g}) > d_{ij}(\mathfrak{g} + \langle kl \rangle)$ *and*

   - $\frac{d_{ij}(\mathfrak{g}+\langle kl \rangle)}{d_{ij}(\mathfrak{g})} > \frac{\Phi_j(\mathfrak{g})}{\Phi_j(\mathfrak{g}+\langle kl \rangle)}$ *then* $\alpha_{ij}(\mathfrak{g}) > \alpha_{ij}(\mathfrak{g} + \langle kl \rangle)$.
   - $\frac{d_{ij}(\mathfrak{g}+\langle kl \rangle)}{d_{ij}(\mathfrak{g})} < \frac{\Phi_j(\mathfrak{g})}{\Phi_j(\mathfrak{g}+\langle kl \rangle)}$ *then* $\alpha_{ij}(\mathfrak{g}) < \alpha_{ij}(\mathfrak{g} + \langle kl \rangle)$.

Proof. The proof follows the following observations:
$\alpha_{ij}(\mathfrak{g}) = \frac{1}{d_{ij}(\mathfrak{g})\Phi_j(\mathfrak{g})}$ and $\alpha_{ij}(\mathfrak{g} + \langle kl \rangle) = \frac{1}{d_{ij}(\mathfrak{g}+\langle kl \rangle)\Phi_j(\mathfrak{g}+\langle kl \rangle)}$. □

6.3 Externalities

6.3.1 Experimental Analysis

In the SSC model, we have seen that externalities are derived by the agents' chance of obtaining storage space from at least one agent in the network (we term this as global resource availability). Addition of a link imposes both positive as well as negative externalities on other agents. A newly added link is beneficial for a set of agents, and at the same time, it is non-beneficial (and even detrimental) for another set of agents. This section attempts to extend the understanding of externalities in the process of network formation by computing both positive and negative externalities as a function of the network structure (in terms of network size and density).

6.3.1.1 Experimental Data:

We are interested in environments with a large number of network configurations where agents form resource sharing connections with others based on harmonic centrality based measure. Since the number of network combinations grows exponentially ($O(2^{\frac{N(N-1)}{2}})$) with the number of agents (N), the complexity of an analytical treatment of the problem also increases. For instance, with 4 agents, there are 64 possible networks and with 6 agents there are 32,768 possible networks. In addition to the issue of manageability, a core difficulty with large networks is that they may encode a tremendous amount of information or a very limited amount of information [98]. In the literature, several authors have studied the role of network structure in determining the outcome of economic activities by taking different sizes of network (number of agents) with different network structures (shape of the network). Table 6.1 lists a few of them.

To generate data to investigate the relation between externalities, and network size and network density we implement the network formation algorithm on different sizes of *connected* ring (circle) networks varying from 4 to 30 agents. The network sizes we consider here are considerably large in the context of endogenous network formation literature. However, we admit that the network size we consider for the analysis is one of the limitations of our study.

Author(s)	Network Size (Agents)	Network Structure
Keser et al. [99]	3, 8	The complete and circle
Vallam et al. [100]	4	The complete equi-bipartite
Gallo and Yan [101]	9, 15	The circle
Carrillo and Gaduh [102]	12	The bipartite and star
Gallo [103]	6	The regular network of degree 4, the star, and the circle

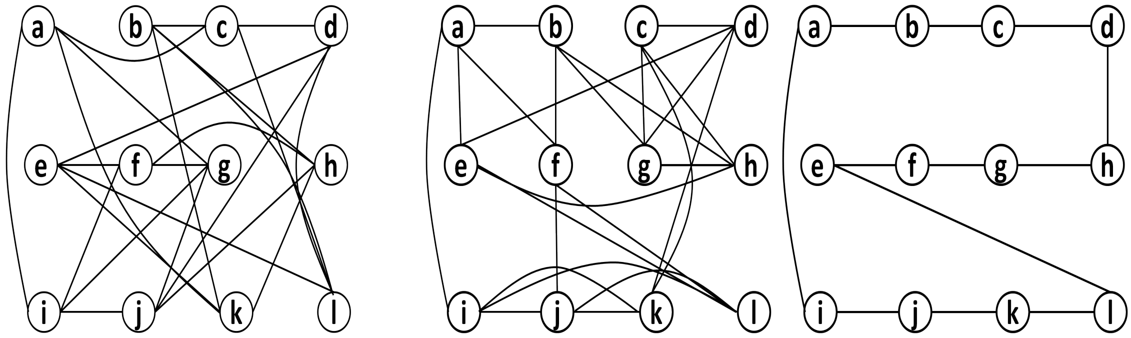
Table 6.1: Studies of network formation experiments

We focus on the ring network structure for two reasons. First, unlike an arbitrary network or a random η-regular network (where each agent has same neighbourhood size η), the ring network has uniform harmonic centrality distribution. That is, each agent has same closeness in the ring network. Second, the structural properties of this network do not significantly affect the centralities' granularity, and hence, do not have impact on the relation between the network size and externalities. Our investigation becomes difficult in the case of non-uniform harmonic centrality and with large network size. For instance, even for a small network of 14 agents and $\eta = 3$ and $\eta = 7$, we have 509 and 21609301 networks[31], respectively [104]. For the given network size and η, the harmonic centrality (closeness) of agents differ from one network structure to another network structure.

For instance, we have three networks of size 12 ($N = 12$) (see Figure 6.3). In the network $\mathfrak{g}_1$ (see Figure 6.3(a)) and $\mathfrak{g}_2$ (see Figure 6.3(b)) each agent has four neighbours (i.e. $\eta = 4$), whereas in the ring network (see Figure 6.3(c)), each agent has two neighbours (i.e. $\eta = 2$). The corresponding closeness of each agent in each network is shown in Table 6.2. From this data, we can observe that although network $\mathfrak{g}_1$ and $\mathfrak{g}_2$ are of the same size and where each agent has the same neighbourhood size, there is non-uniformity in the harmonic centrality distribution, whereas this is not true in the case of ring network as shown in Figure 6.3(c)).

Therefore, it is cumbersome to deal with networks $\mathfrak{g}_1$ and $\mathfrak{g}_2$ in terms of generating the data and further analysing this data for each case. In fact, the non-uniformity in closeness

[31] http://www.mathe2.uni-bayreuth.de/markus/reggraphs.html



((a)) $\mathfrak{g}_1(\eta = 4)$ ((b)) $\mathfrak{g}_2(\eta = 4)$ ((c)) Ring($\eta = 2$)

Figure 6.3: Network structures with $N = 12$

Agent	$\mathfrak{g}_1$	$\mathfrak{g}_1$	Ring
a	7.5000	7.3333	4.7333
b	7.1667	7.3333	4.7333
c	7.3333	7.0000	4.7333
d	7.5000	7.3333	4.7333
e	7.5000	7.5000	4.7333
f	7.3333	7.1667	4.7333
g	7.1667	7.0000	4.7333
h	7.5000	7.1667	4.7333
i	7.1667	7.1667	4.7333
j	7.5000	7.1667	4.7333
k	7.5000	7.3333	4.7333
l	7.1667	7.1667	4.7333

Table 6.2: Closeness (harmonic centrality)

does not serve our aim of investigating the relation between externalities, network size and density.

6.3.1.2 Experimental Approach

In order to obtain experimental data for our study, among practices such as online and offline methods of recruiting subjects, agent-based modelling, and computer program based simulation, we find computer program based simulation methodology the most appropriate. On the one hand, online (for example, Amazon Mechanical Turk[32]) and offline methods of recruiting subjects are laborious and costly, and not necessary in our study. On the other

[32] https://www.mturk.com/

hand, agent-based-models are well suited for a dynamic strategic economic framework.

To obtain data for studying externalities, we adopt the cost-efficient and simple approach of computer program based simulation [105, 106] method[33]. This method is widely used to understand both strategic and non-strategic dynamic network formation. For example, [108] investigate agents' similarity in terms of interests, tests, beliefs, social backgrounds, and so on. in a non-strategic dynamic network. [100] study network formation with localized payoff in order to study the pairwise stability and efficiency of the network. Similarly, network formation with heterogeneous preferences is studied by [109]. The above works implement the network formation procedure in MATLAB, C++ and Java, respectively.

In our study, for the analysis of externalities, we generate social networks (of sizes mentioned above) by using SocNetV (Social Network Analysis and Visualization) software[34]. We obtain data related to distances of agents in these networks by using the same software. The generated network acts as an input to the procedure describe in the following algorithm, which generates the data. Further, we make use of Microsoft Excel for quantitative analysis of closeness and resource availability.

Initially, we compute $\gamma_i(\mathfrak{g})$ for all agents in a ring network $\mathfrak{g}$. Then, we select an agent j arbitrarily and add a link with another agent k whose distance is two hops from j. We compute $\gamma_i(\mathfrak{g}+\langle jk\rangle)-\gamma_i(\mathfrak{g})$, and count the number of beneficiaries (**NOB**) i for whom this difference is positive. That is, NOB is the set of agents who experience positive externalities due to a newly added link. We repeat the above for all agents k who are located at distance two hops from agent j in $\mathfrak{g}$. Then, we increment the distance between j and k by one and follow the same procedure. We do this until we exhaust all agents j.

6.3.2 Findings

This section discusses the findings on the relation between externalities, and network size and density.

[33]This is due to unavailability of data of real world networks like BuddyBackup or CrashPlan, and because of the endogeneity issue [107] faced by various studies

[34]http://socnetv.org/

Algorithm for Analysing Externalities in a Network $\mathfrak{g}$

Input: Network $\mathfrak{g}$

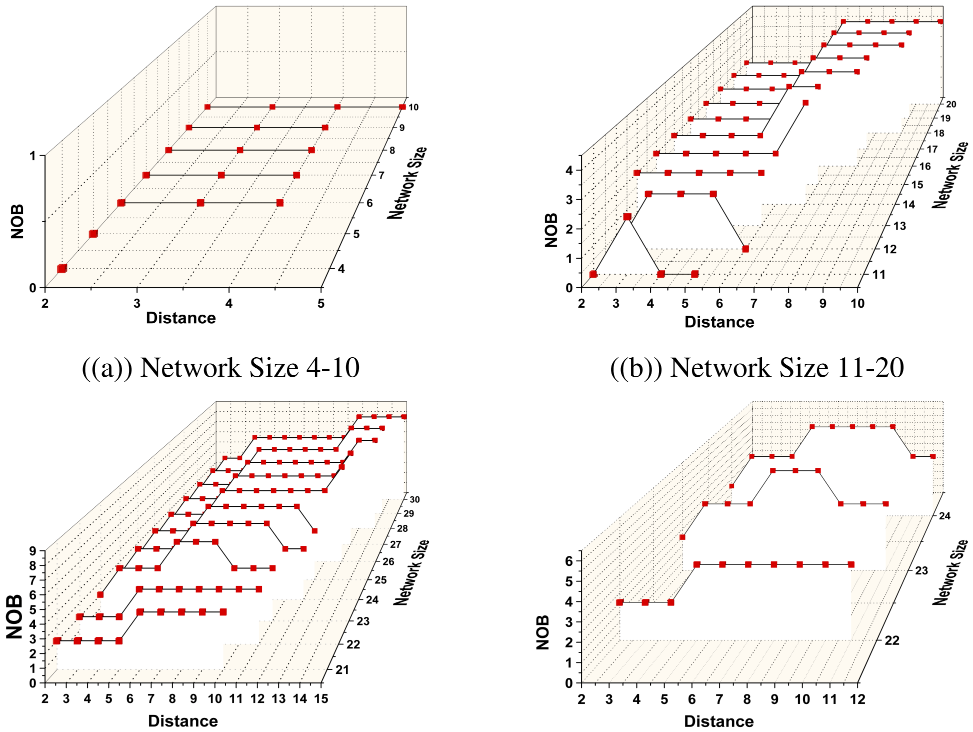
Output: $Total\_NOB$ = Total Number of Beneficiaries

Output: Number of Beneficiaries for an agent i (NOB_i)

for each agent $i \in \mathfrak{g}$ **do**
  Compute $\gamma_i(\mathfrak{g})$ by (Equation 1 in the manuscript)
end for
Compute diameter $\mathscr{D}_{\mathfrak{g}}$ of $\mathfrak{g}$
For an agent $i \in \mathfrak{g}$ set $NOB_i = 0$
$\{NOB_i$ is the number of beneficiaries due to agents i's link formation with other agents$\}$

Set distance = 2
while distance $\leq \mathscr{D}_{\mathfrak{g}}$ **do**
  for each agent $j \in \mathfrak{g}$ such that $d_{ij}(\mathfrak{g}) =$ distance **do**
    Add a link $\langle ij \rangle$ in $\mathfrak{g}$
    $\{$Network $\mathfrak{g}$ is updated and it becomes $\mathfrak{g} + \langle ij \rangle\}$
    Set $NOB_{ij} = 0$
    $\{NOB_{ij}$ is the number of beneficiaries due to newly added link between i and $j\}$
    for each $k \in \mathfrak{g} + \langle ij \rangle \setminus \{i, j\}$ **do**
      Compute $\gamma_k(\mathfrak{g} + \langle ij \rangle)$ by (Equation 1 in the manuscript)
      Compute $\gamma_k(\mathfrak{g} + \langle ij \rangle) - \gamma_k(\mathfrak{g})$
      if $\gamma_k(\mathfrak{g} + \langle ij \rangle) - \gamma_k(\mathfrak{g}) > 0$ **then**
        $NOB_{ij} = NOB_{ij} + 1$
      end if
    end for
    $NOB_i = NOB_i + NOB_{ij}$
    Delete $\langle ij \rangle$ in $\mathfrak{g} + \langle ij \rangle$
  end for
  increase distance by 1
end while

6.3.2.1 Network Size



((a)) Network Size 4-10 ((b)) Network Size 11-20

((c)) Network Size 21-30 ((d)) Network Size 22-24

Figure 6.4: Externalities and network-size.

Figure 6.4 summarises our results. The *x-axis* represents the shortest distance between two agents involved in link formation. The *y-axis* represents the NOB. The *z-axis* represents the network size (the number of agents in the network).

Finding 6.1. *In "less" populated ring networks, no agent experiences positive externalities.*

In networks with size varying from 4 to 10 (Figure 6.4(a)), no agent experiences positive externalities. From from Remark 5.1, positive externalities require an increase in closeness, which is absent here. However, from network size 11 to 30 (Figures 6.4(b) and 6.4(c)), a significant number of agents experience positive externalities.

Finding 6.2. *In ring networks of size greater than 10, as the distance between the agents involved in link addition increases, the number of beneficiaries increases in most cases, and in all cases for small distances.*

Plots in Figures 6.4(b), 6.4(c) and 6.4(d) are of this type. It is clear that, if an agent experiences positive externalities, its closeness should increased (from Remark 5.1). This is intuitive — if a pair of agents who are far from each other in $\mathfrak{g}$ form a link then, this link reduces the mutual distances among other agents, and therefore, their closeness increase. For example, in the network of size 21 (see Figure 6.4(c)), if a pair of agents who have distance 2 form a link, we have 3 NOB. But, if a pair of agents who have distance 6 form a link we have 4 NOB. This indicates that link formation between two agents in $\mathfrak{g}$ brings many agents close to each other, resulting in increments in their closeness. However, merely an increase in closeness is not sufficient for agents to experience positive externalities.

Finding 6.3. *In ring networks, the number of beneficiaries is always less than the number of non-beneficiaries.*

In all our experiments, the percentage of beneficiaries varies from 0% to 26% of the total number of agents in the network.

6.3.2.2 Network Density

Network density of a network $\mathfrak{g}$ is the ratio of the number of existing links in $\mathfrak{g}$ to the maximum number of possible links in $\mathfrak{g}$ [110]. That is, $\triangledown(\mathfrak{g}) = \frac{\ell}{\frac{N(N-1)}{2}}$. Thus, $\triangledown(\mathfrak{g})$ goes from 0 (if each agent is isolated in $\mathfrak{g}$, i.e. $\ell = 0$) to 1 (if each agent has connections with all other $N-1$ agents in $\mathfrak{g}$, i.e. $\ell = \frac{N(N-1)}{2}$).

To study the correlation between externalities and network density, we focus on η-regular networks with varying η' s. As compared to the network size experiments above, where η was 2 (or ring networks), this is done so that different agents have different closeness, which would help in our analysis (as earlier, for ring networks, closeness of all agents was the same). The range of number of agents n is same as above, but for the sake of compactness we report data for the number of agents varying from 11 to 20, which form a representative set.

For a given n and a given η (which also gives the diameter of the network or $\mathscr{D}_{\mathfrak{g}}$), we first compute its density ($\triangledown(\mathfrak{g})$). Computing NOB is more involved here because of the varying closeness of agents. Next, we compute closeness of all agents, and sort them into buckets corresponding to the same closeness[35]. Then, we randomly pick any agent from each bucket and run the above algorithm. We report the maximum of NOB obtained from all buckets. Note that the combination of odd n and odd η is not valid (i.e., a network is not possible for this combination). The results of these experiments are given in Table 6.3.

Finding 6.4. *The network density is inversely proportional to positive externalities.*

From Table 6.3, we observe that the network density and the NOB are inversely proportional. That is, as the network density increases, the number of agents who experience positive externalities decreases. We analyse this finding further by using our earlier conjectures. From Remark 6.2, we know that by every link addition the closeness of all agents either remains same or increases. From Remark 5.1, we know that for an agent to experience positive externalities, its closeness must improve due to this new link (although this is not sufficient).

In a highly connected network (or a dense network, i.e., $\triangledown(\mathfrak{g}) \rightarrow 1$), agents are already very close to each other, and hence, it is less likely that a newly added link improves their closeness. Thus, here the chance of agents experiencing positive externalities is also less.

On the contrary, in a less connected network (or a sparse network, i.e., $\triangledown(\mathfrak{g}) \rightarrow 0$), when a pair of agents form a link, it is very likely that this newly added link improves the closeness of many agents. Thus, here the chance of agents experiencing positive externalities also increases.

Finding 6.5. *The above relation between the network density and positive externalities, to a large extent, is independent of the number of agents in the network.*

As in Table 6.3, for network sizes varying from 11 to 20, we see that for a linear increment in η, the number of neighbours (or a linear decrement in the maximum shortest path), the network density increases and the NOB decrease loosely following different arithmetic progressions, which is independent of the number of agents in the network. A similar behaviour is observed for network sizes less than 11, but that data is not reported in this table for the sake of compactness.

[35]The bucket logic is added for efficiency. Also, for ring networks, we had only one such bucket because all agents had the same closeness.

Finding 6.6. *In a two-diameter network SSC (i.e. $\mathscr{D}_{\mathfrak{g}} = 2$) the NOB is always 0.*

Table 6.3 clearly indicates this. In a two-diameter network, the maximum shortest path between a pair agent is either one or two. Hence, no link addition improves the distance of other pairs of agents, thus their closeness. This finding is consistent with the earlier observation stated in 5.1.

Finding 6.7. *An agent who has lower (higher) closeness is likely to experience positive (negative) externalities.*

The maximum value of closeness of an agent indicates that it is close to every other agent. For instance, if agent (say, i) in $\mathfrak{g}$ is connected with every other agent then $\Phi_i(\mathfrak{g}) = N-1$, which is the maximum closeness for any agent in any network structure. For an agent i who has higher closeness, a newly added link (say, between agents j and k) may not reduce its distances with others to a great extent, therefore, $\Phi_i(\mathfrak{g}) \approx \Phi_i(\mathfrak{g} + \langle jk \rangle)$, and as a result, $\gamma_i(\mathfrak{g})$ likely goes down in $\mathfrak{g} + \langle jk \rangle$. The above results suggest that as the density of a given network increases, the closeness of agents also increases. We observe that as network density increases, the chances that agents experience negative externalities also increase.

6.4 Choice Modelling

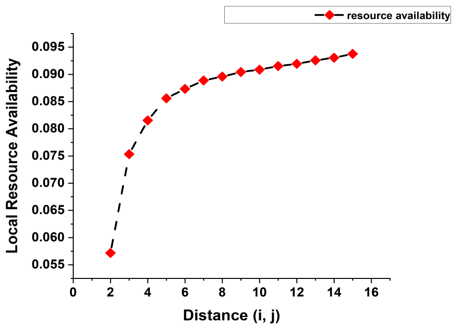
In SSC g, each agent strives for maximizing its resource availability. Let us assume an agent wants to form a link with others. With whom will the agent prefer to form a link (or links) so that it maximizes its resource availability (say, the local resource availability)? This section deals with this aspect. In order to study this, we first understand the relation between distance between agents and local as well as global resource availabilities, we, first, discuss the following example.

Example 6.1. *Suppose $\mathfrak{g}$ is a ring network*[36]*. Then, for all $i \in \mathfrak{g}$,*

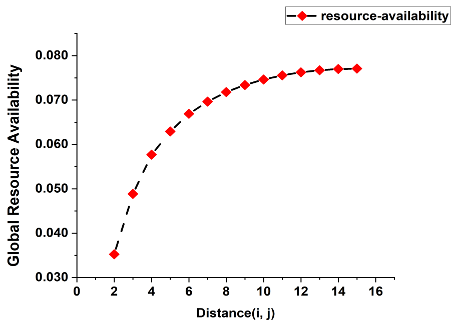
$$\Phi_i(\mathfrak{g}) = \begin{cases} 2(1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{N}), \text{ if } N \text{ is odd} \\ 2(1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{N-1}) + \frac{1}{N}, \text{ if } N \text{ is even.} \end{cases}$$

[36] A ring network $\mathfrak{g}$ is a connected network where $\eta_i(\mathfrak{g}) = 2$ for all $i \in \mathfrak{g}$.

Suppose $i \in \mathfrak{g}$. For $j \in \mathfrak{g}$, suppose $\langle ij \rangle \notin \mathfrak{g}$. We compute the local resource availability, $\alpha_{ij}(\mathfrak{g}+\langle ij \rangle)$ of i from j, and the global resource availability $\gamma_i(\mathfrak{g}+\langle ij \rangle)$ for different j in increasing order of the distance between i and j in $\mathfrak{g}$. Figures 6.5(a) and 6.5(b) show that as the distance between i and j increases, $\alpha_{ij}(\mathfrak{g}+\langle ij \rangle)$ and $\gamma_i(\mathfrak{g}+\langle ij \rangle)$ also increase.



((a)) $\alpha_{ij}(\mathfrak{g}+\langle ij \rangle)$



((b)) $\gamma_i(\mathfrak{g}+\langle ij \rangle)$

Figure 6.5: Local and global resource availabilities of agent *i* in the ring network

We, now, discuss the relation between local resource availability and distance as well as closeness.

Lemma 6.3. *In an SSC $\mathfrak{g}$ with distinct agents i, j, such that $\langle ij \rangle \notin \mathfrak{g}$, the local resource availability of agent i from agent j increases with decrease in the distance, $d_{ij}(\mathfrak{g})$, between them.*

Proof. For any $i, j \in \mathfrak{g}$, $i \neq j$, $0 < d_{ij}(\mathfrak{g}) < d_{ij}(\mathfrak{g})+1$. Then, from Equation (6.2), $\frac{1}{d_{ij}(\mathfrak{g})\Phi_j(\mathfrak{g})} > \frac{1}{(d_{ij}(\mathfrak{g})+1)\Phi_j(\mathfrak{g})}$. □

Lemma 6.4. *Suppose $\mathfrak{g}$ is an SSC with distinct agents i, j and k, such that $\langle ij \rangle, \langle ik \rangle \notin \mathfrak{g}$. If $d_{ij}(\mathfrak{g}) > d_{ik}(\mathfrak{g})$ then, $\alpha_{ij}(\mathfrak{g}) < \alpha_{ik}(\mathfrak{g})$.*

Proof. Follows from Lemma 6.3. □

Lemma 6.5. *In an SSC $\mathfrak{g}$, agent $i \in \mathfrak{g}$ obtains maximum local resource availability from that $k \in \mathfrak{g}$ who is least close to others (that is, with the least harmonic centrality).*

Proof. Let $j, k \in \mathfrak{g}$ and $0 < \Phi_k(\mathfrak{g}) < \Phi_j(\mathfrak{g})$, and $d_{ij}(\mathfrak{g}) = d_{ik}(\mathfrak{g})$. Then, from Equation (6.2), $\frac{1}{d_{ik}(\mathfrak{g})\Phi_k(\mathfrak{g})} > \frac{1}{d_{ij}(\mathfrak{g})\Phi_j(\mathfrak{g})}$. □

We showed, in Lemmas 6.4 and 6.5, that the local resource availability of an agent from another agent increases with decrease in the distance between them and that maximum local resource availability is obtained from the agent with the least closeness (that is, least harmonic centrality). We, now, look at the relation between the global resource availability of an agent and its closeness.

Lemma 6.6. *In an SSC $\mathfrak{g}$, agent i maximizes its global resource availability by maximizing its own closeness or equivalently, by minimizing its distance with others.*

Proof. The proof is in lines similar to that of Lemma 6.5. □

We now discuss results that show which agent to add a link to, under different circumstances.

Proposition 6.5. *Suppose $\mathfrak{g}$ is an SSC and $i \in \mathfrak{g}$. Across all $j \in \mathfrak{g}$, $\langle ij \rangle \notin \mathfrak{g}$, suppose i chooses $j = j_0$ to which to add a link, such that j_0 maximizes the local resource availability of i from j in $\mathfrak{g} + \langle ij \rangle$. Then, j_0 is the agent (or one of the agents) whose closeness is the least, among all $j \in \mathfrak{g}$ such that $\langle ij \rangle \notin \mathfrak{g}$.*

Proof. Suppose agents k and $l \in \mathfrak{g}$ are such that $\langle ik \rangle \notin \mathfrak{g}$ and $\langle il \rangle \notin \mathfrak{g}$. Agent i prefers l over k to add a link, if and only if

$\alpha_{ij}(\mathfrak{g} + \langle ij \rangle) > \alpha_{ik}(\mathfrak{g} + \langle ik \rangle)$, if and only if $\frac{1}{d_{ij}(\mathfrak{g}+\langle ij \rangle)\Phi_j(\mathfrak{g}+\langle ij \rangle)} > \frac{1}{d_{ik}(\mathfrak{g}+\langle ik \rangle)\Phi_k(\mathfrak{g}+\langle ik \rangle)}$.

We have $d_{ij}(\mathfrak{g} + \langle ij \rangle) = d_{ik}(\mathfrak{g} + \langle ik \rangle) = 1$.

Hence, $\alpha_{ij}(\mathfrak{g} + \langle ij \rangle) > \alpha_{ij}(\mathfrak{g} + \langle ik \rangle)$, if and only if $\frac{1}{\Phi_j(\mathfrak{g}+\langle ij \rangle)} > \frac{1}{\Phi_k(\mathfrak{g}+\langle ik \rangle)}$, if and only if

$\Phi_k(\mathfrak{g} + \langle ik \rangle) > \Phi_j(\mathfrak{g} + \langle ij \rangle)$. □

Proposition 6.6. *Suppose $\mathfrak{g}$ is an SSC and $i \in \mathfrak{g}$. Across all $j \in \mathfrak{g}$, $\langle ij \rangle \notin \mathfrak{g}$, suppose i chooses $j = j_0$ to which to add a link, such that j_0 maximizes the global resource availability of i in $\mathfrak{g} + \langle ij \rangle$. Then, j_0 is the agent (or one of the agents) whose closeness is the highest, among all $j \in \mathfrak{g}$ such that $\langle ij \rangle \notin \mathfrak{g}$.*

Proof. Similar to the proof of Proposition 6.5. □

Lemma 6.7. *Agent i prefers j over k to which to add a link, if and only if*

$$\frac{\Phi_k(\mathfrak{g}+\langle ik\rangle)-\Phi_j(\mathfrak{g}+\langle ij\rangle)}{\Phi_k(\mathfrak{g}+\langle ik\rangle)\Phi_j(\mathfrak{g}+\langle ij\rangle)} > \frac{[d_{ik}(\mathfrak{g})\Phi_k(\mathfrak{g})]-d_{ij}(\mathfrak{g})\Phi_j(\mathfrak{g})]}{[d_{ik}(\mathfrak{g})\Phi_k(\mathfrak{g})][d_{ij}(\mathfrak{g})\Phi_j(\mathfrak{g})]}.$$

Proof. $[\alpha_{ij}(\mathfrak{g}+\langle ij\rangle)-\alpha_{ij}(\mathfrak{g})] > [\alpha_{ik}(\mathfrak{g}+\langle ik\rangle)-\alpha_{ik}(\mathfrak{g})]$.

$$\Rightarrow [\alpha_{ij}(\mathfrak{g}+\langle ij\rangle)-\alpha_{ik}(\mathfrak{g}+\langle ik\rangle)] > [\alpha_{ij}(\mathfrak{g})-\alpha_{ik}(\mathfrak{g})].$$

$$\Rightarrow [\frac{1}{\Phi_j(\mathfrak{g}+\langle ij\rangle)} - \frac{1}{\Phi_k(\mathfrak{g}+\langle ik\rangle)}] > [\frac{1}{d_{ij}(\mathfrak{g})\Phi_j(\mathfrak{g})} - \frac{1}{d_{ik}(\mathfrak{g})\Phi_k(\mathfrak{g})}]$$

$$\Rightarrow \frac{\Phi_k(\mathfrak{g}+\langle ik\rangle)-\Phi_j(\mathfrak{g}+\langle ij\rangle)}{\Phi_k(\mathfrak{g}+\langle ik\rangle)\Phi_j(\mathfrak{g}+\langle ij\rangle)} > \frac{[d_{ik}(\mathfrak{g})\Phi_k(\mathfrak{g})]-d_{ij}(\mathfrak{g})\Phi_j(\mathfrak{g})]}{[d_{ik}(\mathfrak{g})\Phi_k(\mathfrak{g})][d_{ij}(\mathfrak{g})\Phi_j(\mathfrak{g})]}.$$ □

Theorem 6.3. *The following holds for any SSC* $\mathfrak{g}$.

1. *An agent i prefers to add a link with j over k if j is farther to others than that of k in* $\mathfrak{g}+\langle ij\rangle$*, given that, both agents (j and k) have the same closeness and they are at the same distance from the agent i in* $\mathfrak{g}$.

2. *An agent i prefer to add a link with j over k if j is far from i than that of k in* $\mathfrak{g}$*, given that, both agents have same closeness in* $\mathfrak{g}$ *and* $\mathfrak{g}+\langle ij\rangle$.

3. *An agent i prefers to add a link with j over k if j is less close to others than that of k in* $\mathfrak{g}$*, given that, both agents have the same closeness in* $\mathfrak{g}+\langle ij\rangle$ *and they are at the same distance from agent i in* $\mathfrak{g}$.

Proof. Follows from Lemma 6.7. □

6.5 Chapter Summary

This chapter examined the impact of link formation between a pair of agents on their local resource availability. It shown that on the one hand, link addition always increases local resource availability involved in the link formation, and on the other hand, deletion decreases it. This study also examined externalities in terms of group well-being — it measures, the number of beneficiaries and non-beneficiaries due to a newly added link. The last section of the chapter studied the preferences of agents in link formation. If link formation is allowed, then with whom will agents want to add a link so that their local resource availability is maximized.

Network Size n	η	$\mathscr{D}_{\mathfrak{g}}$	$\nabla(\mathfrak{g})$	NOB
11	2	5	0.20	8
	4	3	0.40	5
	6	2	0.60	0
12	2	6	0.18	24
	3	4	0.27	9
	4	3	0.36	7
	5	3	0.46	7
	6	2	0.55	0
13	2	6	0.17	24
	4	3	0.33	11
	6	2	0.50	0
14	2	7	0.15	28
	3	4	0.22	22
	4	3	0.31	15
	5	3	0.40	11
	6	2	0.46	0
15	2	7	0.14	32
	4	3	0.29	20
	6	2	0.43	0
16	2	8	0.13	38
	3	5	0.20	30
	4	3	0.27	24
	5	3	0.33	17
	6	3	0.40	8
	7	2	0.47	0
17	2	8	0.13	40
	4	4	0.25	45
	6	3	0.38	13
	8	2	0.50	0
18	2	9	0.12	48
	3	5	0.18	36
	4	3	0.24	30
	5	3	0.29	21
	6	3	0.35	14
	7	2	0.42	0
19	2	9	0.11	48
	4	3	0.22	30
	6	3	0.33	16
	8	2	0.44	0
20	2	10	0.11	56
	4	4	0.21	37
	6	3	0.32	23
	8	2	0.42	0

Table 6.3: Externalities and network density

Chapter 7

Social Compute Cloud: Pairwise Stability

In the previous chapter, we have studied the social storage cloud. These results may be extended to other social cloud networks where computational resources (besides storage) may be shared. In this chapter, we discuss one such extension. Here, we present our initial observations on the stability of symmetric social compute cloud. We first characterize stable networks by deriving the conditions under which an agent has no incentive to add a link or delete one of its links.

The analysis of network formation can be performed either as a forward or a reverse problem. The forward approach looks at which network is likely to emerge for the given cost and benefit of agents in the network. The reverse problem looks at conditions under which the given network structure is pairwise stable [111].

Recall, in a symmetric social compute cloud (SSCC) $\mathfrak{g}$, the utility of agent i is as follows.

$$u_i(\mathfrak{g}) = p(1-q)\xi + q[p + (1-p)\gamma_i(\mathfrak{g})]\theta_i - \varsigma\eta_i(\mathfrak{g}). \tag{7.1}$$

whereas ς the link formation cost, ξ and θ are the benefits associated with providing a resource to others and accomplishing a computational task, respectively. Note that, p is the probability that agent has a resource and q is the probability that agent needs to perform a computational task.

7.0.1 Stable SSCC Characterisation

We now discuss deviation conditions for agent $i \in \mathfrak{g}$, which suggest under what condition agent i has an incentive to add a new or delete an existing resource sharing connection.

Lemma 7.1. *Let $\mathfrak{g}$ be an SSCC. A agent $i \in \mathfrak{g}$ is benefited by adding a link with agent $j \in \mathfrak{g}$, if and only if $q(1-p)[\gamma_i(\mathfrak{g}+\langle ij\rangle)-\gamma_i(\mathfrak{g})] > \frac{\varsigma}{\theta}$.*

Proof. Let us say agent i's utility in $\mathfrak{g}$ is as (7.1).

Let agent i and i forms a link $\langle ij\rangle$, then the structure of $\mathfrak{g}$ changes to $\mathfrak{g}+\langle ij\rangle$ and then the utility of agent i in new structure $\mathfrak{g}+\langle ij\rangle$ is as follows

$$u_i(\mathfrak{g}+\langle ij\rangle) = p(1-q)\xi + q[p+(1-p)\gamma_i(\mathfrak{g}+\langle ij\rangle)]\theta - \varsigma(\eta_i+1)$$

agent i has incentive to form a link with agent j if and only if $u_i(\mathfrak{g}+\langle ij\rangle) > u_i(\mathfrak{g})$.

$$u_i(\mathfrak{g}+\langle ij\rangle) > u_i(\mathfrak{g})$$

$$\Rightarrow [p(1-q)\xi + q[p+(1-p)\gamma_i(\mathfrak{g}+\langle ij\rangle)]\theta - \varsigma(\eta_i+1)] > [p(1-q)\xi + q[p+(1-p)\gamma_i(\mathfrak{g})]\theta - \varsigma\eta_i]$$

$$\Rightarrow \theta q(1-p)[\gamma_i(\mathfrak{g}+\langle ij\rangle)-\gamma_i(\mathfrak{g})] > \varsigma.$$

Hence, we can argue that,

$$u_i(\mathfrak{g}+\langle ij\rangle) > u_i(\mathfrak{g}) \text{ if } q(1-p)[\gamma_i(\mathfrak{g}+\langle ij\rangle)-\gamma_i(\mathfrak{g})] > \frac{\varsigma}{\theta}.$$ □

Corollary 7.1. *A agent $i \in \mathfrak{g}$ has no incentive to add a link with a agent $j \in \mathfrak{g}$, if and only if $q(1-p)[\gamma_i(\mathfrak{g}+\langle ij\rangle)-\gamma_i(\mathfrak{g})] \leq \frac{\varsigma}{\theta}$.*

Lemma 7.2. *Let $\mathfrak{g}$ be an SSCC. A agent $i \in \mathfrak{g}$ benefits by deleting a link with agent j, if and only if $q(1-p)[\gamma_i(\mathfrak{g})-\gamma_i(\mathfrak{g}-\langle ij\rangle)] < \frac{\varsigma}{\theta}$.*

Proof. Let us say agent i's utility in $\mathfrak{g}$ is as (7.1).

Let agent i deletes a link $\langle ij \rangle$ with j, then structure of $\mathfrak{g}$ changes to $\mathfrak{g} - \langle ij \rangle$ and then the utility of agent i in new structure $\mathfrak{g} - \langle ij \rangle$ is as follows

$$u_i(\mathfrak{g} - \langle ij \rangle) = p(1-q)\xi + q[p + (1-p)\gamma_i(\mathfrak{g} - \langle ij \rangle)]\theta - \varsigma(\eta_i - 1).$$

A agent i has incentive to delete a link with agent j if and only if $u_i(\mathfrak{g} - \langle ij \rangle) > u_i(\mathfrak{g})$.

$$u_i(\mathfrak{g} - \langle ij \rangle) > u_i(\mathfrak{g})$$

$$\Rightarrow [p(1-q)\xi + q[p + (1-p)\gamma_i(\mathfrak{g} - \langle ij \rangle)]\theta - \varsigma(\eta_i - 1)] >$$

$$[p(1-q)\xi + q[p + (1-p)\gamma_i(\mathfrak{g})]\theta - \varsigma(\eta_i)]$$

$$\Rightarrow \varsigma > \theta q(1-p)[\gamma_i(\mathfrak{g}) - \gamma_i(\mathfrak{g} - \langle ij \rangle)].$$

Thus, $u_i(\mathfrak{g} - \langle ij \rangle) > u_i(\mathfrak{g})$ if $\frac{\varsigma}{\theta} > q(1-p)[\gamma_i(\mathfrak{g}) - \gamma_i(\mathfrak{g} - \langle ij \rangle)]$. □

Corollary 7.2. *Let $\mathfrak{g}$ be an SSCC. A agent i has no incentive to delete an existing link with a agent j if and only if* $q(1-p)[\gamma_i(\mathfrak{g}) - \gamma_i(\mathfrak{g} - \langle ij \rangle)] \geq \frac{\varsigma}{\theta}$.

Corollary 7.3. *In $\mathfrak{g}$, a agent's decision about link addition and deletion is independent of benefit ξ.*

Although an agent benefits ξ by sharing its resource to others, its decision of adding or deleting a link is independent of benefit ξ. In fact, from Lemma 7.1 and 7.2, we can observe that the structure of the network is defined by the parameters, such as, the cost to maintain a direct link, and the rate at which agents' need to perform a task and the resources available in the network. Therefore, an agent's expected benefit (i.e., $q(1-p)\xi$) is exogenous to the network.

Proposition 7.1. *An SSCC $\mathfrak{g}$ is pairwise stable if*

- *for all* $i, j \in \mathfrak{g}, q(1-p)[\gamma_i(\mathfrak{g}) - \gamma_i(\mathfrak{g} - \langle ij \rangle)] \geq \frac{\varsigma}{\theta}$ *and* $q(1-p)[\gamma_j(\mathfrak{g}) - \gamma_j(\mathfrak{g} - \langle ij \rangle)] \geq \frac{\varsigma}{\theta}$, *and*

- *for all* $i, j \in \mathfrak{g}$, *if* $q(1-p)[\gamma_i(\mathfrak{g} + \langle ij \rangle) - \gamma_i(\mathfrak{g})] > \frac{\varsigma}{\theta}$ *then* $q(1-p)[\gamma_j(\mathfrak{g} + \langle ij \rangle) - \gamma_j(\mathfrak{g})] < \frac{\varsigma}{\theta}$.

Proof. The proof follows Definition 2.2, Lemma 7.1, and Corollary 7.2 . □

From the above result, we can argue that a resource network is a pairwise stable if no agent has incentives to delete an existing link. If the ratio of cost for maintaining the link and the benefit obtained by completing a computational task is less than the probability of becoming a consumer. And the marginal probability of getting a resource in the present network structure and the network obtained by deleting an existing link. On the other hand, no pair of agents want to add a new link between them if the ratio of cost and benefit is greater than the probability of becoming a consumer, the marginal probability of getting a resource in the present network structure and the network obtained by adding a new link.

7.0.2 Stable Network Existence

In this section, we examine network formation as the inverse problem. Specifically, we focus on four special cases of network structures, namely, the two-diameter network, the star network, the wheel network, the complete bipartite network. With deviation conditions discussed earlier and the definition of pairwise stability, we investigate the best response of agents in the given network structures that lead to pairwise stable.

Lemma 7.3. *Let* $\mathfrak{g}$ *be a two-diameter SSCC. Then, for each agent* $i \in \mathfrak{g}$,

1. $\gamma_i(\mathfrak{g} + \langle ij \rangle) - \gamma_i(\mathfrak{g}) = \pi_i^+(\mathfrak{g})[\alpha_{ij}(\mathfrak{g} + \langle ij \rangle) - \alpha_{ij}(\mathfrak{g})]$.
2. $\gamma_i(\mathfrak{g}) - \gamma_i(\mathfrak{g} - \langle ij \rangle) = \pi_i^-(\mathfrak{g})[\alpha_{ij}(\mathfrak{g}) - \alpha_{ij}(\mathfrak{g} - \langle ij \rangle)]$.

Proof. For the proof we consider two cases.
In the first case we assume that in the network $\mathfrak{g}$, a pair of agents $\langle ij \rangle$ is involved in *indirect* resource sharing relationship, while in second we assume that a pair of agents $\langle ij \rangle$ is involved in *direct* relationship.

1. Suppose, in $\mathfrak{g}$ agent i is *indirectly* connected with agent j. From 3.10 the probability that j will obtain a resource from i is $\alpha_{ij}(\mathfrak{g})$, and the probability ($\gamma_i(\mathfrak{g})$) that j will get from at least one agent in $\mathfrak{g}$ is

$$\gamma_i(\mathfrak{g}) = [1 - (\prod_{k \in \mathfrak{g} \setminus \{i,j\}} (1-\alpha_{kj}(\mathfrak{g})) \times \underbrace{(1-\alpha_{ij}(\mathfrak{g}))}_{i \text{ is not a neighbor of } j \text{ in } \mathfrak{g}})] \tag{7.2}$$

Now, let us consider that agent j add link with i in $\mathfrak{g}$ then in new structure $\mathfrak{g} + \langle ij \rangle$ agent i and j are neighbors.

In the network structure $\mathfrak{g} + \langle ij \rangle$ the probability that j will not get resource from i is $1 - \alpha_{ij}(\mathfrak{g} + \langle ij \rangle)$, and $\gamma_i(\mathfrak{g} + \langle ij \rangle)$ is the probability that j will get resource from at least one agent as follows

$$\gamma_i(\mathfrak{g} + \langle ij \rangle) = [1 - (\prod_{k \in \mathfrak{g}+\langle ij \rangle \setminus \{i,j\}} (1-\alpha_{kj}(\mathfrak{g} + \langle ij \rangle)) \times \overbrace{(1-\alpha_{ij}(\mathfrak{g} + \langle ij \rangle))}^{i \text{ is now a neighbor of } j \text{ in } (\mathfrak{g} + \langle ij \rangle)})] \tag{7.3}$$

Let us say, $\pi_i^+(\mathfrak{g}) = \prod_{k \in \mathfrak{g} \setminus \{i,j\}} (1-\alpha_{kj}(\mathfrak{g}))$, and

$\pi_i(\mathfrak{g} + \langle ij \rangle) = \prod_{k \in \mathfrak{g}+\langle ij \rangle \setminus \{i,j\}} (1-\alpha_{kj}(\mathfrak{g} + \langle ij \rangle))$.

We simplifying 7.2 and 7.3 as bellow

$$\gamma_i(\mathfrak{g}) = [1 - (\pi_i^+(\mathfrak{g}) \times \underbrace{(1-\alpha_{ij}(\mathfrak{g}))}_{i \in \hat{\eta}_j(\mathfrak{g})})] \tag{7.4}$$

$$\gamma_i(\mathfrak{g} + \langle ij \rangle) = [1 - (\pi_i(\mathfrak{g} + \langle ij \rangle) \times \underbrace{(1-\alpha_{ij}(\mathfrak{g} + \langle ij \rangle))}_{i \in \eta_i(\mathfrak{g} + \langle ij \rangle)})]. \tag{7.5}$$

Then by subtracting 7.4 from 7.5 we have,

$\gamma_i(\mathfrak{g} + \langle ij \rangle) - \gamma_i(\mathfrak{g})$

$$= [1 - (\pi_i^+(\mathfrak{g} + \langle ij \rangle) \times (1 - \alpha_{ij}(\mathfrak{g} + \langle ij \rangle))] - [1 - (\pi_i^+(\mathfrak{g}) \times (1 - \alpha_{ij}(\mathfrak{g}))].$$

But we know, $\pi_i(\mathfrak{g} + \langle ij \rangle) = \pi_i^+(\mathfrak{g})$, and hence,

$$\gamma_i(\mathfrak{g} + \langle ij \rangle) - \gamma_i(\mathfrak{g}) = \pi_i^+(\mathfrak{g})[\alpha_{ij}(\mathfrak{g} + \langle ij \rangle) - \alpha_{ij}(\mathfrak{g})].$$

This proves the first part. □

2. Now, let us suppose that in $\mathfrak{g}$ agent i is *directly* connected with agent j.

The probability that j will get from at least one agent in $\mathfrak{g}$ is

$$\gamma_i(\mathfrak{g}) = [1 - (\prod_{k \in \mathfrak{g} \setminus \{i,j\}} (1 - \alpha_{kj}(\mathfrak{g})) \times \underbrace{(1 - \alpha_{ij}(\mathfrak{g}))}_{i \in \eta_i(\mathfrak{g})})]. \tag{7.6}$$

Now let us consider that agent j deletes link with i then i and j are not remain neighbours in the new structure $\mathfrak{g} - \langle ij \rangle$.

In this network $(\mathfrak{g} - \langle ij \rangle)$ the probability that j will not get resource from at least one agent is as follows

$$\gamma_i(\mathfrak{g} - \langle ij \rangle) = [1 - (\prod_{k \in \mathfrak{g} - \langle ij \rangle \setminus \{i,j\}} (1 - \alpha_{kj}(\mathfrak{g} - \langle ij \rangle)) \times \underbrace{(1 - \alpha_{ij}(\mathfrak{g} - \langle ij \rangle))}_{i \text{ in } (\mathfrak{g} - \langle ij \rangle) \text{ not a neighbor of } j})]. \tag{7.7}$$

Let us say, $\pi_i^-(\mathfrak{g}) = \prod_{k \in \mathfrak{g} \setminus \{i,j\}} (1 - \alpha_{kj}(\mathfrak{g}))$, and

$$\pi_i(\mathfrak{g} - \langle ij\rangle) = \prod_{k\in\mathfrak{g}-\langle ij\rangle\setminus\{i,j\}} (1 - \alpha_{kj}(\mathfrak{g} - \langle ij\rangle))$$

Then by subtracting 7.7 from 7.6 we have,

$$\gamma_i(\mathfrak{g}) - \gamma_i(\mathfrak{g} - \langle ij\rangle)$$
$$= [1 - (\pi_i^-(\mathfrak{g}) \times (1 - \alpha_{ij}(\mathfrak{g}))] - [1 - (\pi_i(\mathfrak{g} - \langle ij\rangle) \times (1 - \alpha_{ij}(\mathfrak{g} - \langle ij\rangle))]$$

But we know $\pi_i(\mathfrak{g} - \langle ij\rangle) = \pi_i^-(\mathfrak{g})$, and hence,

$$\gamma_i(\mathfrak{g}) - \gamma_i(\mathfrak{g} - \langle ij\rangle) = \pi_i^-(\mathfrak{g})[\alpha_{ij}(\mathfrak{g}) - \alpha_{ij}(\mathfrak{g} - \langle ij\rangle)].$$

This proves the second part. □

Proposition 7.2. *In a two-diameter network* $\mathfrak{g}$*,*

1. *For agent j adding a link with agent i is beneficial if and only if* $q(1-p)\pi_i^+(\mathfrak{g})[\alpha_{ij}(\mathfrak{g}+\langle ij\rangle) - \alpha_{ij}(\mathfrak{g})] > \frac{\varsigma}{\theta}$*, for all* $i,j \in \mathfrak{g}$*, and*
2. *For agent j deleting a link with agent i is beneficial if and only if* $q(1-p)\pi_i^-(\mathfrak{g})[\alpha_{ij}(\mathfrak{g}) - \alpha_{ij}(\mathfrak{g}] - \langle ij\rangle)] < \frac{\varsigma}{\theta}$ *for all* $i,j \in \mathfrak{g}$*.*

From Lemma 7.1, 7.2 and 7.0.2.

Proposition 7.3. *In a two-diameter SSCC* $\mathfrak{g}$ *is pairwise stable if*

1. *for all* $\langle ij\rangle \in \mathfrak{g}$*,*
$min\Big\{q(1-p)\pi_i^-(\mathfrak{g})[\alpha_{ij}(\mathfrak{g}) - \alpha_{ij}(\mathfrak{g}], q(1-p)\pi_j^-(\mathfrak{g})[\alpha_{ji}(\mathfrak{g}) - \alpha_{ji}(\mathfrak{g})]\Big\} > \frac{\varsigma}{\theta}$*, and*
2. *for all* $\langle ij\rangle \in \mathfrak{g}$*,*
$max\Big\{q(1-p)\pi_i^+[\alpha_{ij}(\mathfrak{g}+\langle ij\rangle) - \alpha_{ij}(\mathfrak{g})], q(1-p)\pi_j^+[\alpha_{ji}(\mathfrak{g}+\langle ij\rangle) - \alpha_{ji}(\mathfrak{g})]\Big\} < \frac{\varsigma}{\theta}$*.*

Proof follows Definition 2.2, Lemma and Proposition 7.2.

Claim 7.1. *A null SSCC* $\mathfrak{g}$ *is pairwise stable if* $p(1-q)q(1-p)\theta \leq \varsigma$*.*

This is intuitive. If for an agent the cost of link formation is greater than the probability (that agent be a provider and a consumer) and the benefit (that the agent obtain after completing the task), then, the agent has no incentive to add a link, as a result, the null SSCC is pairwise stable.

Observation 7.1. *Let an SSCC be the complete network* $\mathfrak{g}$ *of n agents such that each agent is adjacent with every other agent. Then* $\mathfrak{g}$ *is pairwise stable if*
$\frac{\varsigma}{q(1-p)\theta} \leq p(1-q)(1-\frac{p(1-q)}{n-1})^{n-2}(\frac{n-2}{(n-1)(2n-3)})$.

As each agent is adjacent with every other agents in the complete network $\mathfrak{g}$, agent can not add one more link in $\mathfrak{g}$. The above condition suggests that no agent has incentives to delete an existing link. Thus, the $\mathfrak{g}$ is pairwise stable. Further, we state our observations on pairwise stability of the star, the wheel, and the complete bipartite network in Appendix D.

7.1 Chapter Summary

The discussion in this chapter is primarily based on our initial observations. Therefore, it has left several issues untouched. In this case, it is important to know, whether, for the given parameters p, q, θ, and ς there exists a pairwise SSCC or not. It is difficult to estimate pairwise stability beyond the two diameter network due to the presence of both positive and negative externalities. In our view, it requires different treatment than the theoretical approach. This we leave for future work.

Chapter 8

Conclusion and Future Work

This study primarily examined social cloud formation in a strategic setting. It has expanded on two untouched aspects of social cloud, namely, endogenous network formation and stability of such networks. In particular, this study formalized the social cloud as a resource sharing network formation game, where rational agents construct their resource sharing connections in order to maximize their utility. Three utility functions (payoff structures) are proposed, which are a combination of the cost and the benefit that agents experience as a function of the established resource sharing network. The proposed utility functions are; a degree-based (where agents receive benefits only from their immediate links), a distance-based (where agents receive benefits from indirect links), and a variant of the two above. These payoff structures are the first of their kind in the social cloud literature. Broadly, this study investigated the stability and efficiency of social storage concerning the proposed utility functions. It also examined externalities (how utility of an agent is affected due to action of other agents) through a theoretical as well as an experimental approach.

This chapter is organized as follows: Section 8.1 summarizes the findings; Section 8.2 relates this study to the existing literature; Section 8.3 outlines the implications of this study; Section 8.4 states limitations; and Section 8.5 discusses the future direction.

8.1 Recapitulation of Findings

In Chapter 3, we have introduced three social cloud formation models, namely, the social storage network model, the social storage cloud model, and the social compute model. In social storage network model agents perform data backup activity with their neighbours (immediate links) only. The utility function defined in this model is degree based, which captures the following important parameters, namely, the benefit associated with data, the cost for link formation, and the chance of disk failure. The social storage cloud model is an the extension of the social storage network model. The utility function defined in this model is the degree-distance based, it incorporates indirect links and relaxes the data loss cause. The social compute model captured the broad class of social cloud systems where agents obtain benefit by providing and consuming resources, however as the earlier, they pay the cost to stay in the network. The utility function in this model (close to the Job Contact Model [60]), is a refinement of the distance-based utility function.

In Chapter 4, we studied the social storage network formation with the solution concept of the bilateral stability (refinement of pairwise stability). We found that for the symmetric version of the utility function, the social storage network formation always leads to a bilateral stable network. However, the structure of the bilateral stable network is not independent of the number of agents. For example, on the one hand, if the number of agents is even then we have a bilateral stable network in which each agent has η number of neighbours (depending upon the value of the cost and the benefit and the disk failure rate). On the other hand, if the number of agents is odd, then we have a bilateral stable network, in which $N-1$ agents have η number of neighbours and one agent has $\eta-1$ neighbours. We also find that an efficient social storage network and a contented social storage network, both, are bilaterally stable. However, not all efficient networks are contented.

In Chapter 5, we have studied the social storage cloud formation, where agents in this network strive for increasing the probability of obtaining storage space by minimizing the distances with others. We studied the social storage cloud formation with the pairwise stability solution concept. We shown that for the symmetric utility function there always exists a unique pairwise stable network, which is also efficient. Therefore, the price of anarchy and the price of stability are, both, one.

Chapter 6 mainly focused on the analysis of externalities in terms of resource availability. With regards to externalities, we found that for an agent to experience positive externalities, an increment in its closeness is necessary, but not sufficient. Through experimental analysis we found that in "less" populated ring networks, no agent experiences positive externalities. Also, the number of beneficiaries is always less than the number of non-beneficiaries. We studied preference modelling, which suggests with whom an agent prefers to add link. For instance, between two candidate agents an agent prefers to add a link with the one who is less close to other agents in the network formed after link addition (given that both the agents have the same closeness and they are at the same distance from the agent in question).

In Chapter **??** the conditions under which a two-diameter, star, complete, wheel, and complete-bipartite networks are pairwise stable.

8.2 Relationship with Previous Research

In this study, we have applied strategic network formation as a tool to study social cloud formation, in particular, social storage systems (which are inspired by the P2P storage). Therefore, the literature source for this study is from two disciplines; network formation and social storage. First, we relate our work with network formation and then to social storage, and specifically, to P2P storage.

In this study, we have presented bilateral stability as a solution concept by refining the pairwise stability to study social storage network formation. Our solution concept of bilateral stability subsumes the concept of bilateral equilibrium proposed by Goyal and Vega-Redondo [32]. The set of all strategies that are bilaterally stable contains the set of all bilateral equilibrium strategies. A network which is bilaterally stable may contain agents who may be better off by deviating, whereas a bilateral equilibrium network does not contain any such agent. Both definitions, however, allow only bilateral deviations (or pairwise addition as well as deletion with mutual consent). Buechel and Hellmann [112] have termed bilateral equilibrium as bilateral stability. Hummon [113] also discussed mutual consent for deletion, but he does not formally define or study the concept of stability with mutual consent for deletion. The author performed agent based simulation of the connection model proposed in [31], and discussed simulation outputs. Other works that focused on agent based simulations are by Falk and Kosfeld [114] and Goeree et al. [115].

Our study is most closely related to P2P systems. Especially, our strategic network formation game has some similarities with peer selection for data placement [116, 117, 118] and topology formation [119, 63] in P2P systems. To start off, in P2P nomenclature, virtual (i.e., logical) topologies (or structures) are built by peers (in-general computers or software modules) on top of physical networks (e.g., Internet).

The articles [116, 117] have studied data placement in a strategic interaction between peers to maximise data availability. Here, peers are involved in a reciprocal replication contract (a pair of agents replicates each others' data to increase data availability). They showed that agents prefer to form contracts with only those who have similar availability. This behaviour of peers makes the system inefficient.

However, by setting cooperation rules and providing incentives to peers, data availability can be increased along with the increase in the efficiency of the system. We believe that these ideas of cooperation rules and incentives would be inspirational to design a more practical social storage system.

Toka and Michiardi [118] studied data placement in a different strategic setting than above. Here, peers selfishly select partners based upon their profiles. The profile of each peer, which includes the online availability, the bandwidth, and global preferences, is considered along with the utility function so that the data storing costs are minimised. Authors have shown here that there exists at least one pairwise stable matching and it can be found in polynomial time. In our study, we have not considered agents availability. This was based upon the assumption that the out of band communication [15] is possible between them. However, this can be further analysed.

Finally, we looked at topology formation in P2P systems. P2P topologies are a mirror image of social connections in our case. The studies [119, 63] have proposed a locality game (inspired by the network creation game proposed by [62]) to study the impact of selfish peers on P2P topologies. In this setting, selfish peers select their partners in such way that the stretch (i.e., the look up performance in terms of latencies) could be minimised. Their three main results are as follows: the topologies built by selfish agents are worse compared to the topologies built by agents in collaboration; the topologies constructed by selfish agents are never stable (i.e., there is always a change in the topology); and determining a pure Nash equilibrium is NP-complete here. This aspect of selfish agents is part of our future work and is discussed in detail in the next section. However, as motivated in the introduction and in the background chapters, for us the solution concept of Nash equilibrium is not useful and we use bilateral stability instead.

When looking at P2P systems more closer to our social storage systems, P2P social networking is one such area [120, 121]. Topology formation is one of the concerns here [121]. We believe that our solution concept of bilateral stability has its theoretical consequences in determining which bilaterally stable topology emerges in P2P social networking.

8.3 Research Implications

As mentioned in Chapter 1, in the social cloud literature, the issues of low service availability (for example, data and storage availability) and imbalanced workload (that lead to low storage utilization) are strongly correlated with the number of social contacts. In fact, the study [3] looks how to balance the workloads between the agents having high and low neighbour sets as an open problem. The studies [23, 24] shown that the small friend set is a cause of low service availability as well as poor storage utilization. However, it is worth noting that these findings are drawn in the context of exogenous social contacts.

We have shown that, in particular, for the given utility function if agents are allowed to select their partners, then in the context of symmetric social storage network (Chapter 4), each agent selects $\hat{\eta}$ friends, therefore, the resource sharing network formation leads to the $\hat{\eta}$-regular network. We conjecture that, in the context of the symmetric social storage network, there is a uniformity in the distribution of the workload as each agent has the same number of neighbours. However, in the context of the symmetric social storage cloud (Chapter 5), we inferred that if agents select their partners by looking at their cost-benefit trade-off, then the issues discussed above are more significant than in the context of exogenous social contacts.

We believe that the analysis of storage availability and network formation performed by us have several advantages from the point of view of storage providers (for example, BuddyBackup, CrashPlan, Friendstore). The analysis of network stability may help design efficient strategies related to data redundancy that suggest how many data pieces are needed on the storage space provided by partners in order to achieve the required level of data availability. It also helps design efficient workload strategies to maximize storage utilization. One of the advantages of endogenous network formation is that it provides more control to agents on their data and in selecting their storage partners. Further, our approach of analysis of network stability is useful for the agents who are part of the Friendstore storage system—it is easy for them to calculate their maintainable capacity [12] so as to maximize their storage utilization and data reliability.

It is widely accepted that network structure plays a crucial role in determining externalities, our findings clearly show that network size is also an equally important factor. Our results relating network size and the distance between agents to externalities would be useful to social cloud business startups for policy making. For instance, the policy may include the use (or recommendation) of friends as backup partners, to avoid negative externalities or to minimize the number of non-beneficiaries (for example, by choosing agents for data backup who are far away from the agent requesting backup). Although our results are for ring networks, they can be extended to other network structures too. Secondly, our approach of relating externalities and group well-being (that is, measuring beneficiaries and non-beneficiaries) enriches the transfer-based network formation model [67] (where an agent subsidizes another agent to form or not form link(s) with others), by incorporating group subsidization. In particular, agents can subsidize a pair of agents involved in a link formation, instead of individual subsidization. For example, in some research and development settings, where a set of beneficiary-firms are willing to pay a pair of firms that would like to collaborate [61]. Using the approach and results discussed in the paper, one can model this situation either as collective subsidization or as bargaining on link formation. In this case, a set of non-beneficiary-firms will pay for the pair of firms to not collaborate with each other, or a set of beneficiary-firms will pay for the pair of firms to collaborate with each other. Alternatively, both beneficiaries and non-beneficiaries may bargain for link formation.

8.4 Limitations

Despite the above advantages and implications, our study has several limitations. Firstly, all the social cloud models, namely, the *social storage network* model, the *social storage cloud model* and the *social compute cloud* model in Chapter 3 stand on the assumption (similar to various network formation models [58, 78, 63, 122]) that agents have complete information about the network structure. Though in the context of the social storage cloud model, we do not require this assumption during network formation as, owing to Theorem 5.7, links form (at most) pairwise, this assumption is crucial for our analysis in Section 5.1 on closeness and distances.

Secondly, although the proposed utility model captures various parameters essential for understanding social storage cloud formation, we cannot rule out that parameters like online availability of agents, trust between them, and the bandwidth they have may influence the network formation.

In this study, though the proposed utility functions are for heterogeneous agents, the analysis is limited to homogeneous agents (or symmetric social storage cloud systems). In the case of heterogeneous agents, it would be interesting to see how externalities will influence social cloud formation. In fact, the analysis of stability, efficiency and externalities (in terms of resource availability) will also be more relevant in this setting.

8.5 Future Work

In this section, we discuss future work regrading utility functions, stability, and efficiency.

Utility Functions

The proposed utility functions can be enriched by taking the above mentioned parameters (that is, online availability, bandwidth, and trust) into account. One can then study social cloud formation with both complete and incomplete information. For example, in the incomplete information setting, agents know neither the network structure nor the online availability and bandwidth of others. Analysis in this context will give more insight into social cloud formation.

For the MO-framework (Chapter 3), we use a convex combination of our two objective functions (maximizing data reliability and minimizing the total cost of the link), and this is no longer a case of Multi-Objective (MO) optimization. Since the solution of the convexly combined problem may not always be the solution of the original MO problem, finding a Pareto frontier (path followed by most MO algorithms) is part of our future work.

Agent Behaviour

In all our discussions, we have assumed that any pair of agents can potentially form a link. In scenarios where agents do not necessarily trust all agents in the network, our results on bilateral stability (discussed in Chapter 4) extend to every clique (of mutually trusting agents) in the network. If not all agents trust each other, we may use an extension of the Hall's marriage theorem [123] to aid independent observers determine whether it is possible to form an efficient network or not.

In our current work, we have not focused on the heterogeneous behaviour of agents in social storage settings. Although incorporating complex and heterogeneous behaviour of agents into our models is closer to real world scenarios, this would make it difficult to deal with the model and as well as predict its outcome. Kuznetsov and Schmid [94] propose a social range matrix, which is a novel approach to deal with heterogeneous behaviour of agents in the network. In particular, social range matrices capture three scenarios: anarchy, monarchy and coalitions. In anarchy, each agent is selfish. In monarchy, agents care about only one agent in the network. In the coalitions scenario, agents support each other within the same coalition but act selfishly or maliciously towards agents in other coalitions. Investigating the applicability of the social range matrix in our earlier proposed frameworks and bilateral stability is part of our future work.

Solution Concepts: Stability Analysis

Coming to the solution concept, if we had used the concept of *Pairwise Nash Stability* as defined by [73] and had applied the mutual consent requirement for deletion too, we would have obtained as for the degree-based utility function proposed in the social storage network context (Chapter 3). This is because the mutual consent requirement for addition and deletion overrides the requirement for Nash equilibrium. We are currently working on modifying the definition of Pairwise Nash Stability to multiple other scenarios. Looking at strong and coalition-proof Nash equilibria [124], strong pairwise stability [72], and farsighted equilibrium [74], are also future research directions.

Inefficiency

In the case the *social storage network model*, we have discussed network efficiencies but not looked at inefficiencies (discussed in Chapter 4). For *the social storage cloud model*, we examine efficiency and its contrapositive, that, the inefficiencies in the network (discussed in Chapter 5). The price of anarchy is an interesting measure to analyse the extent to which a network is inefficient [125, 126]. By definition, this means ratio of the worst sum of the utility of agents in an equilibrium network to the best sum of the utility. For the social storage network model, we have bilaterally stable networks in place of equilibrium networks. In our case, efficient networks are the ones with the best sum of utility that can be compared (using Proposition 4.7). However, knowing the worst sum of utility is non-trivial. The neighbourhood size of every agent that would give us this sum is challenging.

Bibliography

[1] M. Tennekes. *Network formation games*. PhD thesis, Maastricht University, 2010.

[2] K. Chard, S. Caton, O. Rana, and K. Bubendorfer. Social clouds: A retrospective. *IEEE Cloud Computing*, 2(6):30–40, 2015.

[3] X. Zuo and A. Iamnitchi. A survey of socially aware peer-to-peer systems. *ACM Comput. Surv.*, 49(1):9:1–9:28, 2016.

[4] K. Chard, K. Bubendorfer, S. Caton, and O. F. Rana. Social cloud computing: A vision for socially motivated resource sharing. *IEEE Transactions on Services Computing*, 5(4):551–563, 2012.

[5] L. F. Sarmenta. *Volunteer computing*. PhD thesis, Massachusetts Institute of Technology, 2001.

[6] D. Barkai. An introduction to peer-to-peer computing. Technical report, Intel Developer update magazine, February 2 2000.

[7] I. Foster, C. Kesselman, and S. Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. *International Journal of High Performance Computing Applications*, 15(3):200–222, 2001.

[8] C. S. Yeo, M. D. de Assunção, J. Yu, A. Sulistio, S. Venugopal, M. Placek, and R. Buyya. Utility computing on global grids. In H. Bidgoli, editor, *Handbook of Computer Networks*, pages 110–130. John Wiley & Sons, Inc., 2007.

[9] M. Hogan, F. Liu, A. Sokol, and J. Tong. Nist cloud computing standards roadmap. Technical report, National Institute of Standards and Technology, 2011.

[10] N. Yaraghi and S. Ravi. The current and future state of the sharing economy. IMPACT Series 032017, The Brookings Institution India Center, March 2017.

[11] A. Moreno-Martínez, R. Gracia-Tinedo, M. Sánchez-Artigas, and P. Garcia-Lopez. Friendbox: A cloudified f2f storage application. In *2012 IEEE 12th International Conference on Peer-to-Peer Computing (P2P)*, pages 75–76, Tarragona, Spain, 3–5 September 2012. IEEE.

[12] N. Tran, F. Chiang, and J. Li. Efficient cooperative backup with decentralized trust management. *Trans. Storage*, 8(3):8:1–8:25, 2012.

[13] T. D. Nguyen and J. Li. Blockparty: Cooperative offsite backup among friends. In *4th USENIX Symposium on Networked Systems Design & Implementation*, Cambridge, MA, April 11–13 2007. USENIX.

[14] R. Gracia-Tinedo, M. Sánchez-Artigas, and P. Garcia-Lopez. F2box: Cloudifying f2f storage systems with high availability correlation. In *2012 IEEE Fifth International Conference on Cloud Computing*, pages 123–130, Honolulu, HI, USA, 24–29 June 2012. IEEE.

[15] J. Li and F. Dabek. F2f: Reliable storage in open networks. In *The 5th international workshop on peer-to-peer systems (IPTPS'06)*, pages 1–6, Santa Barbara, CA, USA, 27–28 February 2006.

[16] R. Gracia-Tinedo, M. Sánchez-Artigas, and P. Garcia-Lopez. Friendbox: A hybrid f2f personal storage application. In *2012 IEEE Fifth International Conference on Cloud Computing*, pages 131–138, Honolulu, HI, USA, 24–29 June 2012. IEEE.

[17] S. Caton, C. Dukat, T. Grenz, C. Haas, M. Pfadenhauer, and C. Weinhardt. Foundations of trust: Contextualising trust in social clouds. In *2012 Second International Conference on Cloud and Green Computing*, pages 424–429, Xiangtan, China, 1–3 November 2012. IEEE.

[18] C. Haas, S. Caton, and C. Weinhardt. Engineering incentives in social clouds. In *2011 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pages 572–575, Newport Beach, CA, USA, 23–26 May 2011. IEEE.

[19] M. Punceva, I. Rodero, M. Parashar, O. F. Rana, and I. Petri. Incentivising resource sharing in social clouds. *Concurrency and Computation: Practice and Experience*, 27(6):1483–1497, 2015.

[20] S. Caton, C. Haas, K. Chard, K. Bubendorfer, and O. F. Rana. A social compute cloud: Allocating and sharing infrastructure resources via social networks. *IEEE Transactions on Services Computing*, 7(3):359–372, 2014.

[21] C. Haas, S. O. Kimbrough, S. Caton, and C. Weinhardt. Preference-based resource allocation: Using heuristics to solve two-sided matching problems with indifferences. In J. Altmann, K. Vanmechelen, and O.F. Rana, editors, *Economics of Grids, Clouds, Systems, and Services*, volume 8193 of *Lecture Notes in Computer Science*, pages 149–160. Springer International Publishing, 2013.

[22] M. I. S. Oliveira, W. Cirne, F. Brasileiro, and D. Guerrero. On the impact of the data redundancy strategy on the recoverability of friend-to-friend backup systems. In *26th Brazilian Symposium on Computer Networks and Distributed Systems*, 2008.

[23] R. Sharma, A. Datta, M. DeH'Amico, and P. Michiardi. An empirical study of availability in friend-to-friend storage systems. In *2011 IEEE International Conference on Peer-to-Peer Computing*, pages 348–351, Kyoto, Japan, 31 Aug.-2 Sept. 2011. IEEE.

[24] X. Zuo, J. Blackburn, N. Kourtellis, J. Skvoretz, and A. Iamnitchi. The power of indirect ties in friend-to-friend storage systems. In *14-th IEEE International Conference on Peer-to-Peer Computing*, pages 1–5, London, UK, 8–12 September 2014. IEEE.

[25] M. E. Newman, D. J. Watts, and S. H. Strogatz. Random graph models of social networks. *Proceedings of the National Academy of Sciences*, 99(suppl 1):2566–2572, 2002.

[26] P. Erdős and A. Rényi. On the evolution of random graphs. *Publication of the Mathematical Institute of the Hungarian Academy of Science*, 5(1):17–60, 1960.

[27] M. O. Jackson. A survey of network formation models: Stability and efficiency. In G. Demange and M. Wooders, editors, *Group Formation in Economics*, pages 11–57. Cambridge University Press, 2005.

[28] B. Dutta and M. O. Jackson. On the formation of networks and groups. In B. Dutta and M. O. Jackson, editors, *Networks and Groups: Models of Strategic Formation (Studies in Economic Design)*, pages 1–15. Springer Berlin Heidelberg, 2003.

[29] A. M. Marco. Games of coalition and network formation: A survey. In A. K. Naimzada, S. Stefani, and A. Torriero, editors, *Networks, Topology and Dynamics*, pages 67–93. Springer, 2008.

[30] S. Borkotokey, L. Gogoi, and S. Sarangi. A survey of player-based and link-based allocation rules for network games. *Studies in Microeconomics*, 2(1):5–26, 2014.

[31] M. O. Jackson and A. Wolinsky. A strategic model of social and economic networks. *Journal of Economic Theory*, 71(1):44 – 74, 1996.

[32] S. Goyal and F. Vega-Redondo. Structural holes in social networks. *Journal of Economic Theory*, 137(1):460 – 492, 2007.

[33] F. Azzedin and M. Maheswaran. Evolving and managing trust in grid computing systems. In *IEEE Canadian Conference on Electrical and Computer Engineering, (CCECE)*, pages 1424–1429, Winnipeg, Manitoba, Canada,, 12–15 May 2002. IEEE.

[34] D. P. Anderson, E. Korpela, and R. Walton. High-performance task distribution for volunteer computing. In *Proceedings of the First International Conference on e-Science and Grid Computing*, pages 196–203, Melbourne, Vic., Australia, 5–8 July 2005. IEEE.

[35] A. Chandra and J. Weissman. Nebulas: Using distributed voluntary resources to build clouds. In *Proceedings of the 2009 conference on Hot topics in cloud computing*, HotCloud'09, San Diego, California, 14–19 June 2009. USENIX Association.

[36] K. Chard, S. Caton, O. Rana, and K. Bubendorfer. Social cloud: Cloud computing in social networks. In *2010 IEEE 3rd International Conference on Cloud Computing*, pages 99–106, Miami, FL, USA, 5–10 July 2010. IEEE.

[37] K. Bubendorfer, K. Chard, K. John, and A. M. Thaufeeg. eScience in the social cloud. *Future Generation Computer Systems*, 29(8):2143 – 2156, 2013.

[38] A. M. Thaufeeg, K. Bubendorfer, and K. Chard. Collaborative eResearch in a social cloud. In *2011 IEEE Seventh International Conference on eScience*, pages 224–231, Stockholm, Sweden, 5–8 December 2011. IEEE.

[39] K. John, K. Bubendorfer, and K. Chard. A social cloud for public eResearch. In *2011 IEEE Seventh International Conference on eScience*, pages 363–370, Stockholm, Sweden, 5–8 December 2011. IEEE.

[40] C. Batten, K. Barr, A. Saraf, and S. Trepetin. pStore: A secure peer-to-peer backup system. Technical Memo MIT-LCS-TM-632, Massachusetts Institute of Technology Laboratory for Computer Science, October 2002.

[41] L. P. Cox and B. D. Noble. Samsara: Honor among thieves in peer-to-peer storage. In *Proceedings of the nineteenth ACM symposium on Operating systems principles*, SOSP ’03, pages 120–132, NY, USA, 19–22 October 2003. ACM.

[42] M. Landers, H. Zhang, and K. L. Tan. Peerstore: better performance by relaxing in peer-to-peer backup. In *Proceedings Fourth International Conference on Peer-to-Peer Computing*, pages 72–79, Zurich, Switzerland, 27-27 August 2004. IEEE.

[43] S. Elnikety, M. Lillibridge, M. Burrows, and W. Zwaenepoel. Cooperative backup system. In *Proceedings of the 2003 Usenix Annual Technical Conference*, San Antonio, Texas, USA, 9–14 June 2003. USENIX.

[44] M. Lillibridge, S. Elnikety, A. Birrell, M. Burrows, and M. Isard. A cooperative internet backup scheme. In *Proceedings of the annual conference on USENIX Annual Technical Conference*, pages 29–41. San Antonio, Texas, USA, USENIX Association, June 09–14 2003.

[45] P. Hoberg, J. Wollersheim, and H. Krcmar. The business perspective on cloud computing - a literature review of esearch on cloud computing. In *Proceedings of the Eighteenth Americas Conference on Information Systems*, pages 1–12, Seattle, Washington, USA, 9–12 August 2012. AMCIS.

[46] E. Bayrak, J. P. Conley, and S. Wilkie. The economics of cloud computing. *The Korean Economic Review*, 27(2):203–230, 2011.

[47] P. Gayek, R. Nesbitt, H. Pearthree, A. Shaikh, and B. Snitzer. A web content serving utility. *IBM Systems Journal*, 43(1):43–63, 2004.

[48] J. Vykoukal, M. Wolf, and R. Beck. Services grids in industry – on-demand provisioning and allocation of grid-based business services. *Business & Information Systems Engineering*, 1(2):177–184, 2009.

[49] T. Puschmann and R. Alt. Sharing economy. *Business & Information Systems Engineering*, 58(1):93–99, 2016.

[50] V. Katz. Regulating the sharing economy. *Berkeley Technology Law Journal*, 30(4):1067–1126, 2015.

[51] L. Georgiadis, G. Iosifidis, and L. Tassiulas. On the efficiency of sharing economy networks. Available at: arXiv:1703.09669v, 2017.

[52] D. F. Ferguson, C. Nikolaou, J. Sairamesh, and Y. Yemini. Economic models for allocating resources in computer systems. In S. H. Clearwater, editor, *Market-Based Control: A Paradigm for Distributed Resource Allocation*, pages 156–183. World Scientific Publishing Co., Inc., January 1996.

[53] R. Buyya, H. Stockinger, J. Giddy, and D. Abramson. Economic models for management of resources in peer-to-peer and grid computing. In *International Symposium on The Convergence of Information Technologies and Communications*, pages 13–25, Denver, Colorado, USA, 20–24 August 2001. IEEE.

[54] S. K. Garg, C. Vecchiola, and R. Buyya. Mandi: a market exchange for trading utility and cloud computing services. *The Journal of Supercomputing*, 64(3):1153–1174, 2013.

[55] G. Wei, A. V. Vasilakos, Y. Zheng, and N. Xiong. A game-theoretic method of fair resource allocation for cloud computing services. *The Journal of Supercomputing*, 54(2):252–269, 2010.

[56] F. Teng and F. Magoules. Resource pricing and equilibrium allocation policy in cloud computing. In *2010 10th IEEE International Conference on Computer and Information Technology*, pages 195–202, Bradford, UK, 29 June-1 July 2010. IEEE.

[57] S. U. Khan and I. Ahmad. Non-cooperative, semi-cooperative, and cooperative games-based grid resource allocation. In *Proceedings 20th IEEE International Parallel & Distributed Processing Symposium*, Rhodes Island, Greece, 25–29 April 2006. IEEE.

[58] V. Bala and S. Goyal. A noncooperative model of network formation. *Econometrica*, 68(5):1181–1229, 2000.

[59] T. Morrill. Network formation under negative degree-based externalities. *International Journal of Game Theory*, 40(2):367–385, 2011.

[60] A. Calvó-Armengol. Job contact networks. *Journal of Economic Theory*, 115(1):191 – 206, 2004.

[61] M. O. Jackson. *Social and Economic Networks*. Princeton University Press, 2008.

[62] 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*, PODC '03, pages 347–351, New York, USA, 13 - 16 July 2003. ACM.

[63] T. Moscibroda, S. Schmid, and R. Wattenhofer. Topological implications of selfish neighbor selection in unstructured peer-to-peer networks. *Algorithmica*, 61(October):419–446, 2011.

[64] P. Möhlmeier, A. Rusinowska, and E. Tanimura. A degree-distance-based connections model with negative and positive externalities. *Journal of Public Economic Theory*, 18(2):168–192, 2016.

[65] S. Goyal. Social networks on the web. In M. Peitz and J. Waldfogel, editors, *The Oxford Handbook of the Digital Economy*, pages 434–459. Oxford University Press, 2012.

[66] R. Aumann and R. Myerson. Endogenous formation of links between players and coalitions: an application of the shapley value. In A. E. Roth, editor, *The Shapley Value*, pages 175–191. Cambridge: Cambridge University Press, 1988.

[67] F. Bloch and M. O. Jackson. The formation of networks with transfers among players. *Journal of Economic Theory*, 133(1):83 – 110, 2007.

[68] F. Bloch and B. Dutta. Communication networks with endogenous link strength. *Games and Economic Behavior*, 66(1):39 – 56, 2009.

[69] R. B. Myerson. Graphs and cooperation in games. *Mathematics of Operations Research*, 2(3):225–229, 1977.

[70] A. Calvó-Armengol and R. İlkılıç. Pairwise-stability and nash equilibria in network formation. *International Journal of Game Theory*, 38(1):51–79, 2009.

[71] B. D. Bernheim, B. Peleg, and M. D. Whinston. Coalition-Proof Nash equilibria I. Concepts. *Journal of Economic Theory*, 42(1):1 – 12, 1987.

[72] M. O. Jackson and A. Van den Nouweland. Strongly stable networks. *Games and Economic Behavior*, 51(2):420 – 444, 2005. Special Issue in Honor of Richard D. McKelvey.

[73] S. Goyal and S. Joshi. Unequal connections. *International Journal of Game Theory*, 34(3):319–349, 2006.

[74] B. Dutta, S. Ghosal, and D. Ray. Farsighted network formation. *Journal of Economic Theory*, 122(2):143 – 164, 2005.

[75] P. J. J. Herings, A. Mauleon, and V. Vannetelbosch. Farsightedly stable networks. *Games and Economic Behavior*, 67(2):526 – 541, 2009.

[76] S. D. Flåm and C. Horvath. Network games; adaptations to nash-cournot equilibrium. *Annals of Operations Research*, 64(1):179–195, 1996.

[77] R. P. Gilles and S. Sarangi. Network formation under mutual consent and costly communication. *Mathematical Social Sciences*, 60(3):181–185, 2010.

[78] C. Johnson and R. P. Gilles. Spatial social networks. In D. Bhaskar and M. O. Jackson, editors, *Networks and Groups*, Studies in Economic Design, pages 51–77. Springer Berlin Heidelberg, 2003.

[79] G. Doğan, M. A. van Assen, A. van de Rijt, and V. Buskens. The stability of exchange networks. *Social Networks*, 31(2):118–125, 2009.

[80] B. Buechel. *Advances in Strategic Network Formation: Preferences, Centrality, and Externalities*. PhD thesis, Bielefeld University, 2009.

[81] P. Belleflamme and F. Bloch. Market sharing agreements and collusive networks. *International Economic Review*, 45(2):387–411, 2004.

[82] S. Goyal and S. Joshi. Bilateralism and free trade. *International Economic Review*, 47(3):749–778, 2006.

[83] S. Currarini. Network design in games with spillovers. *Review of Economic Design*, 10(4):305–326, 2007.

[84] S. Joshi and A. S. Mahmud. Network formation under multiple sources of externalities. *Journal of Public Economic Theory*, 18(2):148–167, 2016.

[85] A. Mohaisen, H. Tran, A. Chandra, and Y. Kim. Trustworthy distributed computing on social networks. *IEEE Transactions on Services Computing*, 7(3):333 – 345, 2014.

[86] H. Weatherspoon and J. D. Kubiatowicz. Erasure coding vs. replication: A quantitative comparison. In P. Druschel, F. Kaashoek, and A. Rowstron, editors, *Peer-to-Peer*

Systems (IPTPS 2002), volume 2429 of *Lecture Notes in Computer Science*, pages 328–337. Berlin, Heidelberg:Springer, 2002.

[87] J. W. Pratt. Risk aversion in the small and in the large. *Econometrica*, 32(1-2):22–136, 1964.

[88] J. Meyer. Representing risk preferences in expected utility based decision models. *Annals of Operations Research*, 176(1):179–190, 2010.

[89] P. Boldi and S. Vigna. Axioms for centrality. *Internet Mathematics*, 10(3-4):222–262, 2014.

[90] T. Opsahl, F. Agneessens, and J. Skvoretz. Node centrality in weighted networks: Generalizing degree and shortest paths. *Social Networks*, 32(3):245–251, 2010.

[91] M. Marchiori and V. Latora. Harmony in the small-world. *Physica A: Statistical Mechanics and its Applications*, 285(3):539–546, 2000.

[92] R. Gracia-Tinedo, M. S. Artigas, and P. G. López. Analysis of data availability in f2f storage systems: When correlations matter. In *2012 IEEE 12th International Conference on Peer-to-Peer Computing (P2P)*, pages 225–236, Tarragona, Spain, 3–5 September 2012. IEEE.

[93] L. P. Cox, C. D. Murray, and B. D. Noble. Pastiche: Making backup cheap and easy. In *Proceedings of the 5th symposium on Operating systems design and implementation*, pages 285–298, Boston, Massachusetts, USA, 09–11 December 2002. USENIX Association.

[94] P. Kuznetsov and S. Schmid. Towards network games with social preferences. In B. Patt-Shamir and T. Ekim, editors, *17th International Colloquium on Structural Information and Communication Complexity*, pages 14–28, Sirince, Turkey, 7–11 June 2010. Springer Berlin Heidelberg.

[95] H. Jain, G. S. Teja, P. Mane, K. Ahuja, and N. Krishnamurthy. Data backup network formation with heterogeneous agents: (extended abstract). In *2018 10th In-*

ternational Conference on Communication Systems Networks (COMSNETS), pages 418–420, Bengaluru, India, 3–7 January 2018. IEEE.

[96] D. Shuanping. *Networks of Institutions: Institutional Emergence, Social Structure and National Systems of Policies*. Routledge:Taylor & Francis, 2015.

[97] R. L. Breiger and P. E. Pattison. Cumulated social roles: The duality of persons and their algebras. *Social Networks*, 8(3):215 – 256, 1986.

[98] A. Chandrasekhar. Econometrics of network formation. In Bramoullé Y., A. Galeotti, and B. Rogers, editors, *The Oxford Handbook of the Economics of Networks*. Oxford University Press, 2016.

[99] C. Keser, K. M. Ehrhart, and S. K. Berninghaus. Coordination and local interaction: experimental evidence. *Economics Letters*, 58(3):269–275, 1998.

[100] R. D. Vallam, C. A. Subramanian, R. Narayanam, Y. Narahari, and N. Srinath. Strategic network formation with localized pay-offs. *Studies in Microeconomics*, 2(1):63–119, 2014.

[101] E. Gallo and C. Yan. Efficiency and equilibrium in network games: An experiment. Working Paper Series No: 2015/03, Cambridge -INET Institute, January 2015.

[102] J. D. Carrillo and A. Gaduh. The strategic formation of networks: Experimental evidence. CEPR Discussion Papers 8757., 2012.

[103] E. Gallo. Communication networks in markets. https://doi.org/10.17863/CAM.5846, 2014.

[104] N. J. Sloane. An on-line version of the encyclopedia of integer sequences. *The Electronic Journal of Combinatorics*, 1(F1):1–5, 1994.

[105] T. H. Naylor, D. S. Burdick, and W. E. Sasser. Computer simulation experiments with economic systems: The problem of experimental design. *Journal of the American Statistical Association*, 62(320):1315–1337, 1967.

[106] A. Cassar and D. Friedman. Economists go to the laboratory: Who, what, when and why? In A. Cassar and D. Friedman, editors, *Economics Lab: An Intensive Course in Experimental Economics*, pages 26–36. Routledge, 2004.

[107] S. Choi, E. Gallo, and S. Kariv. Networks in the laboratory. In Bramoullé Y., A. Galeotti, and B. Rogers, editors, *The Oxford Handbook on the Economics of Networks*. Oxford University Press, 2016.

[108] M. Scholz. Node similarity as a basic principle behind connectivity in complex networks. *Journal of Data Mining & Digital Humanities*, jdmdh:33(January):1–7, 2015.

[109] A. Lazar, D. Chavalarias, and Ahn T. K. Endogenous network formation and the evolution of preferences. Bloomington: Indiana University, Workshop In Political Theory and Policy Analysisd, 2003.

[110] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*. Cambridge University Press, 1994.

[111] S. Dhamal and Y. Narahari. Formation of stable strategic networks with desired topologies. *Studies in Microeconomics*, 3(2):158–213, 2015.

[112] B. Buechel and T. Hellmann. Under-connected and over-connected networks: the role of externalities in strategic network formation. *Review of Economic Design*, 16(1):71–87, 2012.

[113] N. P. Hummon. Utility and dynamic social networks. *Social Networks*, 22(3):221 – 249, 2000.

[114] A. Falk and M. Kosfeld. It's all about connections: Evidence on network formation. Discussion papers 777, University of Zurich, IZA, 2003.

[115] J. K. Goeree, A. Riedl, and A. Ule. In search of stars: Network formation among heterogeneous agents. *Games and Economic Behavior*, 67(2):445 – 466, 2009.

[116] K. Rzadca, A. Datta, and S. Buchegger. Replica placement in p2p storage: Complexity and game theoretic analyses. In *Proceedings of the 2010 IEEE 30th International*

Conference on Distributed Computing Systems, pages 599–609, Genoa, Italy, 21–25 June 2010. IEEE.

[117] K. Rzadca, A. Datta, G. Kreitz, and S. Buchegger. Game-theoretic mechanisms to increase data availability in decentralized storage systems. *ACM Transactions on Autonomous and Adaptive Systems*, 10(3):14:1–14:32, 2015.

[118] L. Toka and P. Michiardi. Analysis of user-driven peer selection in peer-to-peer backup and storage systems. *Telecommunication Systems*, 47(1):49–63, 2011.

[119] T. Moscibroda, S. Schmid, and R. Wattenhofer. On the topologies formed by selfish peers. In *Proceedings of the Twenty-fifth Annual ACM Symposium on Principles of Distributed Computing*, PODC '06, pages 133–142, New York, NY, USA, 23–26 July 2006. ACM.

[120] S. Buchegger, D. Schiöberg, L. H. Vu, and A. Datta. Peerson: P2p social networking: Early experiences and insights. In *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, SNS '09, pages 46–52, New York, NY, USA, 31–31 March 2009. ACM.

[121] S. Buchegger and A. Datta. A case for p2p infrastructure for social networks - opportunities challenges. In *2009 Sixth International Conference on Wireless On-Demand Network Systems and Services*, pages 161–168, Snowbird, UT, USA, 2–4 February 2009. IEEE.

[122] B. Buechel. Network formation with closeness incentives. In A. K. Naimzada, S. Stefani, and A. Torriero, editors, *Networks, Topology and Dynamics: Theory and Applications to Economics and Social Systems*, pages 95–109. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[123] P. Hall. On representatives of subsets. *Journal of the London Mathematical Society*, 10(1):26–30, 1935.

[124] B. Dutta and S. Mutuswami. Stable networks. *Journal of Economic Theory*, 76(2):322–344, 1997.

[125] C. H. Papadimitriou. Algorithms, games, and the internet. In *Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing*, STOC '01, pages 749–753, Heraklion, Crete, Greece, 06–08 July 2001. ACM.

[126] E. D. Demaine, M. Hajiaghayi, H. Mahini, and M. Zadimoghaddam. The price of anarchy in network creation games. *ACM Transactions on Algorithms*, 8(2):13:1–13:13, 2012.

[127] R. Wooten, R. Klink, F. Sinek, Y. Bai, and M. Sharma. Design and implementation of a secure healthcare social cloud system. In *2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, pages 805–810, Ottawa, ON, Canada, 13–16 May 2012. IEEE.

[128] Y. Zhang and M. van der Schaar. Incentive provision and job allocation in social cloud systems. *IEEE Journal on Selected Areas in Communications*, 31(9):607–617, 2013.

[129] S. Xu and M. Yung. Socialclouds: Concept, security architecture and some mechanisms. In L. Chen and M. Yung, editors, *Proceedings of the First International Conference on Trusted Systems*, INTRUST'09, pages 104–128. Springer-Verlag, 2010.

[130] R. Pezzi. Information technology tools for a transition economy, September 2009. https://socialcloud.net/papers/ITtools.pdf.

[131] Z. Ali, R. ur Rasool, P. Bloodsworth, and S. B. Mansoor. Facebook-based cloud resource sharing. *Computers & Electrical Engineering*, 66(February):162 – 173, 2018.

[132] Z. Ali, R. U. Rasool, and P. Bloodsworth. Social networking for sharing cloud resources. In *Second International Conference on Cloud and Green Computing (CGC)*, pages 160–166, Xiangtan, China, 1–3 November 2012. IEEE.

[133] D. P. Anderson. Boinc: A system for public-resource computing and storage. In *Proceedings of the 5th IEEE/ACM International Workshop on Grid Computing*, GRID '04, pages 4–10, Pittsburgh, PA, USA, 8–8 November 2004. IEEE.

[134] V. Chang. A cybernetics social cloud. *Journal of Systems and Software*, 124(February):195 – 211, 2017.

[135] M. Sato. Creating next generation cloud computing based network services and the contributions of social cloud operation support system (oss) to society. In *2009 18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, pages 52–56, Groningen, Netherlands, 29 June-1 July 2009. IEEE.

[136] F. Moyano, C. Fernandez-Gago, and J. Lopez. A framework for enabling trust requirements in social cloud applications. *Requirements Engineering*, 18(4):321–341, 2013.

[137] K. Levy, B. Sargent, and Y. Bai. A trust-aware tag-based privacy control for ehealth 2.0. In *Proceedings of the 2011 Conference on Information Technology Education*, pages 251–256, West Point, New York, USA, 20–22 October 2011. ACM.

[138] A. Satsiou and L. Tassiulas. Reputation-based resource allocation in p2p systems of rational users. *IEEE Transactions on Parallel and Distributed Systems*, 21(4):466–479, 2010.

[139] M. Gupta, P. Judge, and M. Ammar. A reputation system for peer-to-peer networks. In *Proceedings of the 13th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, NOSSDAV '03, pages 144–152, Monterey, CA, US, 01–03 June 2003. ACM.

[140] K. M. Sim. A survey of bargaining models for grid resource allocation. *SIGecom Exch.*, 5(5):22–32, 2006.

[141] D. Banerjee, S. Saha, S. Sen, and P. Dasgupta. Reciprocal resource sharing in p2p environments. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '05, pages 853–859, The Netherlands, 25–29 July 2005. ACM.

[142] M. Mowbray, F. Brasileiro, N. Andrade, J. Santana, and W. Cirne. A reciprocation-based economy for multiple services in peer-to-peer grids. In *Sixth IEEE International Conference on Peer-to-Peer Computing (P2P'06)*, pages 193–202, Cambridge, UK, 6–8 September 2006. IEEE.

[143] C. Haas, S. Caton, K. Chard, and C. Weinhardt. Co-operative infrastructures: An economic model for providing infrastructures for social cloud computing. In *2013 46th Hawaii International Conference on System Sciences*, pages 729–738, Wailea, Maui, HI, USA, 7–10 January 2013. IEEE.

[144] Z. HAN. *PEERSTORE: A Peer-to-Peer Backup System*. PhD thesis, National University of Singapore, 2004.

[145] A. Chervenak, V. Vellanki, and Z. Kurmas. Protecting file systems: A survey of backup techniques. In *Joint NASA and IEEE Mass Storage Conference*, Maryland, USA, 23–26 March 1998. IEEE.

[146] M. O. Killijian, L. Court'es, and D. Powell. A survey of cooperative backup mechanisms. Technical report, HAL-Inria, 2006. hal-0013969.

[147] B. Studer. Introduction: A survey of p2p backup networks. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.85.2416&rep=rep1&type=pdf.

[148] J. Åslund. Authentication in peer-to-peer systems. Technical report, Linköping University, 2002.

[149] R. G. Tinedo. *On Personal Storage Systems: Architecture and Design Considerations*. PhD thesis, Rovira i Virgili University, 2015.

[150] L. C. Freeman. Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215 – 239, 1978.

[151] S. P. Borgatti and M. G. Everett. A graph-theoretic perspective on centrality. *Social Networks*, 28(4):466 – 484, 2006.

[152] F. Bloch, M. O. Jackson, and P. Tebaldi. Centrality measures in networks. Available at arXiv: https://arxiv.org/abs/1608.05845, 2017.

[153] O. Skibski and J. Sosnowska. Axioms for distance-based centralities. In *Thirty-Second AAAI Conference on Artificial Intelligence*, pages 1218–1225. AAAI Publications, 2018.

[154] A. Bavelas. Communication patterns in task-oriented groups. *The Journal of the Acoustical Society of America*, 22(6):725–730, 1950.

[155] F. Harary. Status and contrastatus. *Sociometry*, 22(1):23–43, 1959.

[156] G. Sabidussi. The centrality index of a graph. *Psychometrika*, 31(4):581–603, 1966.

[157] M. A. Beauchamp. An improved index of centrality. *Behavioral Science*, 10(2):161–163, 1965.

[158] D. L. Rogers. Sociometric analysis of interorganizational relations: Application of theory and measurement. *Rural Sociology*, 39(4):487, 1974.

[159] R. L. Moxley and N. F. Moxley. Determining point-centrality in uncontrived social networks. *Sociometry*, 37(1):122–130, 1974.

[160] M. Wooldridge. Does game theory work? *IEEE Intelligent Systems*, 27(6):76–80, 2012.

[161] R. Graham. *Game Theory: Introduction and Applications*. Oxford University Press, Oxford: UK, 1st edition, 1997.

Appendix A

Social Cloud

In recent years, the idea of Social Cloud has been coming to the forefront of research. Social Cloud has appeared as a metaphor of resource (service) sharing on a social network, where users of the social network share resource available at their end with other users in the social network. In this section, we introduce the concept of social cloud by discussing different views, a few applications (that demonstrate the potential of social cloud), and various other aspects of it such as trust, security and resource trading mechanisms.

A.1 Social Cloud: Different Views

Research and development on Social Cloud is currently at its infancy. In the literature, there seems to be no general definition of Social Cloud (see Table A.1). The different views, as seen in Table A.1, indicates that nearly half of the studies [4, 36, 38, 39, 85, 20] consider Social Cloud as a social network group analogous to dynamic Virtual Organization (VO) [7] in a social network context. A VO like social network group is a set of users who collaboratively pool resources to achieve its certain common goal. A group can achieve their common goal by defining a set of policies regarding group membership, resource sharing, and so on. So one can view Social Cloud paradigm stands on the notion of collaboration between generally smaller, better-connected groups of social network users with more heterogeneous resources to share. The above kind of resource sharing gives an expression of *local resource sharing*.

Authors	View Social Cloud as
Chard et al. [4]	*'...Social Cloud is not representative of point-to-point ex-changes between users, rather it represents multipoint sharing within a whole community group...'*
John et al. [39]	*'...One way of thinking about the Social Cloud is to consider that social network groups are analogous to dynamic Virtual Organizations (VOs) [7]...'.*
Thaufeeg et al. [38]	*"Social Cloud Computing is a resource sharing framework in which resources and services are shared amongst individuals on the premise of the relationships and policies encoded in a social network."*
Mohaisen et al. [85]	*'...Our paradigm* and model are similar in many aspects to the conventional grid-computing paradigm...'.*
Wooten et al. [127]	*'...we describe our design** and prototype implementation of a social healthcare network over the cloud...'.*
Caton et al. [20]	*"A social cloud is a form of community cloud (as defined in NIST's definition of cloud computing [9]), as the resources are owned, provided and consumed by members of a social community".*
Zhang et al. [128]	*'...social cloud systems are constructed with peer-to-peer architectures with resources being owned and managed distributedly by individual users...'*
Xu et al. [129]	*'...SocialClouds are owned and operated by the contributors or providers, who are also SocialClouds users. This is in contrast to PublicClouds where the platforms underlying a cloud are often owned and managed by a single service provider, and to PrivateClouds where the platforms underlying a cloud are often owned and managed by a single enterprise...'*
Rafael Pezzi [130]	*"You may think of the Social Cloud concept as a blend of an auction web-site and a social networking website hosted at a cloud computing environment, which on its turn is running in a peer-to-peer network"*

Table A.1: Different social cloud views
*Here paradigm means Social Cloud, **Healthcare Social Cloud

On the other side, few studies [129, 130, 127, 128] view Social Cloud is in the form of Peer-to-Peer community (or peer-to-peer social networking) where each participant performs the same role as the others and each participant manages its own resources at its end. A participant who joins the network provides resources to others and avails resources provided by others. In this way, Social Cloud is neither subject to centralized control nor owned by a single entity. So one can view Social Cloud paradigm as collaboration between generally bigger, loosely connected group of users present at the edge of Internet sharing varied types of resources. The above kind of resource sharing gives an expression of *global sharing of resources*.

A.2 Social Cloud: Applications

Despite a lack of a clear view, the series of recently reported social cloud systems show the potential of Social Cloud to act as complimentary to various other distributed computing paradigms such as Community Cloud, Grid computing, Volunteer computing, or Networking Services. Note that, we draw such outward forms of Social Cloud on how Social Cloud is described while proposing a particular system and respective researchers explicit arguments. In this subsection, we survey a few existing Social Cloud systems. We classify these systems in different distributed computing settings and tabulated in Table A.2.

Social Storage Cloud (SSC) [4, 36]:
SSC is deployed as a Facebook application, where Facebook users can offer storage as a service to their Facebook friends. These users make use of either posted price or reverse auction market mechanisms for allocating storage space to others. The Facebook application is used for the purpose of currency regulation, serve as a marketplace (so that users can communicate with each other, and hence, resource trading can be possible) and establishment of service level agreement between a storage provider and a storage consumer.

Cloud Resource Bartering (CRB) [131, 132]:
CRB is a social cloud model that enables an on-line social network (e.g., Facebook) users to share a part of their owned *Amazon EC2* infrastructure with other users. Unlike SSC (where users share their storage space to others), users of CRB share a third party Cloud service, which is owned by them with their social network friends. Users perform such a resource sharing in the network through a bartering model.

Collaborative eResearch Social Cloud (CeRSC) [38]:
CeRSC facilitates scientists to collaborate and share computing resources for implementing computation intensive algorithms on large data sets in a virtualised research environment. In CeRSC, an online social network (Facebook) users share either their own Virtual-Machine (VM) images or VM images of third party vendors such as Amazon S3 with others in the

social network. Users or scientists who are members of Facebook, form a dynamic virtual organization for sharing their VM with each other. Users can adopt any market model such as trophy system, reputation, reciprocity model or volunteer contribution for VM sharing.

Public eResearch Social Cloud (PeRSC) [39]:

PeRSC integrates BOINC [133] platform[37] and Facebook so that social network users are able to donate computational resources (e.g., processing power, storage, etc.) to various scientific projects like SETI@Home. The goal of the authors in [39] is to bring out volunteer computing from technical users towards non-technical user domain (such as Facebook users). View of PeRSC from users' perspective and project server are different. From users' point of view, PeRSC is a Facebook application, whereas from project servers' point of view, it is an account management system.

Social Cloud:

Mohaisen et al. [85] presents Social Cloud concept as a grid computing paradigm standing on social networks where network users collaboratively construct a pool of computing resources (such as bandwidth, storage, and computing power). In this system, users perform a local resource sharing (users share resource with only their neighbors). Users altruistically share their resources with each other and perform a computational task on behalf of each other. In this system, users play a dual role; a *task outsourcer* and a *worker*. A task outsourcer is the one who outsources a task to its friends, and a worker is the one who performs an assigned task by an outsourcer.

Husky Healthcare Social Cloud (HHSC) [127]:

HHSC is a third party Cloud (like Amazon EC) assisted social health care network. HHSC facilitates sharing of health related information between standard users (either patients or the users who want to get health knowledge) and health professionals through blogs. In HHSC, each standard user creates its blogs and share it with other trusted standard users and health professionals. Whereas, a health professional can comment on all blogs shared

[37] http://boinc.berkeley.edu/

Social Cloud as a Community Cloud	
SSC [4]	Storage Social Cloud members offer under-utilized storage space to their friends *storage-as-service*.
CRB-Model [132]	Users share Cloud resources with friends within the social network through the Barter Model.
CeRSC [38]	Users form collaboration like VO ([7]) by sharing of computational resource (Virtual Machine).
Social Cloud as a Grid Computing	
Social Cloud [85]	A user (who is a task outsourcer) outsources a computational task to its friends (acts worker for the outsourcer). Then the friends performs computational task on behalf of the friend.
Social Cloud as a Volunteer Computing	
PeRSC [37]	Users in social network forms group and donate their computational resources for scientific computing like BOINC [133] and gain credit as in volunteer computing community.
Cybernetics Social Cloud [134]	Social Cloud system (a combination of BOINC and Facebook) deal with big data processing associated with social networks.
Social Cloud as a Networking Services	
HHSS [127]	Various kinds of users (for example, patients, physicians) share their blogs among each other to improve health care environment.
Social Cloud OSS [135]	The Social Cloud OSS provides a way to deliver social network services that currently are being provided with the help Cloud.

Table A.2: Classification of social cloud systems. Note that the most of the Social Cloud systems are in their prototype phases. Hence, we should not consider any rigid boundary for their classification.

by the standard users regardless of trust rating.

The above social cloud systems can be differentiated according to resources that are considered as a shareable entity, principle stakeholders, deployment mechanism, application and resource trading mechanism (see Table A.3).

Social Cloud Model	**Resource Definition**	**Stakeholder**	**Deployment**	**Application**	**Resource Trading Mechanism**
SSC [4]	Storage	Social Cloud members	Facebook Application	Storage-as-Service	Reverse auction, Posted Price
CRB-Model [132]	Amazon EC2	Social Cloud members	Facebook Application	-	Bartering
CeRSC [38, 37]	VM or Amazon S3	Scientist	Facebook Application	Platform-as-Service (PaaS)	-
PeRSC [39, 37]	VM, Storage, CPU, etc.	BOINC project and Facebook user	amalgamation of Facebook & BOINC	Volunteer Computing on Social Network	-
Social Cloud [85]	Storage	Social Cloud members	Overlay on Social Graph	Task outsourcing	Altruistic
HHSC [127]	blogs	Patients, physicians	Networking Services	Health-care	-

Table A.3: Summary of some existing social cloud systems

A.3 Social Cloud: Current Trends

In the previous subsection, we have glanced a few social cloud systems. Apart from social cloud development and deployment, researchers have been dealing with several other aspects such as trust, incentives for resource sharing, resource management, computational infrastructure, and security. Note that, a general framework of Social Cloud is absent. Researchers are focusing on the above aspects specific to a system, and hence, it is difficult for us to report the literature that covers the above aspects in detail. Therefore, in brief, we overview these aspects.

Trust:

Caton et al. [17], argue that trust plays an important role in motivating and joining social cloud systems. Further, trust is essential to maintain interpersonal relations, measure user's resource sharing ability and adhere to an informal agreement (if any). They also argue that the role of trust is required to finalize an exchange of resource (or service trading) between users.

Trust and reputations keep changing. To deal with the issue of change in trust and reputations, Moyano et al. [136] propose a framework called as *callable framework*. This provides a way for application developers to develop and deploy an application specific trust or reputation model.

In the context of HHSS, the authors make use of the *adaptive trust rating algorithm* [137]. The algorithm assists a user to quantify the trust value of other users by taking various parameters (like users' and their groups' availability and popularity) as an input.

Incentives:

By considering different Social Cloud settings, researchers have studied user incentivization for resource sharing, participation and desired behaviour.

Haas et al. [18] deal with the issue of user involvement and their behaviour in SSC. They identify three different phases namely discovery of participants, encouraging active participation and incentivising social behaviour. Every phase posts some incentive requirement. Further, they hold the view that hard service level agreement may not be an ideal approach to resolve social misbehaviour issue in SSC. They express a need to design a social oriented mechanism to curb misbehaviour of users.

John et al. [39] deal with the issue of user involvement, resource contribution in PeRSC. For this, they suggest a title (award) scheme. They introduce titles such as Project Champions and Social Anchors. The project champions title goes to those who give highest contribution (in terms of resource and time) for a given project. Social Anchors is titled to those who bring maximum new users in Social Cloud [39].

Punceva et al. [19] put a virtual currency in service to incentivize users for resource sharing. They consider a social cloud system where users (service exchanging entities) perform inter- and intra- group service trading by using virtual currency. The value of a user's currency depends upon its reputation value, which is based upon, how the provider offers a service to other users (who are consumers) and what feedback the provider gets from the consumers. Thus, a user's currency value fluctuates as its reputation changes.

Resource Management:

The resource management scheme varies from one resource sharing setup to another. Users who perform cooperative resource sharing can decide on a specific resource trading mechanism. To regulate and facilitate resource sharing and trading, researchers have suggested several models. Chard et al. [4],[36] considered various trading models, for example, reputation points [138][139], trophy [4], bargaining [140], reciprocation [141][142] and market metaphors such as posted price and auction that members of a Social Cloud can use. Haas et al. [21] proposed a preference-based resource allocation method. They use two sided matching approach to match user preferences, which specifies with whom a user would like to share its resources.

Computational Infrastructure:

Like any other cooperative computing, Social Cloud requires computational infrastructure that facilitates primary functionalities such as service advertisement, resource allocation, market-place etc. There are several sources that can provide required computational infrastructure (in terms of hardware, software etc.), for example, Cloud infrastructure or any third party vendor.

Haas et al. [143] present a cooperative economic model to construct computational infrastructure, where each social cloud member i donates resource endowment $\mathscr{X}_i$ to build the infrastructure. They present two contribution schemes: enforced fixed and voluntary varied. In the enforced fixed scheme, $\mathscr{X}_i = \mathbf{X}$ for all users i who are members of the Social Cloud and every user has to contribute $\mathbf{X}$ resource endowment. In the voluntary varied scheme, users contribute $\mathscr{X}_i$ as per their wish, that is, $\mathscr{X}_i \neq \mathbf{X}$ and may not equal to $\mathscr{X}_j$ form some user j.

Security:

Xu et al. [129], identify threat against data confidentiality, threat against data integrity, threats against data availability, threats against resource providers. To deal with these threats, they propose a CryptoOverlay (cryptographic overlaying) security architecture. CryptoOverlay preserve data and resource access controls and also cope with trust dynamics. CryptoOverlay consists of the following components: first, *privacy-preserving access control* enforces access control policies. Second, *periodic pro-activization* to overcome the potential availability weakness. Pro-activization is based on proactive cryptography to mitigate the possible damage by compromised virtual machines of providers. Third, *on demand proactivization* is an intrusion detection system that assures that provider's resource is not compromised by malware. Data confidentiality is ensured through data encryption. An integrity and availability of data is ensured by implementing advanced techniques such as the proof of retrievability.

Appendix B

Social Storage

Nowadays, one of the most important things for user is his (her) data, its loss can be catastrophic. Data backup systems keep important data of users safe from hardware or software failure. Since for more than a decade, significant efforts [40, 44, 93, 41, 144] have been made in building peer-to-peer (P2P) data backup systems. A peer-to-peer storage system allows its users to utilize disk space of each other to backup data. Despite a great number of advantages, due to anonymity and lack of trust in peers, these P2P systems are dealing with a number of challenges. For example, quality of service (e.g., data reliability, availability, storage availability, etc.), data privacy, security and confidentiality, system reliability, free riding, and heterogeneity. Numerous studies have described distinctive features and design issues of P2P backup system. Chervenak et al. [145], characterize a backup system, where they focus on backup techniques for protecting file systems. A detailed examination of backup systems is presented in [146], where they look at various peer-to-peer backup systems. The authors characterize backup system based on functional (e.g. full vs. incremental backup, resource usage, performance, etc.) and non-functional requirements (e.g. availability, privacy, integrity, etc.) of cooperative backup systems. The studies [43, 147] discuss various backup system design issues from security point of view such as data privacy, authentication and different security attacks.

In order to address the aforementioned issues associated with P2P backup systems, researchers have started to take advantage of social knowledge (connections). Numerous storage systems (for example, Friendstore [12], F2Box [14], FriendBox [11], BlockParty

[13], etc.) have proposed, under the tag of *Friend-to-Friend (F2F)* or *Social Storage*, an alternative to P2P storage systems. Researchers believe that the approach of leveraging social connections might be more useful to build stable, reliable and effective storage systems. It is believed that the social fabric incentivises users for proper behaviour in the system.

Recently, the first systematic survey of various P2P systems (which are leveraging social connections) is presented in [3] under the tag of socially-aware P2P systems. Although the survey is extensive and studied various F2F storage systems, there are several aspects of F2F storage that are left out. The aim of this section is to provide an overview of Social Storage or (F2F storage), which includes; the concept, its characteristics, and taxonomy. We refer the reader to the survey [3] for understanding of the current research trends and challenges of this field in details.

B.1 Social Storage: Architecture

The general purpose architecture[38] of Social Storage is shown in Figure B.1. The architecture consists of three components, namely, peer-to-peer (P2P) network, social knowledge and social storage applications.

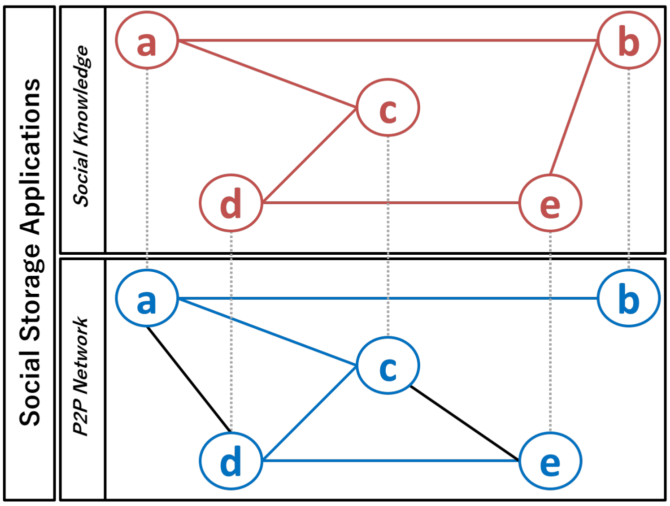


Figure B.1: A general-purpose architecture of Social Storage.

[38]This architecture bears a close resemblance to socially-aware P2P system architecture proposed in [3].

P2P Network:

A peer-to-peer (P2P) network is a set of autonomous peers (computers) that correspond to nodes, where the peers share their resources to perform a function in a distributed fashion and can leave and join the network randomly. Unlike client-server model (where a client makes a request of service and the server responds to the request), in a peer-to-peer (P2P) network, every peer plays a dual role of client and server, at the same time. In the above architecture, the P2P network layer manages resources in the decentralized infrastructure. This layer provides various other functionalities such as membership management, lookup services, message routing, and information retrieval [3].

Social Knowledge:

Social knowledge can be viewed as a social network, which can be obtained in the following three ways.

1. A social network can be obtained from existing online social networks, which is encapsulated in a social graph. Fitzpatrick and Recordon[39] define social graph as “the global mapping of everybody and how they' re related”. Iskold[40] views “social graph no different from a network, which is a more common term for describing the same thing. Graphs consist of nodes and edges, or things and the ways that things relate to each other”.

2. A social network is a result of the actions of the participants regarding whom they want to form social connections and with whom they do not, in the application context. In other words, a system acts as a social storage networking system, which provides a way for users to form data backup connections.

3. A social network can be obtained by integrating the above two approaches. In this scenario, a social graph is extracted for authentication purpose, where users then select their social backup partners who are members of that social graph (online social network).

[39]http://bradfitz.com/social-graph-problem/

[40]https://readwrite.com/2007/09/11/social graph concepts and issues/

B.2 Social Storage Application: Functionalities

Now, we list a few functionalities that social storage application (system) should offer to its users.

Data Placement:

A system distributes (places) agent's data on its backup partners' storage devices. For this, the system follows off the shelf data placement strategies such as replication or erasure coding [22].

Replication: it is a simple data placement technique, in which, a certain number of replicas of a data object are stored on a set of friend-nodes.

Erasure coding: it divides the data file into n blocks and re-codes into m blocks. A number of recorded m blocks are greater than the original n blocks. The reconstruction of the original data is possible with any n blocks. Note that the storage cost increases with factor n/m [86].

Data Encryption:

A data owner's file is made secret on the local machine by using an encryption technique/standards (e.g., Advanced Encryption Standard (AES), Data Encryption Standard (DES), public-private key pair, etc.) before sending data on friends' storage devices. The encryption technique is a choice of application or it depends on the requirement of the application. For example, *Friendstore* [12] system uses exclusive-or operation for data encoding as well as for maximising storage utilisation. *BuddyBackup* system implements AES-256 keys for file encryption. *BlockParty* [13, 15] uses a public-private key pair for data encryption and decryption.

Metadata:

Metadata contains information about backup partners and their IP addresses and storage related information. Data operations like data placement and its recovery require knowledge about data owner's partners, their IP addresses, and port numbers, the uploading bandwidth of data owner and downloading bandwidth of its backup partners. Although major systems allow data owners to store their data on immediate friend-nodes in the graph, however, in some cases, users are allowed to store data on indirect friend-nodes. In this scenario, the system maintains the list of direct and indirect friend-nodes. The above information is essential for the system to implement an efficient data placement.

Marketplace:
Users may offer their storage space as a service to backup partners, for example, *social cloud storage* [4]. This service provision is like commodity market, where storage provider decides prices of their resources on the basis of supply and demand of services in the system. A computational marketplace (like Mandi [54]) facilitates users to share (or trade) storage as a service.

Besides, it supports co-existence of multiple market models like auction, post price or tenders. A marketplace incorporates the following features: 1) storage service advertisement, 2) storage resource discovery, 3) support co-existences of multiple market models like auction, post price, bargaining, and so on. for storage trading, 4) facilitates users to share (or trade) storage for monetary payment or non-monetary way, and 5) service registrations where service providers can advertise their resources for trade and consumer can publish the required services.

B.3 Social Storage: Characteristics

As the idea of Social Storage is inspired by P2P backup systems, we believe that the comparison between these two systems aids us to understand the characteristics of Social Storage.

Storage Partner Selection:
Unlike P2P backup systems, in a social storage systems, users explicitly select their data backup partners (or friend-nodes) rather than randomly or suggested by the system. A data owner selects those users as storage-partners with whom the owner is involved in a real

world social relationship (e.g., friends, colleagues, relatives, and so on).

Authentication:

Authentication is a process of ensuring the identity of an entity. In P2P backup systems, authentication plays a key role in the identification of peers and their authenticity. There is an explicit need that each pair of peers or group of peers performs authentication before storage sharing [148]. Nevertheless, in P2P storage systems, it is difficult for a data owner to verify its storage partner's identity owing to the absence of a central authority. However, in a small and trustful network, various complexities related with authentication can be discounted [147]. From this point of view, in social storage, authentication process is less difficult as compared to P2P backup systems, as owner and storage partner (friend-node) both know each other's identity through an out of band communication (i.e. explicit communication).

Data Control and Location Transparency:

In some scenarios, users require control on their data as well as need to know the location of their data. The absence of data control and data location creates issues related to data privacy and security. In P2P backup systems, the process of backing up data is transparent to data owners. In a typical scenario, a data owner submits a file (that to be backed up) to the system (system software or client) and the client backs up on different peers (unknown to the data owner) through a specific data placement technique. This indicates that a data owner has less control over its data as well as is unaware about the location of its data. On the contrary, in social storage context, as a data owner stores its data on known friend-nodes, it has a high control over its data and is also aware of the location of its data.

B.4 Social Storage: Taxonomy

We classify Social Storage systems into three broad classes: *centralized*, *distributed* and *hybrid*.

Centralized Social Storage:

A centralized Social Storage takes the advantage of decentralised architecture as well to

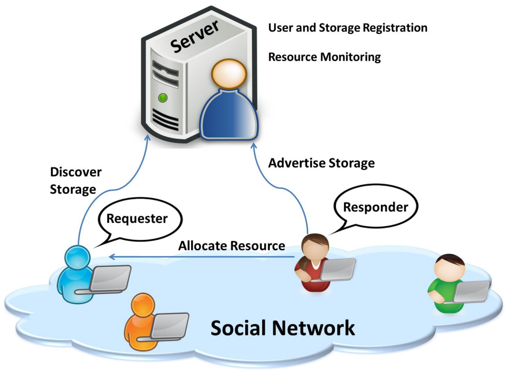


Figure B.2: Centralized social storage system

deploy a group based storage sharing (a set of users in a given social graph form a group). The centralization is similar to Client-Server set-up where one or multiple servers act as a coordinating entity, assist users to register their storage resources to share (or trade) with others. In a particular scenario (Figure B.2), at one side, a storage provider sends a message to the server containing information about its address and storage capacity. On the other side, a data owner sends a message to this server asking for information about storage availability and its providers. The decentralisation is reflected in data owner-storage provider communication. Once the owner gets the information about a storage provider from the server, further the owner can directly communicate with the storage provider without communicating with the server further.

Decentralised Social Storage:

A decentralized Social Storage can be considered as a different form of P2P backup systems [149] as shown in Fig. B.3. Users select their backup partners (whom they trust) explicitly as storage partners implying decentralisation [14]. In this system, each user shares storage space available at its end with other users. This system follows a typical backup scenario, where a user installs backup software on its machine. Next, a user selects other users (socially connected) as backup partners. A user initiates the process of data backup by selecting a file to be backed up. Then, the system places the

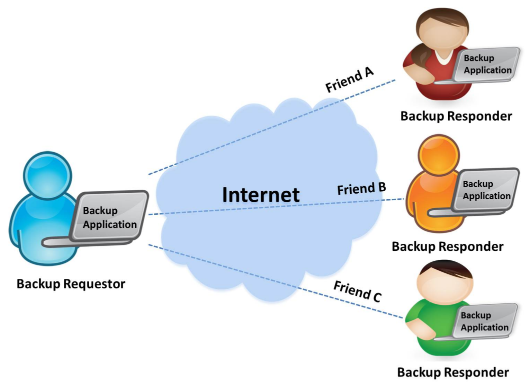


Figure B.3: Decentralized social storage system

data on users partners' storage devices by using a data placement technique like replication or Erasure coding (where replication and encoding are choices of the backup system).

Hybrid Social Storage:

Hybrid Social Storage combines decentralized Social Storage and Cloud storage service

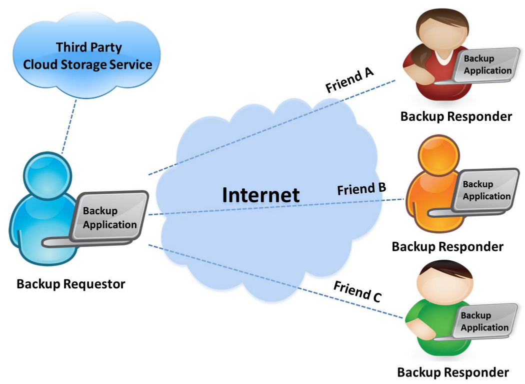


Figure B.4: Hybrid social storage system

(see Figure B.4). A hybrid Social Storage allows users to select their friends as storage partners as well as multiple third party cloud services to store their data. Thus, users can store their data either on friends' storage devices or on Cloud storage or on both. The system benefits those users who wish for high QoS in terms of data availability and long-term data storage. It also helps those who need high control over the data. When users look for data control, they can choose only friends as storage partners. In this sense, the system functions as a decentralised Social Storage. If users experience poor data availability, then the cloud storage can be used to store the part of the data to achieve the desired QoS. Cloud storage serves as a *temporary buffer* [14] to store the data blocks until some friend-nodes are available. The usage of Cloud service as a temporary buffer is advantageous to minimise block transfer time, and hence, the time to schedule. However, embodying a Cloud storage could be costly and would raise data security related issues.

Appendix C

Preview on Network, Centrality and Game Theory

C.1 Network: Basic Concepts and Structures

In this section, we define a few fundamental concepts and standard network structures we used in this thesis.

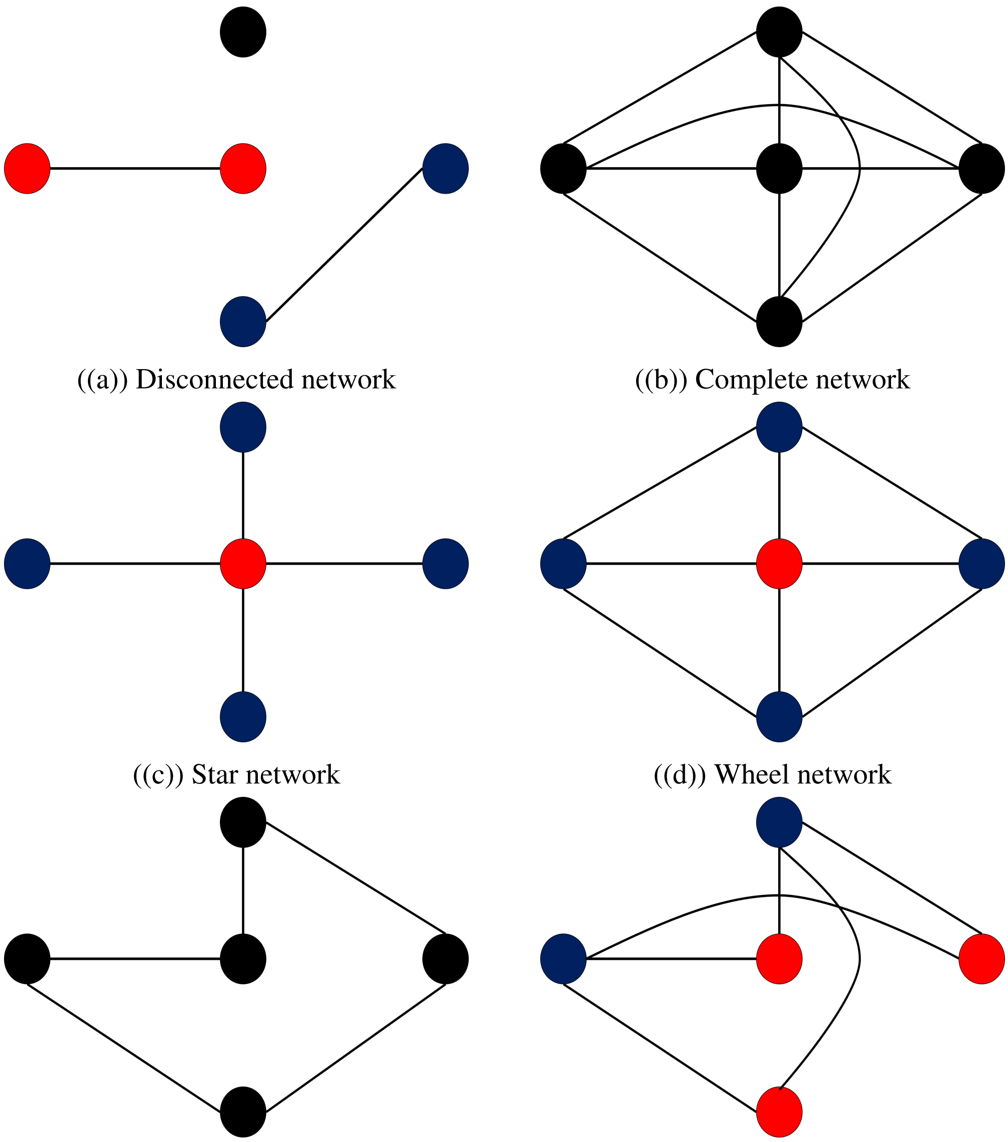
Definition C.1. *A network* $\mathfrak{g} = \{\bar{\mathscr{A}}, \bar{\mathscr{L}}\}$ *is a* **sub-network** *of* $\mathfrak{g} = \{\mathscr{A}, \mathscr{L}\}$ *if* $\bar{\mathscr{A}}$ *is a subset of* $\mathscr{A}$ *and* $\bar{\mathscr{L}}$ *is a subset of* $\mathscr{L}$.

Definition C.2. *A* **complement of network** $\mathfrak{g}$*, denoted by* $\mathfrak{g}^{\mathfrak{c}}$*, is a network on the same set of agents such that* $\langle ij \rangle \in \mathfrak{g}^{\mathfrak{c}}$ *if and only if* $\langle ij \rangle \notin \mathfrak{g}$.

Definition C.3. *A* ***null*** *(or empty) network is the one where there are no links — that is, no agent is connected to any agent.*

Definition C.4. *A* ***complete*** *network is the one where every agent is connected to every other agent (as shown in Figure C.1(b)). In a complete network* $\mathfrak{g}$*, for each agent i,* $\eta_i(\mathfrak{g}) = N - 1$.

Definition C.5. *An r-****regular*** *network is one where each agent has exactly r neighbours. A complete network* $\mathfrak{g}$ *(as shown in Figure C.1(b)) is an example of a regular network, where* $r = N - 1$.



((a)) Disconnected network ((b)) Complete network

((c)) Star network ((d)) Wheel network

((e)) Ring network ((f)) Bipartite network

Figure C.1: Example of standard network structures.

Definition C.6. *An N agent **star** network consists of a single universal agent and $N-1$ pendant agents. A universal agent is the one who is adjacent to other $N-1$ pendant agents. A pendant agent is the one who is adjacent to only the universal agent (as shown in Figure C.1(c)). A star component is a component which is a star (sub-)network.*

Definition C.7. *An N agent **wheel** network consists of a single universal agent and $N-1$ pendant agents where the universal agent is adjacent to other $N-1$ pendant agents and $N-1$ pendant agents are adjacent to exactly three agents (as shown in Figure C.1(d)).*

Definition C.8. *A component* $\mathfrak{g}^{\mathfrak{c}}$ *of* $\mathfrak{g}$ *is a* ***clique*** *if it is complete.*

Definition C.9. *A* **ring (circle)** *network is the one where each agent has exactly two neighbours.*

Definition C.10. *A network* $\mathfrak{g}$ *is a two diameter network such that* $1 \leq d_{ij}(\mathfrak{g}) \leq 2$*, for all* $i, j \in \mathfrak{g}$.

C.2 Preview on Network Centrality

Which agent (node) is ‘at the center' or the most important (prominent or prestigious) one in the network? This is one of the central question in the research field of social network analysis [110]. The notion of *centrality* answers this question. Centrality is an attempt to quantify importance (or closeness) of agents in a network[41]. Formally, a centrality measure is a function that assigns a numeric value to each agent of a network. Centrality measures can be classified into two types: local centrality measures and global centrality measures.

Local Centrality Measures

Local centrality measures only focus direct links in a network. Following are the examples of local centrality measures.

1) Degree Centrality: is an example of local centrality measure. This is the most simplest and straightforward measure of centrality. Degree centrality is defined as the neighbourhood size of an agent in the network, that is, the number of links an agent has. The normalized form of degree centrality is defined below.

For a network $\mathfrak{g}$ with N agents, the degree centrality for agent i is:

$$C_D(i) = \frac{\eta_i(\mathfrak{g})}{N-1}.$$

[41] We refer the reader to the body of the paper[150, 151, 89, 152, 153] for details on centrality measures in networks.

If someone wants to find a popular agent who has the highest number of neighbours, then degree centrality is an appropriate measure.

2) Eigenvector Centrality: it is a relative measure for importance or prestige. It assumes that the importance of agent i is related to the importance of its neighbours. Eigenvector centrality of agent i is proportional to the sum of centrality of its neighbours as follows.

$$E_s(i) = \frac{\sum_{j \in \eta_i(\mathfrak{g})} \mathbf{M}_{ij} E_s(j)}{\rho},$$

where $E_s(i)$ and $E_s(j)$ are the eigenvector score of agent i and its neighbour j, respectively. $\mathbf{M}_{ij}$ is the adjacency matrix of the network $\mathfrak{g}$, and ρ is a constant (positive proportionality factor).

Global Centrality Measures

Global centrality measures focus on both direct and indirect links in a network. In other words, global centrality measures consider the shortest paths to generate centrality scores for agents.

1) Betweenness Centrality: of an agent in a network is determined by how many times the agent interrupts the shortest path between each pair of agents. A betweenness score of an agent i in network $\mathfrak{g}$ is computed as below.

$$C_B(i) = \sum_{i \neq j \neq k \in \mathscr{A}} \frac{\sigma_{jk}(i)}{\sigma_{jk}},$$

where σ_{jk} is the number of shortest paths from agent j to k, and is $\sigma_{jk}(i)$ is the number of shortest paths from j to k that pass through agent i.

For an information flow network (where agents are organizations), if someone wants to find the agent that most frequently controls the flow of information, then betweenness centrality is an appropriate measure.

2) Closeness Centrality: measure reflects how an agent is close to others in a network. There are many variants of closeness centrality available in the social network analysis literature [154, 155, 156, 157, 158, 159], however, the closeness centrality measure proposed in [156] is simple and straightforward, which is defined as below.

$$C_C(i) = \frac{1}{\sum_{i \neq j \in \mathscr{A}} d_{ij}(\mathfrak{g})}.$$

One of the limitations associated with this above closeness centrality measure is that it does not deal with a disconnected network. If a few agents are isolated in a network, then the above centrality measure fails to score their closeness.

3) Harmonic Centrality: is one of the variants of closeness centrality that deals with a disconnectedness of a network. This measure is a slight modification of the centrality measure [156] (stated above) and defined as below.

$$H_C(i) = \sum_{i \neq j \in \mathscr{A}} \frac{1}{d_{ij}(\mathfrak{g})}.$$

C.3 Preview on Game Theory

Game theory is a branch of mathematics that deals with conflict and cooperation between multiple intelligent rational decision-makers (called agents or players). To be specific, this theory focuses on decision making settings where agents are rational who act to achieve their own goals. In such a setting, the decisions of each agent not only influence the outcome of their own but also of others. In game theory, "the term *game* means an abstract mathematical model of a multi-agent decision-making setting" [160]; such a model includes all those details of the domain that are relevant to the decisions that agents must take. A game can be either *non-cooperative* or *cooperative*. We say, a game is non-cooperative if the agents make their decisions independently to maximize their own benefit, and they are not able to negotiate or form binding agreements. On the contrary, in cooperative games agents are free to negotiate and form coalitions based on binding or enforceable agreements to make joint actions. So, non-cooperative games can be seen as a competition between individual agents, unlike cooperative games where there is a competition between coalitions of agents.

Non-cooperative Games

Formally, a non-cooperative game can be characterized with the following elements.

1. **Agents:** A game consists of two or more but finite rational agents who make the relevant decisions.
2. **Actions:** A set of actions available for each agent.
3. **Rules:** these are the complete description of how agents will act on actions. For example, agents choose actions simultaneously or sequentially.
4. **Outcomes:** these are the consequences of the actions taken by all the agents during the game. Every combination of actions (one for each agent) is an outcome of the game.
5. **Payoffs:** A payoff (or utility) function assigns a payoff (which is a value and in general a number) to each agent for each possible outcome. In other words, a payoff is the value (the degree of satisfaction) observed by each agent as the function of the actions chosen by all agents.

Solution Concepts are at the core of game theory as they formulate outcomes of games. A solution of a game can be seen as a rule (or model) that predicts how the game will be played in terms of what action(s) agents will adopt in that game, and hence, the outcome of the game.

In the context of non-cooperative games, one class of solutions to a game is based on the concept of *dominance*, which is determined by attempting to eliminate a set of actions that rational agents would never play. Another class of solutions is based on the notion of *equilibrium*, which take place when no agent has incentive to depart from the predicted solution [161]. For example, *Nash equilibrium*. We say, a given action profile (a combination of actions of all agents) constitute Nash equilibrium if no agent has an incentive to deviate unilaterally, given that all other agents' actions remain unchanged in the given action profile.

Other than Nash equilibrium there are many solution concepts that follow the concept of equilibrium. These include Bayesian Nash equilibrium, Correlated equilibrium, Subgame Perfect Nash Equilibrium, Evolutionary stable strategy, Stackelberg equilibrium, Wardrop Equilibrium, Pareto Equilibrium, ε-Equilibrium, etc.

Appendix D

Proofs

D.1 Proof of Observation 7.1

Proof. Let $\mathbf{C}$ is a set of all $n-1$ agents who form the cycle and every agent in the cycle is connected to one other central agent (say, k). Let we have four distinct agents i, j, k, and l, such that, $i, j, l \in \mathbf{C}$ and k is the central agent.

For the given star network $\mathfrak{g}$, from Eq. (3.5), we have, $\Phi_k(\mathfrak{g}) = n-1$ and $\Phi_i(\mathfrak{g}) = \frac{n}{2}$ for all $i \in \mathbf{C}$.

Then from Eq. (3.6), we have,

$$\alpha_{ij}(\mathfrak{g}) = \frac{p(1-q)}{n}, \text{ for all } j \in \mathbf{C}. \tag{D.1}$$

$$\alpha_{ik}(\mathfrak{g}) = \frac{p(1-q)}{n-1}. \tag{D.2}$$

Then, from Eq, (3.11), (D.1), and (D.2). We have,

$$\gamma_i(\mathfrak{g}) = 1 - \Big[\big(1 - \frac{p(1-q)}{n-1}\big)\big(1 - \frac{p(1-q)}{n}\big)^{n-2}\Big] \tag{D.3}$$

Let agent i and j in $\mathbf{C}$ form a direct link in $\mathfrak{g}$. Then $\mathfrak{g}$ leads to $\mathfrak{g} + \langle ij \rangle$. Then from Eq. (3.5), in $\mathfrak{g} + \langle ij \rangle$, we have, $\Phi_k(\mathfrak{g} + \langle ij \rangle) = n-1$, $\Phi_i(\mathfrak{g} + \langle ij \rangle) = \Phi_j(\mathfrak{g} + \langle ij \rangle) = \frac{n+1}{2}$, and

$\Phi_l(\mathfrak{g}+\langle ij\rangle) = \frac{n}{2}$, for all $l \in \mathbf{C}\setminus\{i,j\}$.

Then from Eq.(3.6), we have,

$$\alpha_{ij}(\mathfrak{g}+\langle ij\rangle) = \frac{2p(1-q)}{n+1}. \tag{D.4}$$

$$\alpha_{il}(\mathfrak{g}+\langle ij\rangle) = \frac{p(1-q)}{n} \text{ for all } l \in \mathbf{C}\setminus\{j\}. \tag{D.5}$$

$$\alpha_{ik}(\mathfrak{g}+\langle ij\rangle) = \frac{p(1-q)}{n-1}. \tag{D.6}$$

Then, from Eq, (3.11), (D.4), (D.5), and (D.6). We have,

$$\gamma_i(\mathfrak{g}) = 1-\Big[\big(1-\frac{p(1-q)}{n-1}\big)\big(1-\frac{p(1-q)}{n}\big)^{n-3}\big(1-\frac{2p(1-q)}{n+1}\big)\Big] \tag{D.7}$$

$$\gamma_i(\mathfrak{g}+\langle ij\rangle)-\gamma_i(\mathfrak{g}) = \big(1-\frac{p(1-q)}{n-1}\big)\big(1-\frac{p(1-q)}{n}\big)^{n-3}\big(\frac{p(1-q)(n-1)}{n(n+1)}\big). \tag{D.8}$$

However, from Corollary 7.1 and Eq. (D.8), agent $i \in \mathbf{C}$ has no incentive to add a new link with j if $\big(1-\frac{p(1-q)}{n-1}\big)\big(1-\frac{p(1-q)}{n}\big)^{n-3}\big(\frac{p(1-q)(n-1)}{n(n+1)}\big) \leq \frac{\varsigma}{q(1-p)\theta}$.

Let agent i delete link with k in $\mathfrak{g}$, then it leads to $\mathfrak{g}-\langle ik\rangle$. Then, in $\mathfrak{g}-\langle ik\rangle$, from Eq.(3.6), we have, $\Phi_i(\mathfrak{g}-\langle ik\rangle)=0$, $\Phi_j(\mathfrak{g}-\langle ik\rangle) = \frac{n-1}{2}$, for all $j \in \mathbf{C}\setminus\{i\}$. Therefore,

$$\alpha_{ij}(\mathfrak{g}-\langle ij\rangle) = 0 \text{ for all } j \in \mathfrak{g}-\langle ik\rangle, \tag{D.9}$$

and

$$\alpha_{kj}(\mathfrak{g}-\langle ij\rangle) = \frac{2p(1-q)}{n-1} \text{ for all } j \in \mathbf{C}\setminus\{i\}. \tag{D.10}$$

From Eq. (D.9), $\gamma_i(\mathfrak{g}-\langle ij\rangle)=0$ and

$$\gamma_k(\mathfrak{g}-\langle ij\rangle) = \big(1-\frac{2p(1-q)}{n-1}\big)^{n-2}. \tag{D.11}$$

$$\gamma_i(\mathfrak{g})-\gamma_i(\mathfrak{g}-\langle ij\rangle) = 1-\Big[\big(1-\frac{p(1-q)}{n-1}\big)\big(1-\frac{p(1-q)}{n}\big)^{n-2}\Big]. \tag{D.12}$$

$$\gamma_i(\mathfrak{g}) - \gamma_i(\mathfrak{g} - \langle ij \rangle) = (1 - \frac{2p(1-q)}{n-1})^{n-2}) - (1 - \frac{2p(1-q)}{n})^{n-1}. \tag{D.13}$$

However, from Corollary 7.2 and Eq. (D.12), agent $i \in \mathbf{C}$ has no incentive to delete an existing link with k if $1 - \left[\left(1 - \frac{p(1-q)}{n-1}\right)\left(1 - \frac{p(1-q)}{n}\right)^{n-2}\right] \geq \frac{\varsigma}{q(1-p)\theta}$. Similarly, from Eq. (D.13), agent k has no incentive to delete an existing link with $i \in \mathbf{C}$ if $(1 - \frac{2p(1-q)}{n-1})^{n-2}) - (1 - \frac{2p(1-q)}{n})^{n-1} \geq \frac{\varsigma}{q(1-p)\theta}$.

Therefore, a direct link between i and k will be deleted if $min\{1 - \left[\left(1 - \frac{p(1-q)}{n-1}\right)\left(1 - \frac{p(1-q)}{n}\right)^{n-2}\right], (1 - \frac{2p(1-q)}{n-1})^{n-2}) - (1 - \frac{2p(1-q)}{n})^{n-1}\}$.

Hence, from Corollary 7.2 and 7.1, Eq. (D.8), (D.13), and (D.12), the star network $\mathfrak{g}$ is pairwise stable if

$(1 - \frac{p(1-q)}{n-1})(1 - \frac{p(1-q)}{n})^{n-3}(\frac{(p(1-q))(n-1)}{n(n+1)})$

$\leq \frac{\varsigma}{q(1-p)\theta} \leq$

$min\{[1 - ((1 - \frac{p(1-q)}{n-1})(1 - \frac{p(1-q)}{n})^{n-2})], [(1 - \frac{2p(1-q)}{n-1})^{n-2}) - (1 - \frac{2p(1-q)}{n})^{n-1}]\}$. □

D.2 Pairwise Stability: Star, Wheel, and Bipartite

In each inequality (stated below), the left hand side suggests when agent has no incentives to add a new link and the right hand side suggests when agent has no incentives to deleted an existing link. Further, each left hand side represents an agent's chance of not getting resource in the existing network and the marginal probability of getting resource due to newly added link. Similarly, each right hand side represents an agent's chance of obtaining the resource after deleting one of its link.

Observation D.1. *Let an SSCC be a star network $\mathfrak{g}$ of n agents such that it contains one central agent who is connected to all $n-1$ pendant agents, and for which every pendant agent is connected to the central agent. Then the star network $\mathfrak{g}$ is pairwise stable if*

$(1-\frac{p(1-q)}{n-1})(1-\frac{p(1-q)}{n})^{n-3}(\frac{(p(1-q))(n-1)}{n(n+1)})$

$\leq \frac{\varsigma}{q(1-p)\theta} \leq$

$min\{[1-((1-\frac{p(1-q)}{n-1})(1-\frac{p(1-q)}{n})^{n-2})],[(1-\frac{2p(1-q)}{n-1})^{n-2})-(1-\frac{2p(1-q)}{n})^{n-1}]\}.$

In the above inequality, the left hand side suggests when a pendant agent has no incentives to add a new link. Whereas the right hand side suggests when the center agent as well as a pendent agent has no incentive to delete one of its link.

Observation D.2. *Let an SSCC be a wheel network $\mathfrak{g}$ of n agents, such that, it contains a cycle of order $n-1$ agents, and for which every agent in the cycle is connected to one other central agent. Then the wheel network $\mathfrak{g}$ is pairwise stable if*

1. $(1-\frac{p(1-q)}{n-1})(1-\frac{2p(1-q)}{2})^2(1-\frac{p(1-q)}{n+2})^{n-5}(\frac{p(1-q)(n+1)}{(n+2)(n+3)}) \leq \frac{\varsigma}{q(1-p)\theta}$

   $\leq (1-\frac{p(1-q)}{n-1})(1-\frac{2p(1-q)}{n+2})(1-\frac{p(1-q)}{n+2})^{n-4}(\frac{np(1-q)}{(n+1)(n+2)})$, *or*

2. $(1-\frac{p(1-q)}{n-1})(1-\frac{2p(1-q)}{2})^2(1-\frac{p(1-q)}{n+2})^{n-5}(\frac{p(1-q)(n+1)}{(n+2)(n+3)}) \leq \frac{\varsigma}{q(1-p)\theta}$

   $\leq min[(1-\frac{2p(1-q)}{n+2})^2(1-\frac{p(1-q)}{n+2})^{n-4}(\frac{p(1-q)(n-2)}{2n-3})],[(1-\frac{2p(1-q)}{n+2})^{n-2}(\frac{p(1-q)(n-1)}{(n+1)(n+2)}))]$

Let $\hat{\mathbf{C}}$ is a set of all $n-1$ agents who form the cycle and every agent in the cycle is connected to one other central agent (say, k).

1 assures that no agent $i \in \hat{\mathbf{C}}$ has incentive to either add a new link or delete an existing link link with $j \in \hat{\mathbf{C}}$.

2 assures that agent $i \in \hat{\mathbf{C}}$ has no incentive to add a new link with another agent $j \in \hat{\mathbf{C}}$, and the central agent k has no incentive to delete an exiting link with one of agents $i \in \hat{\mathbf{C}}$ and vice versa.

Observation D.3. *Let an SSCC be a complete bipartite network $\mathfrak{g}$ of n agents. A complete bipartite $\mathfrak{g}$ consists of two disjoint sets $\bar{s}_i$ and $\hat{s}_j$ such that no two agents within the same* *set are adjacent and every pair of agent in the two sets are adjacent. Let $\bar{s}_i$ and $\hat{s}_j$ be the number of agents in the respective sets and $n = \bar{s}_i + \hat{s}_j$. The complete bipartite $\mathfrak{g}$ is pairwise stable if*

$$max\Big\{\big[p(1-q)(1-\tfrac{2p(1-q)}{2\bar{s}_i+\hat{s}_j-1})^{\hat{s}_j}(1-\tfrac{p(1-q)}{2\hat{s}_j+\bar{s}_i-1})^{(\bar{s}_i-2)}[\tfrac{2\hat{s}_j+\bar{s}_i-2}{(2\hat{s}_j+\bar{s}_i)(2\hat{s}_j+\bar{s}_i-1)}]\big],$$

$$\big[p(1-q)(1-\tfrac{2p(1-q)}{2\hat{s}_j+\bar{s}_i-1})^{\bar{s}_i}(1-\tfrac{p(1-q)}{2\bar{s}_i+\hat{s}_j-1})^{(\hat{s}_j-2)}[\tfrac{2\bar{s}_i+\hat{s}_j-2}{(2\bar{s}_i+\hat{s}_j)(2\bar{s}_i+\hat{s}_j-1)}]\big]\Big\} \leq \frac{\varsigma}{\theta q(1-p)}$$

$$\leq min\Big\{\big[4p(1-q)(1-\tfrac{2p(1-q)}{2\bar{s}_i+\hat{s}_j-1})^{\hat{s}_j-1}(1-\tfrac{p(1-q)}{2\hat{s}_j+\bar{s}_i-1})^{(\bar{s}_i-1)}[\tfrac{2\bar{s}_i+\hat{s}_j-3}{(2\bar{s}_i+\hat{s}_j-1)(6\bar{s}_i+3\hat{s}_j-7)}]\big]$$

$$\big[4p(1-q)(1-\tfrac{2p(1-q)}{2\hat{s}_j+\bar{s}_i-1})^{\bar{s}_i-1}(1-\tfrac{p(1-q)}{2\bar{s}_i+\hat{s}_j-1})^{(\hat{s}_j-1)}[\tfrac{2\hat{s}_j+\bar{s}_i-3}{(2\hat{s}_j+\bar{s}_i-1)(6\hat{s}_j+3\bar{s}_i-7)}]\big]\Big\}$$

We do the similar interpretation for this as above.