

## ABSTRACT

Title of dissertation:      MULTIMEDIA SOCIAL NETWORKS:  
   GAME THEORETIC MODELING  
   AND EQUILIBRIUM ANALYSIS

Yan Chen, Doctor of Philosophy, 2011

Dissertation directed by:   Professor K. J. Ray Liu  
   Department of Electrical and Computer Engineering

Multimedia content sharing and distribution over multimedia social networks is more popular now than ever before: we download music from Napster, share our images on Flickr, view user-created video on YouTube, and watch peer-to-peer television using Coolstreaming, PPLive and PPStream. Within these multimedia social networks, users share, exchange, and compete for scarce resources such as multimedia data and bandwidth, and thus influence each other's decision and performance. Therefore, to provide fundamental guidelines for the better system design, it is important to analyze the users' behaviors and interactions in a multimedia social network, i.e., how users interact with and respond to each other.

Game theory is a mathematical tool that analyzes the strategic interactions among multiple decision makers. It is ideal and essential for studying, analyzing, and modeling the users' behaviors and interactions in social networking. In this thesis, game theory will be used to model users' behaviors in social networks and analyze the corresponding equilibria. Specifically, in this thesis, we first illustrate

how to use game theory to analyze and model users' behaviors in multimedia social networks by discussing the following three different scenarios. In the first scenario, we consider a non-cooperative multimedia social network where users in the social network compete for the same resource. We use multiuser rate allocation social network as an example for this scenario. In the second scenario, we consider a cooperative multimedia social network where users in the social network cooperate with each other to obtain the content. We use cooperative peer-to-peer streaming social network as an example for this scenario. In the third scenario, we consider how to use the indirect reciprocity game to stimulate cooperation among users. We use the packet forwarding social network as an example.

Moreover, the concept of "multimedia social networks" can be applied into the field of signal and image processing. If each pixel/sample is treated as a user, then the whole image/signal can be regarded as a multimedia social network. From such a perspective, we introduce a new paradigm for signal and image processing, and develop generalized and unified frameworks for classical signal and image problems. In this thesis, we use image denoising and image interpolation as examples to illustrate how to use game theory to re-formulate the classical signal and image processing problems.

MULTIMEDIA SOCIAL NETWORKS: GAME THEORETIC  
MODELING AND EQUILIBRIUM ANALYSIS

by

Yan Chen

Dissertation submitted to the Faculty of the Graduate School of the  
University of Maryland, College Park in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
2011

Advisory Committee:  
Professor K. J. Ray Liu, Chair/Advisor  
Professor Min Wu  
Professor Rama Chellappa  
Professor Shuvra S. Bhattacharyya  
Professor Lawrence C. Washington

© Copyright by  
Yan Chen  
2011

## Dedication

To my family.

## Acknowledgments

This thesis is the outcome of a wonderful four-year study and research experience in Signal and Information Group (SIG) at University of Maryland, College Park.

Here, I would like to take this opportunity to express my sincere appreciation and gratitude to my advisor, Prof. K. J. Ray Liu, for his guidance, patience, understanding, and encouragement. During these four years, he inspired me and gave me freedom to work on many exciting cutting-edge research topics. His vision, creativeness, and enthusiasm not only have fueled my current research but also have a lasting influence in my professional and personal development as a role model.

I would like to thank Prof. Min Wu and Prof. Rama Chellappa for not only serving on my thesis committees and reviewing my thesis but also serving as references for my faculty application. I am also grateful to Prof. Shuvra S. Bhattacharyya and Prof. Lawrence C. Washington for their time and effort serving on my committees and reviewing my thesis. I would also like to thank Prof. Oscar Au, Prof. Andre L. Tits, and Prof. Peter Cramton for serving as references for my faculty application.

Thanks should also go to all members in our Signals and Information Group for their friendship, encouragement, and help. I have learned a lot from them for which I am deeply grateful. Special thanks go to Dr. Beibei Wang, Dr. Yongle Wu, Dr. Wan-Yi Lin, Mr. Matt Stamm, Dr. Steve Tjoa, and Dr. Quoc Lai for our research discussions that were all inspiring for me.

My deepest gratitude goes to my wife Yang. She is always there for me,

giving me her love and support. We share all things happened in life, and discuss the problems on research. I would also like to thank my father, my sister and my parents in law, for their constant love, comfort, support and encouragement. Finally, I would like to dedicate my thesis to my mother. I really hope that my thesis would make her proud. I would never forget her, and she is always in my heart.

# Table of Contents

List of Tables	viii
List of Figures	ix
1 Introduction	1
1.1 Motivation . . . . .	1
1.2 Related Works on Social Networks . . . . .	3
1.3 Dissertation Outline . . . . .	5
1.3.1 An Overview of Game Theory (Chapter 2) . . . . .	7
1.3.2 Multiuser Rate Allocation Social Networks (Chapter 3) . . . . .	7
1.3.3 Peer-to-Peer Cooperative Video Streaming Social Networks (Chapter 4) . . . . .	8
1.3.4 Cooperation Stimulation Using Indirect Reciprocity Game Mod- eling (Chapter 5) . . . . .	9
1.3.5 Image Denoising Games (Chapter 6) . . . . .	10
1.3.6 Simultaneous Image Denoising and Interpolation Using Evo- lutionary Games (Chapter 7) . . . . .	11
2 An Overview of Game Theory	13
2.1 Nash Equilibrium and Pareto Optimality . . . . .	14
2.2 Auction Games . . . . .	15
2.3 Evolutionary Games . . . . .	17
2.4 Coalition Formation Games . . . . .	18
2.5 Stochastic Games . . . . .	19
3 Multiuser Rate Allocation Social Networks	22
3.1 The Game-Theoretic Framework . . . . .	26
3.1.1 System Model . . . . .	26
3.1.2 Video Distortion-Rate Model . . . . .	27
3.1.3 User's Utility Function . . . . .	28
3.1.4 Multi-User Rate Allocation Game . . . . .	31
3.2 Analysis of The Multi-User Rate Allocation Game . . . . .	32
3.2.1 Non-Efficient Rate Allocation ( $a > a_0$ ) . . . . .	32
3.2.2 Efficient Rate Allocation ( $0 \leq a < a_0$ ) . . . . .	33
3.2.3 Efficient and Proportionally Fair in Both Utility and PSNR ( $a = a_0$ ) . . . . .	33
3.3 Relation To The Traditional Optimization-Based Approach . . . . .	37
3.4 Clock Auction For Distributed Cheat-Proof Optimal Rate Allocation	38
3.5 Experimental Results . . . . .	46
3.5.1 Parameter Estimation . . . . .	46
3.5.2 Multi-User Rate Allocation . . . . .	49
3.6 Summary . . . . .	54

4	Peer-to-Peer Cooperative Video Streaming Social Networks	55
4.1	The System Model and Utility Function	59
4.1.1	System Model	59
4.1.2	Utility Functions	61
4.2	Agents Selection Within A Homogeneous Group	64
4.2.1	Centralized Agent Selection	64
4.2.2	Distributed Agent Selection	65
4.2.3	Evolutionary Cooperative Streaming Game	66
4.2.4	Analysis of the Cooperative Streaming Game	67
4.3	Agents Selection Within A Heterogeneous Group	74
4.3.1	Two-Player Game	75
4.3.2	Multi-Player Game	77
4.4	A Distributed Learning Algorithm For ESS	78
4.5	Simulation Results	79
4.6	Summary	87
5	Cooperation Stimulation Using Indirect Reciprocity Game Modeling	89
5.1	The System Model	94
5.1.1	Social Norms	96
5.1.2	Action Rules	97
5.2	Optimal Action Rule	98
5.2.1	Reputation Updating Policy	98
5.2.2	Stationary Reputation Distribution	101
5.2.3	Payoff Function	102
5.2.4	Optimal Action Using An Alternative Algorithm	104
5.3	Action Spreading Due To Natural Selection	105
5.3.1	Action Spreading Algorithm Using Wright Fisher Model	106
5.3.2	Action Spreading Algorithm Using Replicator Dynamic Equation	107
5.4	Evolutionarily Stable Strategy and Simulations	108
5.4.1	Binary Reputation Scenario	109
5.4.2	Multi-Level Reputation Scenario	113
5.5	Summary	116
6	Image Denoising Games	119
6.1	The System Model and Coalition Formation Game	122
6.1.1	The System Model	122
6.1.2	The Coalition Formation Game	123
6.2	Candidate Set Selection	125
6.3	Game Theoretical Problem Formulation	128
6.3.1	Stein's unbiased risk estimate (SURE)	128
6.3.2	Confidence and Distortion Trade-off	132
6.3.2.1	Confidence	132
6.3.2.2	Distortion	133
6.3.2.3	Confidence and Distortion Trade-off	133

6.3.3	Utility Function and Solution to the Game . . . . .	134
6.4	Relation to the Traditional Approaches . . . . .	139
6.5	Experimental Results . . . . .	140
6.6	Summary . . . . .	142
7	Simultaneous Image Denoising and Interpolation Using Evolutionary Games	151
7.1	Evolutionary Game Model . . . . .	156
7.2	Image Denosing and Interpolation as an Evolutionary Game . . . . .	158
7.2.1	Problems of Denoising+Interpolation . . . . .	158
7.2.2	Game Theoretic Formulation . . . . .	160
7.2.3	Pure Strategy Set $\Omega_{ij}$ . . . . .	162
7.2.4	Payoff Function . . . . .	163
7.3	Experimental Results . . . . .	166
7.4	Summary . . . . .	169
8	Conclusions and Future Work	177
8.1	Conclusions . . . . .	177
8.2	Future Work . . . . .	180
	Bibliography	182

## List of Tables

3.1	$\gamma_i^*$ , $\beta_i^*$ , $R_i^{min}(kb/s)$ , and $R_i^{max}(kb/s)$ for different sequence by training.	47
4.1	Utility table of a two-player game. . . . .	74

## List of Figures

3.1	System Model. . . . .	27
3.2	The visual quality of Foreman sequence at different PSNR level: (a) 33dB; (b) 34dB; (c) 40dB; (d) 41dB. . . . .	30
3.3	Illustration for the proof of Lemma 2: (a) If $\tilde{R}_i^t \leq R_i^t$ and $R_i^{L_1} > R_i^{min}$ , we can see that $[\ln(\gamma_i + \beta_i R_i^{L_1}) - \frac{\partial \ln(\gamma_i + \beta_i R_i)}{\partial R_i}  _{R_i=R_i^{L_1}} (R_i^{L_1} - R_i^{L_2})] \geq \ln(\gamma_i + \beta_i R_i^{L_2})$ ; (b) If $\tilde{R}_i^t > R_i^t$ and $R_i^{L_1} < R_i^{max}$ , we can see that $\ln(\gamma_i + \beta_i R_i^{L_1}) \geq [\ln(\gamma_i + \beta_i R_i^{L_2}) - \frac{\partial \ln(\gamma_i + \beta_i R_i)}{\partial R_i}  _{R_i=R_i^{L_1}} (R_i^{L_2} - R_i^{L_1})]$ . . . . .	43
3.4	Training $\gamma$ and $\beta$ : (a) Football; (b) Coastguard. . . . .	48
3.5	Allocated rates for Akiyo, Carphone, Coastguard, Foreman, Table, Football, and Mobile using different methods. . . . .	49
3.6	The sum of PSNR vs. the available network bandwidth $R$ . . . . .	50
3.7	Cheat-proof performance: (a) AFD, AFR, and MSPSNR; (b) Proposed Method. . . . .	51
4.1	A cooperative streaming example. . . . .	60
4.2	The deceasing property of $w_i$ . . . . .	68
4.3	The social welfare comparison among <b>Non-Coop</b> , <b>MSW-C</b> , <b>MSW-D</b> , and <b>ESS-D</b> . . . . .	81
4.4	Behavior dynamic of a homogeneous group of peers. . . . .	82
4.5	Behavior dynamic of a heterogeneous group of peers. . . . .	83
4.6	The probability of real-time streaming comparison between <b>Non-Coop</b> and <b>ESS-D</b> . . . . .	84
4.7	The visual quality comparison: (a) <b>Non-Coop</b> ; (b) <b>ESS-D</b> with $N=2$ ; (c) <b>ESS-D</b> with $N=3$ ; (d) <b>ESS-D</b> with $N=4$ . . . . .	85
4.8	Single-source rate comparison between <b>Non-Coop</b> and <b>ESS-D</b> . . . . .	86
4.9	Multi-source rate comparison between <b>Non-Coop</b> and <b>ESS-D</b> . . . . .	87
4.10	The social welfare comparison between <b>Non-Coop</b> and <b>ESS-D</b> when the utility function is defined as (4.42). . . . .	88
5.1	System model. Within every interaction, a pair of transmitter and receiver is randomly sampled from the population. Then, the transmitter will forward a certain amount of packets to the receiver according to the receiver's and his/her own reputations. After the transmission, the transmitter's reputation will be updated by the receiver and the observers. Finally, the transmitter's reputation is propagated to the whole population from the receiver and the observers through a noisy gossip channel. . . . .	93
5.2	Reputation updating policy. . . . .	98
5.3	The population evolution when $L = 1$ , $g = 1$ and $c = 0.1$ : (a) the percentage of the population with reputation $L = 1$ ; (b) the percentage of the population using optimal action shown in (5.24). . . . .	108

5.4	The stable region for the optimal action rule shown in (5.24) when $L = 1$ . . . . .	110
5.5	The population evolution when $L = 1$ , $g = 1$ and $c = 0.6$ : (a) the percentage of the population with reputation $L = 1$ ; (b) the percentage of the population using optimal action shown in (5.24). . .	111
5.6	The stable region for the optimal action rule shown in (5.26) when $L = 4$ . . . . .	113
5.7	The population evolution when $L = 4$ , $g = 1$ and $c = 0.5$ : (a) the percentage of the population with reputation $L = 4$ ; (b) the percentage of the population using optimal action shown in (5.26). . .	114
5.8	The population evolution when $L = 4$ , $g = 1$ and $c = 0.7$ : (a) the percentage of the population with reputation $L = 4$ ; (b) the percentage of the population using optimal action shown in (5.26). . .	115
6.1	An example of optimal candidate set with an edge region: (a) original image; (b) noisy image with $\sigma = 15$ ; (c) noisy image with $\sigma = 25$ ; (d) noisy image with $\sigma = 35$ ; (e) the candidate set used by nonlocal; (f) the ideally optimal candidate set of (b) when the original signal is available; (g) the ideally optimal candidate set of (c) when the original signal is available; (h) the ideally optimal candidate set of (d) when the original signal is available. . . . .	127
6.2	An example of optimal candidate set with a smooth region: (a) original image; (b) noisy image with $\sigma = 15$ ; (c) noisy image with $\sigma = 25$ ; (d) noisy image with $\sigma = 35$ ; (e) the candidate set used by nonlocal; (f) the ideally optimal candidate set of (b) when the original signal is available; (g) the ideally optimal candidate set of (c) when the original signal is available; (h) the ideally optimal candidate set of (d) when the original signal is available. . . . .	129
6.3	The optimal $m^*$ for lena image when $\sigma = 10$ . . . . .	130
6.4	The trade-off between the confidence term $C$ and the distortion term $D$ : (a) the performance of $C$ with different $N$ ; (b) the performance of $D$ with different $N$ . . . . .	132
6.5	Some possible gain functions. . . . .	134
6.6	The four tested images: (a) Lena; (b) Barbara; (c) Boat; (d) Flinstones.	135
6.7	An example of optimal candidate set with an edge region: (a) original image; (b) noisy image with $\sigma = 15$ ; (c) noisy image with $\sigma = 25$ ; (d) noisy image with $\sigma = 35$ ; (e) the candidate set used by nonlocal; (f) (g) and (h) are the ideally optimal candidate sets of (b) (c) and (d) when the original signal is available; (i) (j) and (k) are the candidate sets of (b) (c) and (d) generated by the proposed method. . . . .	144

6.8	An example of optimal candidate set with a smooth region: (a) original image; (b) noisy image with $\sigma = 15$ ; (c) noisy image with $\sigma = 25$ ; (d) noisy image with $\sigma = 35$ ; (e) the candidate set used by nonlocal; (f) (g) and (h) are the ideally optimal candidate sets of (b) (c) and (d) when the original signal is available; (i) (j) and (k) are the candidate sets of (b) (c) and (d) generated by the proposed method. . . . .	145
6.9	The PSNR comparison for different images: (a) Lena; (b) Barbara; (c) Boat; (d) Flinstones. . . . .	146
6.10	The visual quality comparison for Flinstones with $\sigma = 25$ : (a) original image; (b) noisy image; (c) the result generated by the nonlocal method; (d) the result generated by the proposed approach. . . . .	147
6.11	The visual quality comparison for Barbara with $\sigma = 35$ : (a) original image; (b) noisy image; (c) the result generated by the nonlocal method; (d) the result generated by the proposed approach. . . . .	148
6.12	The visual quality comparison for Lena with $\sigma = 45$ : (a) original image; (b) noisy image; (c) the result generated by the nonlocal method; (d) the result generated by the proposed approach. . . . .	149
6.13	The visual quality comparison for Boat with $\sigma = 20$ : (a) original image; (b) noisy image; (c) the result generated by the nonlocal method; (d) the result generated by the proposed approach. . . . .	150
7.1	(a) a region of Lena image that is corrupted with Gaussian noise with noise variance $\sigma^2 = 25$ ; (b) the interpolation result using bicubic; (c) the interpolation result using SAI. . . . .	153
7.2	(a) Denoising before interpolation; (b) Simultaneous denoising and interpolation. . . . .	158
7.3	(a) a region of Lena image that is corrupted by Gaussian noise with noise variance $\sigma^2 = 225$ ; (b) the denoised result using nonlocal; (c) the interpolation result of (b) using bicubic; (d) the interpolation result of (b) using SAI. . . . .	159
7.4	The pure strategy set $\Omega_{ij}$ for $I_h(2m-1, 2n-1)$ , $I_h(2m-1, 2n)$ , $I_h(2m, 2n-1)$ , $I_h(2m, 2n)$ . . . . .	162
7.5	The block diagram of the proposed method. . . . .	165
7.6	The test images: (a) Lena; (b) Boat; (c) Kodim03; (d) Kodim07; (e) Kodim09. . . . .	170
7.7	The PSNR comparison: (a) Lena; (b) Boat; (c) Kodim03; (d) Kodim07; (e) Kodim09. . . . .	171
7.8	The visual quality comparison for Lena: (a) the noisy low resolution image with $\sigma = 10$ ; (b) the result generated by Nonlocal+Bicubic (30.71dB); (c) the result generated by Nonlocal+SAI method (31.11dB); (d) the result generated by the proposed method (31.38dB). . . . .	172
7.9	The visual quality comparison for Boat: (a) the noisy low resolution image with $\sigma = 5$ ; (b) the result generated by Nonlocal+Bicubic (28.52dB); (c) the result generated by Nonlocal+SAI method (28.92dB); (d) the result generated by the proposed method (28.74dB). . . . .	173

7.10	The visual quality comparison for Kodim07: (a) the noisy low resolution image with $\sigma = 10$ ; (b) the result generated by Nonlocal+Bicubic (29.85dB); (c) the result generated by Nonlocal+SAI method (30.18dB); (d) the result generated by the proposed method (30.35dB). . . . .	174
7.11	The visual quality comparison for Kodim09: (a) the noisy low resolution image with $\sigma = 15$ ; (b) the result generated by Nonlocal+Bicubic (29.56dB); (c) the result generated by Nonlocal+SAI method (29.81dB); (d) the result generated by the proposed method (30.41dB). . . . .	175
7.12	The visual quality comparison for Kodim03: (a) the noisy low resolution image with $\sigma = 20$ ; (b) the result generated by Nonlocal+Bicubic (27.85dB); (c) the result generated by Nonlocal+SAI method (28.03dB); (d) the result generated by the proposed method (28.37dB). . . . .	176

## Chapter 1

### Introduction

#### 1.1 Motivation

Multimedia content sharing and distribution over multimedia social networks is more popular now than ever before: we download music from Napster [3], share our images on Flickr [2], view user-created video on YouTube [9], and watch peer-to-peer television using Coolstreaming [133], PPLive [4] and PPStream [5]. Within these multimedia social networks, users share, exchange, and compete for scarce resources such as multimedia data and bandwidth, and thus influence each other's decision and performance. Therefore, to provide fundamental guidelines for the better system design, it is important to analyze the users' behaviors and interactions in a multimedia social network, i.e., how users interact with and respond to each other.

Unlike generic data applications, multimedia applications have time-varying bandwidth requirements, stringent delay deadlines and dynamic characteristics. To enable users in a multimedia social network to successfully participate in the resource competition, the uniquely scalable and delay-sensitive characteristics of multimedia data and the resulting impact on users viewing experiences of multimedia content should be explicitly involved in the system design.

In multimedia social networks, users are intelligent and have the ability to

observe, learn, and make intelligent decisions. Since users usually belong to different authorities and pursue different goals, they will choose the strategies that can maximize their own payoffs. In such a case, traditional centralized optimization-based approaches no longer work well since they only consider the efficiency of the whole system while totally ignore the fairness among users, which is an even more important issue in multimedia social networks. To better design the system, not only the efficiency issue from the system designers' perspective but also the fairness issue from the users' perspective should be taken into account. Moreover, since users in multimedia social networks are rational and thus naturally selfish, they tend to over-claim what they may need and will not truly report their private information if cheating can improve their payoffs. Therefore, enforcing truth-telling is crucial in multimedia social networks.

From the above discussions, we can see that the behavior dynamics among users in a multimedia social network are very complex. To understand the users' complex behavior dynamics and thus lead to a better system design, game theory is a powerful mathematical tool that analyzes the strategic interactions among multiple decision makers [97]. It has been developed for understanding cooperation and conflict between individuals in many fields such as economics, politics, business, social sciences and biology. Thus, game theory is ideal and essential for studying, analyzing, and modeling the users' behaviors and interactions in social networking. Recently, it draws great attentions in cognitive networking [30] [123] and multimedia signal processing [27]. In this thesis, we will illustrate how game theory can be used to model users' behaviors in various multimedia social networks and analyze the

corresponding equilibria.

## 1.2 Related Works on Social Networks

A social network is a social structure made of individuals and/or organizations called “nodes”, which are connected with each other by certain types of interdependency, such as friendship, kinship, financial exchange, conflict, trade, etc. Many methodologies have been studied to formulate the relationships among members at all scales, from interpersonal to international, and social network analysis becomes a popular topic in sociology, economics, information science and many other disciplines.

Most of the existing works on social networks fall into the following three categories [80]: (1) social network properties, (2) social network models, (3) social network dynamics and evolution. In [17] [43], the authors showed that the vertex connectivities in many large networks follow a scale-free power-law distribution. Such a property is found to be a consequence of two generic mechanisms: (i) networks expand continuously by the addition of new vertices, and (ii) new vertices attach preferentially to sites that are already well connected. Another important property of social networks is the “small-world” phenomenon. As pointed out in [125] [18], most real-world networks exhibit relatively small diameter, i.e., the networks are highly clustered.

Besides the study of the social network properties, there are quite a lot of work on building models for social networks. The simplest model is the random graph

model introduced in [37], where given a number of nodes, each pair of nodes has an identical and independent probability of being joined by an edge. However, since it fails to match the real-world social network properties, e.g., it does not produce power law degree distributions, this model is not realistic. A better model that can produce power law degree distributions is the preferential attachment [38] [44] [10], where when a new node  $u$  arrives to the network, the probability of connecting to a node  $v$  is proportional to the degree of  $v$ . Another model that can also produce power law degree distributions is the copying model [103], where a new node joins the networks by uniformly creating random edges or first random choosing a node  $u$  and then linking to  $u$ 's neighbors.

Another important research topic in the field of social network is the study of social network dynamics and evolution where the researchers study how the social network evolve and how information spread over the networks. Many works have been done to investigate the dynamics and evolution of different networks, e.g., trendsetters selecting in viral marketing [49], inoculation targets identification in epidemiology [92], and studying trends in blogosphere [102].

However, most of these existing works study and analyze the social networks at the macroeconomic level, i.e., from system designer's perspective. Few efforts have been made to investigate the social networks at the microeconomic level, i.e., from the users' perspective, which is an important issue in social network analysis since users may only care about their own objectives and their decisions greatly affect the evolution and performance of the social networks. In this thesis, we will study and analyze the social networks from the users' perspective by modeling users' behaviors

and interactions using game theory.

### 1.3 Dissertation Outline

Since users in different multimedia social networks may have different types of interdependency, to effectively model the users' behaviors and interactions, different game models for different multimedia social networks should be employed. The two most common types of users' interdependency in multimedia social networks are competition and cooperation, which leads to non-cooperative social networks and cooperative social networks, respectively. In cooperative social networks, since users are rational thus naturally selfish, they will not cooperate with others unless cooperation can improve their own performance. Therefore, one important issue in cooperative social networks is cooperation stimulation. Without loss of generality, in this thesis, we first illustrate how to use game theory to analyze and model users' behaviors in multimedia social networks by discussing the following three different scenarios:

- In the first scenario, we consider a non-cooperative multimedia social network where users in the social network compete for the same resource. We use multiuser rate allocation social network [29] as an example for this scenario and the details will be described in Chapter 3.
- In the second scenario, we consider a cooperative multimedia social network where users in the social network cooperate with each other to obtain the content. As discussed in Chapter 4, we will use cooperative peer-to-peer streaming

social network [34] as an example for this scenario.

- In the third scenario, we consider how to use the indirect reciprocity game to stimulate cooperation among users. In Chapter 5, we will use the packet forwarding social network [28] as an example.

Moreover, the concept of “multimedia social networks” can be applied into the field of signal and image processing. Although there are seemingly no human factors involved in the algorithmic solution in classical signal/image processing, if we take the view that the pixels/signals of an image are forming a notion of a “social network” to jointly interact to accomplish a common (“processing”) goal, be it filtering, denoising, or segmentation, then the game theoretic approach can offer new views beyond what classical methods can. This completely changes the traditional thinking that we have to decide what a pixel does instead of simply giving some generic rules/guidelines and let the pixels themselves interact/cooperate to decide the best “strategy”. From such a perspective, we introduce a new paradigm for signal and image processing, and develop generalized and unified frameworks for classical signal and image problems. In this thesis, we use image denoising (Chapter 6) and simultaneous image denoising and interpolation (Chapter 7) as examples to illustrate how to use game theory to re-formulate the classical signal and image processing problems. The rest of the dissertation is organized as follows.

### 1.3.1 An Overview of Game Theory (Chapter 2)

Since game theory has been recognized as an important tool in studying, modeling, and analyzing the interaction process among multiple decision makers, in this chapter, we present an overview of some fundamental concepts of game theory that will be used in this thesis.

### 1.3.2 Multiuser Rate Allocation Social Networks (Chapter 3)

In multiuser rate allocation problem, a set of transmitters want to transmit the video sequences to corresponding receivers through a common channel that is shared by all transmitters. Since the transmitters compete for the same resource, i.e., channel bandwidth, they form a non-cooperative social network. The key problem in this social network is how to efficiently and fairly allocate data rate among different users. Most of the existing optimization-based methods, such as minimizing the weighted sum of the distortions or maximizing the weighted sum of the peak signal-to-noise ratios (PSNRs), have their weights heuristically determined. Moreover, those approaches mainly focus on the efficiency issue while there is no notion of fairness. In this chapter, we address this problem by proposing a game-theoretic framework, in which the utility function of each user is jointly determined by the characteristics of the transmitted video sequence and the allocated bit-rate. We show that a unique Nash equilibrium (NE), which is proportionally fair in terms of both utility and PSNR, can be obtained, according to which the controller can efficiently and fairly allocate the available network bandwidth to the users. Moreover, we

propose a distributed cheat-proof rate allocation scheme for the users to converge to the optimal NE using alternative ascending clock auction. We also show that the traditional optimization-based approach that maximizes the weighted sum of the PSNRs is a special case of the game-theoretic framework with the utility function defined as an exponential function of PSNR. Finally, we show several experimental results on real video data to demonstrate the efficiency and effectiveness of the proposed method.

### 1.3.3 Peer-to-Peer Cooperative Video Streaming Social Networks (Chapter 4)

While peer-to-peer (P2P) video streaming systems have achieved promising results, they introduce a large number of unnecessary traverse links, which consequently leads to substantial network inefficiency. To address this problem and achieve better streaming performance, we propose to enable cooperation among group peers, which are geographically neighboring peers with large intra-group upload and download bandwidths. Considering the peers selfish nature, we formulate the cooperative streaming problem as an evolutionary game and derive, for every peer, the evolutionarily stable strategy (ESS), which is the stable Nash equilibrium and no one will deviate from. Moreover, we propose a simple and distributed learning algorithm for the peers to converge to the ESSs. With the proposed algorithm, each peer decides whether to be an agent who downloads data from the peers outside the group or a freerider who downloads data from the agents by simply tossing

a coin, where the probability of being a head for the coin is learned from the peers own past payoff history. Simulation results show that the strategy of a peer converges to the ESS. Compared to the traditional non-cooperative P2P schemes, the proposed cooperative scheme achieves much better performance in terms of social welfare, probability of real-time streaming, and video quality (source rate).

### 1.3.4 Cooperation Stimulation Using Indirect Reciprocity Game Modelling (Chapter 5)

In social networks, since nodes generally belong to different authorities and pursue different goals, they will not cooperate with others unless cooperation can improve their own performance. Thus, how to stimulate cooperation among nodes in social networks is very important. However, most of existing game-theoretic cooperation stimulation approaches rely on the assumption that the interactions between any pair of players are long-lasting. When this assumption is not true, according to the well-known Prisoners Dilemma and the backward induction principle, the unique Nash equilibrium (NE) is to always play non-cooperatively. In this chapter, we propose a cooperation stimulation scheme for the scenario where the number of interactions between any pair of players are finite. The proposed algorithm is based on indirect reciprocity game modelling where the key concept is “I help you not because you have helped me but because you have helped others”. We formulate the problem of finding the optimal action rule as a Markov Decision Process (MDP) and propose a modified value iteration algorithm to find the optimal action rule.

Using the packet forwarding game as an example, we show that with an appropriate cost-to-gain ratio, the strategy of forwarding the number of packets that is equal to the reputation level of the receiver is an evolutionarily stable strategy (ESS). Finally, simulations are shown to verify the efficiency and effectiveness of the proposed algorithm.

### 1.3.5 Image Denoising Games (Chapter 6)

Based on the observation that every small window in a natural image has many similar windows in the same image, the nonlocal denoising methods perform denoising by weighted averaging all the pixels in a nonlocal window and have achieved very promising denoising results. However, the use of a fixed square neighborhood window greatly limits the denoising performance. Therefore, an important issue in pixel-domain image denoising algorithms is how to adaptively choose optimal neighborhoods. Obviously, too large a neighborhood set may cause overly-smooth artifacts, while too small a neighborhood set may not be able to efficiently reduce the noise variance. While the Stein's principle is shown to be able to estimate the true mean square error (MSE) for determining the optimal neighborhoods, there exists a trade-off between the accuracy of the estimate and the minimum of the true MSE. In this chapter, we study the impact of such a trade-off and formulate the image denoising problem as a coalition formation game. In this game, every pixel is treated as a player, who tries to seek partners to form a coalition to achieve better denoising results. By forming a coalition, every player in the coalition can

obtain certain gains by improving the accuracy of the Stein’s estimate, while incurring some costs by increasing the minimum of the true MSE. Moreover, we show that the traditional approaches using a heuristically determined neighborhood set are special cases of the proposed game theoretical framework by choosing the utility function without a cost term. Finally, experimental results show that the proposed game theoretic approach can achieve better performance than the nonlocal method in terms of both PSNR and visual quality.

### 1.3.6 Simultaneous Image Denoising and Interpolation Using Evolutionary Games (Chapter 7)

While the existing image interpolation approaches can achieve promising interpolation results, they are specially designed for the clean images. However, when the low resolution image is noisy, most of the existing interpolation approaches will also boost the noise and introduces severe visual distortions. Therefore, to achieve better reconstruction, we should jointly consider image denoising and interpolation. In this chapter, we study the problem of simultaneous image denoising and interpolation from the game theoretic perspective and formulate the problem as an evolutionary game. In this evolutionary game, the players are the unknown high resolution pixels and the pure strategies of the players are the corresponding noisy low resolution neighbors. By regarding the non-negative weights of the noisy low resolution pixels as the probabilities of selecting the pure strategies, the problem of estimating the high resolution pixels becomes finding the evolutionarily stable

strategies for the evolutionary game. Experimental results show that the proposed game theoretical approach can achieve better performance than the methods that first denoise the noisy low resolution image and then interpolate the denoised image, in terms of both PSNR and visual quality.

## Chapter 2

### An Overview of Game Theory

Game theory [97] is a mathematical tool that analyzes the strategic interactions among multiple decision makers. Its history dates back to 1944 when J. von Neumann and O. Morgenstern publish the book *Theory of Games and Economic Behavior*. In this book, von Neumann and Morgenstern introduced the method of finding mutually consistent solutions for two-person zero-sum games, which lays the foundation of game theory. During the late 1940s, cooperative game theory had been studied to analyze how groups of individuals should cooperate with each other to improve their positions in a game. In early 1950s, J. Nash developed an important criterion, known as Nash equilibrium, to characterize mutually optimal strategies of players. This concept is applicable to non-zero-sum games, and thus is more general than the criterion proposed by von Neumann and Morgenstern and marks a quantum leap forward in the development of non-cooperative game theory. During the 1950s, many important concepts of game theory were developed, such as the concepts of the core, the extensive form games, repeated games, and the Shapley value. Refinement of Nash equilibriums and the concepts of complete information and Bayesian games were proposed in the 1960s. Application of game theory to biology, i.e., the evolutionary game theory, was introduced by J. M. Smith in the 1970s, during which time, the concepts of correlated equilibrium and common knowledge

were introduced by R. Aumann. In nowadays, game theory has been widely recognized as an important tool in many fields, such as economics, politics, business, social sciences, biology, computer science, and engineering, for understanding cooperation and conflict between individuals. In this chapter, we will present a brief overview on some fundamental concepts of game theory that will be used in this thesis to model and analyze users' behaviors and interaction in multimedia social networks. For more extensive concepts of game theory, the readers are referred to.

## 2.1 Nash Equilibrium and Pareto Optimality

A strategic game  $\langle N, (A_i), (u_i) \rangle$  consists of three components: a set of *players*, denoted by  $N$ ; a set of *actions*, denoted by  $A_i$  for player  $i$ ; and *payoff functions*, denoted by  $u_i : A \rightarrow \mathbb{R}$  for player  $i$ , where  $A = \times_{i \in N} A_i$  is the action set of all players. Generally, one player's payoff depends on not only his/her own action, but also other players' actions, and hence there is a strategic interaction between players.

*Nash equilibrium* is the key concept to understand non-cooperative game theory. Informally speaking, it is an equilibrium where everyone plays the best strategy while taking others' decisions into account. Mathematically,  $a^*$  is a Nash equilibrium if for every player  $i \in N$ ,

$$u_i(a_i^*, a_{-i}^*) \geq u_i(a_i, a_{-i}^*), \quad \forall a_i \in A_i, \quad (2.1)$$

where  $a_i$  denotes the strategy of player  $i$  and  $a_{-i}$  is a common notation in game theory representing the strategies of all players other than player  $i$ . Therefore, Nash

equilibrium predicts the outcome of a game when all players are rational. Depending on whether players choose a single action or randomize over a set of actions according to some probability distribution, an equilibrium can be classified as the *pure-strategy* Nash equilibrium or the *mixed-strategy* Nash equilibrium.

*Pareto optimality* is a strategy profile at which no single player can improve his/her own payoff without hurting any other player. Specifically, let  $\mathbf{u}$  be a vector composed of payoffs in one particular game outcome. Then,  $\mathbf{u}$  is *Pareto efficient* if there is no  $\mathbf{u}'$  of another game outcome for which  $u'_i > u_i$  for all  $i \in N$ ;  $\mathbf{u}$  is *strongly Pareto efficient* if there is no  $\mathbf{u}'$  for which  $u'_i \geq u_i$  for all  $i \in N$  and  $u'_i > u_i$  for some  $i \in N$ . The *Pareto frontier* is defined as the set of all  $\mathbf{u}$  that are Pareto efficient.

## 2.2 Auction Games

Auction theory [77] is an applied branch of game theory which analyzes interactions in auction markets. An auction, conducted by an *auctioneer*, is a process of buying and selling products by eliciting *bids* from potential buyers (i.e., *bidders*) and deciding the auction outcome based on the bids and auction rules. The rules of auction, or *auction mechanisms*, determine whom the goods are allocated to (i.e., the allocation rule) and how much they have to pay (i.e., the payment rule).

The well-known four basic forms of auctions are: *English auction*, *Dutch auction*, *Second-price (sealed-bid) auction*, and *First-price (sealed-bid) auction*. *English auction* is a sequential auction where price increases round by round from a low starting price until only one bidder is left, who wins the product and pays his/her

bid. *Dutch auction* is another sequential auction where price decreases round by round from a high starting price until one bidder accepts the price, who wins the product and pays the price at acceptance. *Second-price (sealed-bid) auction* is the auction where each bidder submits a bid in a sealed envelope simultaneously, and the highest bidder wins the product with payment equal to the second highest bid. *First-price (sealed-bid) auction* is the auction where each bidder submits a bid in a sealed envelope simultaneously, and the highest bidder wins the product with payment equal to his/her own bid.

Although the four basic auctions appear quite different at first glance, they are actually equivalent in some sense under certain conditions [119]. As established in [119] by William Vickrey, a Nobel laureate in Economics, the English auction is equivalent to the second-price sealed-bid auction under the private values model while the Dutch auction is equivalent to the first-price sealed-bid auction since for every strategy in the first-price auction, there is an equivalent strategy in the Dutch auction and vice versa; and given symmetric and risk-neutral bidders and private values, all four auctions yield the same expected revenue of the seller. Therefore, it will suffice to study or adopt only one kind of auction out of the four basic forms.

An auction becomes more complicated when more than one item are simultaneously sold and bidders bid for “packages” of products instead of individual products. This is known as the *combinatorial auction* [40]. One possible approach is the Vickrey-Clarke-Groves (VCG) mechanism, which is the generalized version of the second-price mechanism. The basic idea is that the allocation of products maximizes the social welfare and each winner in the auction pays the opportunity

cost that their presence introduces to all the other bidders. Another approach is the alternative ascending clock auction proposed in [16], where the basic idea is to awarded the items to bidders at the price whenever they are “clinched”.

## 2.3 Evolutionary Games

In some games, there can be more than one Nash equilibrium. When there exist multiple Nash equilibria, one interesting and important problem is how to choose an optimal one in some sense. This process is also known as “equilibria refinement” in game theory. In the literature, several refinement criteria have been proposed, e.g. Pareto optimality is defined to compare multi-dimension payoff profiles. However, the establishment of Pareto optimality is based on the assumption that players have the full knowledge of the game they are playing and others players’ actions, and that players are rational and willing to cooperate in their moves. Nevertheless, this assumption may not be true since players may only have limited information about the other players strategies. Moreover, players may take out-of-equilibrium strategies due to the uncertainty of the game and incorrect/noisy estimate of others’ strategies. To overcome such problems, we need to provide a robust stable equilibrium, and evolutionary game theory is such a theory that can provide the desired stable equilibrium – evolutionarily stable strategy.

Evolutionary game theory is an application of the mathematical theory of games to the interaction dependent strategy evolution in populations [110] [41]. Arising from the realization that frequency dependent fitness introduces a strategic

aspect to evolution, evolutionary game theory becomes an essential component of a mathematical and computational approach to biological contexts, such as genes, viruses, cells, and humans. Recently, however, evolutionary game theory has become of increased interest to economists, sociologists, anthropologists, social scientists, and computer science.

Differs from classical game theory, evolutionary game theory focuses on the dynamics of strategy change more than the properties of strategy equilibria. It can tell us how a rational player should behave to approach a best strategy against a small number of players who do not follow the best strategy, and thus evolutionary game theory can better handle the unpredictable behavior of players.

## 2.4 Coalition Formation Games

The coalition formation game is one type of cooperative game [104], which describes how a set of players can cooperate with others by forming cooperating groups and thus improves their payoffs in a game.

A coalition  $S$  is a nonempty subset of  $N$ , the set of all players. Since the players in coalition  $S$  have agreed to cooperate together, they can be viewed as one entity and is associated with a *value*  $v(S)$  which represents the worth of coalition  $S$ . Then, a coalitional game is determined by  $N$  and  $v(S)$ . When the value  $v(S)$  is the total payoff that can be distributed in any way among the members of  $S$ , e.g., using an appropriate fairness rule, this kind of coalitional games is known as games *with transferrable payoff*. However, in some coalitional games, rigid restrictions exist

on the allocation of the payoff. These games fall into the other category known as games *without transferrable payoff*.

In coalition formation games, often the value  $v(S)$  is determined by two terms: the gain of forming a coalition  $g(S)$  and the cost of forming a coalition  $c(S)$ , i.e.

$$v(S) = g(S) - c(S). \quad (2.2)$$

In general, cooperation by forming larger coalitions is beneficial for players in terms of a higher gain. This property is referred to as *superadditivity*, i.e.,

$$g\left(S_1 \cup S_2\right) \geq g(S_1) + g(S_2), \forall S_1, S_2 \subset N, S_1 \cap S_2 = \emptyset. \quad (2.3)$$

However, on the other hand, forming a larger coalition also require a larger cost, i.e. the cost is also superadditive as follows

$$c\left(S_1 \cup S_2\right) \geq c(S_1) + c(S_2), \forall S_1, S_2 \subset N, S_1 \cap S_2 = \emptyset. \quad (2.4)$$

Therefore, forming larger coalitions are not always beneficial due to the cost term, which means that grand coalition is seldom formed. The objective of coalition formation games is to find the optimal coalition structure  $\mathbf{S}^* = \{S_1^*, S_2^*, \dots, S_l^*\}$ ,  $S_1^* \cup S_2^* \cup \dots \cup S_l^* = N$ , that maximizes the total coalition values, i.e.,

$$\mathbf{S}^* = \{S_1^*, S_2^*, \dots, S_l^*\} = \arg \max_{\mathbf{S}} \sum_i v(S_i). \quad (2.5)$$

## 2.5 Stochastic Games

We have discussed various games, but generally speaking, players are assumed to face the same stage game at each time, meaning the game and the players'

strategies are not depending on the current state of the network. However, this is not true for a dynamic environment where players' strategies keep changing over time. In order to study the cooperation and competition behaviors under such a dynamic environment, the theory of stochastic games might be a better fit.

A stochastic game [108] is an extension of Markov decision process (MDP) [101] by considering the interactive competition among different agents. In a stochastic game, there is a set of states, denoted by  $\mathcal{S}$ , and a collection of action sets,  $A_1, \dots, A_{|N|}$ , one for each player in the game. The game is played in a sequence of stages. At the beginning of each stage the game is in a certain state. After the players select and execute their actions, the game then moves to a new random state with some transition probability determined by the current state and actions from all players:  $T : \mathcal{S} \times A_1 \times \dots \times A_{|N|} \mapsto PD(\mathcal{S})$ . Meanwhile, at each stage each player receives a payoff  $u_i : \mathcal{S} \times A_1 \times \dots \times A_{|N|} \mapsto \mathbb{R}$ , which also depends on the current state and all the chosen actions. The game is played continually for a number of stages, and each player attempts to maximize an objective function. Like in the repeated game, the overall payoff function is defined as the expected sum of discounted intermediate payoffs.

The solution, also called a *policy* of a stochastic game is defined as a probability distribution over the action set at any state,  $\pi_i : \mathcal{S} \rightarrow PD(A_i)$ , for all  $i \in N$ . Given the current state  $s^t$  at time  $t$ , if player  $i$ 's policy  $\pi_i^t$  at time  $t$  is independent of the states and actions in all previous time slots, the policy  $\pi_i$  is said to be *Markov*. If the policy is further independent of time, it is said to be *stationary*.

The stationary policy of the players in a stochastic game, i.e., their optimal

strategies, can be obtained by value iteration according to Bellman's optimality condition. For example, in a two-player stochastic game with opposite objectives, let us denote  $V(s)$  as the expected reward (of player 1) for the optimal policy starting from state  $s$ , and  $Q(s, a_1, a_2)$  as the expected reward of player 1 for taking action  $a_1$  against player 2 who takes action  $a_2$  from state  $s$  and continuing optimally thereafter [85]. Then, the optimal strategy for player 1 can be obtained from the following iterations,

$$V(s) = \max_{\pi} \min_{a_2 \in A_2} \sum_{a_1 \in A_1} Q(s, a_1, a_2) \pi_{a_1}, \quad (2.6)$$

$$Q(s, a_1, a_2) = u_1(s, a_1, a_2) + \delta \sum_{s' \in \mathcal{S}} T(s, a_1, a_2, s') V(s'), \quad (2.7)$$

where  $\pi_{a_1}$  denotes player 1's strategy profile, and  $T(s, a_1, a_2, s')$  denotes the transition probability from state  $s$  to  $s'$  when player 1 takes  $a_1$  and player 2 takes  $a_2$ .

## Chapter 3

### Multuser Rate Allocation Social Networks

Nowadays, due to the explosive growth of the Internet and the advance of compression technologies, delay-sensitive multimedia networking applications such as multimedia streaming and multi-camera surveillance become more and more popular. Therefore, a fundamental problem in these applications, how to fairly and efficiently allocate the rate among many users who share the same network bandwidth, becomes more and more important and draws great attention recently.

Rate allocation for a single user has been well investigated in the literature [39] [35] [48]. In single-user rate allocation, the task of the rate controller is to assign the available rate to each frame and each macroblock (MB) to achieve the maximal visual quality. This is also known as rate control. The simplest rate control method is the constant bit-rate allocation (CBR), which equally allocates the bit-rate to each frame. However, CBR often results in quality fluctuation, due to which the overall visual quality is significantly degraded. To overcome this problem, variable bit-rate allocation (VBR) is proposed for constant quality reconstruction by assigning rate according to the complexity of each frame [78]. A core technique in VBR-based rate control methods is rate distortion modelling [64], which highly affects the rate control performance. Many works have been done on rate distortion modelling, including parametric method [137] and non-parametric method [134].

If a channel is shared by multiple users, besides considering the rate allocation within the same user (i.e., frame-level rate allocation and MB-level rate allocation), the rate controller needs to consider the rate allocation among different users. This becomes the multi-user rate allocation problem. Similar to frame-level rate allocation, the simplest multi-user rate allocation is the constant bit-rate allocation (CBR), where the available network bandwidth is equally assigned to each user. A major problem of CBR is that it does not consider the variable bit-rate characteristics of the video sequences. One way to overcome this disadvantage is to optimize a global objective function that involves the characteristics of all the video sequences using conventional optimization methods such as Lagrangian or dynamic programming [95]. For example, a commonly adopted method is for the rate controller to minimize the weighted sum of the distortions or try to maximize the weighted sum of the PSNRs, i.e., the optimization problem becomes:

$$\min_{R_i} \sum_{i=1}^N w_i D_i(R_i), \quad s.t. \sum_{i=1}^N R_i \leq R, \quad (3.1)$$

or

$$\max_{R_i} \sum_{i=1}^N w_i PSNR_i(R_i), \quad s.t. \sum_{i=1}^N R_i \leq R, \quad (3.2)$$

where  $R$  is the available network bandwidth,  $w_i$  is the weight,  $D_i$  is the distortion, and  $PSNR_i$  is the PSNR of the  $i^{th}$  user.

Notice that the solution to the above optimization-based methods is highly related to the selection of the weights  $w_i$ . However, in the literature, the weights  $w_i$ 's are usually heuristically determined, e.g.,  $w_i$  is uniformly set to be  $1/N$  [109]. Moreover, such a formulation can only address the efficiency issue, e.g., how to maximize

the weighted sum of the PSNRs or minimize the weighted sum of the distortions. As such, the fairness issue, which is an important problem for multi-user rate allocation, has been generally ignored in the image/video/multimedia community.

However, in the networking literature, the fairness issue in multi-user rate allocation have been considered in a different setting. In [105], the authors formulated the optimal channel-assignment problem as a convex optimization problem using a max-min fairness criterion for the downlink application. As pointed out in [61], the max-min approach deals with the worst-cast scenario, so it favors users with worse channels and reduces the system efficiency. To overcome the disadvantage, the authors in [61] considered a generalized proportional fairness based on the Nash bargaining solutions and coalitions. While this proportional fairness criterion was successfully employed in networking applications, it cannot be directly used in content-aware multimedia applications since it does not explicitly consider the characteristics of the video content and the resulting impact on video quality. In [98], the authors applied the Nash bargaining solutions to the multimedia multi-user rate allocation problem, where the utility function for each user is defined as the inverse of the distortion. But there are two main drawbacks of that utility function. Firstly, since no cost in video transmission is considered, every user can overclaim his/her need to get more bandwidth regardless the consequence to the system, which is recognized as selfish behavior. Due to the selfish nature, without a cost, all users will become too greedy and want to get as much bit-rate as possible, which is not good to the system [106]. Secondly, since the gain is defined as the inverse of the distortion, i.e., an exponential function of the PSNR, a certain increase of the bit-

rate in the low PSNR region will lead to a less significant gain than that in the high PSNR region. This contradicts with the human visual system (HVS) model since the quality difference in the low PSNR region is easier to be distinguished than that in the high PSNR region (see Section 3.1.3 for details). Moreover, with the utility function defined in [98], the generalized Nash bargaining solution is shown to be the same as the traditional optimization-based approach in (3.2), i.e., to maximize the weighted sum of the PSNRs, while the weights are determined by the bargaining powers, which are still heuristically determined.

In this chapter, we propose a multi-user rate allocation game framework to efficiently and fairly allocate the available network bandwidth to different multimedia users. The utility/payoff function of each user/player is defined according to the characteristics of the transmitted video sequences and the allocated bit-rate. Specifically, motivated by the intuition that the quality difference in the low PSNR region is easier to be distinguished than that in the high PSNR region, we define the gain as a logarithm function of the PSNR. We also introduce a cost term in the utility function, which is linear in the allocated rate, to guide users' behaviors. In this way, the users will be more rational in choosing bit-rate since transmitting data with a higher bit-rate in this case does not necessarily result in a higher payoff, especially when the transmitted video sequence is a fast motion and complex scene sequence. Then, we discuss the Nash equilibrium (NE) of this rate allocation game. We show that with a unique NE, which is proportionally fair in terms of both utility and PSNR, can be obtained, based on which the rate controller can efficiently and fairly allocate the available rate. Moreover, we propose a decentralized cheat-proof

rate allocation scheme for the users to converge to the unique NE using alternative ascending clock auction [16]. We also show that the traditional optimization-based method in (3.2) is a special case of the game-theoretic framework if the utility function is defined as an exponential function of PSNR. This fact indicates that the game-theoretic approach offers a more general and unified solution, especially in a multi-user setting. Finally, we illustrate several experimental results on real video data to demonstrate the efficiency and effectiveness of the proposed game-theoretic multi-user multimedia rate allocation method.

The rest of this chapter is organized as follows. In Section 3.1, we give a detailed description on the proposed method, including the system model, how to define the utility function, and the problem formulation. In Section 3.2, we provide a detailed analysis of the proposed game-theoretic framework. In Section 3.3, we show the relationship between the proposed game-theoretic method and the traditional optimization-based approach. In Section 3.4, we describe in details the proposed distributed cheat-proof rate allocation scheme using alternative ascending clock auction. Finally, we illustrate the experimental results on real video signals in Section 3.5 and draw conclusions in Section 3.6.

## 3.1 The Game-Theoretic Framework

### 3.1.1 System Model

As shown in Figure 3.1, in our system, we assume that there is a controller,  $N$  transmitters,  $u_1, u_2, \dots, u_N$ , and  $N$  receivers,  $r_1, r_2, \dots, r_N$ . User  $u_i$  transmits

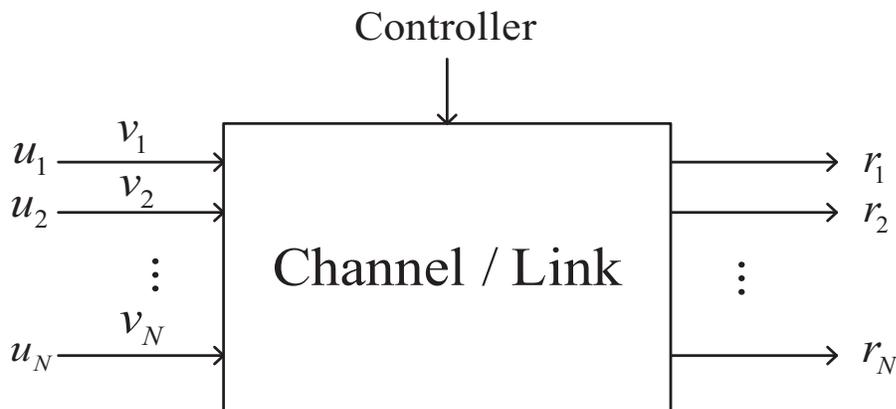


Figure 3.1: System Model.

the video sequence  $v_i$  to the corresponding receiver  $r_i$  through a channel/link that is shared by other users  $u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_N$ . Since the channel has a limited bandwidth, it may not be able to satisfy the bandwidth requirements for all users. The role of the controller is to allocate the channel bandwidth to users  $u_1, u_2, \dots, u_N$ . So, the question is how the controller allocates the bandwidth to the users in an efficient and fair way? We will formally define the notion of fairness later.

### 3.1.2 Video Distortion-Rate Model

Before answering the question raised in the above subsection, let us first discuss the Distortion-Rate (DR) model for the video sequences. In video compression, due to the quantization process, there exists a tradeoff between the distortion ( $D$ ), which is usually defined as the mean squared error (MSE), and bit-rate ( $R$ ), which determines the channel bandwidth or storage space required to transmit or store the coded data. Generally, high bit-rate leads to small distortion while low bit-rate causes large distortion. In the literature, several models have been proposed to

characterize this distortion rate tradeoff for different video coders, such as MPEG2 [63] [48], MPEG4 [39] [35], FGS [46], H.263 [114], H.264 [31] [83], and wavelet-based coders [124]. Without loss of generality, in this chapter, we use a simple two-parameter distortion-rate model, which is widely employed in a medium or high bit-rate situation, and other models can be similarly analyzed. The two-parameter distortion-rate model is described as follows:

$$D(R) = \alpha e^{-\beta R}, \quad (3.3)$$

where  $\alpha$  and  $\beta$  are two positive parameters determined by the characteristics of the video content.

### 3.1.3 User's Utility Function

As shown in Figure 3.1, user  $u_i$  can get gain by successfully transmitting the video  $v_i$  to receiver  $r_i$ , and the gain is determined by the quality of the transmitted video. On the other hand, user  $u_i$  needs to pay for the used bandwidth to transmit  $v_i$ , and the payment is determined by the bit-rate of  $v_i$ . Therefore, given the profile of  $u_i$ , the bit-rate  $R_i$  and distortion  $D_i$ , the utility function of user  $u_i$  can be defined as:

$$U_i(R_i, D_i) = f(D_i) - ag(R_i). \quad (3.4)$$

where  $f(D_i)$  is the gain,  $g(R_i)$  is the cost, and  $a$  is a parameter controlling the balance between the gain and cost.

Generally, since the gain of  $u_i$  will be larger if the distortion  $D_i$  is smaller, the function  $f(\cdot)$  should be a monotonically decreasing function. Similarly, since the

cost of  $u_i$  will be larger if the bit-rate  $R_i$  is larger, the function  $g(\cdot)$  should be a monotonically increasing function. Without loss of generality, we assume that the cost per bit-rate unit is one, which means:

$$g(R_i) = R_i. \quad (3.5)$$

The gain  $f(D_i)$  is generally determined by how much receiver  $r_i$  is satisfied with the received video. In video processing and coding community, the PSNR is a more common objective quality measure than MSE. For any MSE, i.e., the distortion  $D$ , the corresponding PSNR is given by:

$$PSNR = 10 \log_{10} \frac{255^2}{D}. \quad (3.6)$$

Moreover, according to the human visual system (HVS) model, the quality difference in the low PSNR region is easier to be distinguished than that in the high PSNR region, e.g., as shown in Figure 3.2, the 33dB and 34dB images are easier to be distinguished than the 40dB and 41dB images. Therefore, we define the  $f(\cdot)$  function as:

$$f(D_i) = \ln(PSNR_i) = \ln\left[10 \log_{10} \frac{255^2}{D_i}\right]. \quad (3.7)$$

Note that the reason of using  $\ln(\cdot)$  function is that  $\ln(\cdot)$  is a monotonically increasing function in its argument and its second order derivative is negative, due to which a certain increase in the low PSNR region will lead to a more significant gain than that in the high PSNR region. Other functions that have similar properties can also be used. Moreover, if we do not consider the distinct characteristics of video signals, any monotonically decreasing function of the distortion  $D$  can be used, e.g.,

$$f(D) = 255^2/D = e^{\eta PSNR}, \quad (3.8)$$



Figure 3.2: The visual quality of Foreman sequence at different PSNR level: (a) 33dB; (b) 34dB; (c) 40dB; (d) 41dB.

or

$$f(D) = 10 \log_{10} \frac{255^2}{D} = PSNR. \quad (3.9)$$

Combining (3.3)-(3.7) and ignoring the constant term, the utility function of user  $u_i$  becomes:

$$U_i(R_i) = \ln(\gamma_i + \beta_i R_i) - a R_i. \quad (3.10)$$

where  $\gamma_i = 2 \ln 255 - \ln \alpha_i$ .

### 3.1.4 Multi-User Rate Allocation Game

To answer the question raised in Section 3.1.1, we formulate this problem as a multi-user rate allocation game. As shown in Figure 3.1, in this game, there are  $N$  users/players, who share the available network bandwidth with each other. Each user  $u_i$  has his/her own utility function as shown in (3.10), and it also has a minimum desired quality constraint (minimal rate constraint  $R_i^{min}$ ) and a maximum satisfied quality constraint (maximum rate constraint  $R_i^{max}$ ). Since  $R_i^{min}$  is the minimal rate constraint that each user expects by jointing in the game, we assume that the available network rate at least guarantees each user for the minimal desired rate in the game. Obviously, if the available network bandwidth is able to satisfy all the users with the maximum quality constraint  $R_i^{max}$ , the rate allocation problem is trivial since the controller just needs to allocate  $R_i^{max}$  to each user  $u_i$ . However, in the case that the available network bandwidth is not able to satisfy all the user with  $R_i^{max}$ , the problem becomes more interesting: how does the controller fairly and efficiently allocate the available bandwidth to the users? From the users' point of view, they try to maximize their utilities subject to the constraint that the sum of the users' bit-rate does not exceed the available bandwidth. Therefore, the game can be formulated as:

$$\begin{aligned}
 \max_{R_i} \quad & U_i(R_i) = \ln(\gamma_i + \beta_i R_i) - aR_i, \\
 \text{s.t.} \quad & R_i^{min} \leq R_i \leq R_i^{max}, \quad \forall i = 1, 2, \dots, N, \\
 & \sum_{i=1}^N R_i \leq R,
 \end{aligned} \tag{3.11}$$

where  $R$  is the available network bandwidth.

## 3.2 Analysis of The Multi-User Rate Allocation Game

According to (3.10), we can see that the utility function  $U_i(R_i)$  is a concave function in terms of  $R_i$ . By taking the derivative of  $U_i(R_i)$  over  $R_i$ , we have:

$$\frac{\partial U_i(R_i)}{\partial R_i} = \frac{\beta_i}{\gamma_i + \beta_i R_i} - a, \quad \forall i = 1, 2, \dots, N. \quad (3.12)$$

Therefore, user  $u_i$  achieves his/her maximal utility  $U_i^*(R_i^*)$  at  $R_i^*$ , where  $R_i^*$  is defined as:

$$R_i^* = \max[R_i^{min}, \min(\frac{1}{a} - \frac{\gamma_i}{\beta_i}, R_i^{max})], \quad \forall i = 1, 2, \dots, N. \quad (3.13)$$

From (3.13), we can see that the optimal  $R_i^*$  corresponding to the maximal utility is determined by the parameter  $a$ . Therefore, for different choices of  $a$ , the game in (3.11) has different equilibria with different physical meanings. Specifically, in the following, we discuss the Nash equilibrium (NE) in three different cases:  $a > a_0$ ,  $0 \leq a < a_0$ , and  $a = a_0$ , where  $a_0$  is the constant that satisfies the following equation:

$$\sum_{i=1}^N \left( \max[R_i^{min}, \min(\frac{1}{a_0} - \frac{\gamma_i}{\beta_i}, R_i^{max})] \right) = R. \quad (3.14)$$

### 3.2.1 Non-Efficient Rate Allocation ( $a > a_0$ )

If  $a > a_0$ , the game in (3.11) has a unique Nash Equilibrium  $(R_1^*, R_2^*, \dots, R_N^*)$ . Since  $a > a_0$ , from (3.14), we have  $\sum_{i=1}^N R_i^* < R$ , which means that the available network bandwidth is not fully utilized. Therefore, this allocation scheme is not efficient.

### 3.2.2 Efficient Rate Allocation ( $0 \leq a < a_0$ )

If  $0 \leq a < a_0$ , the game in (3.11) has infinitely many NE. For every NE  $(\tilde{R}_1, \tilde{R}_2, \dots, \tilde{R}_N)$ , according to Lemma 1, we have  $\sum_{i=1}^N \tilde{R}_i = R$ , which means that the available network bandwidth is fully utilized. Therefore, this allocation scheme is efficient.

*Lemma 1:* When  $0 \leq a < a_0$ , every NE  $(\tilde{R}_1, \tilde{R}_2, \dots, \tilde{R}_N)$  satisfies  $\sum_{i=1}^N \tilde{R}_i = R$ .

*Proof:* Since  $\sum_{i=1}^N \tilde{R}_i \leq R$ , let us assume that there is a NE  $(\tilde{R}_1, \tilde{R}_2, \dots, \tilde{R}_N)$  such that  $\sum_{i=1}^N \tilde{R}_i = R - \Delta < R$ . Since  $0 \leq a < a_0$ , we have  $\sum_{i=1}^N R_i^* > R$ , which means there exists at least one  $\tilde{R}_j$  such that  $\tilde{R}_j < R_j^*$ . Let  $\hat{R}_j = \min(\tilde{R}_j + \Delta, R_j^*)$ , then  $\sum_{i=1}^{j-1} \tilde{R}_i + \hat{R}_j + \sum_{i=j+1}^N \tilde{R}_i \leq R$  and  $U_j(\hat{R}_j) > U_j(\tilde{R}_j)$  (due to the concavity of the utility function). This contradicts with the assumption that  $(\tilde{R}_1, \tilde{R}_2, \dots, \tilde{R}_N)$  is a NE. Therefore,  $\sum_{i=1}^N \tilde{R}_i = R$ . This completes the proof.

### 3.2.3 Efficient and Proportionally Fair in Both Utility and PSNR

$$(a = a_0)$$

If  $a = a_0$ , the game in (3.11) has a unique Nash Equilibrium  $(R_1^*, R_2^*, \dots, R_N^*)$ . According to (3.14), we have  $\sum_{i=1}^N R_i^* = R$ , which means that the available network bandwidth is fully utilized. Therefore, this allocation scheme is efficient.

Moreover, we will show in the following definition [72] and theorem that when  $a = a_0$ ,  $(R_1^*, R_2^*, \dots, R_N^*)$  is a proportionally fair NE in terms of both utility and PSNR.

*Definition 1:* A utility distribution is said to be proportionally fair when any

change in the distribution of utilities results in the sum of the proportional changes being non-positive, i.e.,

$$\sum_i \frac{U_i - \tilde{U}_i}{\tilde{U}_i} \leq 0, \quad \forall U_i \in \mathbf{S}. \quad (3.15)$$

where  $\tilde{U}_i$  and  $U_i$  are the proportionally fair utility and any other feasible utility for the  $i^{\text{th}}$  user, respectively, and  $\mathbf{S}$  is a closed and convex subset of  $\Re^N$  to represent the set of feasible utility functions that the users can achieve.

**Remark:** The definition of proportional fairness comes from the fact that, if  $(\tilde{U}_1, \tilde{U}_2, \dots, \tilde{U}_N)$  satisfies (3.15), any deviation from  $(\tilde{U}_1, \tilde{U}_2, \dots, \tilde{U}_N)$  will lead to a non-increasing sum of the proportional changes. Moreover, from [72] and [61], we can see that  $(\tilde{U}_1, \tilde{U}_2, \dots, \tilde{U}_N)$  is a proportionally fair utility if and only if  $\prod_{i=1}^N \tilde{U}_i \geq \prod_{i=1}^N U_i$  for any feasible  $(U_1, U_2, \dots, U_N)$ .

*Theorem 1:* When  $a = a_0$ ,  $(R_1^*, R_2^*, \dots, R_N^*)$  is a proportionally fair NE in terms of both utility and PSNR.

*Proof:* According to the above remark,  $(R_1^*, R_2^*, \dots, R_N^*)$  is a proportionally fair NE in terms of both utility and PSNR if and only if it is the solution to the following two optimization problems:

$$\begin{aligned} \max_{R_i} \quad & \prod_{i=1}^N U_i = \prod_{i=1}^N [\ln(\gamma_i + \beta_i R_i) - a R_i] \\ \text{s.t.} \quad & R_i^{\min} \leq R_i \leq R_i^{\max}, \quad \forall i = 1, 2, \dots, N, \\ & \sum_{i=1}^N R_i \leq R, \end{aligned} \quad (3.16)$$

and

$$\begin{aligned}
\max_{R_i} \quad & \prod_{i=1}^N PSNR_i = \prod_{i=1}^N (\gamma_i + \beta_i R_i) \\
s.t. \quad & R_i^{min} \leq R_i \leq R_i^{max}, \quad \forall i = 1, 2, \dots, N, \\
& \sum_{i=1}^N R_i \leq R.
\end{aligned} \tag{3.17}$$

- We first show that  $(R_1^*, R_2^*, \dots, R_N^*)$  is a proportionally fair NE in terms of utility. Since user  $u_i$  achieves his/her maximal utility  $U_i^*(R_i^*)$  at  $R_i^*$ , we have  $U_i(R_i^*) \geq U_i(R_i)$ , for any  $R_i$  satisfies  $R_i^{min} \leq R_i \leq R_i^{max}$  and  $\sum_{i=1}^N R_i \leq R$ . This means that  $\prod_{i=1}^N U_i(R_i^*) \geq \prod_{i=1}^N U_i(R_i)$  for any feasible  $(R_1, R_2, \dots, R_N)$ . Therefore,  $(R_1^*, R_2^*, \dots, R_N^*)$  is a proportionally fair NE in terms of utility.
- We then show that  $(R_1^*, R_2^*, \dots, R_N^*)$  is a proportionally fair NE in terms of PSNR. Since maximizing  $\prod_i x_i$  is the same as maximizing  $\sum_i \ln(x_i)$ , the optimization problem in (3.17) is equivalent to the following optimization problem:

$$\begin{aligned}
\max_{R_i} \quad & \sum_{i=1}^N \ln(\gamma_i + \beta_i R_i) \\
s.t. \quad & R_i^{min} \leq R_i \leq R_i^{max}, \quad \forall i = 1, 2, \dots, N, \\
& \sum_{i=1}^N R_i \leq R,
\end{aligned} \tag{3.18}$$

Since the above optimization problem is convex, the optimal solution can be found by solving the Karush-Kuhn-Tucker (KKT) conditions [19]. We first write the Lagrangian of problem (3.18) as:

$$\begin{aligned}
L(R_i, \lambda, \kappa_i, \nu_i) = & - \sum_{i=1}^N \ln(\gamma_i + \beta_i R_i) + \lambda \left( \sum_{i=1}^N R_i - R \right) \\
& + \sum_{i=1}^N \kappa_i (R_i - R_i^{max}) + \sum_{i=1}^N \nu_i (R_i^{min} - R_i).
\end{aligned} \tag{3.19}$$

Then, the KKT conditions are:

$$\begin{aligned}
-\frac{\beta_i}{\gamma_i + \beta_i R_i} + \lambda + \kappa_i - \nu_i &= 0; \\
\lambda \left( \sum_{i=1}^N R_i - R \right) &= 0; \\
\kappa_i (R_i - R_i^{max}) &= 0; \quad \forall i = 1, \dots, N. \\
\nu_i (R_i^{min} - R_i) &= 0; \quad \forall i = 1, \dots, N. \\
R_i^{min} \leq R_i &\leq R_i^{max}; \quad \forall i = 1, \dots, N. \\
\sum_{i=1}^N R_i &\leq R; \\
\lambda \geq 0, \kappa_i \geq 0, \nu_i \geq 0; &\quad \forall i = 1, \dots, N.
\end{aligned} \tag{3.20}$$

By solving the KKT conditions above, the optimal solution is:

$$R_i^* = \max[R_i^{min}, \min(\frac{1}{\lambda} - \frac{\gamma_i}{\beta_i}, R_i^{max})], \quad \forall i = 1, \dots, N. \tag{3.21}$$

where  $\sum_{i=1}^N \max[R_i^{min}, \min(\frac{1}{\lambda} - \frac{\gamma_i}{\beta_i}, R_i^{max})] = R$ .

Therefore,  $(R_1^*, R_2^*, \dots, R_N^*)$  is the solution to the optimization problem in (3.17), which means that it is a proportionally fair NE in terms of PSNR.

In all, when  $a = a_0$ ,  $(R_1^*, R_2^*, \dots, R_N^*)$  is a proportionally fair NE in terms of both utility and PSNR. This completes the proof.

**Remark:** From the above analysis, we can see that choosing  $a = a_0$  is the best among the three different cases due to the following four reasons: (1) a unique proportionally fair NE in terms of both utility and PSNR can be found when  $a = a_0$ ; (2) with the unique proportionally fair NE, the available network bandwidth will be fully utilized; (3) since the optimal solution shown in (3.21) is very simple, no optimization is needed and the computational complexity is low; and (4) a distributed

algorithm can be designed for the users to converge to the unique NE which will be discussed in Section 3.4.

### 3.3 Relation To The Traditional Optimization-Based Approach

While the task of rate allocation for a single user is to find the best trade-off point on the rate-distortion curve, the traditional optimization-based multi-user rate allocation approach can be seen as first constructing an overall rate-distortion curve by combining rate-distortion curves of all users, and then finding the best trade-off point on the joint rate-distortion curve. However, it is difficult to construct the overall rate-distortion curve from all users' rate-distortion curve. The approach shown in (3.2) is one possible way, but there is no notion of fairness. Furthermore, the weights in (3.2) are hard to determine and are usually defined heuristically.

Instead of focusing on finding a good way of constructing the overall rate-distortion curve, the proposed game-theoretic framework considers each user's rate-distortion trade-off in the utility function. Then, the notion of proportional fairness, is introduced to balance the rate allocation among different users and to make sure that the total rate constraint is satisfied. Moreover, from (3.2), (3.4), and (3.16), we can see that the traditional optimization-based approach shown in (3.2) is actually a special case of the proposed game-theoretic framework by choosing the gain function and the cost function as follows:

$$f(D_i) = e^{w_i PSNR_i}, g(R_i) = 0, \quad (3.22)$$

which means,

$$U_i(R_i, D_i) = e^{w_i PSNR_i}. \quad (3.23)$$

Note that there are mainly three drawbacks of this kind of utility function:

- The parameters  $w_i$  are usually heuristically determined.
- If no cost in video transmission is considered, selfish users may become too greedy and want to get as much bit-rate as possible, which is not good to the system [106].
- Since the gain is defined as an exponential function of the PSNR, a certain increase of the bit-rate in the low PSNR region will lead to a less significant gain than that in the high PSNR region. This contradicts with the human visual system (HVS) model since the quality difference in the low PSNR region is easier to be distinguished than that in the high PSNR region.

### 3.4 Clock Auction For Distributed Cheat-Proof Optimal Rate Allocation

In Section 3.2, we have discussed the NE of the multi-user rate allocation game for different  $a$ 's and found that, when  $a = a_0$ , the game has a proportionally fair NE in both utility and PSNR. However, we have not discussed how to obtain  $a_0$  and how the users converge to the NE yet. There are two possible approaches, centralized approach and distributed approach. For the centralized approach, the controller knows exactly all the private information of each user, i.e.,  $\gamma_i, \beta_i, R_i^{min}$ ,

and  $R_i^{max}$ . Then, the controller can first find  $a_0$  in a collective way by solving (3.14) and then allocate  $R_i^*$  to  $u_i$ .

However, in general, the users can be geographically distributed in many places, it is therefore not feasible for the controller to collect all the private information of each user. Moreover, since the users are selfish, e.g., they tend to overclaim what they may need, they will not truly report their private information if cheating can improve their utilities [62]. To solve this problem, we propose a distributed cheat-proof rate allocation scheme using alternative ascending clock auction [16]. An auction is a decentralized mechanism for allocating resources, where there is an auctioneer and several bidders. The auction processes can be described as follows: the auctioneer announces a price, bidders report to the auctioneer their demands at that price, and the auctioneer raises the price until the total demand meets the supply. In our multi-user rate allocation problem, the controller is the auctioneer and the users are the bidders.

The proposed rate allocation scheme is described in Algorithm 1. As shown in Algorithm 1, before the auction, the controller sets up the step size  $\delta > 0$ , clock index  $t = 0$ , and initializes  $a$  with a small value  $a^0$ . At the beginning of clock  $t$ , the controller first announces  $a^t$  to all the users. Then, each user submits his/her optimal demand to the controller. After collecting all the demands, the controller compares the total demand  $R_{total}$  with the available bandwidth  $R$ . If  $R_{total} > R$ , i.e., the total demand exceeds the supply, the auction is not concluded. The controller continues the auction and goes to next clock  $t + 1$  with an increased  $a$  computed by  $a^{t+1} = a^t + \delta$ . Moreover, the controller computes the *cumulative clinch*, which is

the amount of bit-rate that a user is guaranteed to win at current clock given by

$$C_i^t = \max(0, R - \sum_{j \neq i} R_j^t). \quad (3.24)$$

On the other hand, if  $R_{total} \leq R$ , then the supply can meet all users' demands and the auction is concluded. Let the final clock index be  $L$ . As  $a$  increases discretely, we may have  $R_{total} < R$  and do not fully utilize the bandwidth. To make sure that  $R_{total} = R$ , we modify (3.24) by introducing proportional rationing [16], and the final cumulative clinch of  $u_i$  is given by,

$$C_i^L = R_i^L + \frac{R_i^{L-1} - R_i^L}{\sum_i R_i^{L-1} - \sum_i R_i^L} [R - \sum_i R_i^L], \text{ with } \sum_i C_i^L = R. \quad (3.25)$$

Finally, the rate allocated to  $u_i$  is  $R_i^* = C_i^L$ . The utility of  $u_i$  is obtained as,

$$U_i^* = \ln(\gamma_i + \beta_i R_i^*) - P_i^*, \quad (3.26)$$

where  $P_i^* = C_i^0 a^0 + \sum_{t=1}^L a^t (C_i^t - C_i^{t-1})$  is the payment from user  $u_i$ .

**Remark:** Since  $a^{t+1} > a^t$ , we have  $R_j^{t+1} \leq R_j^t$ . Therefore, at clock  $t$ ,  $u_i$  is guaranteed at least the amount of bit-rate  $C_i^t = \max(0, R - \sum_{j \neq i} R_j^t)$ . This is how (3.24) comes from.

The rate allocation scheme described in Algorithm 1 has several advantages:

- The auction process is transparent to all users and simple enough for all users to understand. Simplicity and transparency are two important factors to stimulate auction since users may not be willing to join in the game if they do not understand the auction process.
- The auction scheme can preserve privacy. Since the scheme is distributed,

---

**Algorithm 1** Cheat-Proof Rate Allocation Scheme Using Clock Auction

---

Given the available bandwidth  $R$ , step size  $\delta > 0$ , and clock index  $t = 0$ , the

controller initializes  $a$  with a small value  $a^0$ .

Repeat:

(1) the controller announces  $a^t$  to all the users.

(2) Each user  $u_i$  submits his/her optimal demand:

$$R_i^t = \max[R_i^{min}, \min(\frac{1}{a^t} - \frac{\gamma_i}{\beta_i}, R_i^{max})].$$

(3) The controller sums up all the demand  $R_{total}^t = \sum_i R_i^t$  and compares  $R_{total}^t$

with  $R$ :

If  $R_{total}^t > R$ , compute  $C_i^t = \max(0, R - \sum_{j \neq i} R_j^t)$ , set  $a^{t+1} = a^t + \delta$ ,  $t = t + 1$ ,

and go to (1).

Else, conclude the auction, set  $L = t$ , compute  $C_i^L = R_i^L + \frac{R_i^{L-1} - R_i^L}{\sum_i R_i^{L-1} - \sum_i R_i^L} [R - \sum_i R_i^L]$ ,

and allocate  $R_i^* = C_i^L$  to  $u_i$ .

Finally, the payment of  $u_i$  is  $P_i^* = C_i^0 a^0 + \sum_{t=1}^L a^t (C_i^t - C_i^{t-1})$  and the utility of

$u_i$  is  $U_i^* = \ln(\gamma_i + \beta_i R_i^*) - P_i^*$ .

---

users do not need to report their private information. Instead, they only need to submit their demands.

- The computational complexity of each user is low since what the users need to do is to submit their optimal demands calculated by  $R_i = \max[R_i^{min}, \min(\frac{1}{a} - \frac{\gamma_i}{\beta_i}, R_i^{max})]$  for any given  $a$ .
- The computational complexity of the controller is low in that the controller only needs to sum up the demands from the users, compare it with the available bandwidth, and compute the cumulative clinch for each user.
- Through the auction, each user will converge to the unique proportionally fair NE shown in Section 3.2. This is trivial due to the following two reasons: (1) since the auction concludes if and only if  $\sum_i R_i^t \leq R$ , when  $\delta$  is sufficient small, the auction will conclude at  $\sum_i R_i^t = R$ . (2) At each clock  $t$ ,  $u_i$  chooses  $R_i^t = \max[R_i^{min}, \min(\frac{1}{a^t} - \frac{\gamma_i}{\beta_i}, R_i^{max})]$ .
- The scheme is cheat-proof, meaning that the best strategy of each user is to report his/her true optimal demand at every clock. There is no incentive for  $u_i$  to deviate, and the proof is shown in Theorem 2.

Let  $R_i^t$  be user  $u_i$ 's true optimal demand at clock  $t$ , and  $\tilde{R}_i^t$  be the claimed demand that  $u_i$  reports to the controller at clock  $t$ . Note that  $\tilde{R}_i^t$  can be any value in  $[R_i^{min}, R_i^{max}]$  if  $u_i$  cheats at clock  $t$ . Let  $\Gamma(t, L) = \{L, \tilde{R}_i^0, \dots, \tilde{R}_i^t, R_i^{t+1}, \dots, R_i^L; C_i^0, \dots, C_i^L; a^0, \dots, a^L\}$  be the profile of  $u_i$  at the following scenario: from clock 0 to clock  $t$ ,  $u_i$  reports  $\tilde{R}_i^0, \dots, \tilde{R}_i^t$ , and from clock  $t + 1$  to the final clock,  $u_i$  reports

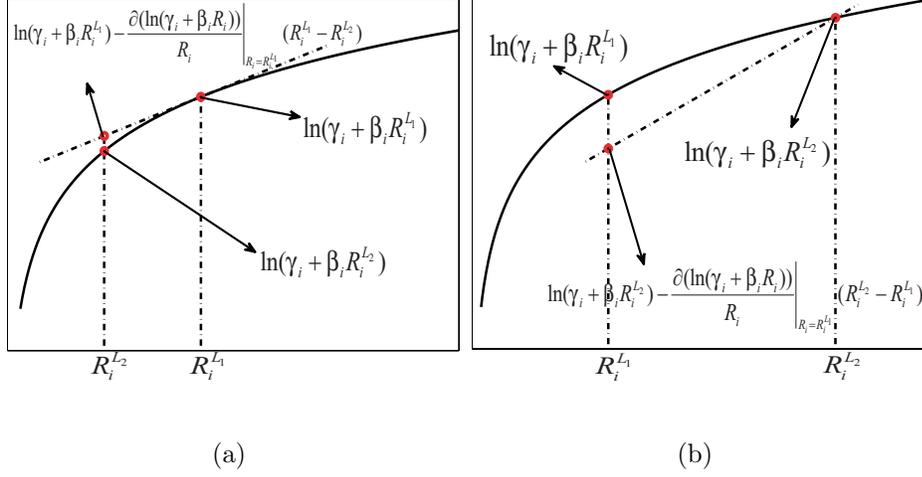


Figure 3.3: Illustration for the proof of Lemma 2: (a) If  $\tilde{R}_i^t \leq R_i^t$  and  $R_i^{L1} > R_i^{min}$ , we can see that  $[\ln(\gamma_i + \beta_i R_i^{L1}) - \frac{\partial \ln(\gamma_i + \beta_i R_i)}{\partial R_i} \Big|_{R_i=R_i^{L1}} (R_i^{L1} - R_i^{L2})] \geq \ln(\gamma_i + \beta_i R_i^{L2})$ ; (b) If  $\tilde{R}_i^t > R_i^t$  and  $R_i^{L1} < R_i^{max}$ , we can see that  $\ln(\gamma_i + \beta_i R_i^{L1}) \geq [\ln(\gamma_i + \beta_i R_i^{L2}) - \frac{\partial \ln(\gamma_i + \beta_i R_i)}{\partial R_i} \Big|_{R_i=R_i^{L1}} (R_i^{L2} - R_i^{L1})]$ .

$R_i^{t+1}, \dots, R_i^L$ , where  $L$  is the final clock index,  $C_i^0, \dots, C_i^L$  is the corresponding cumulative clinch of  $u_i$  from clock 0 to clock  $L$ , and  $a^0, \dots, a^L$  is the corresponding value of  $a$  at each clock. Let  $U_i[\Gamma(t, L)]$  be the utility of  $u_i$  in this scenario. Let  $\Gamma(-1, L) = \{L, R_i^0, \dots, R_i^L; C_i^0, \dots, C_i^L; a^0, \dots, a^L\}$  and  $\Gamma(L, L) = \{L, \tilde{R}_i^0, \dots, \tilde{R}_i^L; C_i^0, \dots, C_i^L; a^0, \dots, a^L\}$  be two special cases of  $\Gamma(t, L)$ .

*Lemma 2:* If all other users report their true optimal demands at every clock, then  $U_i[\Gamma(t-1, L_1)] \geq U_i[\Gamma(t, L_2)]$ .

*Proof:* From (3.26), we have,

$$\begin{aligned}
 U_i[\Gamma(t-1, L_1)] &= \ln(\gamma_i + \beta_i R_i^{L1}) - a^0 C_i^0 - \sum_{k=1}^{L_1} a^k (C_i^k - C_i^{k-1}), \\
 U_i[\Gamma(t, L_2)] &= \ln(\gamma_i + \beta_i R_i^{L2}) - a^0 C_i^0 - \sum_{k=1}^{L_2} a^k (C_i^k - C_i^{k-1}). \quad (3.27)
 \end{aligned}$$

- If  $\tilde{R}_i^t \leq R_i^t$ , according to Algorithm 1, we have  $L_2 \leq L_1$  and  $R_i^{L2} \leq R_i^{L1}$ .

Then,

$$\begin{aligned}
& U_i[\Gamma(t-1, L_1)] - U_i[\Gamma(t, L_2)] \\
&= \ln(\gamma_i + \beta_i R_i^{L_1}) - \ln(\gamma_i + \beta_i R_i^{L_2}) - \sum_{k=L_2+1}^{L_1} a^k (C_i^k - C_i^{k-1}) \\
&\geq \ln(\gamma_i + \beta_i R_i^{L_1}) - \ln(\gamma_i + \beta_i R_i^{L_2}) - a^{L_1} (C_i^{L_1} - C_i^{L_2}). \tag{3.28}
\end{aligned}$$

When  $\delta$  is sufficiently small,  $C_i^{L_1} = R_i^{L_1}$  and  $C_i^{L_2} = R_i^{L_2}$ . Since  $R_i^t = \max[R_i^{min}, \min(\frac{1}{a^t} - \frac{\gamma_i}{\beta_i}, R_i^{max})]$ ,

– if  $R_i^{L_1} > R_i^{min}$ , according to (3.12), we get  $a^{L_1} \leq \frac{\partial \ln(\gamma_i + \beta_i R_i)}{\partial R_i} \Big|_{R_i=R_i^{L_1}}$ .

Thus, (3.28) becomes

$$\begin{aligned}
& U_i[\Gamma(t-1, L_1)] - U_i[\Gamma(t, L_2)] \\
&\geq [\ln(\gamma_i + \beta_i R_i^{L_1}) - \frac{\partial \ln(\gamma_i + \beta_i R_i)}{\partial R_i} \Big|_{R_i=R_i^{L_1}} (R_i^{L_1} - R_i^{L_2})] - \ln(\gamma_i + \beta_i R_i^{L_2}) \\
&\geq 0 \text{ (see Figure 3.3(a)).} \tag{3.29}
\end{aligned}$$

– if  $R_i^{L_1} = R_i^{min}$ , since  $R_i^{min} \leq R_i^{L_2} \leq R_i^{L_1} = R_i^{min}$ , we have  $R_i^{L_2} = R_i^{min} = R_i^{L_1}$ . Therefore,

$$U_i[\Gamma(t-1, L_1)] = U_i[\Gamma(t, L_2)]. \tag{3.30}$$

So, if  $\tilde{R}_i^t \leq R_i^t$ , we have  $U_i[\Gamma(t-1, L_1)] \geq U_i[\Gamma(t, L_2)]$ .

- If  $\tilde{R}_i^t > R_i^t$ , according to Algorithm 1, we have  $L_2 \geq L_1$  and  $R_i^{L_2} \geq R_i^{L_1}$ .

Then,

$$\begin{aligned}
& U_i[\Gamma(t-1, L_1)] - U_i[\Gamma(t, L_2)] \\
&= \ln(\gamma_i + \beta_i R_i^{L_1}) - \ln(\gamma_i + \beta_i R_i^{L_2}) + \sum_{k=L_1+1}^{L_2} a^k (C_i^k - C_i^{k-1}) \\
&\geq \ln(\gamma_i + \beta_i R_i^{L_1}) - \ln(\gamma_i + \beta_i R_i^{L_2}) + a^{L_1} (C_i^{L_2} - C_i^{L_1}). \tag{3.31}
\end{aligned}$$

When  $\delta$  is sufficiently small,  $C_i^{L_1} = R_i^{L_1}$  and  $C_i^{L_2} = R_i^{L_2}$ . Since  $R_i^t = \max[R_i^{min}, \min(\frac{1}{a^t} - \frac{\gamma_i}{\beta_i}, R_i^{max})]$ ,

– if  $R_i^{L_1} < R_i^{max}$ , according to (3.12),  $a^{L_1} \geq \frac{\partial \ln(\gamma_i + \beta_i R_i)}{\partial R_i} \Big|_{R_i=R_i^{L_1}}$ . Thus,

(3.31) becomes

$$\begin{aligned} & U_i[\Gamma(t-1, L_1)] - U_i[\Gamma(t, L_2)] \\ & \geq \ln(\gamma_i + \beta_i R_i^{L_1}) - [\ln(\gamma_i + \beta_i R_i^{L_2}) - \frac{\partial \ln(\gamma_i + \beta_i R_i)}{\partial R_i} \Big|_{R_i=R_i^{L_1}} (R_i^{L_2} - R_i^{L_1})] \\ & \geq 0 \text{ (see Figure 3.3(b)).} \end{aligned} \quad (3.32)$$

– if  $R_i^{L_1} = R_i^{max}$ , since  $R_i^{max} \geq R_i^{L_2} \geq R_i^{L_1} = R_i^{max}$ , we have  $R_i^{L_2} = R_i^{max} = R_i^{L_1}$ . Therefore,

$$U_i[\Gamma(t-1, L_1)] = U_i[\Gamma(t, L_2)]. \quad (3.33)$$

So, if  $\tilde{R}_i^t > R_i^t$ , we still have  $U_i[\Gamma(t-1, L_1)] \geq U_i[\Gamma(t, L_2)]$ .

In all, we can show that  $U_i[\Gamma(t-1, L_1)] \geq U_i[\Gamma(t, L_2)]$ . This completes the proof.

With Lemma 2, we can now show that the best strategy of each user is to report his/her true optimal demand at every clock.

*Theorem 2 (Cheat-Proof):* Reporting true optimal demand at every clock is a mutually best response for every user, i.e.,  $U_i[\Gamma(L_3, L_3)] \leq U_i[\Gamma(-1, L_4)] \forall i$ .

*Proof:* If all the other users report their true optimal demands in every clock, according to Lemma 2, we have  $U_i[\Gamma(L_3, L_3)] \leq U_i[\Gamma(L_3 - 1, \tilde{L}_3)] \leq \dots \leq U_i[\Gamma(-1, L_4)]$ , where  $\tilde{L}_3$  stands for the final clock index of the following scenario: from clock 0 to clock  $L_3 - 1$ ,  $u_i$  reports  $\tilde{R}_i^0, \dots, \tilde{R}_i^{L_3-1}$ , and from clock  $L_3$  to the final clock  $\tilde{L}_3$ ,  $u_i$  reports  $R_i^{L_3}, \dots, R_i^{\tilde{L}_3}$ . Since all users are non-collaborative, reporting true

optimal demand at every clock is a mutually best response for every user. There is no incentive for the users to cheat since any cheating will lead to a loss in utility. Therefore, the proposed scheme is cheat-proof. This completes the proof.

In the above theorem, we give a theoretical proof for the cheat-proof strategy. In the following section, we will verify this cheat-proof strategy through experimental results.

### 3.5 Experimental Results

In order to evaluate the proposed game-theoretic multi-user rate allocation game, we conduct experiments on real video data. Seven video sequences: Akiyo, Mobile, Table, Carphone, Coastguard, Foreman, and Football in QCIF format, are tested. Notice that these video sequences include slow, medium or fast motion, and smooth or complex scene. We use the state-of-art H.264 JM 9.0 video codec to encode the video sequences [6]. By changing the quantization parameter (QP) or using the rate control feature, we are able to compress the video sequences at different bit-rate and achieve different quality requirements.

#### 3.5.1 Parameter Estimation

From Section 3.1, we can see that there are several parameters in our framework,  $\gamma_i$ ,  $\beta_i$ ,  $R_i^{min}$ , and  $R_i^{max}$ . In this subsection, we will discuss how to estimate these parameters.

Table 3.1:  $\gamma_i^*$ ,  $\beta_i^*$ ,  $R_i^{min}(kb/s)$ , and  $R_i^{max}(kb/s)$  for different sequence by training.

Sequence	$\gamma_i^*$	$\beta_i^*$	$R_i^{min}(kb/s)$	$R_i^{max}(kb/s)$
Akiyo	6.8449	0.0416	1.5119	84.5447
Carphone	6.6759	0.0114	20.2554	322.0153
Coastguard	6.6796	0.0043	28.4987	878.8011
Foreman	6.7418	0.0093	17.8168	388.7091
Football	6.2201	0.0024	286.311	1720
Mobile	6.3464	0.0025	225.0682	1610
Table	6.8135	0.0074	12.7781	481.1014

According to (3.3) and (3.6), we have:

$$\begin{aligned}
 PSNR_i &= 10(\log_{10} e) \ln \frac{255^2}{D_i} \\
 &= (10 \log_{10} e)(\gamma_i + \beta_i R_i)
 \end{aligned} \tag{3.34}$$

Therefore, we can estimate  $\gamma_i$  and  $\beta_i$  using off-line training. For each video sequence, we first generate a set of  $(PSNR_i, R_i)$  by encoding the sequence using H.264 JM 9.0 with different QP. Then, the optimal  $\gamma_i^*$  and  $\beta_i^*$  can be computed by:

$$(\gamma_i^*, \beta_i^*) = \min_{\gamma_i, \beta_i} \sum_j [PSNR_i(j) - (10 \log_{10} e)(\gamma_i + \beta_i R_i(j))]^2, \tag{3.35}$$

where  $j$  is the index of the training set.

Through the training data and equation above, we get the optimal  $\gamma_i^*$  and  $\beta_i^*$  for different video sequences and show them in Table 3.1. As shown in Figure 3.4,

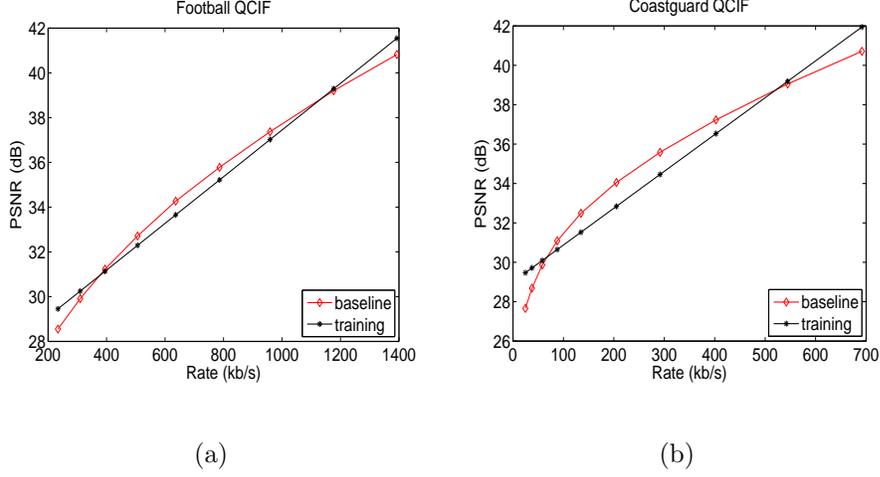


Figure 3.4: Training  $\gamma$  and  $\beta$ : (a) Football; (b) Coastguard.

with the optimal  $\gamma_i^*$  and  $\beta_i^*$ , the  $(10 \log_{10} e)(\gamma_i + \beta_i R_i)$  can approximate  $PSNR_i$  well. Due to the page limitation, we only show the results for Football and Coastguard. Similar results are observed for other sequences.

After finding the optimal  $\gamma_i^*$  and  $\beta_i^*$ , we derive the values for  $R_i^{min}$  and  $R_i^{max}$ . Suppose that the minimal desired PSNR (quality) constraint is  $PSNR^{min}$ , e.g., 30dB, and the maximal satisfied PSNR (quality) constraint is  $PSNR^{max}$ , e.g., 45dB. According to (3.34), we have:

$$\begin{aligned}
 R_i^{min} &= \frac{1}{\beta_i^*} \left( \frac{PSNR^{min}}{10 \log_{10} e} - \gamma_i^* \right), \\
 R_i^{max} &= \frac{1}{\beta_i^*} \left( \frac{PSNR^{max}}{10 \log_{10} e} - \gamma_i^* \right).
 \end{aligned} \tag{3.36}$$

According the equations above, the  $R_i^{min}$  and  $R_i^{max}$  for different sequences are obtained and shown in Table 3.1. From Table 3.1, we can see that the tested video sequences can be classified to four categories according to  $\beta^*$ ,  $R_i^{min}$  and  $R_i^{max}$ : slow motion and smooth scene (Akiyo), medium motion and smooth scene (Carphone, Foreman, and Table), medium motion and complex scene (Coastguard), and fast or

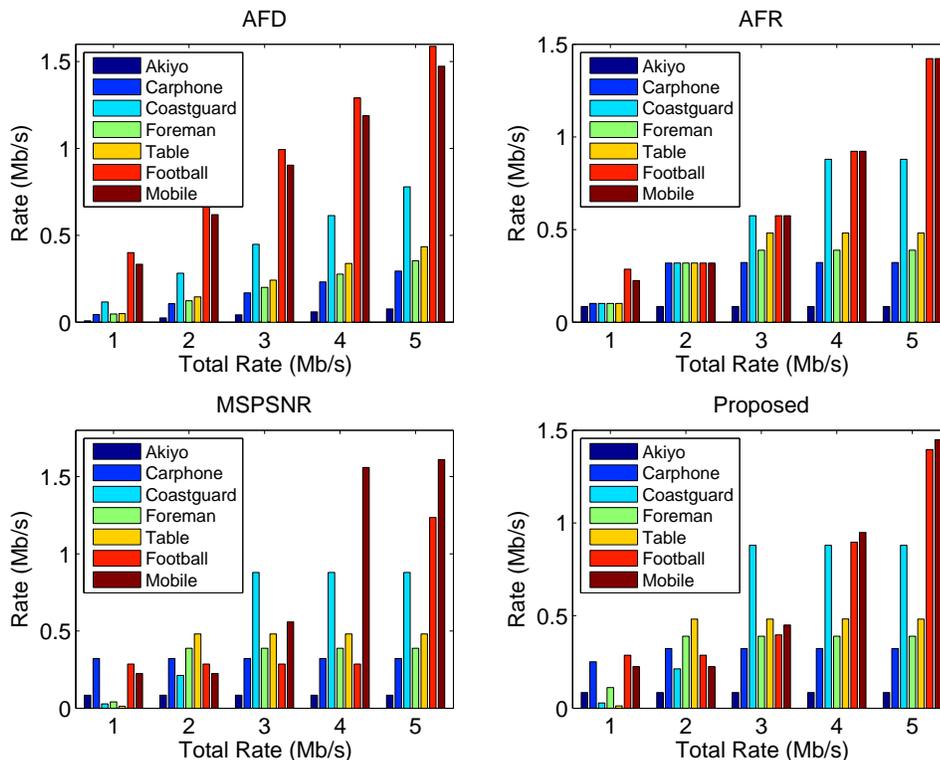


Figure 3.5: Allocated rates for Akiyo, Carphone, Coastguard, Foreman, Table, Football, and Mobile using different methods.

complex motion (Football and Mobile).

### 3.5.2 Multi-User Rate Allocation

We compare the proposed method with three approaches: the Absolute Fairness in Rate (**AFR**), which equally divides the available bandwidth to all the users, the Absolute Fairness in Distortion (**AFD**), which minimizes the maximal distortion of all the users, i.e., min-max fairness, and the approach Maximizing the Sum of the PSNRs (**MSPSNR**), i.e. the traditional optimization-based approach shown in (3.2) with uniform weights. Notice that for **AFR**, **AFD**, and **MSPSNR**, the

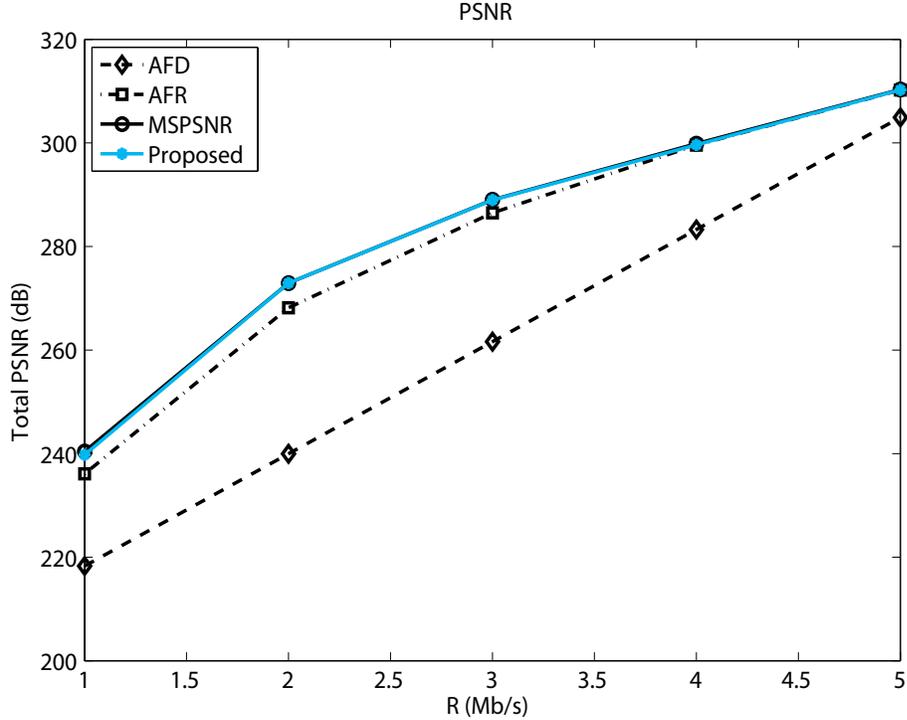


Figure 3.6: The sum of PSNR vs. the available network bandwidth  $R$ .

allocated rate should be within  $[R_i^{min}, R_i^{max}]$ . Otherwise, we set it to be  $R_i^{min}$  or  $R_i^{max}$  and re-allocate the rest rate for the other users. Given the video sequences to be transmitted, the available bandwidth  $R$ , we can compute the rate allocated to each video sequence using different methods, i.e., **AFD**, **AFR**, **MSPSNR**, and the proposed method. Then, setting the allocated bit-rate as the target bit-rate, we compress the video sequence using the rate control feature in H.264 JM 9.0 reference software. Finally, each user transmits the compressed bitstream to the corresponding receiver.

In the experiments, we assume that there are seven users  $u_1, u_2, \dots, u_7$ . They transmit Akiyo, Carphone, Coastguard, Foreman, Table, Football, and Mobile to seven receivers  $r_1, r_2, \dots, r_7$ , respectively. We test  $R$  at 1000, 2000, 3000, 4000 and

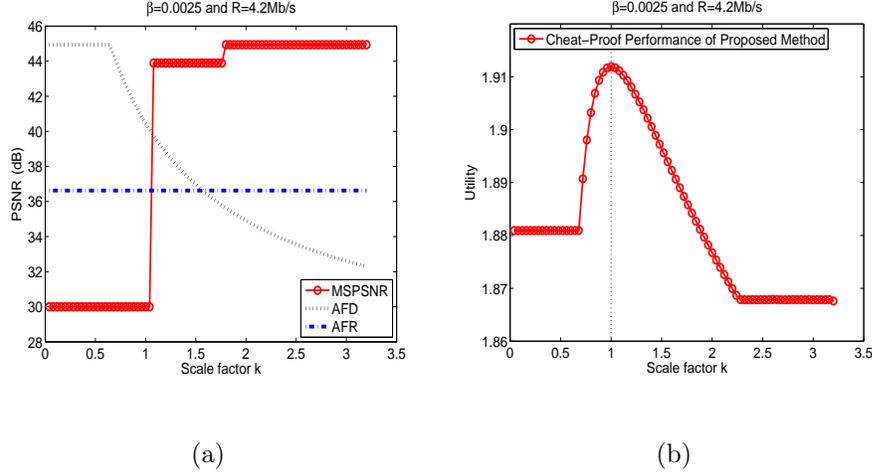


Figure 3.7: Cheat-proof performance: (a) AFD, AFR, and MSPSNR; (b) Proposed Method.

5000 kb/s. The allocated bit-rate for each video sequence in different situations (i.e., different  $R$ ) using different methods (i.e., **AFD**, **AFR**, **MSPSNR**, and the proposed method) are shown in Figure 3.5. From Figure 3.5, we can see that **AFR** equally allocates the bandwidth to each users if the allocated bit-rate is within  $[R_i^{min}, R_i^{max}]$ . **AFD** tries to allocate more bit-rate to the video sequence that has more complex motion and/or scene (a smaller  $\beta^*$ ) to preserve constant quality among different users. On the contrary, **MSPSNR** favors the video sequence that has a larger  $\beta^*$  since allocating more bit-rate to the sequence with a larger  $\beta^*$  leads to a greater increase in the sum of the PSNRs. However, with **MSPSNR**, the sequence with  $\beta_i^*$  will not be allocated more bit-rate than  $R_i^{min}$  if there is a sequence with  $\beta_j^* > \beta_i^*$  who has not been allocated its maximal rate requirement  $R_j^{max}$  yet. Specifically, the rate controller will first allocate each user with  $R_i^{min}$ . Then, the remaining rates will be first allocated to Akiyo until the bit-rate of Akiyo achieves its maximal requirement. If there are still some unused rates, then Carphone will

be satisfied first. And the bit-rate of Football with the smallest  $\beta^*$  stays at its minimal requirement until all other sequences with higher  $\beta^*$  have achieved their maximal rate requirements. Obviously, this is not fair to the users who transmit the sequences with smaller  $\beta^*$ . By taking the proportional fairness into account, the proposed method can avoid this disadvantage and balance the rate allocation between the sequences with a larger  $\beta^*$  and a smaller  $\beta^*$ . For example, as shown in Figure 3.5, when the total available network bandwidth  $R$  increases from 3000kb/s to 4000kb/s, both the bit-rate of Mobile and Football increase. This is because the proposed method with the proportional fairness criterion aims at maximizing the product of the utility function  $U_i$ , and keeping a certain balance between the sequences with a larger  $\beta^*$  and a smaller  $\beta^*$  leads to an increase in the product.

Let  $T\_PSNR = \sum_{i=1}^N PSNR_i$  be the sum of the analytical  $PSNR_i$  computed by (3.34) of all the users. In Figure 3.6, we show  $T\_PSNR$  versus the available network bandwidth  $R$ . We can see that there is a big gap between the performance of **AFD**, **AFR** and **MSPSNR**, which means using **AFD** or **AFR** leads to a big loss in the system performance. However, the performance of the proposed method is almost the same as that of **MSPSNR**, which fully demonstrates the efficiency of the proposed method. Therefore, while achieving a fair rate allocation among different users, the proposed method still performs well in terms of total PSNR.

Finally, we evaluate the cheat-proof property of different methods. As shown in Table 3.1, since  $\beta$  is the most important parameter representing the characteristics of video sequences, the best way for  $u_i$  to pretend as another user  $u_j$  is to use  $\beta_j$  rather than  $\beta_i$  in calculating optimal demand. Therefore, we evaluate the cheat-

proof property in terms of  $\beta$ . In this experiment, the available network bandwidth  $R$  is set to be 4.2Mb/s. We assume that  $u_6$  who transmits Mobile sequence will cheat while other users are honest. In **AFD**, **AFR** and **MSPSNR**,  $u_6$  reports a false  $\tilde{\beta}$  to the controller by scaling the original  $\beta$  with a factor  $k$ , i.e.,  $\tilde{\beta} = k\beta$ . In the proposed method, at each clock  $t$  of the auction,  $u_6$  uses  $\tilde{\beta}$  to generate the “optimal” demand  $\tilde{R}_6^t$  using  $\tilde{R}_6^t = \max[R_6^{min}, \min(\frac{1}{a^t} - \frac{\gamma_6}{k\beta_6}, R_6^{max})]$  and reports  $\tilde{R}_6^t$  to the controller. As shown in Figure 3.7(a), the PSNR performance of **AFR** is independent of the scale factor  $k$ . This is because **AFR** does not care about  $\beta$  and just equally allocates the bandwidth to each user if the allocated bit-rate is within  $[R_i^{min}, R_i^{max}]$ . The PSNR performance of **AFD** decreases as  $k$  increases. This is because **AFD** tries to allocate more bit-rate to the video sequence with a smaller  $\beta$  to preserve constant quality among different users. Therefore, with **AFD**, all users tend to report a smaller  $\beta$  to the controller to obtain a better PSNR performance. On the contrary, the PSNR performance of **MSPSNR** is an increasing piecewise constant function in terms of  $k$ . This is because, with **MSPSNR**, the sequence with  $\beta_i$  will not be allocated more bit-rate than  $R_i^{min}$  if there is a sequence with  $\beta_j > \beta_i$  who has not been allocated its maximal rate requirement  $R_j^{max}$  yet. To be allocated more rate and obtain a higher PSNR,  $u_6$  should increase  $k$  until at least  $k\beta_6 > \beta_j$  where  $\beta_j = \min_l(\beta_l > \beta_6)$ . Therefore, with **MSPSNR**, all users tend to report a larger  $\beta$  to the controller to obtain a better PSNR performance. However, with the proposed method, as shown in Figure 3.7(b), reporting the optimal demand generated by the true  $\beta$  ( $k=1$ ) will lead to the best utility. Any deviation will lead to a loss in terms of utility, which means that the proposed method is cheat-proof.

Therefore, the proposed method ensures all users will be honest about their private information.

### 3.6 Summary

In this chapter, we proposed a game-theoretic framework for multi-user multimedia rate allocation and a distributed cheat-proof scheme for users to converge to the NE of the game. Different from the traditional optimization-based approaches, which mainly focus on the efficiency issue, e.g. maximizing the system performance, the proposed method not only considers the efficiency issue but also the fairness issue. From the experimental results on real video sequences, we can see that with the proportional fairness criterion, the proposed game-theoretic method can efficiently and fairly allocate bit-rate to different users by allocating more bit-rate to the sequence with slower motion and/or simpler scene while keeping an eye on the fast motion and/or complex scene sequence. We also find that, with the proposed distributed cheat-proof rate allocation scheme, reporting the true optimal demand at every clock is the mutual best response for every user. Moreover, we show that the traditional optimization-based method that maximizes the weighted sum of the PSNRs is a special case of the game-theoretic framework with the utility function defined as an exponential function of PSNR.

## Chapter 4

### Peer-to-Peer Cooperative Video Streaming Social Networks

With the rapid development of signal processing, communication, and networking technologies, video-over-IP applications become more and more popular and have attracted millions of users over the Internet [1] [9]. One simple solution to video streaming over Internet is the client-server service model [47] [76], where the video is streamed directly from a server to clients. However, with the client-server service model, the upload bandwidth of the server grows proportionally with the number of clients [86], which makes the large-scale video streaming impractical.

To reduce the workload of the server, Peer-to-Peer (P2P) service model is proposed [36] [133], where a peer not only acts as a client to download data from the network, but also acts as a server to upload data for the other peers in the network. The upload bandwidth of the peers reduces the workload placed on the server dramatically, which makes large-scale video streaming possible. Recently, several industrial large-scale P2P video streaming systems have been developed, including Coolstreaming [133], PPLive [4], PPStream [5], UUSee [8] and Sopcast [7]. Studies show that these systems can support hundreds of thousands of users simultaneously [65].

While P2P video streaming systems have achieved promising results, they have several drawbacks. First, there is a large number of unnecessary traverse links within

a provider's network. As observed in [129], each P2P bit on the Verizon network traverses 1000 miles and takes 5.5 metro-hops on average. Second, there is a huge number of cross Internet Service Provider (ISP) traffic. The studies in [71] [107] showed that 50%-90% of the existing local pieces in active peers are downloaded externally. Third, the differences in playback time among peers can be as high as 140 seconds [65], and the lag can be greater if the source rate is higher. Fourth, most of the current P2P systems assume that all peers are willing to contribute their resources. However, this assumption may not be true since the P2P systems are self-organizing networks and the peers are selfish by nature [128] [60]. Note that the selfish peers will act as free-riders if being free-riders can improve their utilities.

In the literature, many approaches have been proposed to overcome these drawbacks. Karagiannis et al. [71] and Madhyastha et al. [87] proposed to use locality-aware P2P schemes to reduce the unnecessary traverse links within and cross ISPs and thus reduce the download time. Purandare and Guha [100] proposed an alliance based peering scheme to reduce the playback time lag and improve the Quality of Service (QoS). Xie et al. [129] proposed a P4P architecture that allows cooperative traffic control between applications and network providers. To stimulate selfish peers to contribute their resources, payment mechanisms [121] [57] and reputation schemes [88] [59] are proposed, where peers pay points to receive data and earn points by forwarding data to others. However, such payment or reputation based mechanisms often demand a centralized architecture and thus hinder their scalability.

Game theory is a mathematical tool that analyzes the strategic interactions

among multiple decision makers. Recently, it draws great attentions in cognitive networking [123], multimedia social networking [136], and is being applied to many multimedia signal processing problems such as video coding [12] and multimedia communications [29]. In P2P networks, peers make intelligent decisions on their strategies of requesting and forwarding packets based on their needs and other peers' actions. Moreover, since peers are rational and thus naturally selfish, they have no incentive to contribute their resources for other peers. Therefore, it is natural to study the intelligent behaviors and interactions of selfish peers in P2P networks from a game theoretic perspective [128] [84]. Using a mental cost to describe the level of the peer's altruism, the authors in [128] presented a game theoretical model to analyze nodes' behaviors and the influence of incentive mechanism. In [84], a game theoretic framework is proposed for designing distributed, cheat-proof and attack-resistant cooperation stimulation strategies for P2P live streaming social networks.

Most of the existing schemes treat every peer as an independent individual. However, in reality, every peer can have a large number of geographically neighboring peers with large intra-group upload and download bandwidths, e.g. the peers in the same lab, building, or campus. Here, we name those geographically neighboring peers with large intra-group upload and download bandwidths as *group peers*. To reduce the unnecessary traverse links and improve network efficiency, instead of considering each peer's strategy independently, we investigate possible cooperation among the *group peers*. Moreover, since peers are naturally selfish, they will act as free-riders if doing so can improve their utilities. In such a case, full cooperation cannot be guaranteed. Instead, to achieve better payoff, rational peers will adjust

their degree of cooperation by learning from their payoff history. Therefore, a key question to answer is: “*how a group of selfish peers should cooperate with each other to achieve better streaming performance?*”

The main contributions of this chapter are summarized as follows.

- We propose a cooperative streaming scheme to enable cooperation among *group peers* to achieve better streaming performance.
- In the proposed scheme, we define the utility function of a peer by taking into account the possibility of real-time streaming and the cost of acting as a server to upload data for the other peers.
- Due to their selfish nature, peers tend to act as free riders to improve their own utilities. Moreover, the peers may take out-of-equilibrium strategies due to the uncertainty of the strategies of the other peers. Therefore, a robust Nash equilibrium (NE) solution is desired for every peer. In this chapter, we formulate the cooperative streaming problem as an evolutionary game and derive the evolutionarily stable strategy (ESS) for every peer, which is the desired stable NE.
- To stimulate cooperation, the cooperative streaming scheme should be simple since peers may not be willing to join the cooperative streaming if the protocol is complicated. The proposed cooperative streaming scheme is very simple. Each peer tosses a coin to decide whether to be an agent or a free rider. If the outcome is head, the peer acts as an agent to download data from the peers outside the group. Otherwise, the peer acts as a free-rider to download data

from the agents. And the probability of being a head for the coin is learned from the peer’s own past payoff history.

- Due to the highly dynamic behaviors of the peers, i.e., the peers may join or leave the P2P network at any time, the cooperative streaming scheme should be distributed. We propose a distributed algorithm for every peer to approach the ESS by learning from the peer’s own past payoff history.

The rest of this chapter is organized as follows. In Section 4.1, we describe the system model and the utility function. Then, we show in details how to select agents in a homogeneous group in Section 4.2. We extend the analysis to the heterogeneous case in Section 4.3. In Section 4.4, we propose a distributed learning algorithm for ESS. Finally, we show the simulation results in Section 4.5 and draw conclusions in Section 4.6.

## 4.1 The System Model and Utility Function

### 4.1.1 System Model

As shown in Figure 4.1, there is a set of *group peers*<sup>1</sup> (three in this example) who want to view a real-time video streaming simultaneously. Within a group, every peer can choose either to be an **agent** or a normal peer. If the peer serves as an agent, he/she not only needs to act as a client to download video data from the

---

<sup>1</sup>How to group the peers itself is an interesting problem. However, in this chapter, we assume that the peers have already been grouped and mainly focus on how the *group peers* cooperate with each other to achieve better streaming performance.

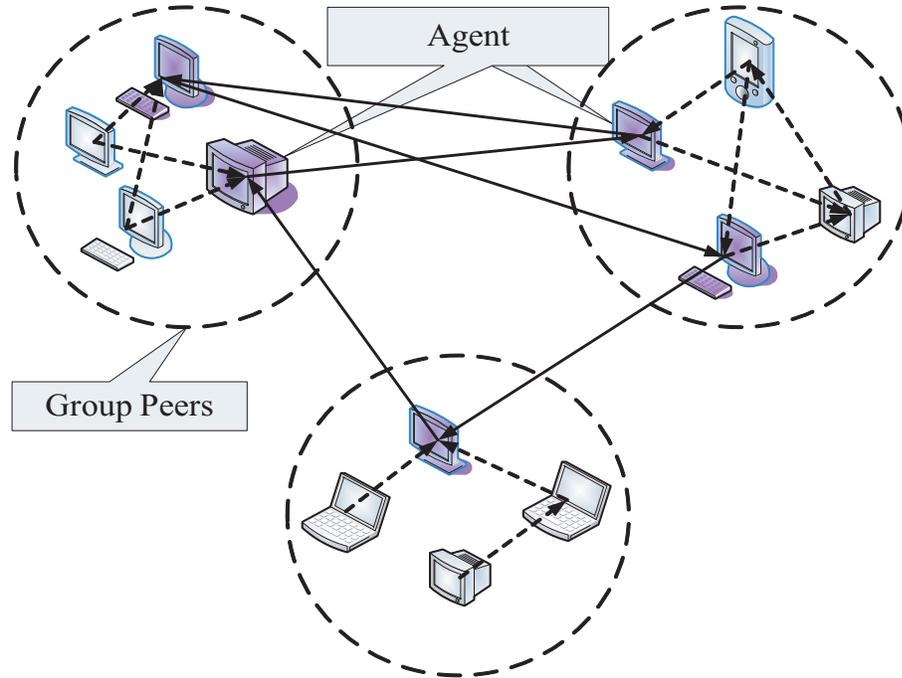


Figure 4.1: A cooperative streaming example.

agents in other groups, but also needs to act as a server to upload video streams for both the agents in other groups and the peers in the same group. However, if the peer chooses not to be an agent, he/she only needs to download/upload data from/to the peers in the same group. We assume that the upload and download bandwidth within the group is larger than those cross groups. In such a case, peers tend to be a normal peer due to the selfish nature. Nevertheless, the normal peers, on the other hand, take a risk of receiving degraded streaming performance since there may not be sufficient agents to download data from other groups. In order to achieve good streaming performance through cooperation, a question need to be addressed: *given a group of peers, which peers should serve as agents.*

### 4.1.2 Utility Functions

In a P2P network, a peer not only acts as a client to download video data from the other peers but also acts as a server to upload video data for the other peers. Therefore, while a peer can benefit from downloading data from the other peers, he/she also incurs a cost in uploading data for the other peers, where the cost can be resource spending on uploading data, e.g. bandwidth, buffer size.

Given the *group peers*,  $u_1, u_2, \dots, u_N$ , we assume that  $k$  peers are willing to serve as agents to download multimedia data from the peers outside the group. Let the download rate be the transmission speed between an agent and a corresponding peer outside the group. If we denote that the download rates of the  $k$  agents are  $r_1, r_2, \dots, r_k$ , then the total download rate of the *group peers* is given by

$$y_k = \sum_{i=1}^k r_i. \quad (4.1)$$

Since the agents randomly and independently select peers outside the group for downloading data, the download rate  $r_i$ 's are random variables. According to [68], the Cumulative Distribution Function (CDF) of a peer's download bandwidth can be modelled as a linear function, which means that the PDF of a peer's download bandwidth can be modelled as a uniform distribution, i.e.,  $r_i$ 's are uniformly distributed.

To provide more insight into the cooperative streaming problem, we first consider a simple scenario without buffering. Then, we extend our discussion to the case when there is buffering effect in Section 4.5. For the scenario without buffering, if the total download rate  $y_k$  is not smaller than the source rate  $r$ , then the group

peers can have a real-time streaming, and all the group peers can obtain a certain gain  $G$ . Otherwise, there will be some delay, and in this case we assume the gain is zero. Therefore, given the total download rate  $y_k$  and the source rate  $r$ , if peer  $u_i$  chooses to be an agent, then the utility function of  $u_i$  is given by

$$U_{A,i}(k) = Pr(y_k \geq r)G - C_i, \forall k \in [1, N], \quad (4.2)$$

where  $C_i$  is the cost of  $u_i$  when he/she serves as an agent, and  $Pr(y_k \geq r)$  is the probability of achieving a real-time streaming which can be computed according to

**Theorem 1.**

Since the upload and download bandwidths within the group is large, the cost of uploading data to the other peers within the group can be negligible. In such a case, if peer  $u_i$  chooses not to be an agent, then there is no cost for  $u_i$  and the utility function becomes

$$U_{N,i}(k) = \begin{cases} Pr(y_k \geq r)G, & \text{if } k \in [1, N - 1]; \\ 0, & \text{if } k = 0. \end{cases} \quad (4.3)$$

**Theorem 1:** If  $r_1, r_2, \dots, r_k$  are i.i.d. uniformly distributed within  $[r^L, r^U]$ , then  $Pr(y_k \geq r)$  is given by

$$Pr(y_k \geq r) = \frac{1}{2k!} \sum_{l=0}^k (-1)^l \binom{k}{l} \left[ (k-l)^k - \text{sgn}(\hat{r} - l)(\hat{r} - l)^k \right], \quad (4.4)$$

and when  $k$  is sufficiently large,  $Pr(y_k \geq r)$  can be approximated as

$$Pr(y_k \geq r) \approx Q \left( \frac{\hat{r} - \frac{k}{2}}{\sqrt{\frac{k}{12}}} \right), \quad (4.5)$$

where  $\hat{r} = \frac{r - kr^L}{r^U - r^L}$  and  $Q(x)$  is the Gaussian tail function  $\int_x^\infty \frac{1}{\sqrt{2\pi}} \exp^{-\frac{x^2}{2}} dx$ .

*Proof:* Let  $\hat{r}_l = \frac{r_l - r^L}{r^U - r^L}$ ,  $\forall l$ , then  $\hat{r}_1, \hat{r}_2, \dots, \hat{r}_k$  are i.i.d. uniformly distributed with  $[0, 1]$ . And the characteristic function of  $\hat{r}_l$  is given by

$$\phi(t) = \frac{i(1 - e^{it})}{t}. \quad (4.6)$$

Let  $\hat{y}_k = \sum_{l=1}^k \hat{r}_l$ , then the characteristic function of  $\hat{y}_k$  can be computed by

$$\phi_{\hat{y}_k}(t) = \left( \frac{i(1 - e^{it})}{t} \right)^k. \quad (4.7)$$

Therefore, the density function of  $\hat{y}_k$  is

$$\begin{aligned} f_{\hat{y}_k}(y) &= \mathcal{F}_t^{-1} \left[ \left( \frac{i(1 - e^{it})}{t} \right)^k \right] (y) \\ &= \frac{1}{2(k-1)!} \sum_{l=0}^k (-1)^l \binom{k}{l} \text{sgn}(y-l)(y-l)^{k-1}. \end{aligned} \quad (4.8)$$

Since  $Pr(y_k \geq r) = Pr(\hat{y}_k \geq \hat{r})$ , according to (4.8), we have

$$\begin{aligned} Pr(y_k \geq r) &= Pr(\hat{y}_k \geq \hat{r}) = \int_{\hat{r}}^{\infty} f_{\hat{y}_k}(y) dy \\ &= \frac{1}{2k!} \sum_{l=0}^k (-1)^l \binom{k}{l} \left[ (k-l)^k - \text{sgn}(\hat{r}-l)(\hat{r}-l)^k \right]. \end{aligned} \quad (4.9)$$

When  $k$  is sufficiently large, according to the Central Limit Theory, the distribution of  $\hat{y}_k$  can be approximated as Gaussian distribution  $N(\frac{k}{2}, \frac{k}{12})$ . Therefore, we have

$$Pr(y_k \geq r) = Pr(\hat{y}_k \geq \hat{r}) \approx Q \left( \frac{\hat{r} - \frac{k}{2}}{\sqrt{\frac{k}{12}}} \right). \quad (4.10)$$

■

## 4.2 Agents Selection Within A Homogeneous Group

In the previous section, we have discussed the system model and the peer's utility function. To optimize the streaming performance, proper peers should serve as agents to download data from the peers outside the group. In this section, we will discuss how to select agents within a homogeneous group where the cost of all peers serving as an agent is assumed to be the same.

### 4.2.1 Centralized Agent Selection

If there is a central controller who can choose which peers should act as agents, then a straightforward criterion of selecting proper agents is to maximize the social welfare, which is the sum of all peers' utilities.

Let  $C_i = C$  be the cost of a peer serving as an agent in a homogeneous group. Then the social welfare of an  $N - peer$  group with  $k$  agents can be calculated by

$$SW(k) = Pr(y_k \geq r)GN - kC. \quad (4.11)$$

Based on (4.11), the agent selection problem to maximize the social welfare can be formulated as

$$\max_k SW(k) = \max_k [Pr(y_k \geq r)GN - kC], \quad (4.12)$$

where  $k \in \{1, 2, \dots, N\}$ .

By solving (4.12), we can find the optimal  $k^*$  that maximizes the social welfare. Then, the central controller can choose  $k^*$  peers from the group as agents to download data from the peers outside the group based on some mechanism, e.g.

the peers take turns to serve as agents. However, since peers' behaviors are highly dynamic, they may join in or leave the P2P network at any time. In such a case, the centralized approach may not be practical.

## 4.2.2 Distributed Agent Selection

To overcome the drawback of the centralized approach, it is possible to consider a distributed approach where each peer acts as an agent with probability  $x$ . Then, according to (4.2) and (4.3), the group's social welfare can be computed by

$$U_{total}(x) = \sum_{i=1}^N \binom{N}{i} x^i (1-x)^{N-i} [Pr(y_i \geq r)GN - iC]. \quad (4.13)$$

The problem of finding an optimal  $x$  to maximize the social welfare can be formulated as

$$\begin{aligned} \max_x \quad & \sum_{i=1}^N \binom{N}{i} x^i (1-x)^{N-i} [Pr(y_i \geq r)GN - iC] \\ \text{s.t.} \quad & 0 \leq x \leq 1. \end{aligned} \quad (4.14)$$

However, since peers are selfish by nature, they are not as cooperative as a system designer/controller desires. By solving (4.14), we can find the optimal  $x^*$  that maximizes the social welfare, but  $x^*$  can not maximize each peer's own utility. Therefore, the social welfare maximizer  $x^*$  is not attainable when peers are selfish. Moreover, the solution to the optimization problem shown in (4.14) is not stable since any perturbation will lead to a new solution.

### 4.2.3 Evolutionary Cooperative Streaming Game

In order to provide a robust equilibrium strategy for the selfish peers, we adopt the concept of Evolutionarily Stable Strategy (ESS) [110] [122], which is defined as follows.

**Definition 1:** A strategy  $a^*$  is an ESS if and only if,  $\forall a \neq a^*$ ,  $a^*$  satisfies

- equilibrium condition:  $U_i(a, a^*) \leq U_i(a^*, a^*)$ , and
- stability condition: if  $U_i(a, a^*) = U_i(a^*, a^*)$ ,  $U_i(a, a) < U_i(a^*, a)$ ,

where  $U_i(a_1, a_2)$  is the utility of player  $i$  when he/she uses strategy  $a_1$  and another player uses strategy  $a_2$ .

Since all peers are selfish, they will cheat if cheating can improve their payoffs, which means that all peers are uncertain of other peers' actions and utilities. In such a case, to improve their utilities, peers will try different strategies in every play and learn from the strategic interactions using the methodology of understanding-by-building. During the process, the percentage of peers using a certain pure strategy may change. Such a population evolution can be modelled by replicator dynamics. Specifically, let  $x_a$  stand for the probability of a peer using pure strategy  $a \in \mathcal{A}$ , where  $\mathcal{A} = \{A, N\}$  is the set of pure strategies including being an agent ( $A$ ) and not being an agent ( $N$ ). By replicator dynamics, the evolution dynamics of  $x_a$  are given by the following differential equation

$$\dot{x}_a = \eta[\bar{U}(a, x_{-a}) - \bar{U}(x_a)]x_a, \quad (4.15)$$

where  $\bar{U}(a, x_{-a})$  is the average payoff of the peers using pure strategy  $a$ ,  $x_{-a}$  is the

set of peers who use pure strategies other than  $a$ ,  $\bar{U}(x_a)$  is the average payoff of all peers, and  $\eta$  is a positive scale factor.

From (4.15), we can see that if adopting pure strategy  $a$  can lead to a higher payoff than the average level, the probability of a peer using  $a$  will grow and the growth rate  $\dot{x}_a/x_a$  is proportional to the difference between the average payoff of using strategy  $a$  (i.e.,  $\bar{U}(a, x_{-a})$ ) and the average payoff of all peers (i.e.,  $\bar{U}(x_a)$ ).

#### 4.2.4 Analysis of the Cooperative Streaming Game

According to (4.2) and (4.3), the average payoff of a peer if he/she choose to be an agent can be computed by

$$\bar{U}_A(x) = \sum_{i=0}^{N-1} \binom{N-1}{i} x^i (1-x)^{N-1-i} [Pr(y_{i+1} \geq r)G - C], \quad (4.16)$$

where  $x$  is the probability of a peer being an agent, and  $\binom{N-1}{i} x^i (1-x)^{N-1-i}$  is the probability that there are  $i$  agents out of  $N-1$  other peers.

Similarly, the average payoff of a peer if he/she chooses not to be an agent is given by

$$\bar{U}_N(x) = \sum_{i=1}^{N-1} \binom{N-1}{i} x^i (1-x)^{N-1-i} Pr(y_i \geq r)G. \quad (4.17)$$

According to (4.16) and (4.17), the average payoff of a peer is

$$\bar{U}(x) = x\bar{U}_A(x) + (1-x)\bar{U}_N(x). \quad (4.18)$$

Substituting (4.18) back to (4.15), we have

$$\dot{x} = \eta x(1-x)[\bar{U}_A(x) - \bar{U}_N(x)]. \quad (4.19)$$

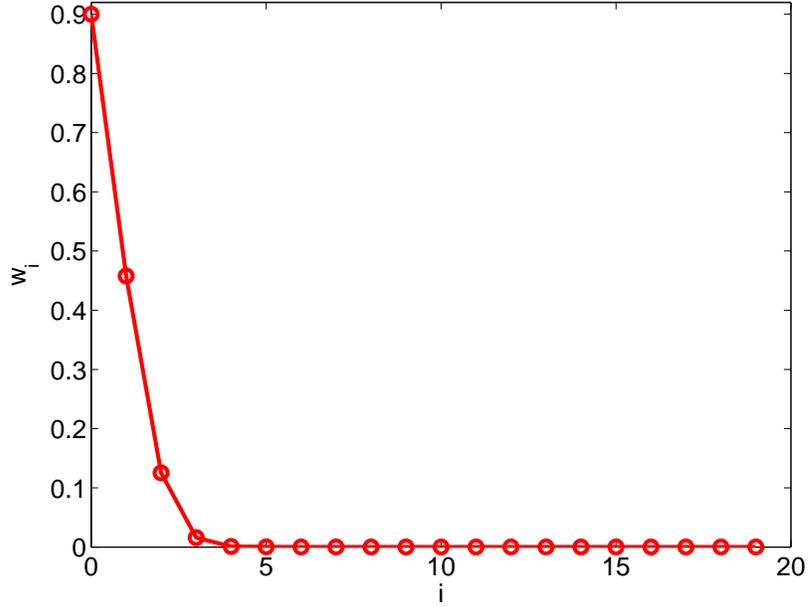


Figure 4.2: The decreasing property of  $w_i$ .

At equilibrium  $x^*$ , no player will deviate from the optimal strategy, which means  $\dot{x}^* = 0$ , and we can obtain  $x^* = 0, 1$ , or the solutions to  $\bar{U}_A(x) = \bar{U}_N(x)$ . However, since  $\dot{x}^* = 0$  is only the necessary condition for  $x^*$  to be ESS, we examine the sufficient condition for each ESS candidate and draw the following conclusions with the proofs shown in **Theorem 2-4**.

- $x^* = 0$  is an ESS only when  $Pr(y_1 \geq r)G - C \leq 0$ .
- $x^* = 1$  is an ESS only when  $Pr(y_N \geq r)G - Pr(y_{N-1} \geq r)G \geq C$ .
- Let  $x^*$  be the solution to  $\bar{U}_A(x) = \bar{U}_N(x)$ , and  $x^* \in (0, 1)$ . Then,  $x^*$  is an ESS.

**Lemma 1:** Let  $f(x) = \bar{U}_A(x) - \bar{U}_N(x)$ , then  $f'(x) < 0, \forall x \in [0, 1]$ . *Proof:*

According to (4.16) and (4.17), we have

$$f(x) = \sum_{i=0}^{N-1} \binom{N-1}{i} x^i (1-x)^{N-1-i} w_i - C, \quad (4.20)$$

where  $w_i = [Pr(y_{i+1} \geq r) - Pr(y_i \geq r)]G$ .

- $\forall x \in (0, 1)$ , by taking the derivative of  $f(x)$  over  $x$ , we have

$$\begin{aligned} f'(x) &= \sum_{i=0}^{N-1} \binom{N-1}{i} x^{i-1} (1-x)^{N-2-i} [i - (N-1)x] w_i, \\ &= \sum_{i=0}^{i_1} \binom{N-1}{i} x^{i-1} (1-x)^{N-2-i} [i - (N-1)x] w_i \\ &\quad + \sum_{i=i_1+1}^{N-1} \binom{N-1}{i} x^{i-1} (1-x)^{N-2-i} [i - (N-1)x] w_i, \end{aligned} \quad (4.21)$$

where  $i_1$  is the integer such that  $i_1 \leq (N-1)x$  and  $i_1 + 1 > (N-1)x$ .

Since  $w_i$  stands for the additional gain by introducing one more agent into the  $i$ -agent system, as shown in Fig. 4.2, it is a decreasing function in terms of  $i$ , which means that  $w_i \geq w_{i_1}, \forall i \leq i_1$  and  $w_i \leq w_{i_1}, \forall i > i_1$ . Therefore,

according to (4.21), we have

$$\begin{aligned}
f'(x) &< \sum_{i=0}^{i_1} \binom{N-1}{i} x^{i-1} (1-x)^{N-2-i} [i - (N-1)x] w_{i_1} \\
&+ \sum_{i=i_1+1}^{N-1} \binom{N-1}{i} x^{i-1} (1-x)^{N-2-i} [i - (N-1)x] w_{i_1}, \\
&= w_{i_1} \sum_{i=0}^{N-1} \binom{N-1}{i} x^{i-1} (1-x)^{N-2-i} [i - (N-1)x], \\
&= w_{i_1} \frac{d \left[ \sum_{i=0}^{N-1} \binom{N-1}{i} x^i (1-x)^{N-1-i} \right]}{dx}, \\
&= 0.
\end{aligned} \tag{4.22}$$

Therefore,  $f'(x) < 0, \forall x \in (0, 1)$ .

- The derivative of  $f(x)$  over  $x$  at  $x = 0$  can be computed by

$$\begin{aligned}
f'(0) &= \lim_{\varepsilon \rightarrow 0} \frac{f(\varepsilon) - f(0)}{\varepsilon} \\
&= \lim_{\varepsilon \rightarrow 0} \frac{\sum_{i=0}^{N-1} \binom{N-1}{i} \varepsilon^i (1-\varepsilon)^{N-1-i} w_i - w_0}{\varepsilon} \\
&= \lim_{\varepsilon \rightarrow 0} \frac{(1-\varepsilon)^{N-1} w_0 - w_0}{\varepsilon} + \lim_{\varepsilon \rightarrow 0} \frac{(N-1)\varepsilon(1-\varepsilon)^{N-2} w_1}{\varepsilon} \\
&= (N-1)(w_1 - w_0) \\
&< 0.
\end{aligned} \tag{4.23}$$

where the last inequality comes from the fact that  $w_i$  is a decreasing function in terms of  $i$ .

- Similarly, the derivative of  $f(x)$  over  $x$  at  $x = 1$  can be computed by

$$\begin{aligned}
f'(1) &= \lim_{\varepsilon \rightarrow 0} \frac{f(1) - f(1 - \varepsilon)}{\varepsilon} \\
&= \lim_{\varepsilon \rightarrow 0} \frac{w_{N-1} - \sum_{i=0}^{N-1} \binom{N-1}{i} (1 - \varepsilon)^i \varepsilon^{N-1-i} w_i}{\varepsilon} \\
&= \lim_{\varepsilon \rightarrow 0} \frac{w_{N-1} - (1 - \varepsilon)^{N-1} w_{N-1}}{\varepsilon} + \lim_{\varepsilon \rightarrow 0} \frac{-(N-1)(1 - \varepsilon)^{N-2} \varepsilon w_{N-2}}{\varepsilon} \\
&= (N-1)(w_{N-1} - w_{N-2}) \\
&< 0.
\end{aligned} \tag{4.24}$$

where the last inequality comes from the fact that  $w_i$  is a decreasing function in terms of  $i$ .

In all,  $f'(x) < 0, \forall x \in [0, 1]$ . This completes the proof of the lemma.  $\blacksquare$

**Theorem 2:** The condition for  $x^* = 0$  to be an ESS is  $Pr(y_1 \geq r)G - C \leq 0$ .

*Proof:* According to (4.16-4.18), the utility that a peer using mixed strategy  $x$  and the other peers use mixed strategy  $x^* = 0$  can be written as

$$\bar{U}(x, 0) = \bar{U}_N(0) + (\bar{U}_A(0) - \bar{U}_N(0))x,$$

where  $\bar{U}_A(0) = Pr(y_1 \geq r)G - C$  and  $\bar{U}_N(0) = 0$ .

- If  $Pr(y_1 \geq r)G - C > 0$ , i.e.  $\bar{U}_A(0) > \bar{U}_N(0)$ , every peer will deviate to  $x = 1$  to obtain  $\bar{U}_A(0)$  rather than  $\bar{U}_N(0)$ .
- If  $Pr(y_1 \geq r)G - C < 0$ , i.e.  $\bar{U}_A(0) < \bar{U}_N(0)$ , every peer will stay at  $x = 0$  to obtain  $\bar{U}_N(0)$  rather than  $\bar{U}_A(0)$ .

- If  $Pr(y_1 \geq r)G - C = 0$ , i.e.  $\bar{U}_A(0) = \bar{U}_N(0)$ , then  $\bar{U}(x, 0) = 0 \forall x$ , and  $f(0) = \bar{U}_A(0) - \bar{U}_N(0) = 0$ . According to **Lemma 1**, we know that  $f'(x) < 0 \forall x \in [0, 1]$ , so  $f(x) = \bar{U}_A(x) - \bar{U}_N(x) < 0$ . In such a case,  $\bar{U}(0, x) = \bar{U}_N(x) > \bar{U}(x, x) = \bar{U}_N(x) + (\bar{U}_A(x) - \bar{U}_N(x))x$ , which means  $x^* = 0$  is an ESS according to **Definition 1**.

Therefore,  $x^* = 0$  is an ESS only when  $Pr(y_1 \geq r)G - C \leq 0$ . ■

**Theorem 3:** The condition for  $x^* = 1$  to be an ESS is  $Pr(y_N \geq r)G - Pr(y_{N-1} \geq r)G \geq C$ .

*Proof:* According to (4.16-4.18), the utility that a peer using mixed strategy  $x$  and the other peers use mixed strategy  $x^* = 1$  can be written as

$$\bar{U}(x, 1) = \bar{U}_N(1) + (\bar{U}_A(1) - \bar{U}_N(1))x,$$

where  $\bar{U}_A(1) = Pr(y_N \geq r)G - C$  and  $\bar{U}_N(1) = Pr(y_{N-1} \geq r)G$ .

- If  $Pr(y_N \geq r)G - Pr(y_{N-1} \geq r)G < C$ , i.e.,  $\bar{U}_N(1) > \bar{U}_A(1)$ , every peer will deviate to  $x = 0$  to obtain  $\bar{U}_N(1)$  rather than  $\bar{U}_A(1)$ .
- If  $Pr(y_N \geq r)G - Pr(y_{N-1} \geq r)G > C$ , i.e.,  $\bar{U}_N(1) < \bar{U}_A(1)$ , every peer will stay at  $x = 1$  to obtain  $\bar{U}_A(1)$  rather than  $\bar{U}_N(1)$ .
- If  $Pr(y_N \geq r)G - Pr(y_{N-1} \geq r)G = C$ , i.e.  $\bar{U}_N(1) = \bar{U}_A(1)$ , then  $\bar{U}(x, 1) = \bar{U}_N(1) \forall x$ , and  $f(1) = \bar{U}_A(1) - \bar{U}_N(1) = 0$ . According to **Lemma 1**, we know that  $f'(x) < 0 \forall x \in [0, 1]$ , so  $f(x) = \bar{U}_A(x) - \bar{U}_N(x) > 0$ . In such a case,  $\bar{U}(1, x) = \bar{U}_N(x) + (\bar{U}_A(x) - \bar{U}_N(x))1 > \bar{U}(x, x) = \bar{U}_N(x) + (\bar{U}_A(x) - \bar{U}_N(x))x$ , which means  $x^* = 1$  is an ESS according to **Definition 1**.

Therefore,  $x^* = 1$  is an ESS only when  $Pr(y_N \geq r)G - Pr(y_{N-1} \geq r)G \geq C$ . ■

**Theorem 4:** If  $x^* \in (0, 1)$  is a solution to  $\bar{U}_A(x) = \bar{U}_N(x)$ , then  $x^*$  is an ESS.

*Proof:* Let  $\bar{U}_i(x, x^*)$  be the utility of player  $i$  when player  $i$  uses mixed strategy  $x$  and other users use mixed strategy  $x^*$ . Then, we have

$$\bar{U}_i(x, x^*) = x\bar{U}_A(x^*) + (1-x)\bar{U}_N(x^*). \quad (4.25)$$

Since  $x^*$  is a solution to  $\bar{U}_A(x) = \bar{U}_N(x)$ , we have  $\bar{U}_A(x^*) = \bar{U}_N(x^*)$ . Therefore, (4.25) becomes

$$\bar{U}_i(x, x^*) = \bar{U}_A(x^*) = \bar{U}_i(x^*, x^*), \quad (4.26)$$

which means  $x^*$  satisfies the equilibrium condition shown in **Definition 1**.

Moreover, according to (4.18), we have

$$\bar{U}_i(x, x) = \bar{U}_N(x) + (\bar{U}_A(x) - \bar{U}_N(x))x, \quad (4.27)$$

and

$$\bar{U}_i(x^*, x) = \bar{U}_N(x) + (\bar{U}_A(x) - \bar{U}_N(x))x^*. \quad (4.28)$$

Therefore, we have

$$\bar{U}_i(x^*, x) - \bar{U}_i(x, x) = (\bar{U}_A(x) - \bar{U}_N(x))(x^* - x). \quad (4.29)$$

From **Lemma 1**, we know that  $f(x) = \bar{U}_A(x) - \bar{U}_N(x)$  is a monotonically decreasing function. Since  $\bar{U}_A(x^*) = \bar{U}_N(x^*)$ ,  $\bar{U}_A(x) - \bar{U}_N(x) > 0$  if  $x < x^*$ , and  $\bar{U}_A(x) - \bar{U}_N(x) < 0$  if  $x > x^*$ . Therefore,  $(\bar{U}_A(x) - \bar{U}_N(x))(x^* - x) > 0, \forall x \neq x^*$ , i.e.

$$\bar{U}_i(x^*, x) > \bar{U}_i(x, x), \forall x \neq x^*, \quad (4.30)$$

which means  $x^*$  satisfies the stability condition shown in **Definition 1**.

According to (4.26) and (4.30), we know that  $x^*$  is an ESS. ■

### 4.3 Agents Selection Within A Heterogeneous Group

In this section, we will discuss how to select agents within a heterogeneous group where the costs of the peers acting as agents are different.

Let  $x_{i,a_i}$  stands for the probability of peer  $u_i$  using pure strategy  $a_i \in \mathcal{A}$ . By replicator dynamics, the evolution dynamics of  $x_{i,a_i}$  are given by the following differential equation

$$\dot{x}_{i,a_i} = \eta[\bar{U}_i(a_i, x_{-i}) - \bar{U}_i(x_i)]x_{i,a_i}, \quad (4.31)$$

where  $\bar{U}_i(a_i, x_{-i})$  is the average payoff of peer  $u_i$  using pure strategy  $a_i$ ,  $\bar{U}_i(x_i)$  is the average payoff of peer  $u_i$  using mixed strategy  $x_i$ , and  $\eta$  is a positive scale factor.

Since it is generally very difficult to represent  $\bar{U}_i(a_i, x_{-i})$  and  $\bar{U}_i(x_i)$  in a compact form, in the following, we first analyze a two-player game to gain some insight. Then, we generalize the observation in the two-player game to the multi-player game.

Table 4.1: Utility table of a two-player game.

	“A”	“N”
“A”	$(B_2 - C_1, B_2 - C_2)$	$(B_1 - C_1, B_1)$
“N”	$(B_1, B_1 - C_2)$	$(0, 0)$

### 4.3.1 Two-Player Game

Let  $x_1$  and  $x_2$  be the probability of  $u_1$  and  $u_2$  being an agent, respectively. Let  $B_1 = Pr(y_1 \geq r)G$  and  $B_2 = Pr(y_2 \geq r)G$ . Then, the payoff matrix of  $u_1$  and  $u_2$  can be written as in Table 4.1. Therefore, the average payoff  $\bar{U}_1(A, x_2)$  can be computed by

$$\bar{U}_1(A, x_2) = (B_2 - C_1)x_2 + (B_1 - C_1)(1 - x_2), \quad (4.32)$$

and the average payoff  $\bar{U}_1(x_1)$  becomes

$$\bar{U}_1(x_1) = (B_2 - C_1)x_1x_2 + (B_1 - C_1)x_1(1 - x_2) + B_1(1 - x_1)x_2. \quad (4.33)$$

According to (4.31), the replicator dynamics equation of  $u_1$  is given by

$$\dot{x}_1 = \eta x_1(1 - x_1) [B_1 - C_1 - (2B_1 - B_2)x_2]. \quad (4.34)$$

Similarly, the replicator dynamics equation of  $u_2$  can be computed by

$$\dot{x}_2 = \eta x_2(1 - x_2) [B_1 - C_2 - (2B_1 - B_2)x_1]. \quad (4.35)$$

At equilibrium, we know that  $\dot{x}_1 = 0$  and  $\dot{x}_2 = 0$ . According to (4.34) and (4.35), we can get five equilibria:  $(0, 0)$ ,  $(0, 1)$ ,  $(1, 0)$ ,  $(1, 1)$ , and the mixed strategy equilibrium  $\left(\frac{B_1 - C_2}{2B_1 - B_2}, \frac{B_1 - C_1}{2B_1 - B_2}\right)$ .

According to [41], if an equilibrium of the replicator dynamics equations is a locally asymptotically stable point in a dynamic system, it is an ESS. Therefore, by viewing (4.34) and (4.35) as a nonlinear dynamic system and analyzing the corresponding Jacobian matrix, we can examine whether the five equilibria are ESSs.

By taking partial derivatives of (4.34) and (4.35), the Jacobian matrix can be written as

$$J = \begin{bmatrix} \frac{\partial \dot{x}_1}{\partial x_1} & \frac{\partial \dot{x}_1}{\partial x_2} \\ \frac{\partial \dot{x}_2}{\partial x_1} & \frac{\partial \dot{x}_2}{\partial x_2} \end{bmatrix} = \eta \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix}, \quad (4.36)$$

where  $J_{11} = (1 - 2x_1)(B_1 - C_1 - (2B_1 - B_2)x_2)$ ,  $J_{12} = -x_1(1 - x_1)(2B_1 - B_2)$ ,  $J_{21} = -x_2(1 - x_2)(2B_1 - B_2)$ , and  $J_{22} = (1 - 2x_2)(B_1 - C_2 - (2B_1 - B_2)x_1)$ .

The asymptotical stability requires that  $\det(J) > 0$  and  $\text{tr}(J) < 0$  [41]. Substituting the five equilibria, i.e.  $(0, 0)$ ,  $(0, 1)$ ,  $(1, 0)$ ,  $(1, 1)$ , and  $\left(\frac{B_1 - C_2}{2B_1 - B_2}, \frac{B_1 - C_1}{2B_1 - B_2}\right)$ , to (4.36), we conclude that

- If  $B_2 - B_1 - C_1 > 0$  and  $B_2 - B_1 - C_2 > 0$ , there is a unique ESS  $(1, 1)$ , where both  $u_1$  and  $u_2$  converge to be agents.
- Elseif  $B_2 - B_1 - C_1 > 0$  and  $B_2 - B_1 - C_2 < 0$ , there is a unique ESS  $(1, 0)$ , where  $u_1$  converges to be an agent and  $u_2$  converges to be a free-rider.
- Elseif  $B_2 - B_1 - C_1 < 0$  and  $B_2 - B_1 - C_2 > 0$ , there is a unique ESS  $(0, 1)$ , where  $u_2$  converges to be an agent and  $u_1$  converges to be a free-rider.
- Else, there are two ESSs  $(0, 1)$  and  $(1, 0)$ , where the converged strategy profiles depends on the initial strategy profiles.

From the above analysis, we can see that when the gain of being an agent  $(B_2 - B_1)$  is greater than the cost of being an agent  $(C_1$  or  $C_2)$ , the peer tends to be an agent. And the peer with a higher cost tends to be a free-rider and rely on the peer with a lower cost.

---

**Algorithm 2 : A Distributed Learning Algorithm For ESS**

---

1. Given the step size  $\eta$  and the slot index  $t = 0$ , each peer  $u_i$  initializes  $x_i$  with  $x_i(0)$ .
  2. During slot  $t$ , for  $q = 1 : M$ ,
    - $u_i$  tosses a coin with probability  $x_i(t)$  being head. If the outcome is head,  $u_i$  serves as an agent and downloads data from the peers outside the group with download rate  $r_i(t, q)$ . On the other hand, if the outcome is tail,  $u_i$  acts as a free-rider and downloads the data from the agents.
    - $u_i$  computes his/her utility using (4.39).
    - $u_i$  computes the indicator function using (4.38).
  3. Then,  $u_i$  approximates  $\bar{U}_i(A, x_{-i}(t))$  and  $\bar{U}_i(x_i(t))$  using (4.40) and (4.41).
  4. Finally,  $u_i$  updates the probability of being an agent  $x_i(t)$  using (4.37).
- 

### 4.3.2 Multi-Player Game

From the analysis of the two-player game, we can infer that the peer with a higher cost ( $C_i$ ) tends to take advantage of the peer with a lower cost. This observation can be extended to multi-player game. If there are more than two peers in the game, the strategy of the peers with higher  $C_i$ 's will converge to “N” with a greater probability. The peers with lower  $C_i$ 's tend to be agents since they suffer relatively heavier losses if no one serves as an agent.

## 4.4 A Distributed Learning Algorithm For ESS

From the previous two sections, we can see that the ESS can be found by solving the replicator dynamics equations ((4.19) or (4.31)). However, solving the replicator dynamics equations require the exchange of private information and strategies adopted by other peers. In this section, we will present a distributed learning algorithm that can gradually converge to ESS without information exchange.

We first discretize the replicator dynamics equation shown in (4.31) as

$$x_i(t+1) = x_i(t) + \eta [\bar{U}_i(A, x_{-i}(t)) - \bar{U}_i(x_i(t))] x_i(t), \quad (4.37)$$

where  $t$  is the slot index and  $x_i(t)$  is the probability of  $u_i$  being an agent during slot  $t$ . Here, we assume that each slot can be further divided into  $M$  subslots and each peer can choose to be an agent or not at the beginning of each subslot.

From (4.37), we can see that in order to update  $x_i(t+1)$ , we need to first compute  $\bar{U}_i(A, x_{-i}(t))$  and  $\bar{U}_i(x_i(t))$ . Let us define an indicator function  $\mathbf{1}_i(t, k)$  as

$$\mathbf{1}_i(t, q) = \begin{cases} 1, & \text{if } u_i \text{ is an agent at subslot } q \text{ in slot } t, \\ 0, & \text{else,} \end{cases} \quad (4.38)$$

where  $q$  is the subslot index.

The immediate utility of  $u_i$  at subslot  $q$  in slot  $t$  can be computed by

$$U_i(t, q) = \begin{cases} G - C_i, & \text{if } u_i \text{ is an agent and } r^t \geq r, \\ -C_i, & \text{if } u_i \text{ is an agent and } r^t < r, \\ G, & \text{if } u_i \text{ is not an agent and } r^t \geq r, \\ 0, & \text{if } u_i \text{ is not an agent and } r^t < r, \end{cases} \quad (4.39)$$

where  $r^t$  is the total download rate of the agents and  $r$  is the source rate.

Then,  $\bar{U}_i(A, x_{-i}(t))$  can be approximated using

$$\bar{U}_i(A, x_{-i}(t)) = \frac{\sum_{q=1}^M U_i(t, q) \mathbf{1}_i(t, q)}{\sum_{q=1}^M \mathbf{1}_i(t, q)}, \quad (4.40)$$

Similarly  $\bar{U}_i(x_i(t))$  can be approximated as

$$\bar{U}_i(x_i(t)) = \frac{1}{M} \sum_{q=1}^M U_i(t, q). \quad (4.41)$$

Based on (4.37-4.41),  $u_i$  can gradually learn the ESS. In Algorithm 2, we summarize the detailed procedures of the proposed distributed learning algorithm.

## 4.5 Simulation Results

In all simulations, the parameters  $G$ ,  $r^L$ , and  $r^U$  are set to be 1, 50, and 800, respectively. For convenience, in the rest of this chapter, we denote the centralized approach maximizing the social welfare shown in (4.12) as **MSW-C**, the distributed approach maximizing the social welfare shown in (4.14) as **MSW-D**, and the ESS-based approach as **ESS-D**. We compare the proposed methods with the traditional P2P non-cooperation method, denoted as **Non-Coop**. In **Non-Coop**, each peer acts as an individual and randomly selects some peers for downloading video streams. Such a protocol has been widely used in the existing P2P systems, e.g., Coolstreaming [133] and PPLive [4].

In the first simulation, we show the social welfare (the sum of all peers' utilities) comparison among different approaches, where we assume that there are 20 homogenous peers and the cost  $C$  is 0.1. As show in Fig. 4.3, **MSW-C** achieves the

best social welfare performance since its objective function is to maximize the social welfare with pure strategy. By using the mixed strategy to maximize the social welfare, **MSW-D** achieves the second best social welfare performance. However, as discussed in Section 4.2.2, the solution to **MSW-D** is not stable. With **ESS-D**, a stable NE solution can be obtained at the cost of a slight loss in social welfare. Nevertheless, all three proposed algorithms perform much better than the **Non-Coop** method. In **Non-Coop**, the social welfare performance decreases linearly in terms of the source rate. With cooperation and adaptively selecting the proper number of agents, all three proposed algorithms can preserve a high social welfare performance even with a large source rate.

In the second simulation, we evaluate the convergence property of the **ESS-D**. In Fig. 4.4, we show the replicator dynamic of the cooperation streaming game with homogeneous peers, where  $C = 0.1$  and  $r = 500$ . We can see that starting from a high initial value, all peers gradually reduce their probabilities of being an agent since being a free-rider more often can bring a higher payoff. However, since too low a probability of being an agent increases the chance of having no peer be an agent, the probability of being an agent will finally converge to a certain value which is determined by the number of peers.

In Fig. 4.5, we show the replicator dynamic of the cooperation streaming game with 20 heterogeneous peers, where  $r = 500$  and the cost  $C_i$  is randomly chosen from  $[0.1, 0.3]$ . We further assume that  $C_i$  is monotonically increasing in  $i$  where  $u_1$  has the lowest cost and  $u_{20}$  has the highest cost. From Fig. 4.5, we can see that the peers with lower costs ( $u_1$ ,  $u_2$ , and  $u_3$  in this simulation) converge to be an agent while

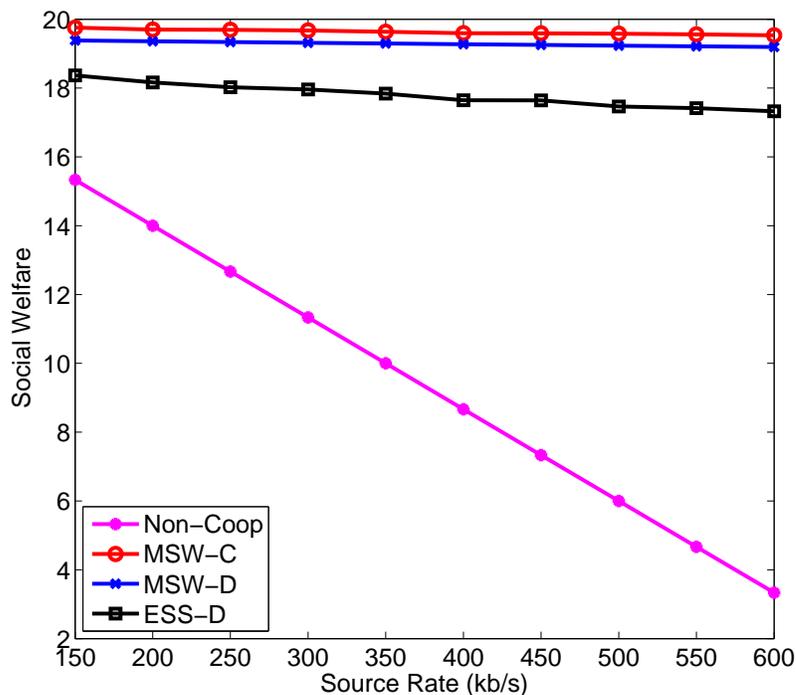


Figure 4.3: The social welfare comparison among **Non-Coop**, **MSW-C**, **MSW-D**, and **ESS-D**.

the peers with higher costs ( $u_4 - u_{20}$  in this simulation) converge to be a free-rider. This observation coincides with our conclusion in Section 4.3.2, which is “*the peers with lower costs tend to be an agent since they suffer relatively higher losses if no one serves as an agent*”. Note that due to the space limitation, we only show the behavior dynamics of  $u_1 - u_4$ . All other peers  $u_5 - u_{20}$  have the similar behavior dynamics with  $u_4$ , and they all converge to be free-riders.

In the third simulation, we compare the performance of **Non-Coop** and **ESS-D** in terms of the probability of real-time streaming, which is defined as the probability that the total download rate is greater than the source rate. The simulation results are shown in Fig. 4.6. We can see that with cooperation, the probability of

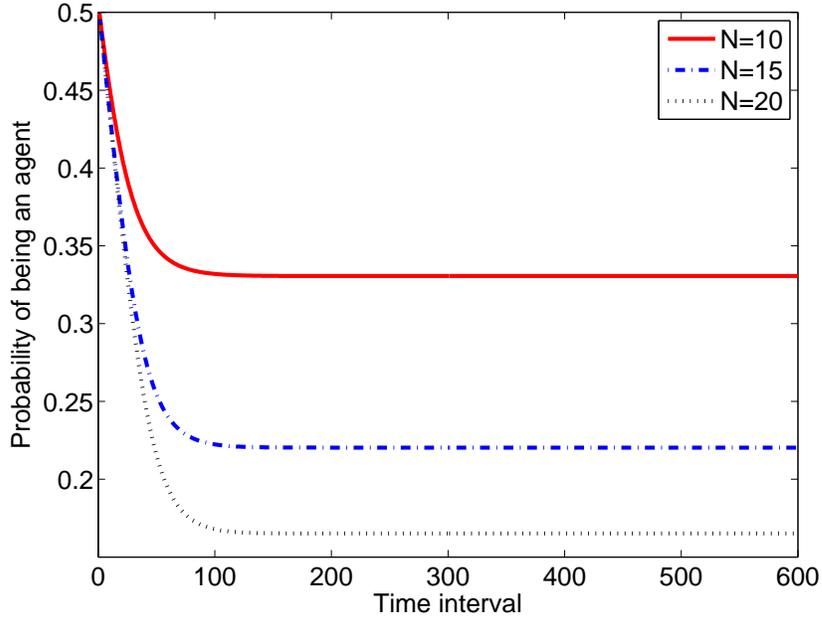


Figure 4.4: Behavior dynamic of a homogeneous group of peers.

real-time streaming can be significantly improved especially at the high source rate region. We also find that at the high source rate region, the probability of real-time streaming increases as  $N$  increases.

The visual quality comparison between **Non-Coop** and **ESS-D** is shown in Fig. 4.7. In this simulation, we fix the probability of real-time streaming to be 0.85. According to Fig. 4.6, we can see that the corresponding source rates for “**Non-Coop**”, “**ESS-D** with  $N=2$ ”, “**ESS-D** with  $N=3$ ”, and “**ESS-D** with  $N=4$ ” are around 100kb/s, 300kb/s, 520kb/s, and 720kb/s, respectively. By setting the above source rates as the target bitrates, we encode the Foreman sequence with CIF format using H.264 encoder. From Fig. 4.7, we can see that the video visual quality with the proposed **ESS-D** is much better than that with **Non-Coop**.

Then, we show the simulation result of the source rate versus the utility. As

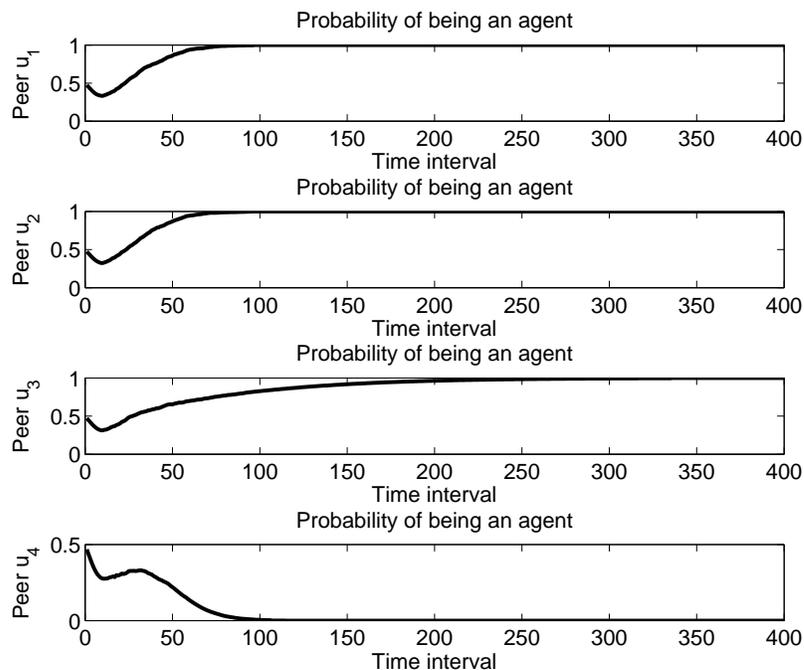


Figure 4.5: Behavior dynamic of a heterogeneous group of peers.

shown in Fig. 4.8, without cooperation, if the peer requires a utility around 0.8, the source rate can not be larger than 130 kb/s. However, with cooperation, the source rate can be more than 400 kb/s even when there are only 2 peers. Therefore, with cooperation, the peers can enjoy much higher quality video with the same utility.

In the fourth simulation, we consider the case that the peers in the same group are viewing multiple channels with  $L$  being the number of the channels. We assume that the source rate is the same for all channels and there are 20 homogenous peers with the cost  $C = 0.1$ . Similar to the View-Upload Decoupling (VUD) scheme [127], the uploading and downloading are decoupled in the proposed **ESS-D** algorithm in this case. We allow cooperation among all the peers where the agent may download

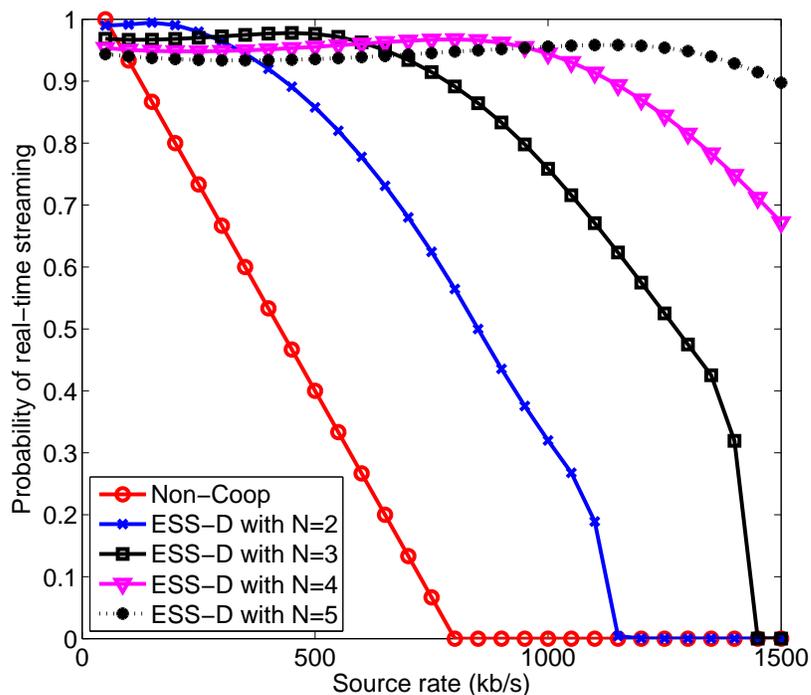


Figure 4.6: The probability of real-time streaming comparison between **Non-Coop** and **ESS-D**.

source data that he/she is not viewing. As shown in Fig. 4.9, without cooperation, if the peer requires a utility around 0.8, the source rate can not be larger than 130 kb/s in the **Non-Coop** method. However, with the proposed **ESS-D** algorithm, the source rate can be around 240kb/s even when the peers are view 8 different channels. This phenomenon fully demonstrates the efficiency of the proposed method.

In the last simulation, we consider the scenario when there is buffering effect. In such a scenario, the gain in the utility will not drop to zero when the total download rate is smaller than the source rate. Instead, the gain should maintain a positive value due to the existence of buffers. One possible utility function that

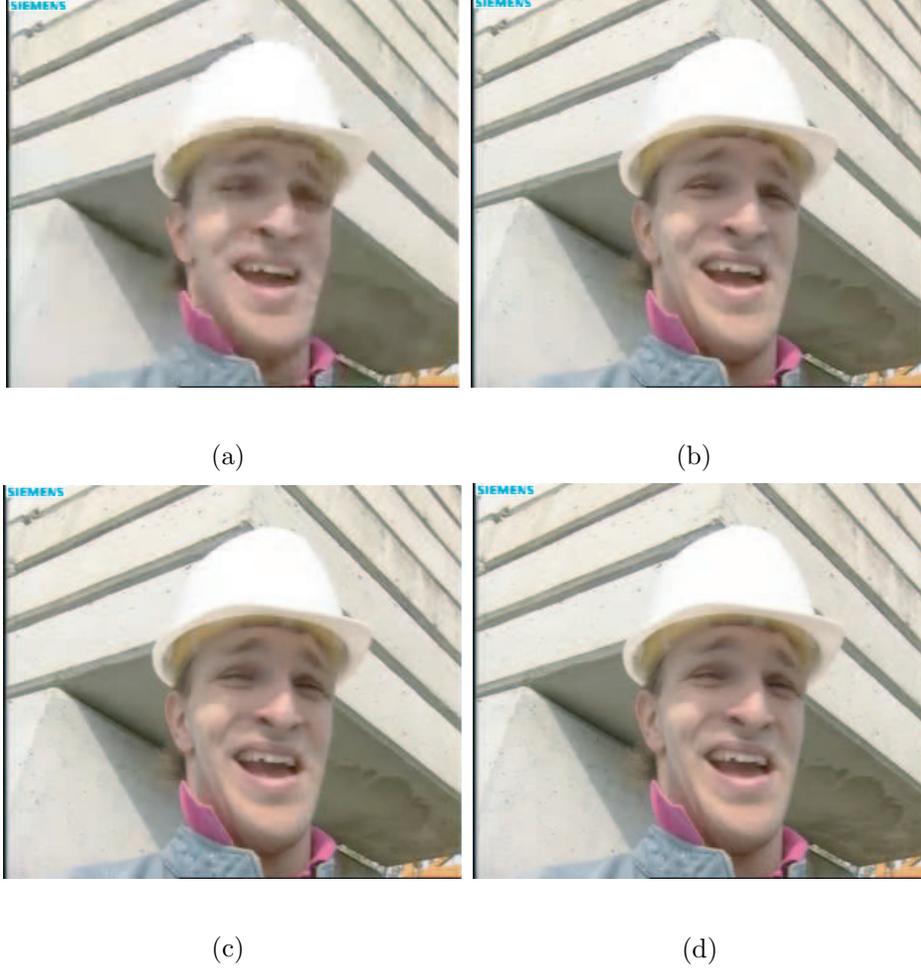


Figure 4.7: The visual quality comparison: (a) **Non-Coop**; (b) **ESS-D** with  $N=2$ ; (c) **ESS-D** with  $N=3$ ; (d) **ESS-D** with  $N=4$ .

considers the buffering effect is

$$\begin{aligned}
 U_{A,i}(k) &= \frac{1}{\ln(r)} E[\ln(y_k)] G - C_i, \forall k \in [1, N], \\
 U_{N,i}(k) &= \begin{cases} \frac{1}{\ln(r)} E[\ln(y_k)] G, & \text{if } k \in [1, N-1]; \\ 0, & \text{if } k = 0. \end{cases} \quad (4.42)
 \end{aligned}$$

From the above utility function, we can see that for any given source rate  $r$ , the gain increases as the total download rate  $y_k$  increases. Moreover, since the probability of playback delay becomes smaller with more data in the buffer, a certain

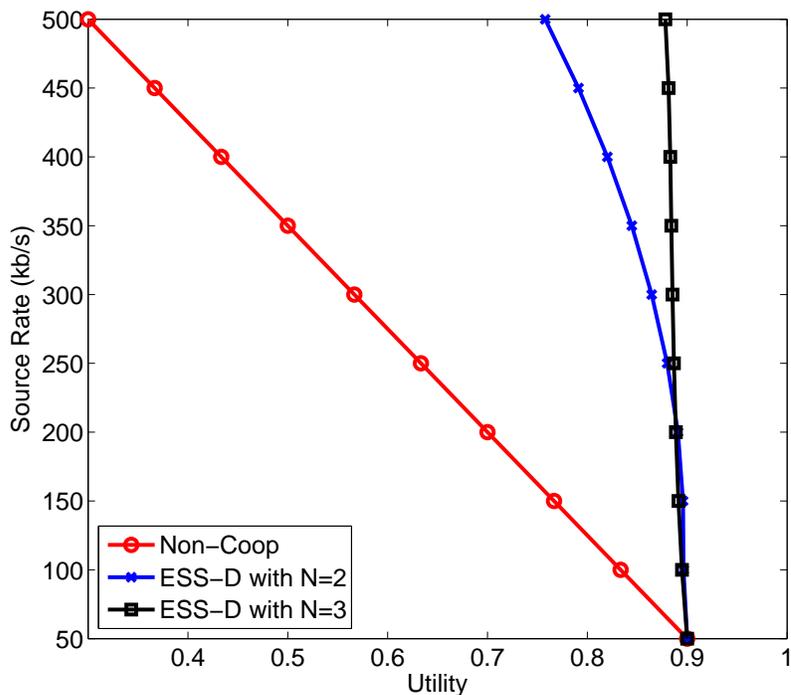


Figure 4.8: Single-source rate comparison between **Non-Coop** and **ESS-D**.

increase in the high  $y_k$  region should lead to a less significant gain than that in the low  $y_k$  region [30]. Here, we use the  $\ln(\cdot)$  function to characterize such properties. Nevertheless, other functions that have similar properties can also be used.

The social welfare comparison between **Non-Coop** and **ESS-D** with the utility function in (4.42) is shown in Fig. 4.10. From Fig. 4.10, we can see that when the utility function in (4.42) is used, the social welfare performance of **Non-Coop** no longer decreases linearly in terms of the source rate. This phenomenon is mainly because, with the existence of buffers, the gain will not drop to zero when the total download rate is smaller than the source rate. Nevertheless, **ESS-D** can still lead to a much higher social welfare performance for all source rates, compared with

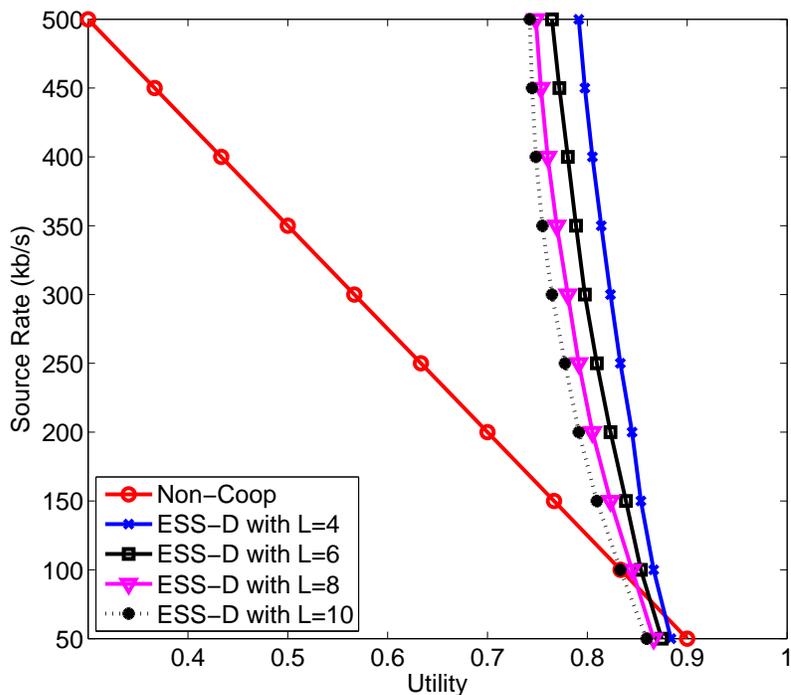


Figure 4.9: Multi-source rate comparison between **Non-Coop** and **ESS-D**.

**Non-Coop**. Moreover, we should notice that all the analysis in Section 4.2 is still applicable to the utility function in (4.42).

## 4.6 Summary

In this chapter, we propose a cooperative streaming scheme to address the network inefficiency problem encountered by the traditional non-cooperative P2P schemes. We answer the question of “*how a group of selfish peers with large intra-group upload and download bandwidths cooperate with each other to achieve better streaming performance*” by formulating the problem as an evolutionary game and deriving the ESS for every peer. We further propose a distributed learning algorithm

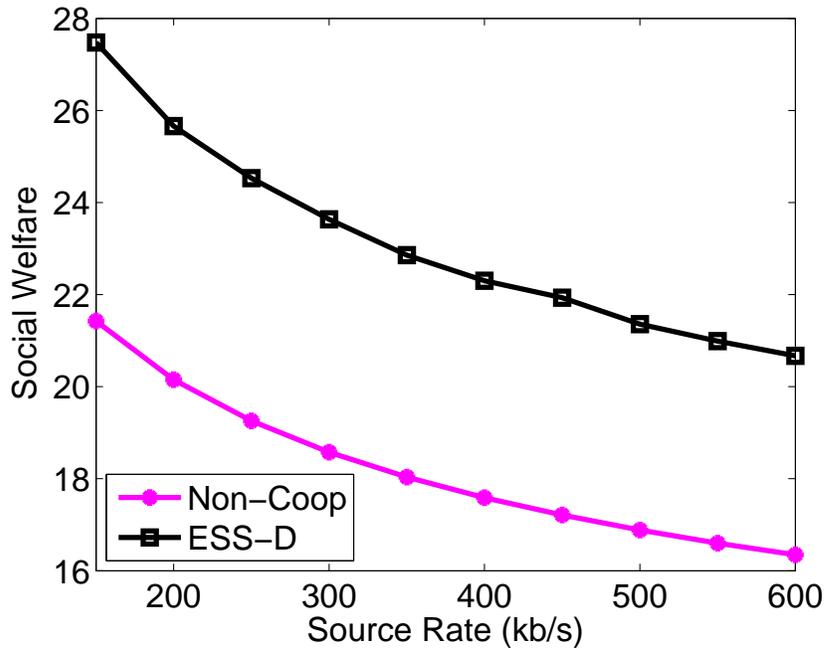


Figure 4.10: The social welfare comparison between **Non-Coop** and **ESS-D** when the utility function is defined as (4.42).

for each peer to converge to the ESS by learning from his/her own past payoff history. From the simulation results, we can see that compared with the traditional non-cooperative P2P schemes, the proposed algorithm achieves much better social welfare, higher probability of real-time streaming, and better video quality (higher source rate). Moreover, by incorporated with the recent proposed View-Upload Decoupling (VUD) scheme, the proposed cooperative streaming scheme allows the peers who are viewing different videos to cooperate with each other and mutually improve the streaming performance.

## Chapter 5

### Cooperation Stimulation Using Indirect Reciprocity Game Modeling

A cognitive network is a network composed of elements that can dynamically adapt to varying network conditions to optimize end-to-end performance through learning and reasoning [116]. In such a network, nodes are intelligent and have the ability to observe, learn, and act to optimize their performance. Since nodes generally belong to different authorities and pursue different goals, fully cooperative behaviors, such as unconditionally forwarding packets for each other, cannot be taken for granted. Instead, nodes will only cooperate with others when cooperation can improve their own performance. We regard the nodes with such behaviors as selfish nodes. Therefore, a key problem in cognitive networks is how to stimulate cooperation among selfish nodes.

In the literature, many schemes have been proposed to stimulate node cooperation for different cognitive networks, such as [23] [138] for ad hoc networks and [121] [57] for peer-to-peer networks. One way to stimulate cooperation among selfish nodes is to use payment based methods [139] [14]. Although these schemes can achieve promising cooperation stimulation results, the requirement of tamper-proof hardware or central billing services greatly limits their potential applications.

Another way to stimulate cooperation among selfish nodes is to use reputation-based methods with necessary monitoring [130] [88] [59]. Marti et. al [89] propose

a mechanism, called “watchdog”, to identify the misbehaving nodes and another mechanism, called “pathrater”, to deflect the traffics around them. The major drawback of their method is that misbehaving nodes are not punished. Therefore, there is no incentive for the nodes to cooperate. To overcome this problem, Buchegger and Boudec [22] as well as Michiardi and Molva [90] propose reputation-based mechanisms to enforce node cooperation. In both approaches, nodes observe the behavior of each other, store this information locally, and distribute this information in reputation reports. According to their observations, nodes isolate the misbehaving nodes by denying forwarding packets to them. However, there is no theoretical justification about the optimality of such approaches.

Recently, efforts have been made to mathematically analyzing cooperation in cognitive networks using game theory [91] [42] [84] [128]. Srinivasan et al. [112] propose to use generous TIT-FOR-TAT strategy while Urpi et al. [118] propose to use Bayesian games. In [53], Felegyhazi et al. investigate equilibrium conditions of packet forwarding strategies based on game theory and graph theory by taking into account the network topology. In [131], Yu and Liu propose a game theoretic framework to jointly analyze cooperation stimulation and security in autonomous mobile ad hoc networks. Their results show that, for a two-player packet forwarding game, the unique cheat-proof Nash equilibrium for every node is not to help the opponent more than the opponent has helped him/her.

However, most of the existing game theoretical frameworks rely on the assumption that the game between a pair of players is directly played for infinite times. In reality, due to mobility or changes of environment, nodes will periodically update

their partners to achieve better performance, which means that any pair of players are supposed to play for only finite times with the termination time are either known or can be estimated by both players. Note that every player can experience infinite times with many players but never always with the same partner. In such a case, according to the well known Prisoner's Dilemma and backward induction principle [96], the only optimal strategy is to always play non-cooperatively. The major reason causing such a non-cooperative optimal strategy is the implicit assumption of direct reciprocity in most games, where the action of a player taking towards his/her opponent is purely determined by the history of how the opponent treats him/her. Obviously, under such a scenario, all players have no incentive to play cooperatively since their behaviors will not be evaluated by other players except their opponents.

To stimulate the plays' incentive to play cooperatively, not only the evaluations from the opponents but also the evaluations from other observers should be taken into account, which leads to the notion of "indirect reciprocity". Indirect reciprocity is a key mechanism for the evolution of human cooperation and has recently drawn a lot of attentions in the area of social science and evolutionary biology [93] [94]. The key concept of indirect reciprocity is "*I help you not because you have helped me but because you have helped others*". In this chapter, we propose to use the indirect reciprocity game modelling to stimulate cooperation among selfish nodes for the scenario where the number of interactions between any pair of players are finite. The main contributions of this chapter are summarized as follows.

- We propose a cooperation stimulation scheme to stimulate cooperation among

selfish users in cognitive networks using indirect reciprocity game modelling. Different from the existing game-theoretic approaches, our proposed scheme does not rely on the assumption that the number of interactions between a pair of players are infinite.

- In the proposed scheme, we first develop the concept of reputation distribution to capture not only the mean behavior of the transmitter's reputation but also all likelihoods of the transmitter's reputation that may be. Then, we develop a reputation updating policy for the receiver and observers to update the transmitter's reputation distribution based on the transmitter's previous reputation distribution and his/her action toward the receiver. We also propose a gradient descent algorithm to find the stationary reputation distribution of the whole population for any given optimal action rule.
- In the proposed scheme, we formulate the problem of finding the optimal action rule as a Markov Decision Process (MDP) and proposed a modified value iteration algorithm to find the optimal action rule.
- We show that with an appropriate cost-to-gain ratio, the strategy of forwarding the number of packets that is equal to the reputation level of the receiver is an evolutionarily stable strategy (ESS). We also show that even with only 60 percentage of population adopting the optimal action rule at the beginning, by natural selection, the optimal action rule will quickly spread over the whole population. And once the whole population use the optimal action rule, no one will deviate. Moreover, we find that such an ESS will lead to a "good"

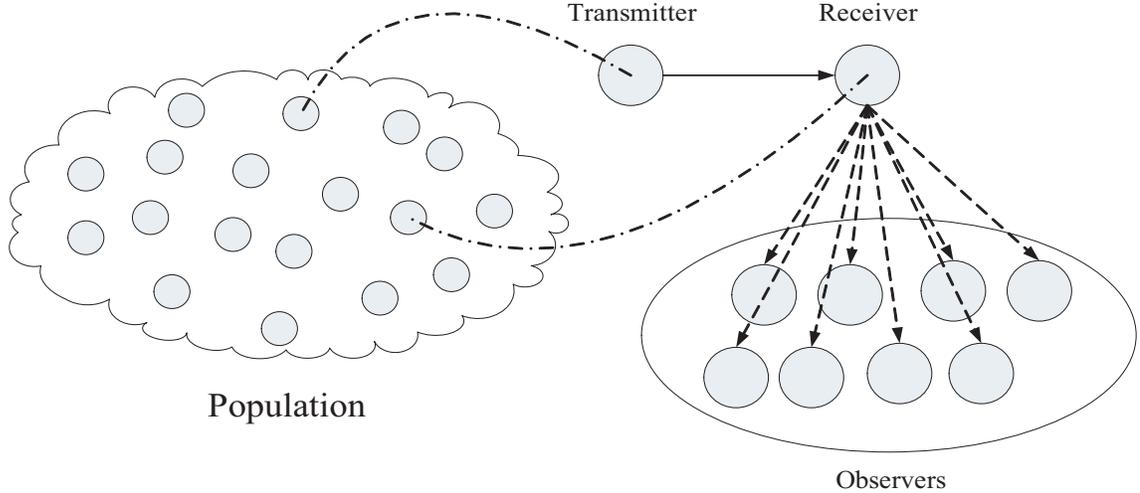


Figure 5.1: System model. Within every interaction, a pair of transmitter and receiver is randomly sampled from the population. Then, the transmitter will forward a certain amount of packets to the receiver according to the receiver’s and his/her own reputations. After the transmission, the transmitter’s reputation will be updated by the receiver and the observers. Finally, the transmitter’s reputation is propagated to the whole population from the receiver and the observers through a noisy gossip channel.

society with more than 90 percentage of the population have good reputation.

The rest of this chapter is organized as follows. In Section 5.1, we describe the problem formulation and introduce basic components in our system model. Then, we show in details how to find the optimal action rule in Section 5.2. In Section 5.3, we describe two action spreading algorithms due to natural selection. Finally, we show the simulation results in Section 5.4 and draw conclusions in Section 5.5.

## 5.1 The System Model

As shown in Figure 5.1, let us consider a cognitive network with sufficiently large population of nodes. Due to mobility and/or changes of environment, short interactions rather than long-lasting associations between anonymous partners are dominant. At each time slot, a fraction of players is chosen from the population to form pairs to forward packets. Within each pair, one player acts as a transmitter and the other player as a receiver. Let  $\mathbf{A} = \{0, 1, \dots, L\}$  stand for the action set that the transmitter may choose, where the action  $i \in \mathbf{A}$  stands for the transmitter forwards  $i$  packets to the receiver.

In the simplest model with  $L = 1$ , the receiver can obtain a gain  $g$  at a cost  $c$  to the transmitter. We should always assume that the gain  $g$  is greater than the cost  $c$ . Otherwise, no transmission will occur. In such a case, if both players cooperate with each other and forward one packet to the other player, both players receive  $g - c$ , which is better than what they would obtain by both defecting, namely 0. However, a unilateral defector would earn  $g$ , which is the highest payoff, and the exploited cooperator would pay the cost  $c$  without receiving any benefit. The payoff structure yields an instance of the well-known Prisoner's Dilemma game and the unique Nash equilibrium (NE) is defecting, i.e. both players will not forward the packet to the other player. Moreover, with backward deduction, the NE remains the same even the game is played a finite number of times. Such a non-cooperative optimal strategy is mainly because of the use of direct reciprocity, where the action of a transmitter taking towards a receiver is purely determined by the history of

how the receiver treats him/her. Obviously, under such a scenario, all transmitters have no incentive to forward packets since their behaviors will not be evaluated by other players except their corresponding receivers.

To stimulate the cooperation under such a scenario, we use the indirect reciprocity game modelling, where the essential concept is: *“I help you not because you have helped me but because you have helped others”*. Therefore, a key concept in indirect reciprocity game is the establishment of the notion of reputation, which is the evaluation of the history of the players’ action. Here, to simplify the analysis, we assume that the reputation is quantized to  $L + 1$  levels with “0” being the worst reputation and “ $L$ ” being the best reputation, i.e., the reputation set can be represented as  $\mathbf{T} = \{0, 1, \dots, L\}$ . However, the results can be easily extended to the case that the reputation set has different size from the action set. Here, we also assume that everyone agrees on the reputation of an individual and no private opinions are allowed. However, errors in assigning reputation are possible. During each interaction, the transmitter determines his action, i.e. how many packets to forward to the receiver, based on the receiver’s and his/her own reputations. After each interaction, the reputation of the receiver remains the same, while the reputation of the transmitter is first updated by the receiver and the observers, and propagated to the whole population through a noisy gossip channel. Then, each participant (including both the transmitter and receiver) goes back to the population with probability  $\delta$  or leaves the population with probability  $1 - \delta$ . The parameter  $\delta$  can be treated as a discounting factor of the future. For every player who leaves the population, a new individual enters with an initial reputation randomly chosen from the reputation set

with equal probability  $\frac{1}{L+1}$ .

### 5.1.1 Social Norms

A social norm,  $\mathbf{Q}$ , is a matrix used for updating the immediate reputation of players, where the immediate reputation is the reputation that a transmitter can immediately obtain by taking an action. Each element  $Q_{i,j}$  in the social norm stands for the immediate reputation assigned to a transmitter who has taken the action  $i$  toward a receiver whose reputation is  $j$ . Without loss of generality, we assume that all players in the population share the same norm. Although the immediate reputation is only determined by the action of the transmitter and the reputation of the receiver, we can see from the later discussion, the final reputation updating rule also involves the reputation of the transmitter.

Since both the cardinalities of the action set and the reputation set are  $L + 1$ , there are  $(L + 1)^{(L+1) \times (L+1)}$  possible social norms. Based on the intuition that forwarding packets to the receiver with good reputation or denying forwarding packets to the receiver with bad reputation should receive good reputation, here, we define the immediate reputation  $Q_{i,j}$  as follows

$$Q_{i,j} = L - |i - j|, \quad (5.1)$$

which means that the social norm is

$$\mathbf{Q} = \begin{pmatrix} L & L-1 & \dots & 1 & 0 \\ L-1 & L & \dots & \vdots & 1 \\ \vdots & L-1 & \ddots & L-1 & \vdots \\ 1 & \vdots & \dots & L & L-1 \\ 0 & 1 & \dots & L-1 & L \end{pmatrix}. \quad (5.2)$$

For the special case when  $L = 1$ , the  $2 \times 2$  social norm can be written as

$$\mathbf{Q}^{2 \times 2} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad (5.3)$$

where “1” stands for good reputation and “0” stands for bad reputation.

With such a social norm shown in (5.3), we can see that the transmitter can obtain a good immediate reputation by either forwarding packets to the receiver with good reputation or denying forwarding packets to the receiver with bad reputation. On the other hand, the transmitter will obtain a bad immediate reputation if he/she either denies forwarding packets to the receiver with good reputation or forwards packets to the receiver with bad reputation.

### 5.1.2 Action Rules

An action rule,  $\mathbf{a}$ , is an action table of the transmitter, where the  $i^{th}$  row and  $j^{th}$  column element  $a_{i,j}$  stands for the action of the transmitter based on his/her own reputation  $i$  and the corresponding receiver’s reputation  $j$ . Since both the cardinalities of the action set and the reputation set are  $L + 1$ , there are  $(L +$

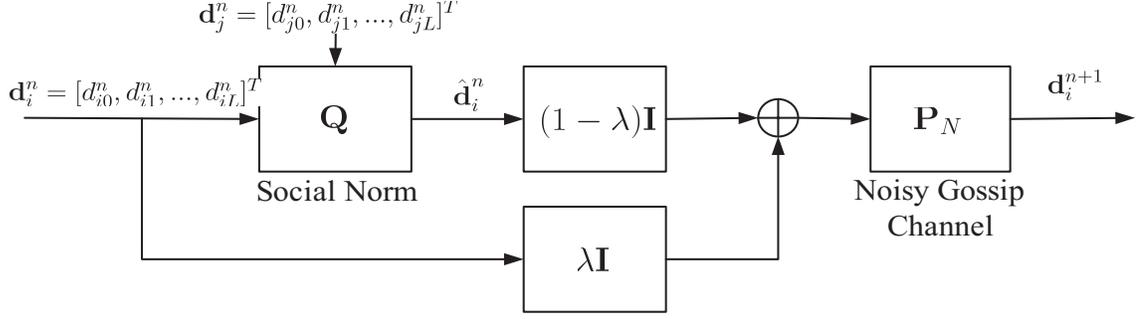


Figure 5.2: Reputation updating policy.

$1)^{(L+1) \times (L+1)}$  possible action rules. The optimal action rule,  $\mathbf{a}^*$ , should be the one that maximizes the payoff function as discussed later.

## 5.2 Optimal Action Rule

### 5.2.1 Reputation Updating Policy

A key concept in indirect reciprocity game is reputation [94]. There is a similar notion of trust [115], however, which is mostly based on direct reciprocity. Players monitor the social interactions within their group and help others establish the reputation of being a helpful player. Therefore, one important step in indirect reciprocity game modelling is how to update reputation based on players' actions. In this subsection, we develop a reputation updating policy based on the action of the transmitter, the reputation of the transmitter and the reputation of the receiver.

To capture not only the mean behavior of the transmitter's reputation but also all likelihoods of the transmitter's reputation that may be, we assign a reputation distribution for each player. Let  $\mathbf{d} = [d_0, d_1, \dots, d_L]^T$  be a reputation distribution for a specific player. Then  $d_i$  stands for the likelihood of the player being assigned with

reputation  $i$ .

The proposed reputation updating policy is shown in Fig. 5.2. Suppose, at time index  $n$ , a transmitter with a reputation distribution  $\mathbf{d}_i^n$  is matched with a receiver with a reputation distribution  $\mathbf{d}_j^n$ . By taking a certain action, the transmitter is assigned with an immediate reputation  $\hat{\mathbf{d}}_i^n$  based on the social norm. Then, the receiver and the observers will update the transmitter's reputation distribution using a linear combination of the transmitter's original and immediate reputations, where the weight  $\lambda$  can be treated as a discounting factor of the past reputation. Finally, the transmitter's reputation is propagated among the population by the receiver and observers through a noisy gossip channel.

In a simple example, we assume that the transmitter's reputation distribution is  $\mathbf{d}_i^n = \mathbf{e}_i$  and the receiver's reputation distribution is  $\mathbf{d}_j^n = \mathbf{e}_j$ , where  $\mathbf{e}_i$  and  $\mathbf{e}_j$  are the standard basis vectors. Let  $a_{i,j}$  be the action the transmitter takes towards the receiver. Then, the immediate reputation of the transmitter is  $\mathbf{e}_{Q_{a_{i,j},j}}$ . According to the reputation updating policy in Fig. 5.2, after the transmission, the transmitter's reputation distribution becomes

$$\mathbf{d}_i^{n+1} = \mathbf{P}_N \left( \lambda \mathbf{e}_i + (1 - \lambda) \mathbf{e}_{Q_{a_{i,j},j}} \right), \quad (5.4)$$

where  $\mathbf{P}_N$  is the transition matrix of the noisy channel. Without loss of generality<sup>1</sup>,

---

<sup>1</sup>Note that the analysis in this chapter are also applicable to the  $\mathbf{P}_N$  with other forms.

we define  $\mathbf{P}_N$  as follows

$$\mathbf{P}_N = \begin{pmatrix} 1 - \mu & \mu/L & \dots & \mu/L & \mu/L \\ \mu/L & 1 - \mu & \dots & \vdots & \mu/L \\ \vdots & \mu/L & \ddots & \mu/L & \vdots \\ \mu/L & \vdots & \dots & 1 - \mu & \mu/L \\ \mu/L & \mu/L & \dots & \mu/L & 1 - \mu \end{pmatrix}, \quad (5.5)$$

with  $\mu \in [0, 0.5]$  being a constant.

The  $\mathbf{d}_i^{n+1}$  in (5.4) is the updated reputation distribution of the transmitter after the transmitter with an original reputation  $\mathbf{e}_i$  takes an action  $a_{i,j}$  towards the receiver with a reputation  $\mathbf{e}_j$ . Since this updated reputation distribution will be used later in the analysis for finding the optimal action rule, we use a specific symbol  $\tilde{\mathbf{d}}_{i \rightarrow j}$  to denote it, i.e.,

$$\tilde{\mathbf{d}}_{i \rightarrow j} = \mathbf{P}_N \left( \lambda \mathbf{e}_i + (1 - \lambda) \mathbf{e}_{Q_{a_{i,j},j}} \right). \quad (5.6)$$

For the general case that  $\mathbf{d}_i^n \neq \mathbf{e}_i$  and/or  $\mathbf{d}_j^n \neq \mathbf{e}_j$ , the transmitter's updated reputation distribution cannot be simply expressed using (5.4) since, given an action rule, different combinations of the transmitter's and receiver's reputations may lead to the same immediate reputation. In such a case, we need to first find the immediate reputation using

$$\hat{\mathbf{d}}_i^n(k) = \sum_p \sum_{q: Q_{a_{p,q}}=k} \mathbf{d}_i^n(p) \mathbf{d}_j^n(q). \quad (5.7)$$

Then, according to Fig. 5.2, the transmitter's updated reputation distribution can be computed by

$$\mathbf{d}_i^{n+1} = \mathbf{P}_N \left( \lambda \mathbf{d}_i^n + (1 - \lambda) \hat{\mathbf{d}}_i^n \right). \quad (5.8)$$

## 5.2.2 Stationary Reputation Distribution

Let  $\mathbf{x} = [x_0, x_1, \dots, x_L]^T$  stand for the reputation distribution of the entire population, where  $x_i$  is the portion of the population that have the reputation  $i$ . Since every pair of transmitter and receiver is chosen from the population, given the transmitter with reputation  $i$ , the probability of matching with the receiver with reputation  $k$  is  $x_k$ . After the transmission, the reputation of the transmitter is updated using the policy shown in Fig.5.2. Therefore, the evolution of  $\mathbf{x}$  can be described by the following differential equation

$$\frac{d\mathbf{x}}{dt} = \mathbf{x}^{new} - \mathbf{x}, \quad (5.9)$$

where  $\mathbf{x}^{new}$  is the new reputation distribution of the entire population and can be computed by

$$\mathbf{x}^{new} = \mathbf{P}_N (\lambda \mathbf{I} + (1 - \lambda) \mathbf{P}_T) \mathbf{x}, \quad (5.10)$$

with the  $i^{th}$  row and  $j^{th}$  column element of the matrix  $\mathbf{P}_T$  being defined as

$$\mathbf{P}_T(j, i) = \sum_{k: Q_{a^*, k}^* = j} x_k. \quad (5.11)$$

According to (5.9), (5.10), and (5.11), the stationary reputation distribution  $\mathbf{x}^*$  is the solution to the following equation

$$\mathbf{P}_N (\lambda \mathbf{I} + (1 - \lambda) \mathbf{P}_T) \mathbf{x}^* = \mathbf{x}^*. \quad (5.12)$$

From (5.11) and (5.12), we can see that, given the optimal action  $a^*$ , the stationary reputation distribution can be found by solving the nonlinear equations in (5.12). In Algorithm 3, we propose a gradient descent algorithm for finding the stationary reputation distribution given the optimal action rule.

---

**Algorithm 3 : Finding Stationary Reputation Distribution Using Gradient Descent**

---

1. Given the optimal action  $a_{i,j}^*, \forall i, \forall j$ , the tolerance  $\eta_0 = 0.01$ , the index  $t = 0$ , and step size  $\alpha = 0.1$ , initialize  $\mathbf{x} = [x_0, x_1, \dots, x_L]^T$  with  $\mathbf{x}^0 = [x_0^0, x_1^0, \dots, x_L^0]^T$ , set  $\epsilon = 1$ , and let  $\mathbf{F}(\mathbf{x}) = \mathbf{P}_N (\lambda \mathbf{I} + (1 - \lambda) \mathbf{P}_T(\mathbf{x})) \mathbf{x} - \mathbf{x}$ .

2. while  $\epsilon > \eta_0$

- Compute the updating vector  $\Delta \mathbf{x}^{t+1}$  using  $\Delta \mathbf{x}^{t+1} = -\alpha \times \nabla \mathbf{F}(\mathbf{x}^t) \times \mathbf{F}(\mathbf{x}^t)$ .
- Update  $\mathbf{x}^{t+1}$  by  $\mathbf{x}^{t+1} = \mathbf{x}^t + \Delta \mathbf{x}^{t+1}$ .
- Normalize  $\mathbf{x}^{t+1}$  using  $\mathbf{x}^{t+1} = \frac{\mathbf{x}^{t+1}}{\|\mathbf{x}^{t+1}\|_2}$ .
- Update the parameter  $\epsilon$  by  $\epsilon = \|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2$ .
- Update the index  $t = t + 1$ .

End

3. The stationary reputation distribution is  $\mathbf{x}^* = \mathbf{x}^t$ .

---

### 5.2.3 Payoff Function

Suppose that the cost of forwarding a packet is a constant,  $c$ , the total cost of the transmitter with reputation  $i$  taking action  $a_{i,j}$  towards a receiver with reputation  $j$  is given by

$$C(a_{i,j}) = a_{i,j}c. \quad (5.13)$$

Similarly, if the gain of receiving a packet is a constant,  $g$ , the total gain of the receiver with reputation  $i$  can be computed by

$$G(a_{j,i}) = a_{j,i}g, \quad (5.14)$$

where  $a_{j,i}$  is the action of the corresponding transmitter with reputation  $j$ .

Let  $W_{i,j}$  denote the maximum payoff that a player, currently having reputation  $i$  and being matched with a player with reputation  $j$ , can gain from this interaction to future. Obviously, if the player with reputation  $i$  serves as a transmitter, then the long-term expected payoff that he/she can obtain by taking action  $a_{i,j}$  would be

$$f_1(a_{i,j}) = -a_{i,j}c + \delta \sum_k \sum_l \tilde{\mathbf{d}}_{i \rightarrow j}(k) \mathbf{x}^*(l) W_{k,l}, \quad (5.15)$$

where the first term  $a_{i,j}c$  is the immediate cost the transmitter incurred by taking action  $a_{i,j}$ , and the second term  $\sum_k \sum_l \tilde{\mathbf{d}}_{i \rightarrow j}(k) \mathbf{x}^*(l) W_{k,l}$  stands for the benefit he gains in the future with a discounting factor  $\delta$ . According to (5.6), by taking action  $a_{i,j}$ , the reputation distribution of the transmitter will change from  $\mathbf{e}_i$  to  $\tilde{\mathbf{d}}_{i \rightarrow j}$ . Since his opponent in the next round is randomly sampled from the population with a stationary reputation distribution  $\mathbf{x}^*$ , with probability  $\tilde{\mathbf{d}}_{i \rightarrow j}(k) \mathbf{x}^*(l)$ , the transmitter's reputation becomes  $k$  and his opponent's reputation is  $l$ .

On the other hand, if the player with reputation  $i$  serves as a receiver, the long-term expected payoff that he/she can obtain is

$$f_2 = a_{j,i}^*g + \delta \sum_l x_l^* W_{i,l}, \quad (5.16)$$

where the first term  $a_{j,i}^*g$  is the immediate gain he/she can obtain when the transmitter takes the optimal action  $a_{j,i}^*$ , and the second term  $\sum_l x_l^* W_{i,l}$  stands for the benefit he gains in the future with a discounting factor  $\delta$ . As a receiver, the reputation will not change after the transmission. Since his opponent in the next round is randomly sampled from the population with a stationary reputation distribution  $\mathbf{x}^*$ , with probability  $\mathbf{x}^*(l)$ , the receiver's reputation is  $i$  and his opponent's reputation is  $l$ .

With each interaction, the play acts either as a transmitter or as a receiver with equal probability  $\frac{1}{2}$ . Therefore, the Bellman equation of  $W_{i,j}$  can be written as

$$\begin{aligned}
W_{i,j} = & \max_{a_{i,j}} \left[ \frac{1}{2} \left( -a_{i,j}c + \delta \sum_k \sum_l \tilde{\mathbf{d}}_{i \rightarrow j}(k) \mathbf{x}^*(l) W_{k,l} \right) \right. \\
& \left. + \frac{1}{2} \left( a_{j,i}^* g + \delta \sum_l \mathbf{x}^*(l) W_{i,l} \right) \right], \tag{5.17}
\end{aligned}$$

and the optimal action  $a_{i,j}^*$  can be computed by

$$\begin{aligned}
a_{i,j}^* = & \arg \max_{a_{i,j}} W_{i,j} \\
= & \arg \max_{a_{i,j}} \left[ \frac{1}{2} \left( -a_{i,j}c + \delta \sum_k \sum_l \tilde{\mathbf{d}}_{i \rightarrow j}(k) \mathbf{x}^*(l) W_{k,l} \right) \right]. \tag{5.18}
\end{aligned}$$

From (5.17) and (5.18), we can see that the problem of finding the optimal action rule is a Markov Decision Process (MDP), where the state is the reputation pair  $(i, j)$ , the action is  $a_{i,j}$ , the transition probability is determined by  $\tilde{\mathbf{d}}_{i \rightarrow j}$  and  $\mathbf{x}^*$ , and the reward is determined by  $c$  and  $g$ . Therefore, given the stationary reputation distribution, the optimal action can be found by solving (5.18) using dynamic programming. In this chapter, we propose a modified value iteration algorithm to find the optimal action given stationary reputation distribution, which is shown in Algorithm 4.

## 5.2.4 Optimal Action Using An Alternative Algorithm

From the previous two subsections, we can see that given the optimal action, the stationary reputation distribution can be found using Algorithm 3, and given

the stationary reputation distribution, the optimal action can be found using Algorithm 4. Therefore, we can obtain the optimal action and the stationary reputation distribution alternatively by iteratively fixing one and solving the other. The detailed processes are summarized in Algorithm 5. Note that the convergence speed of Algorithm 5 is highly determined by the initial action rule  $\mathbf{a}^0$ . Nevertheless, it will converge since the number of the possible action rules is finite. Moreover, Algorithm 5 can also be used to test the evolutionary stability of any action rule. The idea is to set the tested action rule as the initial action rule and see whether it can converge in one iteration. The details will be discussed in Section 5.4.

### 5.3 Action Spreading Due To Natural Selection

Based on Algorithm 5, we can find the optimal action rule and the stationary reputation distribution. However, during the above analysis, we do not include the perturbation effect, where players may take non-optimal action rule due to uncertainty of the system and/or the incorrect (noisy) parameters. Taking the perturbation effect into account, we need to evaluate the stability of the optimal action rule. Here, we adopt the concept of evolutionarily stable strategy (ESS) [110], which is “a strategy such that, if all members of the population adopt it, then no mutant strategy could invade the population under the influence of natural selection”. In the following subsections, we first discuss, by natural selection, how the action rules spread over the population. Specifically, we discuss two action spreading algorithms: one is action spreading algorithm using Wright-Fisher model

[54] and the other is action spreading algorithm using replicator dynamic equation [110]. Then, we examine, in Section 5.4, the stability of the optimal action rule derived by Algorithm 5 by the simulations.

Let  $M$  be the number of action rules,  $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_M$ , used in the population. Let  $p_i^t$  be the percentage of the population that uses action rule  $\mathbf{a}_i$  at time  $t$ . Then, we have  $\sum_{i=1}^M p_i^t = 1$ . Let  $U_i^t$  be the average payoff using action rule  $\mathbf{a}_i$  at time  $t$ .

### 5.3.1 Action Spreading Algorithm Using Wright Fisher Model

The Wright-Fisher model is by far the most popular stochastic model for reproduction in population genetics [54]. It is based on the assumption that the probability of an individual adopting a certain strategy is proportional to the expected payoff of the population using that strategy. Due to its simplicity and capability of capturing the essence of the biology involved, we use the Wright-Fisher model here to characterize how action rules spread over the population.

Let  $y_i$  be the probability of an individual using action  $\mathbf{a}_i$ . Then, we have  $\sum_{i=1}^M y_i = 1$ . With the Wright-Fisher Model, we assume that  $y_i$  is proportional to the total payoff of the users using  $\mathbf{a}_i$ . Therefore,  $y_i$  can be computed by

$$y_i = \frac{p_i^t U_i^t}{\sum_{j=1}^M p_j^t U_j^t}, \quad (5.19)$$

where the numerator  $p_i^t U_i^t$  is the total payoff of the users using action  $\mathbf{a}_i$ , and the denominator  $\sum_{j=1}^M p_j^t U_j^t$  is the total payoff of the whole population, which is the normalization term that ensures  $\sum_{i=1}^M y_i = 1$ .

Based on the assumption that the population size is sufficiently large, the per-

centage of the population using action  $\mathbf{a}_i$  is equal to the probability of an individual using  $\mathbf{a}_i$ . Therefore, the action spreading equation can be written as

$$p_i^{t+1} = \frac{p_i^t U_i^t}{\sum_{j=1}^M p_j^t U_j^t}. \quad (5.20)$$

### 5.3.2 Action Spreading Algorithm Using Replicator Dynamic Equation

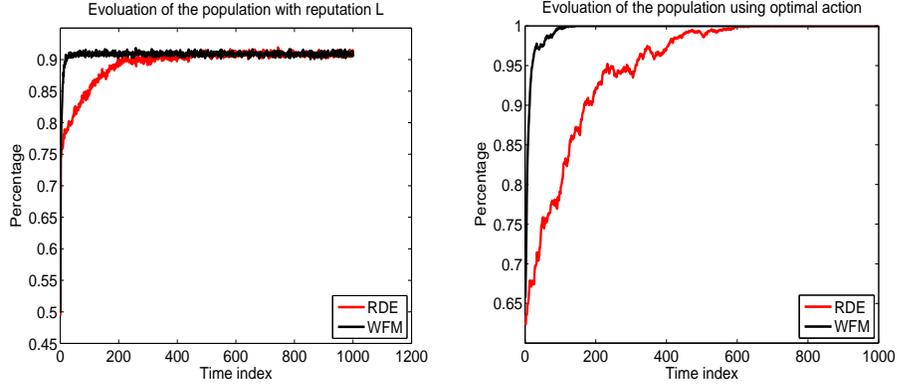
Replicator dynamic equation is widely used to characterize the population evolution in evolutionary game theory [110]. It is based on the following intuition: if a certain strategy results in a higher payoff than the average level, the population share using that strategy will grow with the growth rate proportional to the difference between the expected payoff of the population using that strategy and the expected payoff of the entire population. In this subsection, we use the replicator dynamic equation to model the evolution of the percentage of the population using a certain action rule, which means that the evolution of  $p_i$  is given by the following equation

$$\frac{dp_i}{dt} = \eta \left( U_i - \sum_{j=1}^M p_j U_j \right) p_i, \quad (5.21)$$

where  $\eta$  is a scale factor controlling the speed of the evolution.

By discretizing the replicator dynamic equation in (5.21), we have the action spreading equation

$$\begin{aligned} p_i^{t+1} &= p_i^t + \eta \left( U_i^t - \sum_{j=1}^M p_j^t U_j^t \right) p_i^t \\ &= p_i^t \left[ 1 + \eta \left( U_i^t - \sum_{j=1}^M p_j^t U_j^t \right) \right]. \end{aligned} \quad (5.22)$$



(a)

(b)

Figure 5.3: The population evolution when  $L = 1$ ,  $g = 1$  and  $c = 0.1$ : (a) the percentage of the population with reputation  $L = 1$ ; (b) the percentage of the population using optimal action shown in (5.24).

## 5.4 Evolutionarily Stable Strategy and Simulations

To verify the proposed algorithm, we simulate the packet forwarding game. We study a fixed-size population,  $N = 1000$ . Each new player receives an initial reputation, which is randomly chosen from  $\{0, 1, \dots, L\}$  with equal probability  $\frac{1}{L+1}$ . Each player uses one of  $(L + 1)^{(L+1) \times (L+1)}$  possible action rules. All players in the population share the fixed social norm defined in (5.2). Before any one elementary step of action updating, each individual has exactly 20 interactions with other randomly chosen individuals. Individuals act as transmitter and receiver on average 10 times each. After each interaction, the reputation of the transmitter is updated according to the reputation updating policy shown in Fig. 5.2. We assume that every player in the population agrees on the reputation generated by the reputation updating policy. No private lists of reputation are considered. After all 20 interac-

tions have taken place, each participant including both the transmitter and receiver goes back to the population with probability  $\delta$  or leaves the population with probability  $1 - \delta$ . For every player who leaves, a new individual enters the population to keep the total population size constant. The initial reputation of the new coming is randomly chosen from  $\{0, 1, \dots, L\}$  with equal probability  $\frac{1}{L+1}$ . Then, the players in the population, including the old players who stay in the population and the new players who enter the population, choose their new action rules according to previous payoff history of the whole population. There are two possible action spreading algorithms as shown in the previous section. One is the action spreading algorithm using Wright Fisher Model, which is denoted as “**WFM**”, and the other one is the action spreading algorithm using Replicator Dynamic Equation, which is denoted as “**RDE**”. After updating the action rule, the payoffs of all players are reset to zero. Therefore, older players do not accumulate their payoffs. In all the following simulations, the parameters  $\lambda$ ,  $\delta$ , and  $\mu$  are set to be 0.5, 0.9, and 0.95 respectively. The parameter  $\eta$  that controls the speed of the evolution in **RDE** is set to be 0.1.

#### 5.4.1 Binary Reputation Scenario

To give more insights into the proposed algorithm, we first evaluate the binary reputation scenario where  $L = 1$ . We assume that the gain per unit is 1 and the cost per unit is 0.1, i.e.  $g = 1$  and  $c = 0.1$ . According to Algorithm 5, with different initial conditions, we can find three pairs of stationary reputation distribution  $\mathbf{x}^*$

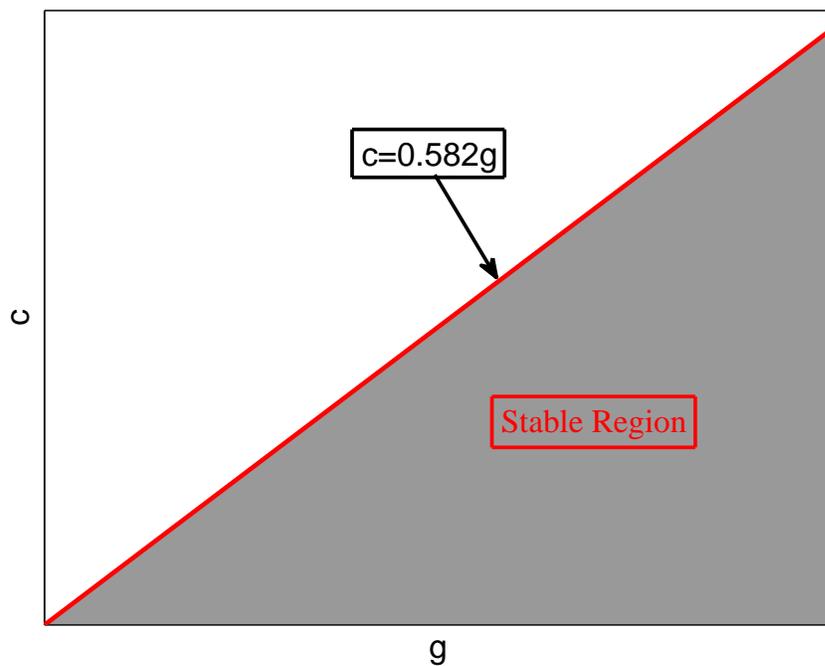


Figure 5.4: The stable region for the optimal action rule shown in (5.24) when  $L = 1$ .

and the optimal action rule  $\mathbf{a}^*$ , which are

$$\mathbf{x}_1^* = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}, \quad \mathbf{a}_1^* = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \quad (5.23)$$

$$\mathbf{x}_2^* = \begin{pmatrix} 0.0909 \\ 0.9091 \end{pmatrix}, \quad \mathbf{a}_2^* = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}. \quad (5.24)$$

$$\mathbf{x}_3^* = \begin{pmatrix} 0.9091 \\ 0.0909 \end{pmatrix}, \quad \mathbf{a}_3^* = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}. \quad (5.25)$$

With  $(\mathbf{x}_1^*, \mathbf{a}_1^*)$ , the transmitter will not forward any packet to the receiver regardless his/her own reputation and the corresponding receiver's reputation. Obvi-

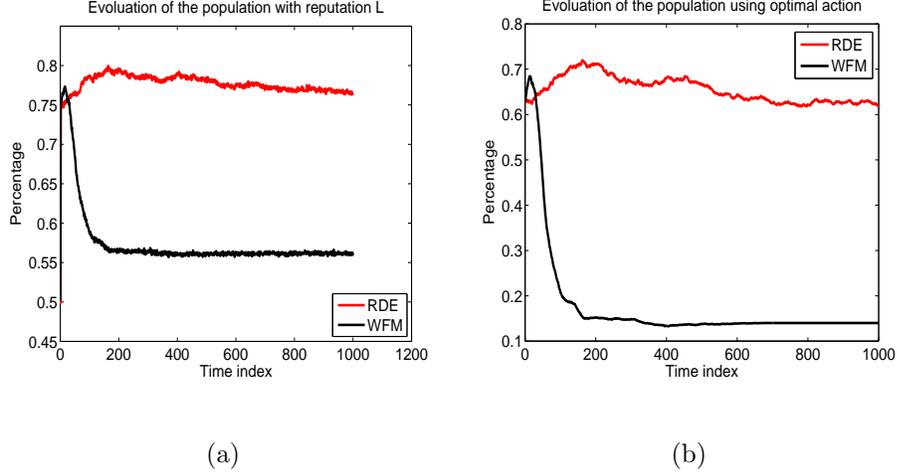


Figure 5.5: The population evolution when  $L = 1$ ,  $g = 1$  and  $c = 0.6$ : (a) the percentage of the population with reputation  $L = 1$ ; (b) the percentage of the population using optimal action shown in (5.24).

ously, it is a bad strategy since, with such a strategy, there is no cooperation and the payoff of every player is zero. The pairs  $(\mathbf{x}_2^*, \mathbf{a}_2^*)$  and  $(\mathbf{x}_3^*, \mathbf{a}_3^*)$  are symmetric where with the former pair, the transmitter will always forward packets to the receiver who has good reputation, and with the latter pair, the transmitter will always forward packets to the receiver who has bad reputation. We can also find that the pair  $(\mathbf{x}_2^*, \mathbf{a}_2^*)$  leads to a population with more than 90 percentage of the players are good reputation while  $(\mathbf{x}_3^*, \mathbf{a}_3^*)$  leads to a population with more than 90 percentage of the players are bad reputation. Here, we prefer  $(\mathbf{x}_2^*, \mathbf{a}_2^*)$  since it leads to a “good” society with more than 90 percentage of the population are good reputation.

Then, we evaluate the evolutionary stability of  $(\mathbf{x}_2^*, \mathbf{a}_2^*)$ . In the simulation, the initial frequency of the optimal action rule  $\mathbf{a}^*$  shown in (5.24) is set to be 0.6. The initial frequencies of the other action rules are randomly chosen. The initial reputation of new players is randomly chosen from  $\{0, 1\}$  with equal probability  $\frac{1}{2}$ . In

Fig. 5.3 (a), we show the evolutionary results of the percentage of the population with reputation level  $L = 1$ . From Fig. 5.3 (a), we can see that for both **WFM** and **RDE**, the reputation distribution converges to the stationary reputation distribution  $\mathbf{x}_2^*$ . Compared with **WFM**, the convergence speed of **RDE** is a bit slower since a small speed controlling parameter  $\eta = 0.1$  is used in **RDE**.

The evolutionary results of the percentage of the population using the action rule  $\mathbf{a}_2^*$  are shown in Fig. 5.3 (b). From Fig. 5.3 (b), we can see that for both **WFM** and **RDE**, the action rule  $\mathbf{a}_2^*$  will spread over the whole population. And once the whole population adopt  $\mathbf{a}_2^*$ , no one will deviate. Therefore, the action rule  $\mathbf{a}_2^*$  is an evolutionarily stable strategy (ESS) [110] in this case.

From (5.17), we can see that the optimal action rule is determined by the values of  $g$  and  $c$ . Intuitively, if  $g \gg c$ , every player is willing to cooperate with other players since in such a scenario, the potential cooperation gain will be greater than the immediate cooperation cost. On the other hand, if  $c \gg g$ , every player tends not to cooperate with other players since the potential cooperation gain will be smaller than the immediate cooperation cost in such a scenario. Based on the intuition, there should exist a critical cost-to-gain ratio  $\gamma$  such that the optimal action rule  $\mathbf{a}_2^*$  is stable if  $c < \gamma g$  and is not stable otherwise.

By setting  $\mathbf{a}_2^*$  as the initial action rule  $\mathbf{a}^0$  in Algorithm 5 and varying  $g$  and  $c$ , we find that if  $\frac{c}{g} \leq 0.582$ , the optimal action rule found by Algorithm 5 is  $\mathbf{a}_2^*$ . On the other hand, if  $\frac{c}{g} > 0.582$ , the optimal action rule changes to be  $\mathbf{a}_1^*$ . Therefore, the critical cost-to-gain ratio  $\gamma$  is equal to 0.582 in this case, which means that the stable region for  $\mathbf{a}_2^*$  is the shadow region shown in Fig. 5.4.

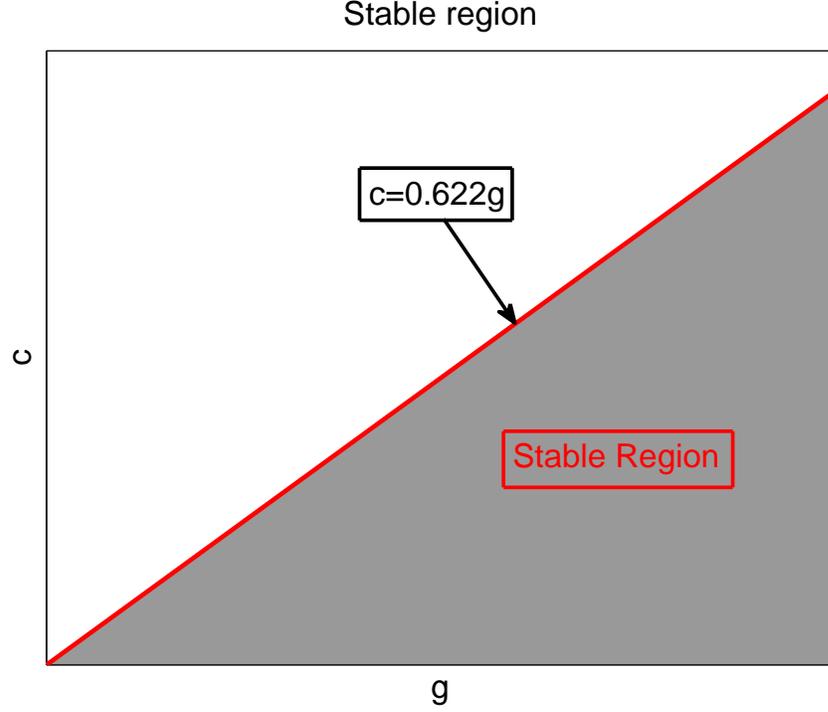
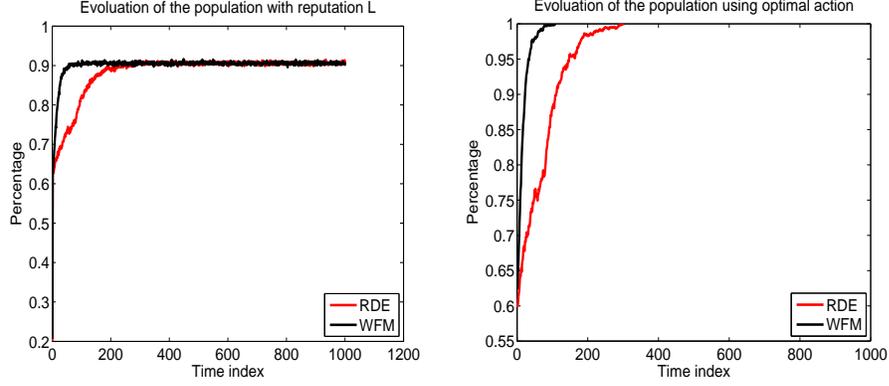


Figure 5.6: The stable region for the optimal action rule shown in (5.26) when  $L = 4$ .

We verify the above statement by evaluating the stability of  $\mathbf{a}_2^*$  when  $g = 1$  and  $c = 0.6$ . The corresponding evolutionary results are shown in Fig. 5.5. From Fig. 5.5 (b), we can see that when  $\frac{c}{g} = 0.6 > 0.582$ , the percentage of the population using action rule  $\mathbf{a}_2^*$  does not converge to 1 for both **WFM** and **RDE**. Therefore,  $\mathbf{a}_2^*$  is not stable in this case. Correspondingly, we can also see from Fig. 5.5 (a) that the reputation distribution does not converge to  $\mathbf{x}_2^*$  in this case.

#### 5.4.2 Multi-Level Reputation Scenario

For the multi-level reputation scenario where  $L \geq 2$ , due to the large dimension of the action space  $((L + 1)^{(L+1) \times (L+1)})$ , it is difficult to find all the possible pairs of



(a)

(b)

Figure 5.7: The population evolution when  $L = 4$ ,  $g = 1$  and  $c = 0.5$ : (a) the percentage of the population with reputation  $L = 4$ ; (b) the percentage of the population using optimal action shown in (5.26).

stationary reputation distribution  $\mathbf{x}^*$  and optimal action rule  $\mathbf{a}^*$ . However, based on the results in the binary reputation scenario, we can infer that one possible optimal action rule  $\mathbf{a}_0^*$  is to forward  $i$  packets to the receiver with reputation  $i$ , i.e.  $\mathbf{a}_0^*$  can be written as

$$\mathbf{a}_0^* = \begin{pmatrix} 0 & 1 & \dots & L \\ 0 & 1 & \dots & L \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & \dots & L \end{pmatrix}. \quad (5.26)$$

According to Algorithm 3, we can find the corresponding stationary reputation distribution  $\mathbf{x}_0^*$ . For the special case with  $L = 4$ ,  $\mathbf{x}_0^*$  is

$$\mathbf{x}_0^* = \left( 0.0235 \quad 0.0235 \quad 0.0235 \quad 0.0235 \quad 0.906 \right)^T. \quad (5.27)$$

Then, similar to the binary reputation scenario, we obtain the stable region for the optimal action rule  $\mathbf{a}_0^*$ . By setting  $\mathbf{a}_0^*$  as the initial action rule  $\mathbf{a}^0$  in Algorithm

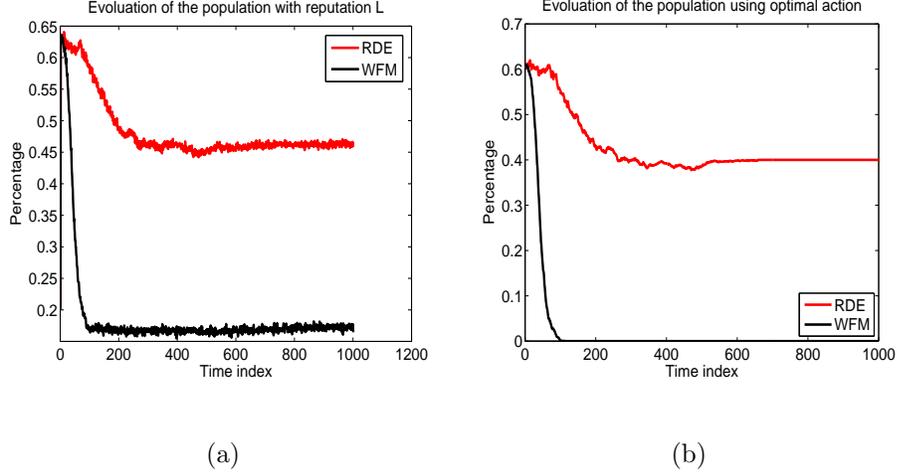


Figure 5.8: The population evolution when  $L = 4$ ,  $g = 1$  and  $c = 0.7$ : (a) the percentage of the population with reputation  $L = 4$ ; (b) the percentage of the population using optimal action shown in (5.26).

5 and varying  $g$  and  $c$ , we find that if  $\frac{c}{g} \leq 0.622$ , the optimal action rule found by Algorithm 5 is still  $\mathbf{a}_0^*$ . On the other hand, if  $\frac{c}{g} > 0.622$ , the optimal action rule changes. Therefore, the critical cost-to-gain ratio  $\gamma$  in this case is equal to 0.622, which means that the stable region for  $\mathbf{a}_0^*$  is the shadow region shown in Fig. 5.6.

We then verify the above statement by simulating the packet forwarding game with two different cost-to-gain ratio settings. One is  $g = 1$  and  $c = 0.5$ , i.e.  $\frac{c}{g} = 0.5 < 0.622$ , and the other is  $g = 1$  and  $c = 0.7$ , i.e.  $\frac{c}{g} = 0.7 > 0.622$ . The evolutionary results for the former setting are shown in Fig. 5.7. From Fig. 5.7, we can see that when the cost-to-gain ratio is set to be  $\frac{c}{g} = 0.5 < 0.622$ , the reputation distribution converges to  $\mathbf{x}_0^*$  and the optimal action rule  $\mathbf{a}_0^*$  spreads over the whole population for both **WFM** and **RDE**, which verifies that  $\mathbf{a}_0^*$  is an ESS in this case.

The evolutionary results for the latter cost-to-gain ratio setting are different and shown in Fig. 5.8. From Fig. 5.8, we can see that when the cost-to-gain ratio

is set to be  $\frac{c}{g} = 0.7 > 0.622$ , for both **WFM** and **RDE**, the action rule  $\mathbf{a}_0^*$  does not spread over the whole population and the reputation distribution does not converge to  $\mathbf{x}_0^*$ . Therefore,  $\mathbf{a}_0^*$  is not stable in this case.

## 5.5 Summary

In this chapter, we propose a cooperation stimulation scheme for cognitive networks using indirect reciprocity game modelling. Different from the existing game theoretic approaches, our proposed scheme does not rely on the assumption that the number of interactions between a pair of players are infinite. From the simulation results, we can see that with a proper cost-to-gain ratio, the action rule of forwarding  $i$  packets to the receiver with reputation level  $i$  is an ESS. Even starting with only 60 percentage of population adopting the optimal action rule, the optimal action rule will quickly spread over the whole population by natural selection. And once the whole population use the optimal action rule, no one will deviate. Moreover, such an ESS will lead to a “good” society where more than 90 percentage of the population have good reputation.

---

## Algorithm 4 : Modified Value Iteration For Optimal Action Selection

---

### Given Stationary Reputation Distribution

---

1. Given the stationary reputation  $\mathbf{x}^*$ , tolerance  $\eta_0 = 0.01$ , initialize  $a_{i,j}^*$  with  $a_{i,j}^0 \forall i \forall j$ , set  $\epsilon_1 = 1$  and  $\epsilon_2 = 1$ .

2. while  $\epsilon_1 > \eta_0$

- Set  $\epsilon_2 = 1$ .

- Initialize  $W_{i,j} = 0 \forall i \forall j$ .

- while  $\epsilon_2 > \eta_0$

- Compute  $\tilde{\mathbf{d}}_{i \rightarrow j}$  using  $\tilde{\mathbf{d}}_{i \rightarrow j} = \mathbf{P}_N \left( \lambda \mathbf{e}_i + (1 - \lambda) \mathbf{e}_{Q_{a_{i,j},j}} \right)$ .

- Compute  $\hat{W}_{i,j}$  using

$$\hat{W}_{i,j} = \max_{a_{i,j}} \left[ \frac{1}{2} \left( -a_{i,j}c + \delta \sum_k \sum_l \tilde{\mathbf{d}}_{i \rightarrow j}(k) \mathbf{x}^*(l) W_{k,l} \right) + \frac{1}{2} \left( a_{j,i}g + \delta \sum_l \mathbf{x}^*(l) W_{i,l} \right) \right].$$

- Compute  $\hat{a}_{i,j}$  using

$$\hat{a}_{i,j} = \arg \max_{a_{i,j}} \left[ \frac{1}{2} \left( -a_{i,j}c + \delta \sum_k \sum_l \tilde{\mathbf{d}}_{i \rightarrow j}(k) \mathbf{x}^*(l) W_{k,l} \right) \right].$$

- Update the parameter  $\epsilon_2$  by  $\epsilon_2 = \|\hat{\mathbf{W}} - \mathbf{W}\|_2$ .

- Update  $\mathbf{W}$  by  $\mathbf{W} = \hat{\mathbf{W}}$ .

- End

- Update the parameter  $\epsilon_1$  by  $\epsilon_1 = \|\hat{\mathbf{a}} - \mathbf{a}^*\|_2$ .

- Update  $\mathbf{a}^*$  by  $\mathbf{a}^* = \hat{\mathbf{a}}$ .

End

3. The optimal action is  $\mathbf{a}^*$ .

---

---

**Algorithm 5 : An Alternative Algorithm For Finding Stationary Reputation Distribution And Optimal Action**

---

1. Given the tolerance  $\eta_0 = 0.01$ , initialize  $\mathbf{a}^*$  with  $\mathbf{a}^0$  and set  $\epsilon = 1$ .
2. while  $\epsilon > \eta_0$ 
  - Given the optimal action  $\mathbf{a}^*$ , finding the stationary reputation distribution  $\mathbf{x}^*$  using Algorithm 3.
  - Given the stationary reputation distribution  $\mathbf{x}^*$ , finding the optimal action  $\hat{\mathbf{a}}^*$  using Algorithm 4.
  - Update the parameter  $\epsilon$  by  $\epsilon = \|\hat{\mathbf{a}}^* - \mathbf{a}^*\|_2$ .
  - Update  $\mathbf{a}^*$  by  $\mathbf{a}^* = \hat{\mathbf{a}}^*$ .

End

3. The stationary reputation distribution is  $\mathbf{x}^*$  and the optimal action is  $\mathbf{a}^*$ .
-

## Chapter 6

### Image Denoising Games

During the processes of being captured, digitized, recorded, and transmitted, an image is usually distorted and noisy. Such a noisy image is visually annoying and often not suited to further perform tasks such as segmentation, recognition and compression. Therefore, image denoising is a very important issue to reconstruct a good estimate of the original image from the noisy observations.

Many approaches have been proposed in the literature to reconstruct the original image by exploiting the inherently spatial correlation. By assuming that the image locally satisfies a stationary Gaussian process, Woods and Radewan [126] propose to estimate the original image from the noisy image using Kalman filter while Jin et al [70] propose to use adaptive Wiener filter. In both approaches, the first-order and second-order statistics used in the filters are calculated based on the noisy samples within a local window. In [111] [51] [117], the authors propose to use bilateral filtering over the local neighborhood samples, where the weights of the bilateral filters are computed based on the intensity and radiometric distances between the center sample and the neighboring samples. Another class of locally adaptive image denoising approaches are derived by considering image processing as a variational problem where the restored image is computed by minimizing a carefully designed energy function [26] [55] [56]. Typically, such energy functions consist

of a fidelity term that is determined by the difference between the reconstructed image and the noisy image, and a regularization penalty term that is determined by the image prior.

To further exploit the spatial correlation, Buades et al [20] proposed to average, in a weighted manner, all the pixels in a nonlocal window instead of only involving the locally neighboring pixels, where the weights are determined by the differences between the region centered by the target pixel and the regions centered by the candidate pixels. Since the weights are not determined by the radiometric (physical) distance, similar pixels that are far away from the target pixel can still be awarded large weights. In such a way, the denoising performance is greatly improved. Several extensions of the nonlocal approach are also proposed [21] [73] [74].

Besides the pixel-domain approaches, transform-domain approaches are also investigated [50] [99] [58] [66]. The transform-domain approaches are mainly based on the assumption that the original signal can be well approximated by a linear combination of few basis, i.e., the original signal is sparse in the transform-domain. In such a case, the original signal can be well estimated by preserving the few high-magnitude transform coefficients that convey mostly the energy of the original signal and discarding the rest which are mainly introduced by the noise. Therefore, one important issue in the transform-domain approaches is how to threshold the transform coefficients. Many threshold rules have been proposed from different speculations [50] [11] [52]. A combination of the nonlocal and transform-domain thresholding ideas is proposed in [45]. The basic idea is to first group similar 2D image blocks into 3D data arrays, then perform 3D wavelet transform, and finally

shrinkage the transform spectrum.

Most of the existing schemes focus on how to choose good weights for some given neighborhoods to achieve better reconstructions. However, how to adaptively choose optimal neighborhoods can be even more important since too large a neighborhood set may cause overly-smooth artifacts, while too small a neighborhood set may not be able to efficiently reduce noise variance. Due to the absence of the original image, the Stein's principle [113] is used to estimate the true MSE for determining the optimal neighborhoods. Nevertheless, we find that there exists a trade-off between the accuracy of the estimate and the minimum of the true MSE. In this chapter, we study the impact of this trade-off and formulate the image denoising problem as a coalition formation game. In this game, every pixel is treated as a player, who tries to seek partners to form a coalition to improve the accuracy of the Stein's estimate while incurring a cost of increasing the minimum of the true MSE. Since finding the optimal coalition structures is NP-hard, we propose a heuristically distributed algorithm in solving the coalition formation game. We also show that the traditional approaches that use a heuristically determined candidate set are special cases of the proposed game theoretical framework by choosing the utility function without a cost term. Finally, experimental results show that the proposed game theoretical approach can achieve better performance than the non-local method in terms of both PSNR and visual quality. Note that the proposed game is also applicable in other scenarios besides the nonlocal method as long as 1) there exist some locally adaptive parameters to be estimated, and 2) the estimation accuracy will be improved when more samples are involved in the estimate process.

The rest of this chapter is organized as follows. In Section 6.1, we give a brief description of the system model and the coalition formation game. Then, we discuss how to choose a good candidate set in Section 6.2. In Section 6.3, we study the trade-off between the accuracy of the estimate and the minimum of the true MSE and provide a detailed analysis of the proposed game-theoretic image denoising framework. In Section 6.4, we show the relationship between the proposed game-theoretic framework and the traditional candidate set selection approaches. Finally, we illustrate the experimental results on real images in Section 6.5 and draw conclusions in Section 6.6.

## 6.1 The System Model and Coalition Formation Game

### 6.1.1 The System Model

In this chapter, we consider the problem of restoring images degraded by additive white Gaussian noise. The degraded process can be modelled as

$$I^n(\mathbf{k}) = I(\mathbf{k}) + n(\mathbf{k}), \quad (6.1)$$

where  $I$  is the original image,  $I^n$  is the noisy observation of the image, and  $n$  is the additive Gaussian noise with zero mean and  $\sigma^2$  noise variance. The  $\mathbf{k} = (k_1, k_2)$  is the coordinate of a pixel. The problem is to find an estimate  $\hat{I}$  of the original image based on the noisy observation  $I^n$ .

It is well known that the image denoising problem is ill-posed. To reconstruct the original image from the noisy observation, we need to use some prior information

such as the correlations among spatial neighboring pixels. In this chapter, we focus on the spatially adaptive linear filtering approach. For the pixel located at  $\mathbf{k}$ , we find the estimate  $\hat{I}(\mathbf{k})$  using the weighted average of the spatially neighboring pixels, i.e.,

$$\hat{I}(\mathbf{k}) = \frac{\sum_{\mathbf{l} \in S(\mathbf{k})} w_{\mathbf{k},\mathbf{l}} I^n(\mathbf{l})}{\sum_{\mathbf{l} \in S(\mathbf{k})} w_{\mathbf{k},\mathbf{l}}}, \quad (6.2)$$

where  $S(\mathbf{k})$  is the candidate set that contains the spatially neighboring pixels for  $\mathbf{k}$ , and  $w_{\mathbf{k},\mathbf{l}}$  is the weight for pixel  $I^n(\mathbf{l})$ .

### 6.1.2 The Coalition Formation Game

Game theory is a mathematical tool that analyzes the strategic interactions among multiple decision makers. A game is mainly composed by three components:

- a finite set of players, denoted by  $u_1, u_2, \dots, u_N$ ;
- a set of actions, denoted by  $A_i$ , for each player  $u_i$ ;
- payoff/utility function, denoted by  $U_i$ , which measures the outcome for player  $u_i$  determined by the actions of all players.

A coalition formation game is a game where the players seek to form cooperative groups, i.e., coalitions, to strengthen their positions in the game. The players' actions in the coalition formation game are whom to cooperate with, i.e., which players to form coalitions with. The payoff/utility function in the coalition formation game is defined over coalitions, called coalition value. The coalition value, which quantifies the worth of a coalition, is mainly determined by two terms: the

gain and the cost. By forming a coalition, every player in the coalition can obtain a gain through cooperation within the coalition. However, the gain is limited by a cooperation cost for forming the coalition, e.g. the negotiation cost or information exchange cost.

Given the player set and the coalition value, the coalition formation game is uniquely defined, and the outcome of the game is a set of coalitions, which is the optimal partitions of the player set. To obtain the optimal partitions, there are two possible approaches: centralized approach and distributed approach. For the centralized approach, the centralized controller needs to search over all the partitions of the player set to find the optimal partitions, which is NP-complete and impractical when the size of the player set is large [104]. For the distributed approach, the players will make their own decisions as to whether or not they join a coalition. One typical approach is to use the merge and split rules proposed in [15]. This approach starts with an initial partition and repeats alternatively the merge and split rule, 1) merge rule: merge any set of coalitions into a single coalition if the new coalition can provide larger total coalition values; 2) split rule: split a coalition into smaller coalitions if the resulting smaller coalitions can provide larger total coalition values.

In Section 6.3, we will discuss in details how to use the coalition formation game to formulate the image denoising problem, where each pixel will be treated as a player seeking to form coalitions to achieve optimal denoising performance.

## 6.2 Candidate Set Selection

From (6.2), we can see that the reconstruction performance are determined by the selection of the weights  $w_{\mathbf{k},\mathbf{l}}$  and the candidate set  $S(\mathbf{k})$ . For any given  $S(\mathbf{k})$ , the optimal weights  $w_{\mathbf{k},\mathbf{l}}^*(S(\mathbf{k}))$  are determined by the correlation between pixels  $I(\mathbf{k})$  and  $I(\mathbf{l})$ , and should be chosen to minimize the difference between the estimation  $\hat{I}(\mathbf{k})$  and the original pixel  $I(\mathbf{k})$  as below.

$$w_{\mathbf{k},\mathbf{l}}^*(S(\mathbf{k})) = \arg \min_{w_{\mathbf{k},\mathbf{l}}} \left( \frac{\sum_{\mathbf{l} \in S(\mathbf{k})} w_{\mathbf{k},\mathbf{l}} I^n(\mathbf{l})}{\sum_{\mathbf{l} \in S(\mathbf{k})} w_{\mathbf{k},\mathbf{l}}} - I(\mathbf{k}) \right)^2 \quad (6.3)$$

Note that when the optimal weights in (6.3) are used, the selection of the candidate set  $S(\mathbf{k})$  is trivial since the accuracy of the reconstruction improves as the candidate set  $S(\mathbf{k})$  becomes larger. The proof can be found in Theorem 1.

**Theorem 1:** When the optimal weights in (6.3) are used, the accuracy of the reconstruction improves as the candidate set  $S(\mathbf{k})$  becomes larger. *Proof:* Let  $S_1(\mathbf{k})$  and  $S_2(\mathbf{k})$  be two candidate sets with  $S_1(\mathbf{k}) \subset S_2(\mathbf{k})$ . Let  $w_{\mathbf{k},\mathbf{l}}^*(S_1(\mathbf{k}))$  and  $w_{\mathbf{k},\mathbf{l}}^*(S_2(\mathbf{k}))$  be the corresponding optimal weights computed by (6.3). Suppose  $\tilde{w}_{\mathbf{k},\mathbf{l}}(S_2(\mathbf{k}))$  are the weights for  $S_2(\mathbf{k})$  and are defined as follows

$$\tilde{w}_{\mathbf{k},\mathbf{l}}(S_2(\mathbf{k})) = \begin{cases} w_{\mathbf{k},\mathbf{l}}^*(S_1(\mathbf{k})), & \mathbf{l} \in S_1(\mathbf{k}); \\ 0, & \text{else.} \end{cases} \quad (6.4)$$

Then, according to the optimality of  $w_{\mathbf{k},\mathbf{l}}^*(S_2(\mathbf{k}))$ , we have

$$\begin{aligned} & \left( \frac{\sum_{\mathbf{l} \in S_2(\mathbf{k})} w_{\mathbf{k},\mathbf{l}}(S_2(\mathbf{k})) I^n(\mathbf{l})}{\sum_{\mathbf{l} \in S_2(\mathbf{k})} w_{\mathbf{k},\mathbf{l}}(S_2(\mathbf{k}))} - I(\mathbf{k}) \right)^2 \\ & \leq \left( \frac{\sum_{\mathbf{l} \in S_2(\mathbf{k})} \tilde{w}_{\mathbf{k},\mathbf{l}}(S_2(\mathbf{k})) I^n(\mathbf{l})}{\sum_{\mathbf{l} \in S_2(\mathbf{k})} \tilde{w}_{\mathbf{k},\mathbf{l}}(S_2(\mathbf{k}))} - I(\mathbf{k}) \right)^2 \\ & = \left( \frac{\sum_{\mathbf{l} \in S_1(\mathbf{k})} w_{\mathbf{k},\mathbf{l}}(S_1(\mathbf{k})) I^n(\mathbf{l})}{\sum_{\mathbf{l} \in S_1(\mathbf{k})} w_{\mathbf{k},\mathbf{l}}(S_1(\mathbf{k}))} - I(\mathbf{k}) \right)^2. \end{aligned} \quad (6.5)$$

Therefore, when optimal weights are used, the reconstruction using candidate set  $S_2(\mathbf{k})$  is more accurate than that using candidate set  $S_1(\mathbf{k})$ .  $\blacksquare$

However, due to the absence of the original pixel  $I(\mathbf{k})$ , it is impossible for us to find the optimal weights using (6.3). One possible approximation is to use the similarity between the neighborhoods around  $\mathbf{k}$  and  $\mathbf{l}$  [20], which is defined as follows

$$w_{\mathbf{k},\mathbf{l}} = \exp \left\{ -\frac{\sum_{\mathbf{b} \in B} [I^n(\mathbf{k} + \mathbf{b}) - I^n(\mathbf{l} + \mathbf{b})]^2}{h^2} \right\}, \quad (6.6)$$

where  $B$  is a predefined neighborhood and  $h$  is the parameter related to the noise's variance.

Nevertheless, since the weights in (6.6) are not optimal, the selection of the candidate set  $S(\mathbf{k})$  for the reconstruction becomes critically important. On one hand, if the size of the candidate set is too small, then the noise may not be effectively removed. On the other hand, if the size of the candidate set is too large, then the reconstruction may be overly-smooth. Moreover, according to (6.6), we can see that the pixels that are more similar to the target pixel would have larger weights. To prevent the reconstruction from being overly-smooth, we will only involve the pixels that have relatively large weights. Let  $\Omega(m)$  stand for the subset of  $S(\mathbf{k})$  which contains the pixels with the first  $m$  largest weights. Then, the reconstruction  $\hat{I}(\mathbf{k}, m)$  using  $\Omega(m)$  can be written as

$$\hat{I}(\mathbf{k}, m) = \frac{\sum_{\mathbf{l} \in \Omega(m)} w_{\mathbf{k},\mathbf{l}} I^n(\mathbf{l})}{\sum_{\mathbf{l} \in \Omega(m)} w_{\mathbf{k},\mathbf{l}}}. \quad (6.7)$$

Obviously, the parameter  $m$  in (6.7) should be chosen in such a way that the difference between  $\hat{I}(\mathbf{k}, m)$  and  $I(\mathbf{k})$  is minimized, i.e., the optimal  $m^*$  can be found

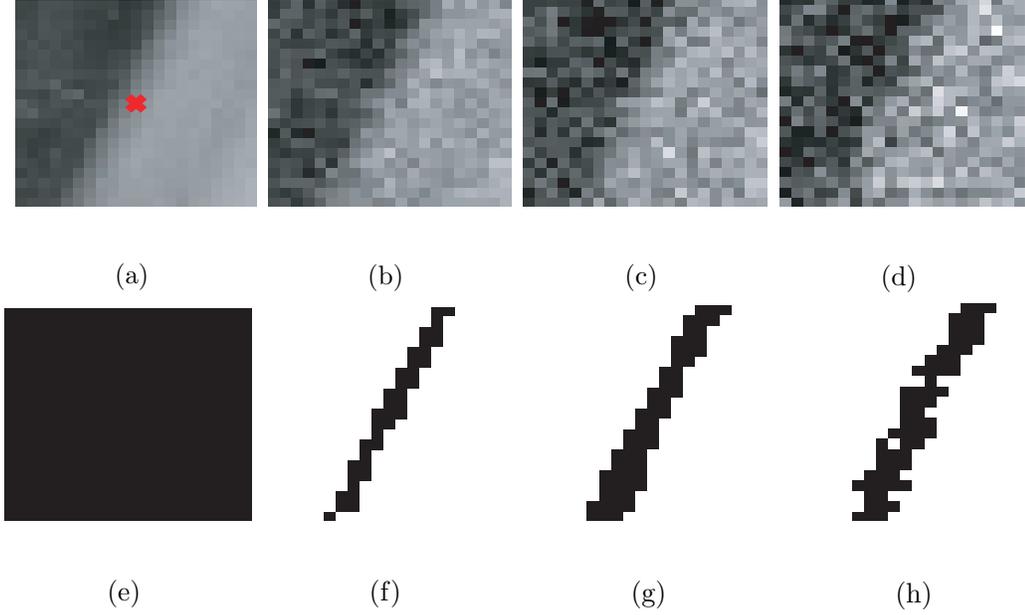


Figure 6.1: An example of optimal candidate set with an edge region: (a) original image; (b) noisy image with  $\sigma = 15$ ; (c) noisy image with  $\sigma = 25$ ; (d) noisy image with  $\sigma = 35$ ; (e) the candidate set used by nonlocal; (f) the ideally optimal candidate set of (b) when the original signal is available; (g) the ideally optimal candidate set of (c) when the original signal is available; (h) the ideally optimal candidate set of (d) when the original signal is available.

by

$$m^* = \arg \min_m |\hat{I}(\mathbf{k}, m) - I(\mathbf{k})|^2. \quad (6.8)$$

In general,  $m^*$  is content dependent, i.e.,  $m^*$  may be different for different  $\mathbf{k}$  and/or different noise variances. Even with the same  $m^*$ , the structure of the candidate set  $\Omega(m^*)$  may be different for different pixels. In Figs. 6.1 and 6.2, we show the structure of the optimal candidate set for two different scenarios: 1) the target pixel is centered within an edge region, and 2) the target pixel is centered within a smooth region. For illustration purpose, we assume that the original image

is available for finding  $m^*$  in these two examples. Later in Section 6.3, we will discuss how to find  $m^*$  using game theory under the scenario that the original image is not available. As shown in Figs. 6.1 and 6.2, (a) is the original image centered by the target pixel, which is denoted by red “x”, (b)-(d) are the noisy images with  $\sigma$  being 15, 25, and 35 respectively. (e) is the candidate set using in [20], which is a square window. (f)-(h) are the optimal candidate sets generated using (6.8) for (b)-(d) respectively. Note that the black pixels in (f)-(h) stand for the pixels in the candidate set. From Figs. 6.1 and 6.2, we can see that for the scenario where the target pixel is centered within an edge region, the candidate set has an edge structure, while for the scenario where the target pixel is centered within a smooth region, the structure of the candidate set is unpredictable. Moreover, we can also see that with different noise variance, the candidate sets are quite different. Therefore, the candidate set should not be pre-defined in a fixed way such as using a square window in [20]. Instead, the candidate set should be chosen adaptively to minimize the difference between the estimate and the original signal.

## 6.3 Game Theoretical Problem Formulation

### 6.3.1 Stein’s unbiased risk estimate (SURE)

Since  $I(\mathbf{k})$  is unknown, the optimal  $m^*$  can not be explicitly computed using (6.5). Fortunately, we can first use the Stein’s unbiased risk estimate (SURE) [113] to estimate the true mean squared error (MSE) from the noisy observation and then use the estimated MSE to find the optimal  $m^*$ .

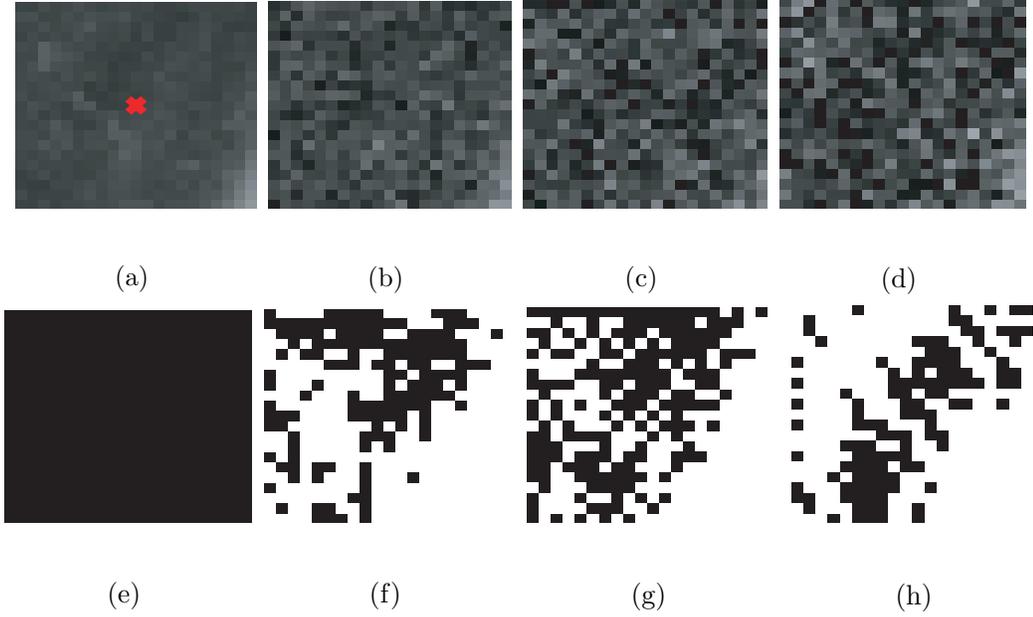


Figure 6.2: An example of optimal candidate set with a smooth region: (a) original image; (b) noisy image with  $\sigma = 15$ ; (c) noisy image with  $\sigma = 25$ ; (d) noisy image with  $\sigma = 35$ ; (e) the candidate set used by nonlocal; (f) the ideally optimal candidate set of (b) when the original signal is available; (g) the ideally optimal candidate set of (c) when the original signal is available; (h) the ideally optimal candidate set of (d) when the original signal is available.

In Fig. 6.3, we show the optimal  $m^*$  obtained using (6.8) for lena image when the standard deviation of the noise is  $\sigma = 10$ , where the intensity stands for the optimal  $m^*$  value. From Fig. 6.3, we can see that there are many pixels have the similar  $m^*$  value, which can be grouped together for finding  $m^*$ . For example, the pixels in the red circles have the  $m^*$  value near 60 can be grouped together. Suppose that the whole image is partitioned into  $M$  subsets  $\Phi = \{\Phi_1, \Phi_2, \dots, \Phi_M\}$ , where each subset  $\Phi_i$  contains a set of pixels that may not be physical neighboring but have the

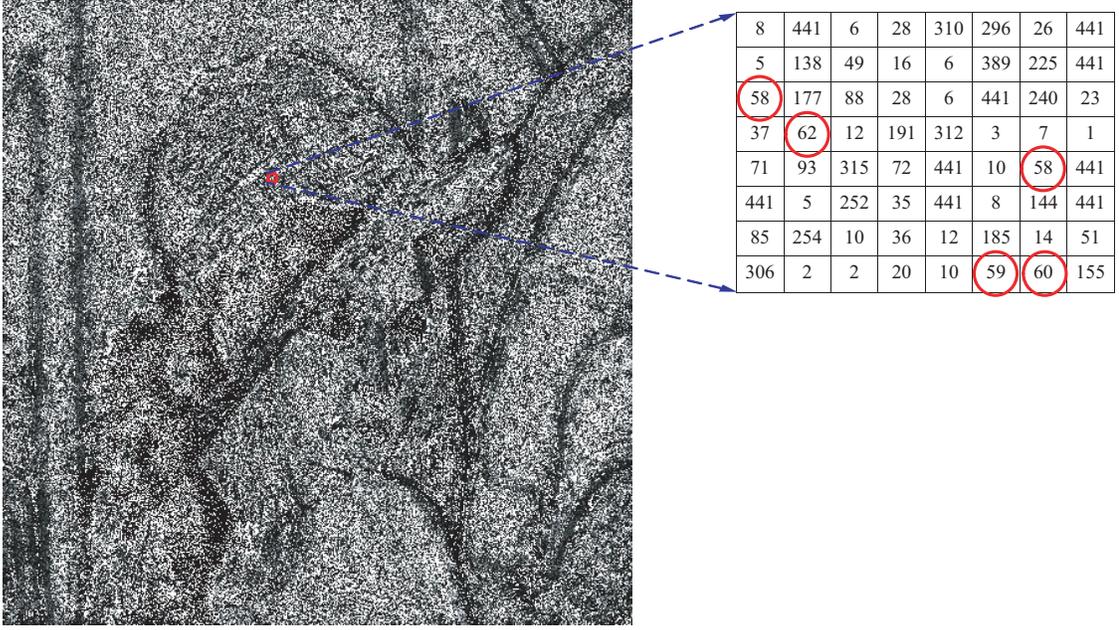


Figure 6.3: The optimal  $m^*$  for lena image when  $\sigma = 10$ .

same optimal parameter  $m_i^*$ , i.e.,

$$m_i^* = \arg \min_m \sum_{\mathbf{k} \in \Phi_i} |\hat{I}(\mathbf{k}, m) - I(\mathbf{k})|^2. \quad (6.9)$$

With the optimal  $m_i^*$ , the mean square error (MSE) for the subset  $\Phi_i$  can be computed by

$$mse_i = \frac{1}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} |\hat{I}(\mathbf{k}, m_i^*) - I(\mathbf{k})|^2, \quad (6.10)$$

and such a MSE can be approximated using SURE, according to Theorem 2, as follows

$$SURE_i = \frac{1}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} |\hat{I}(\mathbf{k}, m_i^*) - I^n(\mathbf{k})|^2 + \sigma^2 \left( \frac{2}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} \frac{\partial \hat{I}(\mathbf{k}, m_i^*)}{\partial I^n(\mathbf{k})} - 1 \right), \quad (6.11)$$

where  $\frac{\partial \hat{I}(\mathbf{k}, m_i^*)}{\partial I^n(\mathbf{k})}$  can be found by

$$\begin{aligned} \frac{\partial \hat{I}(\mathbf{k}, m_i^*)}{\partial I^n(\mathbf{k})} &= \frac{1}{\sum_{\mathbf{l} \in \Omega(m_i^*)} w_{\mathbf{k}, \mathbf{l}}} \times \\ &\left( \sum_{\mathbf{l} \in \Omega(m_i^*)} \frac{\partial w_{\mathbf{k}, \mathbf{l}}}{\partial I^n(\mathbf{k})} I^n(\mathbf{l}) + 1 - \hat{I}(\mathbf{k}, m_i^*) \sum_{\mathbf{l} \in \Omega(m_i^*)} \frac{\partial w_{\mathbf{k}, \mathbf{l}}}{\partial I^n(\mathbf{k})} \right), \end{aligned} \quad (6.12)$$

with  $\frac{\partial w_{\mathbf{k}, \mathbf{l}}}{\partial I^n(\mathbf{k})}$  being defined as

$$\frac{\partial w_{\mathbf{k}, \mathbf{l}}}{\partial I^n(\mathbf{k})} = \begin{cases} 2w_{\mathbf{k}, \mathbf{l}} \frac{I^n(\mathbf{l}) - I^n(\mathbf{k})}{h^2}, & \mathbf{l} - \mathbf{k} \notin B; \\ 2w_{\mathbf{k}, \mathbf{l}} \frac{I^n(\mathbf{l}) + I^n(2\mathbf{k} - \mathbf{l}) - 2I^n(\mathbf{k})}{h^2}, & \mathbf{l} - \mathbf{k} \in B. \end{cases} \quad (6.13)$$

**Theorem 2:** The  $SURE_i$  in (6.11) is an unbiased estimator of the true MSE  $mse_i$  in (6.10), i.e.,

$$E[SURE_i] = E[mse_i]. \quad (6.14)$$

*Proof:* By substituting  $I(\mathbf{k})$  with  $I^n(\mathbf{k}) - n(\mathbf{k})$ , we can re-write  $E[mse_i]$  as follows

$$\begin{aligned} E[mse_i] &= E \left[ \frac{1}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} |\hat{I}(\mathbf{k}, m_i^*) - I(\mathbf{k})|^2 \right] \\ &= E \left[ \frac{1}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} \left( |\hat{I}(\mathbf{k}, m_i^*) - I^n(\mathbf{k})|^2 \right. \right. \\ &\quad \left. \left. + 2n(\mathbf{k})\hat{I}(\mathbf{k}, m_i^*) - n(\mathbf{k})^2 \right) \right] \\ &= \frac{1}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} E \left[ |\hat{I}(\mathbf{k}, m_i^*) - I^n(\mathbf{k})|^2 \right] \\ &\quad + 2 \frac{1}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} E \left[ n(\mathbf{k})\hat{I}(\mathbf{k}, m_i^*) \right] - \sigma^2. \end{aligned} \quad (6.15)$$

According to Stein's Lemma [113], we have

$$E \left[ n(\mathbf{k})\hat{I}(\mathbf{k}, m_i^*) \right] = \sigma^2 E \left[ \frac{\partial \hat{I}(\mathbf{k}, m_i^*)}{\partial I^n(\mathbf{k})} \right]. \quad (6.16)$$

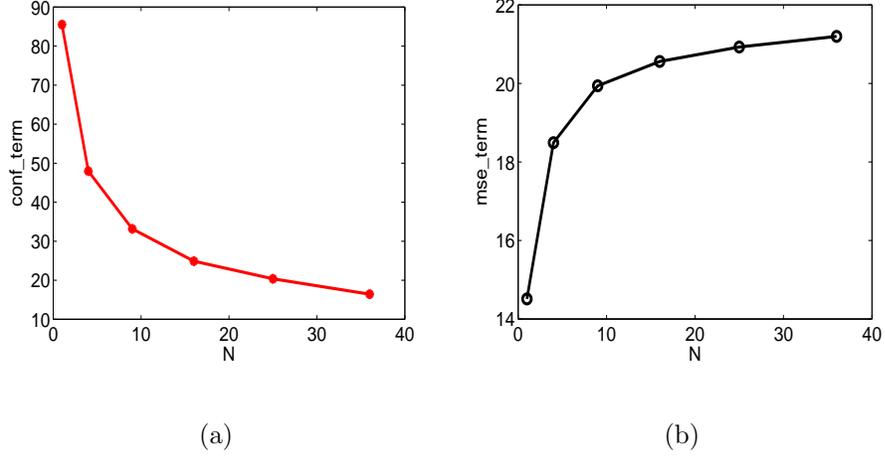


Figure 6.4: The trade-off between the confidence term  $C$  and the distortion term  $D$ : (a) the performance of  $C$  with different  $N$ ; (b) the performance of  $D$  with different  $N$ .

Then, by substituting (6.19) back to (6.18), we have

$$\begin{aligned}
 E[mse_i] &= E \left[ \frac{1}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} |\hat{I}(\mathbf{k}, m_i^*) - I^n(\mathbf{k})|^2 \right. \\
 &\quad \left. + \sigma^2 \left( \frac{2}{|\Phi_i|} \sum_{\mathbf{k} \in \Phi_i} \frac{\partial \hat{I}(\mathbf{k}, m_i^*)}{\partial I^n(\mathbf{k})} - 1 \right) \right] \\
 &= E[SURE_i].
 \end{aligned}$$

■

## 6.3.2 Confidence and Distortion Trade-off

### 6.3.2.1 Confidence

From Theorem 2, we can see that  $SURE_i$  is an unbiased estimator of  $mse_i$ . However, there can be some mismatch between  $SURE_i$  and  $mse_i$  for each realization (noise observation), i.e.,  $SURE_i$  is just an approximation of  $mse_i$ . To measure the

accuracy of the approximation, let us define the confidence term,  $C$ , as the average difference between  $SURE_i$  and  $mse_i$  over the whole image

$$C = \frac{1}{|\Phi|} \sum_{i=1}^M |\Phi_i| \times |mse_i - SURE_i|. \quad (6.17)$$

According to [113], the estimator  $SURE_i$  becomes closer to  $mse_i$  as  $|\Phi_i|$  increases, which means that the confidence term  $C$  in (6.17) decreases as  $|\Phi_i|$  increases.

### 6.3.2.2 Distortion

With the partition  $\Phi = \{\Phi_1, \Phi_2, \dots, \Phi_M\}$  and the optimal parameters  $\{m_1^*, m_2^*, \dots, m_M^*\}$ , we can compute the mean square error for the whole image,  $D$ , as follows

$$D = \frac{1}{|\Phi|} \sum_{i=1}^M |\Phi_i| \times mse_i. \quad (6.18)$$

According to the analysis in Section 6.3.1, we group the pixels with similar  $m^*$  values together and assign a common  $m_i^*$  to all pixels in subset  $\Phi_i$ . In such a case, as  $|\Phi_i|$  increases, the probability that the pixels in  $\Phi_i$  have different true  $m^*$  values increases, which leads to the increase of  $mse_i$ . Therefore, the distortion term  $D$  in (6.18) increases as  $|\Phi_i|$  increases.

### 6.3.2.3 Confidence and Distortion Trade-off

From the above discussion, we can see that as  $|\Phi_i|$  increases, the confidence term  $C$  decreases but the distortion term  $D$  increases. Therefore, there exists a trade-off between  $C$  and  $D$ . To verify such a trade-off, we conduct a simple experiment by setting  $|\Phi_i| = N, \forall i$ . As shown in Figure 6.4, the confidence term  $C$

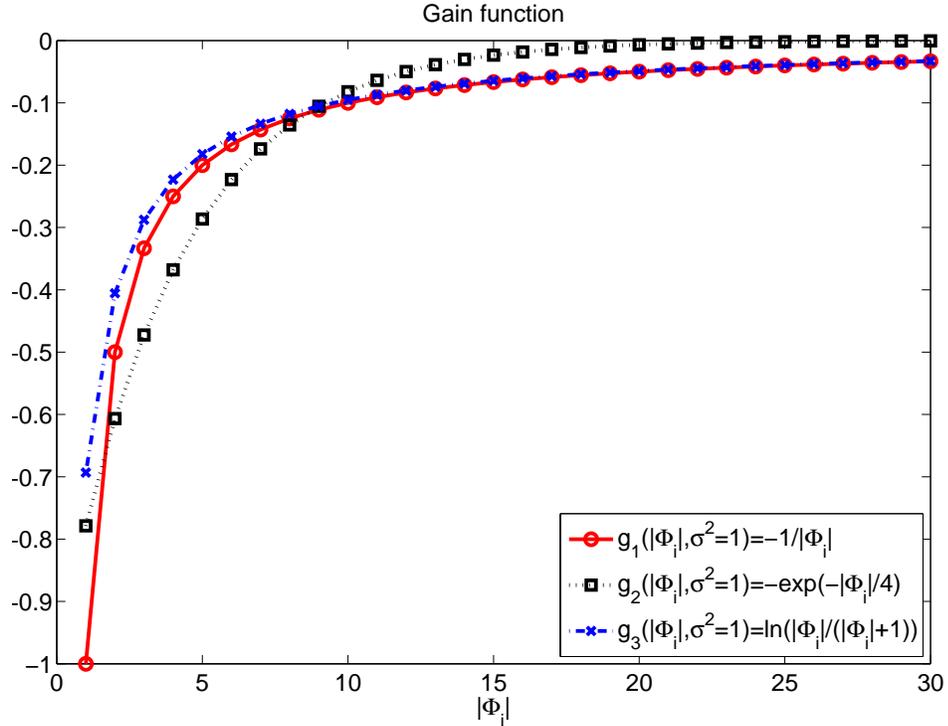


Figure 6.5: Some possible gain functions.

decrease as  $N$  increases while the distortion term  $D$  increases as  $N$  increases, which are consistent with our analysis.

### 6.3.3 Utility Function and Solution to the Game

From the previous subsections, we can see that given the partition  $\Phi = \{\Phi_1, \Phi_2, \dots, \Phi_M\}$ , SURE can be used to approximate the true MSE to find the optimal  $m^*$ . However, how to find a good partition is not trivial since the number of the partition is not fixed and the size of each partition can vary. Due to the uncertainty of the number of the partition, the traditional segmentation and clustering methods may not work. To study the complex interactions among different pixels and



(a)



(b)



(c)



(d)

Figure 6.6: The four tested images: (a) Lena; (b) Barbara; (c) Boat; (d) Flinstones.

the dynamic partition formation process, we propose to use the coalition formation game.

In this game theoretical formulation, every pixel is treated as a player, who tries to seek partners to form coalitions to achieve better reconstruction. By forming a coalition, every player in the coalition can obtain a gain of reducing the difference between the SURE and the true estimate, i.e., the confidence term in (6.17), while incurring a cost of increasing the minimum of the MSE. With this idea in mind, we

define the utility for a coalition as:

$$U(\Phi_i) = -|\Phi_i| \times SURE_i + g(|\Phi_i|, \sigma^2), \quad (6.19)$$

where the first term of the right hand side is the cost and the second term  $g(|\Phi_i|, \sigma^2)$  is the gain.

The function  $g(|\Phi_i|, \sigma^2)$  in (6.19) characterizes the gain of forming a coalition, which is the reduction of the difference between the SURE and the true estimate due to the increase of the coalition size. Therefore,  $g(|\Phi_i|, \sigma^2)$  should satisfy the following properties

1.  $g(|\Phi_i|, \sigma^2)$  should be an increasing function in terms of  $|\Phi_i|$  since the gain increases as the coalition size  $|\Phi_i|$  increases, i.e.,  $\frac{\partial g(|\Phi_i|, \sigma^2)}{\partial |\Phi_i|} > 0$ .
2.  $g(|\Phi_i|, \sigma^2)$  should be a concave function in terms of  $|\Phi_i|$  since a certain increase of the coalition size in the low  $|\Phi_i|$  region should lead to a more significant gain than that in the high  $|\Phi_i|$  region, i.e.,  $\frac{\partial^2 g(|\Phi_i|, \sigma^2)}{\partial |\Phi_i|^2} < 0$ .
3.  $g(|\Phi_i|, \sigma^2)$  should be a superadditive function since the gain of a large coalition should be no smaller than that of two sub coalitions, i.e.,  $g(|\Phi_i + \Phi_j|, \sigma^2) \geq g(|\Phi_i|, \sigma^2) + g(|\Phi_j|, \sigma^2)$ .
4.  $g(|\Phi_i|, \sigma^2)$  should be a decreasing function in terms of  $\sigma^2$  since the gain decreases as noise variance  $\sigma^2$  increases, i.e.,  $\frac{\partial g(|\Phi_i|, \sigma^2)}{\partial \sigma^2} < 0$ .

There are many functions that can satisfy the above property. In the following, we list three possible functions

$$g_1(|\Phi_i|, \sigma^2) = \lambda_1 \sigma^2 \left( \frac{-1}{|\Phi_i|} \right), \quad (6.20)$$

$$g_2(|\Phi_i|, \sigma^2) = \lambda_2 \sigma^2 \left[ -\exp\left(\frac{-|\Phi_i|}{4}\right) \right], \quad (6.21)$$

$$g_3(|\Phi_i|, \sigma^2) = \lambda_3 \sigma^2 \left[ \ln\left(\frac{|\Phi_i|}{|\Phi_i| + 1}\right) \right], \quad (6.22)$$

where  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$  are fixed parameters.

In Fig. 6.5, we plot the three possible gain functions versus  $|\Phi_i|$  by setting  $\sigma^2 = 1$ . We can see that all the three functions meet our requirements and are therefore valid gain functions. Moreover, we can see that all three functions behave similarly. Therefore, in this chapter, we only evaluate the first gain function, i.e.,  $g(|\Phi_i|, \sigma^2)$  in (6.19) is set to be  $g_1(|\Phi_i|, \sigma^2)$ . Nevertheless, similar results can be obtained with the other two functions ( $g_2(|\Phi_i|, \sigma^2)$  and  $g_3(|\Phi_i|, \sigma^2)$ ) and any other functions with similar properties.

With the utility function in (6.19), we can see that as the size of the coalition increases, the members in the coalition can obtain gains from  $g(|\Phi_i|, \sigma^2)$ . However, the gains are limited by the a cost of forming the coalition, which is  $-|\Phi_i| \times SURE_i$ . The problem now is to find the optimal coalition structures based on the utility function in (6.19). One possible approach is to use the merge and split rules proposed in [15], where the authors prove that their algorithm will converge to a unique solution with arbitrary merge and split iterations. However, the computation complexity is still very large since all possible sub-partitions need to be evaluated during the split process. To make the problem traceable, in this chapter, we propose a heuristic algorithm in solving the coalition formation game. As shown in Algorithm 6, the proposed heuristic algorithm starts with a randomly chosen pixel and finds the coalition by selecting the neighborhoods that can give best average utility. Then, all

the pixels in the coalition are denoised with the corresponding optimal  $m^*$ . Finally, all the pixels in the coalition are excluded from the un-denoised set. The above procedures are repeated until all pixels are denoised.

The proposed heuristic algorithm is distributive, and only locally neighboring information is required for finding the coalition. Moreover, since it is not an iterative algorithm, there is no convergence issue. Compared with the merge and split rules [15], the computation complexity is greatly reduced since the split process is avoided. From the experimental results shown in Section 6.5, we can see that the proposed heuristic algorithm performs quite well.

---

**Algorithm 6** A Heuristic Algorithm For Coalition Formation

---

Initialization: let the set of denoised pixel  $S_D = \emptyset$  and its complement  $\bar{S}_D = \Phi$ , let  $N_1 = 800$ ,

$N_2 = 21 \times 21$ , and  $i = 0$ .

While  $\bar{S}_D \neq \emptyset$

- $i = i + 1$
- randomly choose  $\mathbf{k} \in \bar{S}_D$ , let  $\Phi^0 = \{\mathbf{k}\}$  and set  $j = N_1$
- While  $j > 0$ 
  - $j = j - 1$
  - $(\mathbf{I}^*, m^*) = \arg \min_{\mathbf{I} \in \bar{S}_D \setminus \Phi^0, 1 \leq m \leq N_2} SURE(\Phi^0 \cup \{\mathbf{I}\}, m)$
  - set  $\Phi^0 = \Phi^0 \cup \{\mathbf{I}^*\}$
  - compute  $u(|\Phi^0|) = \frac{g(|\Phi^0|)}{|\Phi^0|} - SURE(\Phi^0, m^*)$

End

- let  $n_i^* = \arg \max_n u(n)$ ,  $\Phi_i = \Phi^0(1 : n_i^*)$
- compute  $m_i^* = \arg \min_{1 \leq m \leq N_2} SURE(\Phi_i, m)$
- set  $\bar{S}_D = \bar{S}_D \setminus \Phi_i$  and  $S_D = S_D \cup \Phi_i$
- denoise the pixel in  $\Phi_i$  using (6.4) with  $m = m_i^*$

End

---

## 6.4 Relation to the Traditional Approaches

In the traditional pixel-domain image denoising approaches, every pixel is denoised using (6.2) with a heuristically pre-defined candidate set  $S(\mathbf{k})$ . For example, a fixed-size square window centered by the target pixel  $\mathbf{k}$  is chosen as the candidate set in the nonlocal image denoising method [20]. Such kinds of approaches have a performance limitation due to the self-constrained use of a pre-defined candidate set. As shown in Figs. 6.1 and 6.2, we can see that the candidate set should be adaptively chosen for different neighborhoods and/or noise variances. Moreover, we will show in the following analysis that the traditional methods such as the nonlocal method [20] is actually a special case of the proposed game theoretical framework by choosing a utility function without a cost term

$$U(\Phi_i) = g(|\Phi_i|, \sigma^2). \quad (6.23)$$

According to the discussion in Section 6.3.3, we know that a valid gain function  $g(|\Phi_i|, \sigma^2)$  should be monotonically increasing, concave, and superadditive in terms of  $|\Phi_i|$ . In such a case, if the utility function only involves the gain function as in (6.23), then all pixels will form a grand coalition and use the same candidate set. In such a case, it return to the traditional ad-hoc approaches where a fixed candidate set is used for all pixels. In this sense, we can say that the traditional ad-hoc approaches are special cases of the proposed game theoretical framework.

## 6.5 Experimental Results

We evaluate the proposed game theoretical image denoising approach by comparing it with the nonlocal method [20]. Four  $512 \times 512$  images shown in Fig. 6.6: Lena, Barbara, Boat and Flinstones, are tested. The neighborhood  $B$  and the parameter  $h$  in (6.6) are set to be  $11 \times 11$  and  $10\sigma$  respectively. The candidate set for the nonlocal method is set to be a  $21 \times 21$  square window. The parameter  $\lambda_1$  for the proposed method in the gain function in (6.20) is set to be 0.875. Note that this parameter  $\lambda_1$  is fixed for all four tested images.

We first examine the candidate set generated by the proposed approach. In Figs. 6.1 and 6.2, we show the ideally optimal candidate sets of two different image patches by assuming the original signal is available. Obviously, due to the absence of the original signal, we are not able to get the ideally optimal candidate sets. Nevertheless, with the proposed game theoretical approach and the SURE estimate, we can find approximate candidate sets and the results are shown in Figs. 6.7 and 6.8. From Figs. 6.7 and 6.8, we can see that the approximate candidate sets are much more similar to the ideally optimal candidate sets compared with the fixed square window candidate set used by [20].

Then, we evaluate the PSNR comparison versus the standard derivation of the noise. The PSNR comparison between the nonlocal method and the proposed method for the tested images at different noise levels are shown in Fig. 6.9. From Fig. 6.9, we can see that the proposed method always performs better than, if not equal to, the nonlocal method for all tested images at all different noise variances.

When the standard deviation of the noise is no larger than 20, i.e.  $\sigma \leq 20$ , the PSNR performance of the proposed approach is just a bit better than the nonlocal method. This is because when  $\sigma$  is small, the weights of the pixels outside the optimal candidate set are too small. In such a case, the reconstruction using a heuristically determined square window candidate set is similar to that using an optimal candidate set, i.e., the performance of the nonlocal method is similar to the performance of the proposed approach when  $\sigma$  is small. However, we should notice that the performance of the nonlocal method is always upper bounded by that of the proposed method since the adaptively chosen candidate set is always better than, if not equal to, the fixed square window candidate set.

When  $\sigma$  becomes larger, the superiority of the proposed approach becomes more significant. This phenomenon is mainly because when  $\sigma$  is large, the weights of the pixels outside the optimal candidate is relatively large and is no longer negligible. In such a case, the reconstruction using all pixels in a heuristically determined square window tends to lead to over-smooth artifacts. On the other hand, since the candidate set is adaptively chosen for every pixel in the proposed method, we are able to preserve details of the original image and avoid over-smooth artifacts. From Fig. 6.9, we can see that the gain of the proposed approach over the nonlocal method can be up to 2.16dB for the Flinestones image when  $\sigma = 60$ , which fully demonstrates the effectiveness of the proposed approach.

Finally, we evaluate the visual quality of the reconstructions. In Fig. 6.10, we show the visual quality comparison for Flinestones. As shown in Fig. 6.10, (a) is the original patch of Flinestones and (b) is the noisy patch with  $\sigma = 25$ . The results

generated by the nonlocal method and the proposed approach are shown in (c) and (d) respectively. We can see that the result generated by the nonlocal method is over-smooth. This phenomenon is because the nonlocal method involves too many dis-similar pixels in the averaging process. With the proposed approach, every pixel (player) seeks partners to form coalition to determine the best number of neighborhoods to perform denoising, which can rule out the dis-similar neighborhoods and avoid over-smooth artifacts. Therefore, the details can be well-preserved in the proposed approach. Similar phenomenons can be observed in Figs. 6.11, 6.12 and 6.13 for Barbara, Lena and Boat at noise level  $\sigma = 35$ ,  $\sigma = 45$ , and  $\sigma = 20$  respectively. Due to the page limitation, we only show the results of one  $\sigma$  for each image in this chapter. Similar results are observed for different  $\sigma$ 's.

## 6.6 Summary

In this chapter, we study the trade-off between the accuracy of the Stein's estimate and the minimum of the true MSE and formulate the image denoising problem as a coalition formation game. With the proposed game, every player (pixel) seek partners to form coalitions to obtain better decision for the optimal neighborhoods selection and thus lead to better denoising results. The experimental results show that compared with the nonlocal method [20], the proposed game theoretical approach can achieve not only better PSNR performance but also better visual quality. Note that the proposed game is also applicable in other scenarios besides the nonlocal method as long as 1) there exist some locally adaptive parameters to be estimated,

and 2) the estimation accuracy will be improved when more samples are involved in the estimate process. Moreover, we showed that the traditional approaches using a heuristically determined candidate set are special cases of the game theoretical framework by choosing the utility function without a cost term.

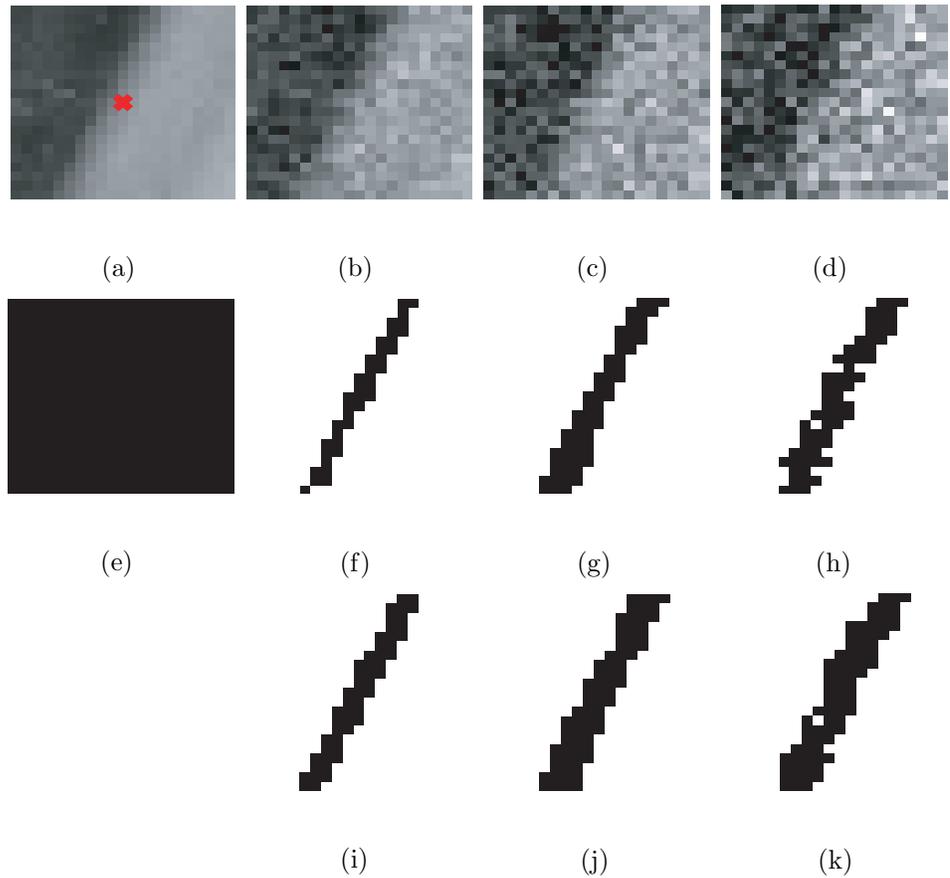


Figure 6.7: An example of optimal candidate set with an edge region: (a) original image; (b) noisy image with  $\sigma = 15$ ; (c) noisy image with  $\sigma = 25$ ; (d) noisy image with  $\sigma = 35$ ; (e) the candidate set used by nonlocal; (f) (g) and (h) are the ideally optimal candidate sets of (b) (c) and (d) when the original signal is available; (i) (j) and (k) are the candidate sets of (b) (c) and (d) generated by the proposed method.

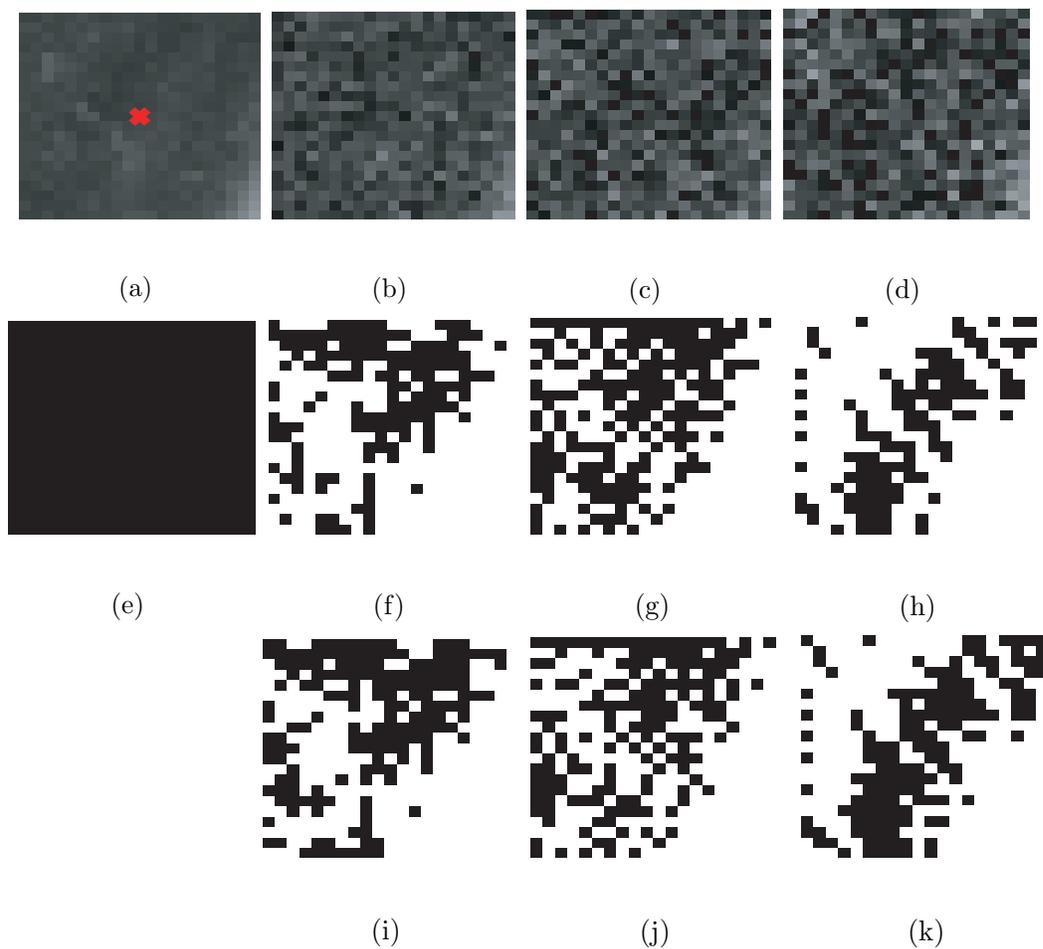
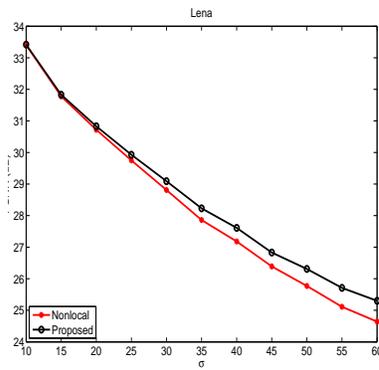
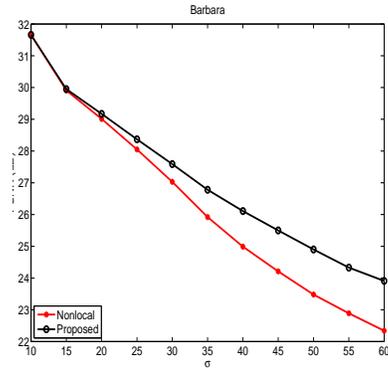


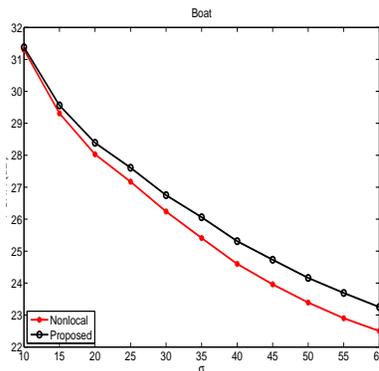
Figure 6.8: An example of optimal candidate set with a smooth region: (a) original image; (b) noisy image with  $\sigma = 15$ ; (c) noisy image with  $\sigma = 25$ ; (d) noisy image with  $\sigma = 35$ ; (e) the candidate set used by nonlocal; (f) (g) and (h) are the ideally optimal candidate sets of (b) (c) and (d) when the original signal is available; (i) (j) and (k) are the candidate sets of (b) (c) and (d) generated by the proposed method.



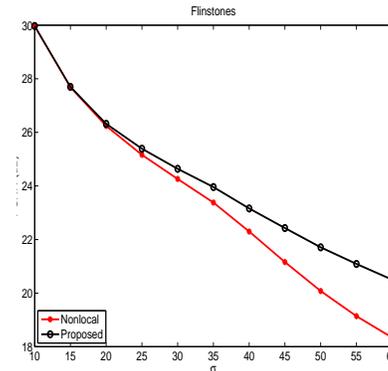
(a)



(b)



(c)



(d)

Figure 6.9: The PSNR comparison for different images: (a) Lena; (b) Barbara; (c) Boat; (d) Flinstones.

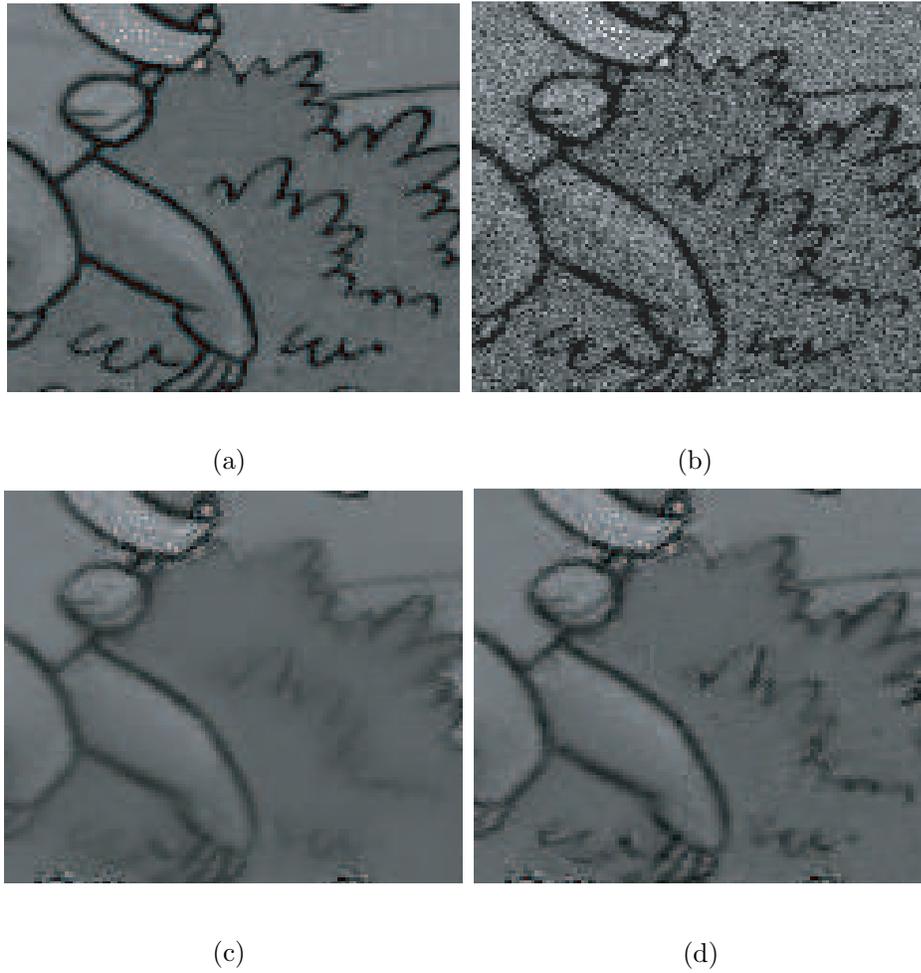
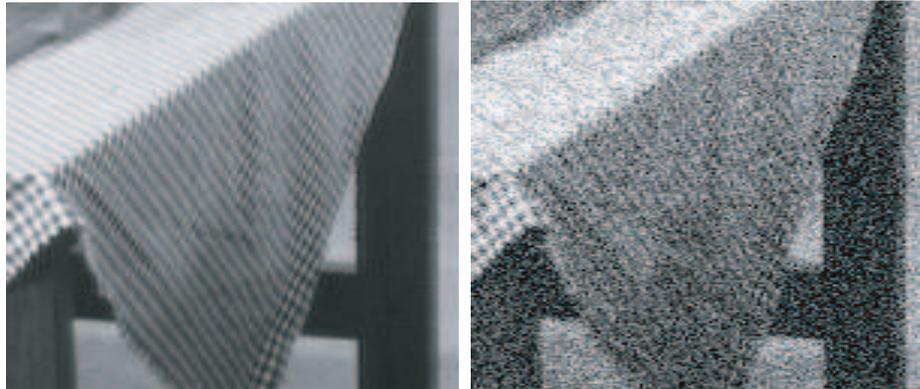
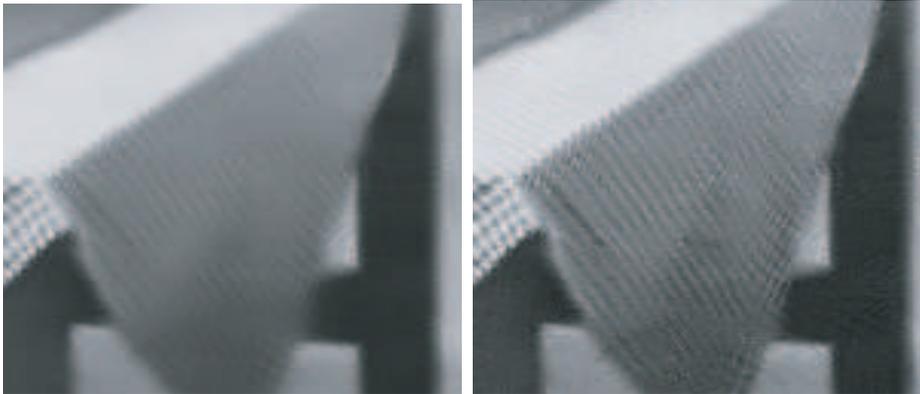


Figure 6.10: The visual quality comparison for Flinstones with  $\sigma = 25$ : (a) original image; (b) noisy image; (c) the result generated by the nonlocal method; (d) the result generated by the proposed approach.



(a)

(b)



(c)

(d)

Figure 6.11: The visual quality comparison for Barbara with  $\sigma = 35$ : (a) original image; (b) noisy image; (c) the result generated by the nonlocal method; (d) the result generated by the proposed approach.



(a)

(b)



(c)

(d)

Figure 6.12: The visual quality comparison for Lena with  $\sigma = 45$ : (a) original image; (b) noisy image; (c) the result generated by the nonlocal method; (d) the result generated by the proposed approach.



(a)

(b)



(c)

(d)

Figure 6.13: The visual quality comparison for Boat with  $\sigma = 20$ : (a) original image; (b) noisy image; (c) the result generated by the nonlocal method; (d) the result generated by the proposed approach.

## Chapter 7

### Simultaneous Image Denoising and Interpolation Using Evolutionary Games

Spatial resolution up-conversion is one of the most important tasks in the field of image processing. Image interpolation is the technique addressing the problem of spatial resolution up-conversion. It generates a high resolution image from the input low resolution image by exploiting the inherent relationship between them.

The commonly used image interpolation methods are the conventional linear interpolation schemes such as bilinear and bicubic interpolation [75]. These methods generate the high resolution image using a spatial-invariant linear interpolation filter. Although the computational complexity is low, these methods are not favored since they introduce a lot of blurring and ringing artifacts.

To overcome the drawbacks of conventional linear interpolation schemes, many more sophisticated adaptive image interpolation methods have been proposed. Jensen and Anastassiou proposed to first detect edges and then fit them with some templates to improve the interpolation result [69], while Carrato and Tenze optimized the interpolation parameters by using some predetermined edge pattern [24]. In [13], Allebach and Wong proposed to first estimate the high resolution edge map from the low resolution image using a subpixel edge estimation technique, and then correct the interpolated high resolution pixels based on the high resolution edge map.

Based on the assumption that the covariance matrix of the high resolution image can be well estimated from the covariance matrix of the low resolution image, Li and Orchard proposed an edge-preserved interpolation scheme [82]. Cha and Kim in [25] proposed a modified bilinear method by amending the error based on the interpolation error theorem in an edge-adaptive way. In [81], an MRF model-based edge-directed interpolation method is proposed by formulating the image interpolation problem as an energy minimization problem over a 2-D Markov Random field, where the edge direction information generated by a statistical based approach is incorporated in the energy function. For better interpolation, Zhang and Wu proposed to first generate multiple reconstructions from different directions, and then fuse the results by minimum mean square error estimation [132]. To further improve the interpolation results, a soft-decision adaptive interpolation (SAI) technique is proposed in [135] by combining the piecewise 2-D autoregressive modeling and block estimation.

Although the existing approaches can achieve promising interpolation results, they are designed for the noisy-free images, i.e., clean images. However, if the low resolution image is noisy, most of the existing interpolation approaches will also boost the noise and introduces severe visual distortions such as fake edge artifacts shown in Figure 7.1. In reality, due to the quality of the sensors or the conditions of the environment, sensor noise is introduced during the image acquisition processes. Moreover, additional noise and distortions will be introduced during the processes of being digitized, recorded, and transmitted. To avoid the distortion caused by the undesired noise boosting, one may perform the denoising before the interpolation.



denoising image can in adversely introduce severe visual artifacts. Therefore, to achieve better results, it is very important for us to jointly consider image denoising and interpolation together. In the literature, there are some prior works on jointly perform denoising and demosaicking – a special case of interpolation that reconstructs the missing color component due to color-filtered image sensors. For example, the authors in [67] proposed to use total least square technique, while the authors in [79] proposed to combine directional filter with wavelet-based denoising method. However, since these approaches are specially designed for demosaicking, they cannot be directly applied into the general interpolation problem.

The essential problem of simultaneous denoising and interpolation is to estimate the unknown clean pixels in the high resolution image based on the low resolution noisy image by exploiting the inherent relationship between them, e.g., estimating the unknown high resolution pixels using weighted average of a set of neighboring low resolution noisy pixels. However, since the original high resolution pixels are unknown, the optimal weights are not achievable. Most of the existing interpolation approaches find the approximated weights based on the assumption that the covariance matrix of the high resolution image can be well-estimated from the covariance matrix of the low resolution image [82]. Nevertheless, when this assumption is not true, the interpolation performance will be greatly degraded. Moreover, since the low resolution image is corrupted by noise, the reconstruction performance can be further degraded. Here, instead of directly estimating the weights in one step, we propose to progressively refine the weights by alternatively estimating the weights based on the reconstruction and finding the reconstruction using the weights. Such

a refinement process of the weights is actually an evolutionary process and can be naturally formulated as an evolutionary game. Thus, in this chapter, we consider the problem of simultaneous denoising and interpolation from the game theoretic perspective and formulate the problem as an evolutionary game.

Evolutionary game theory is an application of the mathematical theory of games to the interaction dependent strategy evolution in populations. Arising from the realization that frequency dependent fitness introduces a strategic aspect to evolution, evolutionary game theory becomes an essential component of a mathematical and computational approach to biological contexts, such as genes, viruses, cells, and humans. There are three basic components in an evolutionary game: players, strategies, and payoff functions. Players are the entities who play the game. Strategies, which can be divided into pure strategies and mixed strategies, are the complete plans of actions players may take in the game. A pure strategy is a deterministic plan of how a player will play a game while a mixed strategy is an assignment of a probability to each pure strategy. Payoff functions determine the payoffs players can obtain by adopting a certain strategy. In the proposed evolutionary game for simultaneous image denoising and interpolation, the players are the unknown high resolution pixels, and their pure strategies are the neighboring low resolution noisy pixels. The probabilities in the mixed strategy are the non-negative normalized weights of the low resolution noisy pixels. In this sense, the simultaneous image denoising and interpolation problem is no longer ill-posed. Instead, the problem becomes well-defined, and the objective of the player is to find the evolutionarily stable strategy, i.e., the optimal combination of the low resolution noisy pixels, to

achieve better denoising or interpolation performance.

The rest of this chapter is organized as follows. We first give an introduction about evolutionary game model in Section 7.1. Then, we show the problems of the methods that first perform denoising and then interpolate the denoising image, and describe in details how to formulate the problem of simultaneous image denoising and interpolation as an evolutionary game, how to choose the pure strategy set and how to define the payoff function in Section 7.2. Experimental results are shown in Section 7.3. Finally, we draw conclusions in Section 7.4.

## 7.1 Evolutionary Game Model

Before discussing how to use evolutionary game to simultaneously perform denoising and interpolation, in this section, we first briefly give an introduction about evolutionary games. A game  $\mathbf{G} = \{\mathbf{U}, \mathbf{A}, \mathbf{F}\}$  is generally defined as follows.  $\mathbf{U} = \{u_1, u_2, \dots, u_N\}$  are the players who play the game.  $\mathbf{A} = A_1 \times A_2 \times \dots \times A_N$  are the pure strategy sets, where  $A_i$  is the pure strategy set containing all possible pure strategies for user  $u_i$ . Let  $a_i \in A_i$  be one possible strategy for  $u_i$ , then  $\mathbf{a} = (a_1, a_2, \dots, a_N) \in \mathbf{A}$  is a strategy profile of  $\mathbf{U}$ .  $\mathbf{F} = \{f_1, f_2, \dots, f_N\}$  are the players' payoff/utility functions, where  $f_i$  is the payoff function of  $u_i$ . In general,  $f_i$  is determined by all players' strategies  $\mathbf{a}$  rather than  $a_i$  only, i.e.,  $f_i = f_i(\mathbf{a})$ . Also,  $a_{-i} = (a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_N)$  denotes the players' strategy profile except player  $u_i$ . Besides pure strategies, players can also take a mixed strategy by randomizing among different pure strategies. Suppose there are  $M_i$  pure strategies in the pure

strategy set  $A_i = \{a_i^1, \dots, a_i^{M_i}\}$ , and let  $p_{ij}, j = 1, \dots, M_i$  be the probability of  $u_i$  choosing the  $j^{\text{th}}$  pure strategy  $a_i^j$ , then  $\mathbf{p}_i = (p_{i1}, \dots, p_{iM_i})$  is a mixed strategy.

Since the payoff function  $f_i$  is generally not only determined by  $a_i$  but also by  $a_{-i}$ , to maximize the payoff, every player needs to know all other players' strategies. However, it is generally very difficult or even impossible for players to know other player strategies and payoffs. In such a case, to improve their payoffs, players will try different strategies in every play and learn from the strategic interactions using the methodology of understanding-by-building, which leads to the concept of "Evolutionary Game" [34] [110].

An evolutionary game is a game that studies the evolution of the interaction dependent strategy in populations, and was first articulated by John Maynard Smith and G. R. Price for evolutionary biology. It is based on the idea that an organism's genes largely determine its fitness in a given environment. Organisms that are more fit to the environment will tend to produce more offspring, due to which genes that provide greater fitness have more representation in the population. Therefore, fitter genes tend to win over time and drive out other genes. If we treat organisms as players and genes as strategies, then the genes which persist in the population are the evolutionarily stable strategy (ESS) of an evolutionary game, which can be formally defined as "a strategy such that, if all members of the population adopt it, then no mutant strategy could invade the population under the influence of natural selection". The evolution of the strategies over the population by natural selection can be characterized by the Wright-Fisher model [54], which is by far the most popular stochastic model for reproduction in population genetics. It is based on

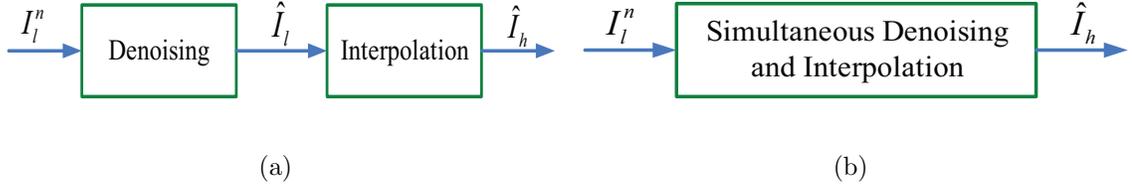


Figure 7.2: (a) Denoising before interpolation; (b) Simultaneous denoising and interpolation.

the assumption that the probability of an individual adopting a certain strategy is proportional to the expected payoff of the population using that strategy, and the strategy updating equation can be written as

$$p_{ij}^{t+1} = \frac{p_{ij}^t f_i^t(a_i^j, a_{-i})}{\sum_{k=1}^{M_i} p_{ik}^t f_i^t(a_i^k, a_{-i})}, \quad (7.1)$$

where the numerator  $p_{ij}^t f_i^t(a_i^j, a_{-i})$  is the expected payoff of  $u_i$  using strategy  $a_i^j$ , and the denominator  $\sum_{k=1}^{M_i} p_{ik}^t f_i^t(a_i^k, a_{-i})$  is the total expected payoff of  $u_i$  using different strategies, which is the normalization term that ensures  $\sum_{j=1}^{M_i} p_{ij}^{t+1} = 1$ .

## 7.2 Image Denosing and Interpolation as an Evolutionary Game

### 7.2.1 Problems of Denoising+Interpolation

As discussed in the introduction, when the input low resolution image is noisy, most of the existing interpolation approaches will also boost the noise and introduces severe visual distortions. To avoid such kinds of distortions, one may consider performing the denoising before the interpolation as shown in Figure 7.2 (a). The noisy low resolution image  $I_l^n$  is first passed through a denoising process and an estimate of the original low resolution image  $\hat{I}_l$  is found. Then, the estimated low



(a)



(b)



(c)



(d)

Figure 7.3: (a) a region of Lena image that is corrupted by Gaussian noise with noise variance  $\sigma^2 = 225$ ; (b) the denoised result using nonlocal; (c) the interpolation result of (b) using bicubic; (d) the interpolation result of (b) using SAI.

resolution image  $\hat{I}_l$  goes through a interpolation process and a reconstruction of the original high resolution image  $\hat{I}_h$  is obtained. Nevertheless, since the denoising problem itself is ill-posed [20], the low resolution estimate  $\hat{I}_l$  is not perfect and there will be some differences between the estimate  $\hat{I}_l$  and the original  $I_l$ . On one hand, the edge structures and textures may be removed during the denoising process, e.g., the details of the hair shown in Figure 7.3 (b). On the other hand, some noise may not be efficiently suppressed during the denoising process, e.g., the region around the nose and mouth shown in Figure 7.3 (b). In such a case, if we directly

perform interpolation on  $\hat{I}_l$ , the lost edge structures and textures will never be reconstructed back and the remained noise will be boosted which introducing severe visual distortions such as fake edge artifacts as shown in Figure 7.3 (c) and (d). Therefore, we should jointly consider image denoising and interpolation to achieve better reconstruction as illustrated in Figure 7.2 (b).

## 7.2.2 Game Theoretic Formulation

In this simultaneous denoising and interpolation problem, we have the noisy observation of the low resolution pixels,  $I_l^n(m, n), 1 \leq m \leq N_H, 1 \leq n \leq M_W$ . The objective of this problem is to estimate the high resolution image  $I_h(i, j), 1 \leq i \leq 2 \times N_H, 1 \leq j \leq 2 \times M_W$  based on the noisy low resolution image  $I_l^n(m, n)$ . Obviously, this problem is ill-posed. To find a good estimate for the unknown high resolution pixels, we need to exploit the correlation among the low resolution pixels and between the low resolution pixels and the high resolution pixels. One possible approach is to use the spatially varying linear filter, i.e., each unknown high resolution pixel can be estimated using weighted average of a set of neighboring noisy low resolution pixels.

$$\hat{I}_h(i, j) = \sum_{(m, n) \in \Omega_{ij}} w_{mn}^{ij} I_l^n(m, n), \forall i, \forall j, \quad (7.2)$$

where  $\Omega_{ij}$  is the candidate set of neighboring noisy low resolution pixels for  $I_h(i, j)$  and  $w_{mn}^{ij}$  is the weight of candidate pixel  $I_l^n(m, n)$ . Usually, we have the constraints that  $0 \leq w_{mn}^{ij} \leq 1$  and  $\sum_{(m, n) \in \Omega_{ij}} w_{mn}^{ij} = 1$ .

Obviously, the optimal weights in (7.2) are not achievable since the original

high resolution pixels are unknown. Most of the existing approaches find the approximated weights based on the assumption that the covariance matrix of the high resolution image can be well-estimated from the covariance matrix of the low resolution image [82]. However, when this assumption is not true, the denoising and interpolation performance will be greatly degraded. Here, instead of directly estimating the weights in one step, we propose to progressively refine the weights by alternatively estimating the weights based on the reconstruction and finding the reconstruction using the weights. Such a refinement process of the weights is actually an evolutionary process and can be naturally formulated as an evolutionary game [110]. In the proposed evolutionary game for image denoising and interpolation, the players are the unknown high resolution pixels to be estimated and the pure strategies are the correspondingly neighboring noisy low resolution pixels. By regarding the non-negative weights of the neighboring noisy low resolution pixels as the probabilities of selecting the pure strategies, the problem of estimating the high resolution pixels becomes finding the evolutionarily stable strategies for the evolutionary game. In this sense, the simultaneous image denoising and interpolation problem is no longer ill-posed. Instead, the problem becomes well-defined, and the objective of the problem is to find the evolutionarily stable strategies to achieve good denoising and interpolation performance.

In summary, the estimation problem in (7.2) can be formulated as an evolutionary game as follows.

- Players: unknown high resolution pixels  $I_h(i, j)$

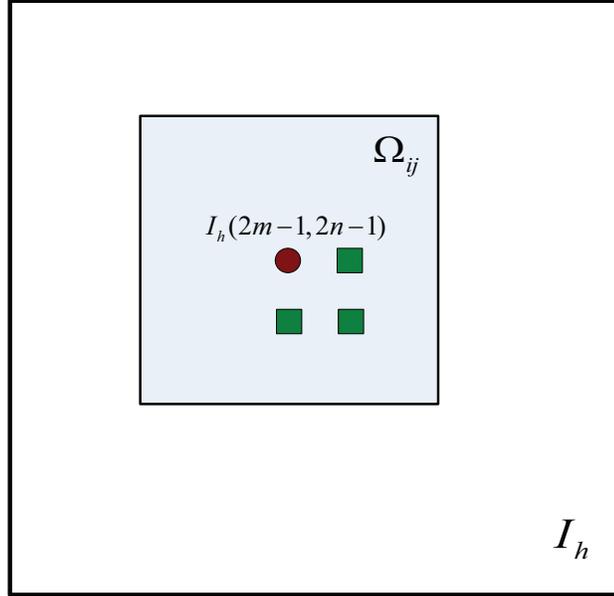


Figure 7.4: The pure strategy set  $\Omega_{ij}$  for  $I_h(2m-1, 2n-1)$ ,  $I_h(2m-1, 2n)$ ,  $I_h(2m, 2n-1)$ ,  $I_h(2m, 2n)$ .

- Pure strategies: noisy low resolution pixels  $I_l^n(m, n)$
- Pure strategy set: the candidate set  $\Omega_{ij}$
- Mixed strategy: the estimate  $\hat{I}_h(i, j)$
- Probabilities in the mixed strategy: the non-negative normalized weights  $w_{mn}^{ij}$

### 7.2.3 Pure Strategy Set $\Omega_{ij}$

In most of the previous interpolation approaches, the high resolution image is reconstructed in two steps: in the first step, the unknown high resolution pixels surrounded by four low resolution pixels, i.e.,  $I_h(2m, 2n)$ ; and in the second step, other unknown high resolution pixels  $I_h(2m-1, 2n)$  and  $I_h(2m, 2n-1)$  are reconstructed

with the help of reconstructed high resolution pixels  $I_h(2n, 2m)$ . Different from previous approaches, in our game theoretic framework, pixels are treated as players and they have the same priority. The pure strategy set  $\Omega_{ij}$  is the same for all four high resolution pixels  $I_h(2m - 1, 2n - 1)$ ,  $I_h(2m - 1, 2n)$ ,  $I_h(2m, 2n - 1)$ ,  $I_h(2m, 2n)$  and is defined as a square window centered by  $I_h(2m - 1, 2n - 1)$  in Figure 7.4. As shown in Figure 7.4, the red circle stands for pixel  $I_h(2m - 1, 2n - 1)$  and the three green squares stand for pixels  $I_h(2m - 1, 2n)$ ,  $I_h(2m, 2n - 1)$ , and  $I_h(2m, 2n)$  respectively. The gray square window centered by  $I_h(2m - 1, 2n - 1)$ , denoted as  $\Omega_{ij}$ , is the pure strategy set for  $I_h(2m - 1, 2n - 1)$ ,  $I_h(2m - 1, 2n)$ ,  $I_h(2m, 2n - 1)$ , and  $I_h(2m, 2n)$ .

#### 7.2.4 Payoff Function

After choosing the pure strategy set, we now discuss how to define the payoff function. The payoff function  $f_{ij}^t(a_{ij}, a_{-ij})$  measures the player's payoff of taking strategy  $a_{ij}$  when other players' strategies are  $a_{-ij}$  at time  $t$ . Let  $\hat{I}_h^t$  stand for the estimate of the high resolution image  $I_h$  at time  $t$ , and  $B(\hat{I}_h^t(i, j))$  stand for the patch centered by  $\hat{I}_h^t(i, j)$ . The neighborhood similarity between pixel  $\hat{I}_h^t(i, j)$  and  $\hat{I}_h^t(m, n)$  at time  $t$  can be measured by  $D^t(\hat{I}_h^t(i, j) \leftrightarrow \hat{I}_h^t(m, n))$  as follows

$$D^t(\hat{I}_h^t(i, j) \leftrightarrow \hat{I}_h^t(m, n)) = \|B(\hat{I}_h^t(i, j)) - B(\hat{I}_h^t(m, n))\|^2. \quad (7.3)$$

Moreover, let us define  $\bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n))$  as the weighted average of the neighborhood similarity up to time  $t - 1$  as follow

$$\bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n)) = \frac{\sum_{k=0}^{t-1} \beta^k D^k(\hat{I}_h^k(i, j) \leftrightarrow \hat{I}_h^k(m, n))}{\sum_{k=0}^{t-1} \beta^k}, \quad (7.4)$$

where  $\beta$  is a discounting factor, and when  $\beta = 1$ ,  $\bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n))$  reduces to the simple averaging.

Note that  $\bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n))$  can be treated as an estimate of  $D(I_h(i, j) \leftrightarrow I_h(m, n))$  at time  $t - 1$  by taking into account all previous reconstruction  $\hat{I}_h^0, \dots, \hat{I}_h^{t-1}$ , and  $D^t(\hat{I}_h^t(i, j) \leftrightarrow \hat{I}_h^t(m, n))$  is an estimate of  $D(I_h(i, j) \leftrightarrow I_h(m, n))$  at time  $t$  using the reconstruction  $\hat{I}_h^t$ . If  $\hat{I}_h^t(i, j)$  is a good estimate,  $D^t(\hat{I}_h^t(i, j) \leftrightarrow \hat{I}_h^t(m, n))$  tends to be close to  $\bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n))$ . If  $D^t(\hat{I}_h^t(i, j) \leftrightarrow \hat{I}_h^t(m, n)) < \bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n))$ , the neighborhood similarity between pixel  $\hat{I}_h(i, j)$  and  $\hat{I}_h(m, n)$  is larger than what we anticipate at time  $t - 1$ , which means that a larger payoff should be received by adopting strategy  $\hat{I}_h^t(m, n)$  at time  $t$ . On the other hand if  $D^t(\hat{I}_h^t(i, j) \leftrightarrow \hat{I}_h^t(m, n)) > \bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n))$ , the neighborhood similarity between pixel  $\hat{I}_h(i, j)$  and  $\hat{I}_h(m, n)$  is smaller than what we anticipate at time  $t - 1$ , which means that a smaller payoff should be received by adopting strategy  $\hat{I}_h^t(m, n)$  at time  $t$ . Therefore, the payoff function should be an increasing function of  $(\bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n)) - D^t(\hat{I}_h^t(i, j) \leftrightarrow \hat{I}_h^t(m, n)))$ . Here, we use the following payoff function

$$f_{ij}^t(a_{ij}, a_{-ij}) = \exp \left( \frac{\alpha_t \left( \bar{D}^{t-1}(\hat{I}_h(i, j) \leftrightarrow a_{ij}) - D^t(\hat{I}_h^t(i, j) \leftrightarrow a_{ij}) \right)}{\gamma} \right), \quad (7.5)$$

where  $\alpha_t$  and  $\gamma$  are parameters.

Interestingly, if we set  $\alpha_t = \frac{\beta^t}{\sum_{k=0}^t \beta^k}$  and substitute (7.5) back into (7.1), the probability of choosing strategy  $a_{ij}$  at time  $t + 1$  can be simplified as

$$p_{a_{ij}}^{t+1} = \frac{\exp \left( -\frac{\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow a_{ij})}{\gamma} \right)}{\sum_{a_{ij}} \exp \left( -\frac{\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow a_{ij})}{\gamma} \right)}. \quad (7.6)$$

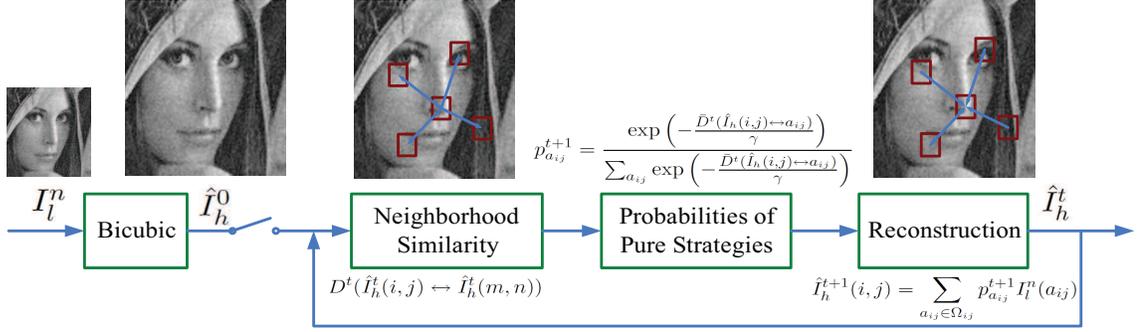


Figure 7.5: The block diagram of the proposed method.

where  $\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n))$  is defined as

$$\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n)) = \frac{\sum_{k=0}^t \beta^k D^k(\hat{I}_h^k(i, j) \leftrightarrow \hat{I}_h^k(m, n))}{\sum_{k=0}^t \beta^k}, \quad (7.7)$$

The (7.6) can be interpreted as follows: the true similarity between  $I_h(i, j)$  and  $a_{ij}$  is approximately measured by  $\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow a_{ij})$ , and the larger  $\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow a_{ij})$ , the less the similarity; therefore, the pixels with smaller  $\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow a_{ij})$  should make more contribution during the denoising and interpolation process, i.e., the probability of choosing  $a_{ij}$  should be a decreasing function of  $\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow a_{ij})$ , and here we use an exponential function.

According to the above discussions, the proposed simultaneous image denoising and interpolation using evolutionary games can be summarized as in Figure 7.5. As shown in Figure 7.5, the noisy low resolution image  $I_l^n$  is first interpolated using Bicubic [75] to obtain a noisy estimate of the high resolution image  $\hat{I}_h^0$ . Then, at each evolution time index  $t$ , we compute the neighborhood similarity  $D^t(\hat{I}_h^t(i, j) \leftrightarrow \hat{I}_h^t(m, n))$  using (7.3). Then, the probability of using a certain pure strategy  $a_{ij}$  can be updated using (7.6) and (7.7). Finally, the estimate of the high resolution image

at time index  $t + 1$  can be found as follows

$$\hat{I}_h^{t+1}(i, j) = \sum_{a_{ij} \in \Omega_{ij}} p_{a_{ij}}^{t+1} I_l^n(a_{ij}). \quad (7.8)$$

Note that the outcome of the proposed game theoretic algorithm is an evolutionarily stable strategy, and the proof can be found in the following **Theorem 1**.

**Theorem 1:** For any  $\beta \in [0, 1)$ , the outcome of the proposed game theoretic algorithm shown in Figure 7.5 is an evolutionarily stable strategy (ESS).

*Proof:* Since  $\beta \in [0, 1)$ ,  $\beta^k$  goes to zero as  $k$  goes to infinite. According to (7.7),  $\bar{D}^t(\hat{I}_h(i, j) \leftrightarrow \hat{I}_h(m, n))$  converges for sufficiently large  $t$ . Therefore, according to (7.6),  $p_{a_{ij}}^{t+1}$  converges for sufficiently large  $t$ , which means that the probability distribution of using the pure strategies  $\mathbf{p}_{ij} = (p_1, \dots, p_{|\Omega_{ij}|})$  converges. Since such a strategy is chosen under the influence of Wright-Fisher natural selection model [54], according to the definition of ESS, the strategy is an ESS. ■

### 7.3 Experimental Results

To evaluate the proposed game theoretic approach for simultaneous image denoising and interpolation, we compare with the methods that first perform denoising using nonlocal [20] and then perform interpolation using either bicubic method [75] or the soft-decision adaptive interpolation (SAI) method [135], which are denoted as “**Nonlocal+Bicubic**” and “**Nonlocal+SAI**”. Five images shown in Figure 7.6: Lena, Boat, Kodim03, Kodim07, and Kodim09, are tested. For convenience, all tested image are truncated to 512x512. The noisy low resolution images are gen-

erated by first adding additive white Gaussian noise to the high resolution images and then directly perform downsample.

We first evaluate the PSNR performance for different approaches. In Figure 7.7, we show the PSNR comparison among Nonlocal+Bicubic, Nonlocal+SAI and the proposed method. We can see that the proposed method outperforms both Nonlocal+Bicubic and Nonlocal+SAI when  $\sigma \geq 10$ , and the gain becomes larger and larger as  $\sigma$  increases. We can also see that as  $\sigma$  increases, the advantage of using SAI diminishes. This is mainly because too many details are removed during the denoising process when  $\sigma$  is high. This phenomenon fully demonstrates the importance of performing joint denoising and interpolation. When  $5 < \sigma < 10$ , the PSNR performance of the proposed method is similar to that of Nonlocal+SAI (the proposed method has slightly better performance for Lena and Kodim03, and slightly worse performance for Boat and Kodim09). However, both the proposed method and Nonlocal+SAI are better than Nonlocal+Bicubic. When  $\sigma \leq 5$ , the proposed method performs slightly worse than Nonlocal+SAI. This is mainly because the proposed method not only removes the additive noise but also removes the sensor noise in the original image, and the PSNR is computed based on the original images which contains some sensor noise. Nevertheless, the proposed method can achieve much better visual quality even in the low  $\sigma$  region as shown in Figure 7.9.

We also evaluate the visual quality performance and show the visual quality comparison in Figures 7.8, 7.9, 7.10, 7.11, and 7.12, respectively, for different images at different noise variances. The results for Lena image are shown in Figure 7.8, where (a) is the noisy low resolution Lena image with  $\sigma = 10$ , (b) is the result

generated by Nonlocal+Bicubic with PSNR 30.71dB, (c) is the result generated by Nonlocal+SAI method with PSNR 31.11dB, and (d) is the result generated by the proposed method with PSNR 31.38dB. By comparing (b) (c) and (d) in Figure 7.8, we can see that both Nonlocal+Bicubic and Nonlocal+SAI cannot well suppress noise in the edge and texture regions and introduce some visually annoying artifacts such as fake edge artifacts in the face region. Since the proposed game theoretic method can simultaneously perform denoising and interpolation, we can avoid the artifacts caused by the separation of denoising and interpolation and generate the reconstruction with much better performance in terms of both PSNR and visual quality.

Moreover, the proposed method has the ability to automatically remove the visually annoying sensor noise. In Figure 7.9, we show the reconstructed results for Boat image. Note that the original Boat image contains some sensor noise especially in the sky region as shown in Figure 7.6 (b). In Figure 7.9, (a) is the noisy low resolution Boat image with  $\sigma = 5$ , (b) is the result generated by Nonlocal+Bicubic with PSNR 28.52dB, (c) is the result generated by Nonlocal+SAI method with PSNR 28.92dB, and (d) is the result generated by the proposed method with PSNR 28.74dB. By comparing (b) (c) and (d) in Figure 7.9, we can see that the proposed method automatically removes the sensor noise in the sky region while Nonlocal+SAI and Nonlocal+Bicubic cannot. In such a case, although the proposed method has a lower PSNR performance compared with Nonlocal+SAI, the visual quality of the result generated by the proposed method is still much better than that of Nonlocal+SAI.

The visual quality of the reconstructions of Kodim07, Kodim09 and Kodim03 are also evaluated at noise level  $\sigma = 10$ ,  $\sigma = 15$ , and  $\sigma = 20$  in Figures 7.10, 7.11, and 7.12, respectively. Similar to previous experiments, the proposed method can greatly suppress the noise and restore the image with not only better PSNR performance but also better visual quality, especially in the regions around edges and textures.

## 7.4 Summary

In this chapter, we investigated the problem of simultaneous image denoising and interpolation from a completely new angle: game theoretic perspective. We treat each unknown high resolution pixel as an individual player and formulate the joint image denoising and interpolation problem as an evolutionary game. From such a perspective, the problem of estimating the high resolution pixels becomes finding the evolutionarily stable strategies for the evolutionary game. The experimental results show that compared with the methods that first denoise the noisy low resolution image and then interpolate the denoised image, the proposed game theoretic approach can achieve not only better PSNR performance but also better visual quality.



(a)



(b)



(c)



(d)



(e)

Figure 7.6: The test images: (a) Lena; (b) Boat; (c) Kodim03; (d) Kodim07; (e) Kodim09.

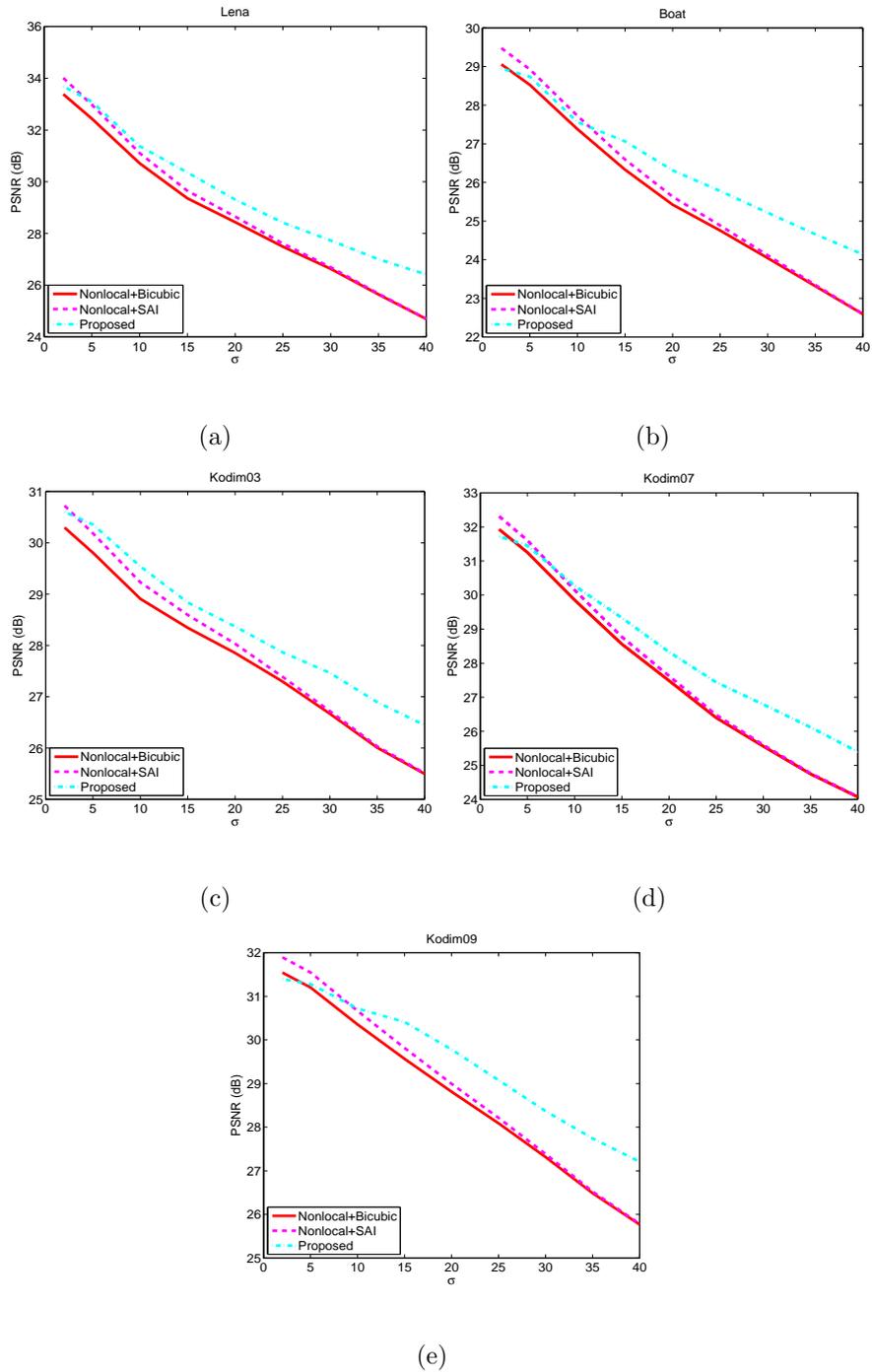


Figure 7.7: The PSNR comparison: (a) Lena; (b) Boat; (c) Kodim03; (d) Kodim07; (e) Kodim09.



(a)

(b)



(c)



(d)

Figure 7.8: The visual quality comparison for Lena: (a) the noisy low resolution image with  $\sigma = 10$ ; (b) the result generated by Nonlocal+Bicubic (30.71dB); (c) the result generated by Nonlocal+SAI method (31.11dB); (d) the result generated by the proposed method (31.38dB).



(a)



(b)



(c)



(d)

Figure 7.9: The visual quality comparison for Boat: (a) the noisy low resolution image with  $\sigma = 5$ ; (b) the result generated by Nonlocal+Bicubic (28.52dB); (c) the result generated by Nonlocal+SAI method (28.92dB); (d) the result generated by the proposed method (28.74dB).



(a)



(b)

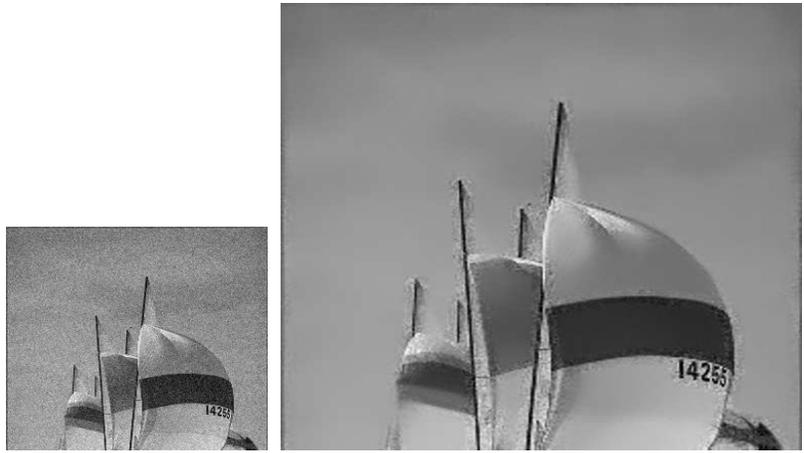


(c)



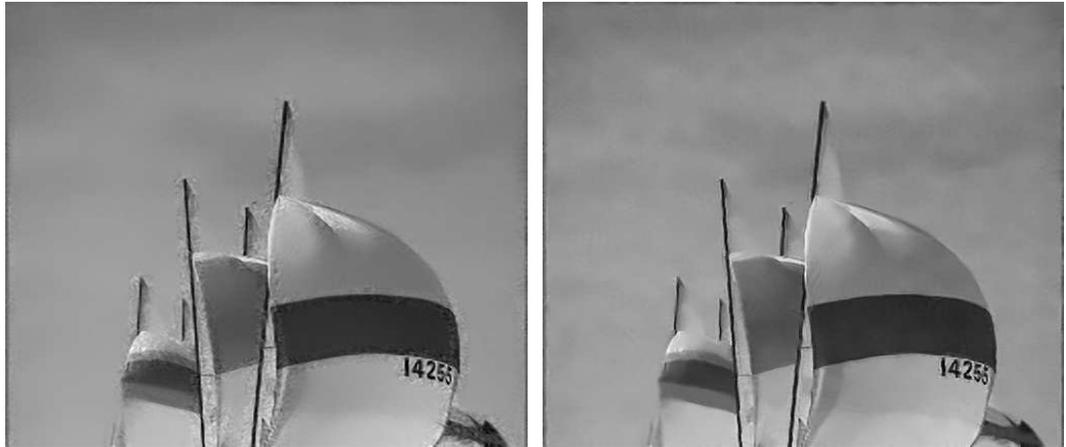
(d)

Figure 7.10: The visual quality comparison for Kodim07: (a) the noisy low resolution image with  $\sigma = 10$ ; (b) the result generated by Nonlocal+Bicubic (29.85dB); (c) the result generated by Nonlocal+SAI method (30.18dB); (d) the result generated by the proposed method (30.35dB).



(a)

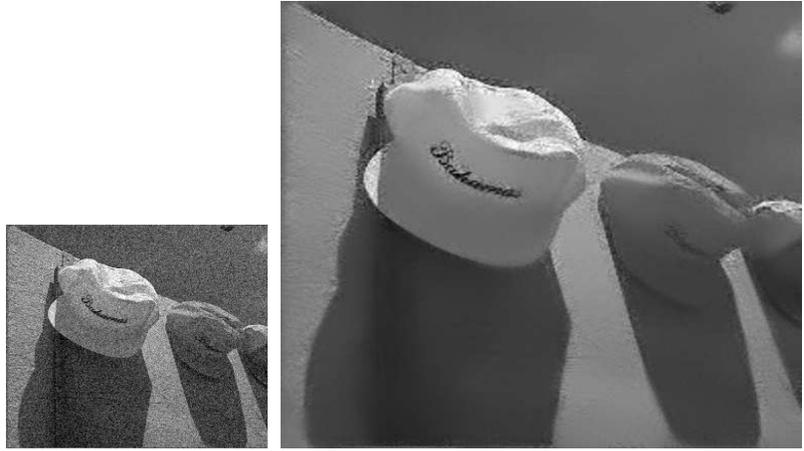
(b)



(c)

(d)

Figure 7.11: The visual quality comparison for Kodim09: (a) the noisy low resolution image with  $\sigma = 15$ ; (b) the result generated by Nonlocal+Bicubic (29.56dB); (c) the result generated by Nonlocal+SAI method (29.81dB); (d) the result generated by the proposed method (30.41dB).



(a)

(b)



(c)

(d)

Figure 7.12: The visual quality comparison for Kodim03: (a) the noisy low resolution image with  $\sigma = 20$ ; (b) the result generated by Nonlocal+Bicubic (27.85dB); (c) the result generated by Nonlocal+SAI method (28.03dB); (d) the result generated by the proposed method (28.37dB).

## Chapter 8

### Conclusions and Future Work

#### 8.1 Conclusions

In this thesis, we develop a game-theoretic framework that enables us to first analyze and model human behaviors in multimedia social networks, and then better design the multimedia systems by taking into account the impact of human factors. We have showed that understanding the human behaviors and dynamics in a multimedia social network can ultimately offer better system performance and thus is essential for its continued progress, and such analysis and modeling can be applied to any social networks.

Moreover, we extend the concept of multimedia social networking into classical signal/image processing problems to liberate pixels/signals as players to develop a game-theoretic framework that can, not only overcome some of the undesired ill-posed formulations in traditional approaches, but also obtain a more general paradigm beyond what can be accomplished by using traditional optimization tools. With the notion of learning and cognitive process inherent in a game theoretic formulation, we show that many classical approaches are basically special cases of the proposed game-theoretic framework. Therefore, the proposed framework offers new directions, insight, and methodologies in further advancing of the science of signal/image processing.

The broader impact of this thesis is

1. Social networks have pervaded our daily life. By illustrating that game theory can be used to understand human behavior and dynamics in a multimedia social network with better system performance, this thesis can motivate similar new ideas in many social networks.
2. Signal and image processing has been a fundamental tool in many scientific and engineering disciplines. By introducing the proposed new game-theoretic paradigm to classical signal/image problems with new insight and significant performance improvement, this thesis can trigger similar new thinking to many scientific areas that use signal/image processing.

Specifically, in Chapter 3, we consider a non-cooperative multimedia social network and discuss how a group of users compete for the same resource. We use multiuser rate allocation social network as an example and show that game theory can provide a more general framework by theoretically proving that the traditional optimization-based approach is a special case of the proposed game theoretical framework. Moreover, with the proposed method, we can find, in a distributed manner, a NE that is not only efficient from system designer's perspective but also fair from users' perspective. Then, in Chapter 4, we consider a cooperative multimedia social network and discuss how a group of selfish users cooperate with each other to better obtain the content. We use cooperative peer-to-peer streaming social network as an example and show that evolutionary game can be used in such a scenario and ESS is the desired cooperative strategy. Moreover, we propose a distributed learn-

ing algorithm for users to converge to the ESS by learning from their own payoff history. In Chapter 5, we discuss how to stimulate cooperation in cooperative social networks. We first show that most of the existing game theoretic cooperation stimulation approaches fail when the number of interaction between a pair of players is finite, and the major reason is the use of direct reciprocity. Then, we propose to use indirect reciprocity games to stimulate cooperation in such a scenario by taking into account the indirect opinions. With such a modeling, we show with simulations that an evolutionarily stable cooperative strategy can be achieved with a proper cost-to-gain ratio. In Chapter 6, the image denoising problem is formulated as a coalition formation game, where every pixel is treated as a player who tries to seek partners to form a coalition to find the optimal neighborhoods for better denoising results. By forming a coalition, every player in the coalition can obtain certain gains by improving the accuracy of the distortion estimation, while incurring some costs by increasing the true distortion. With such a formulation, the traditional image denoising approaches using a heuristically determined neighborhood set can be seen as special cases of the proposed game theoretical framework by choosing the utility function without a cost term. Another example is formulating the problem of simultaneous image denoising and interpolation as an evolutionary game in Chapter 7, where the players are the unknown high resolution pixels and the pure strategies of the players are the corresponding noisy low resolution neighbors. By regarding the nonnegative weights of the noisy low resolution pixels as the probabilities of selecting the pure strategies, the problem of estimating the high resolution pixels becomes finding the evolutionarily stable strategies for the evolutionary game. In

this sense, we say that the simultaneous image denoising and interpolation problem is no longer ill-posed. Instead, the problem becomes well-defined, and the objective of the problem is to find the evolutionarily stable strategies to achieve good denoising and interpolation performance.

## 8.2 Future Work

Recently, the area of human and social dynamics has been identified by the U.S. National Science Foundation as one of its five priority areas, which shows the importance of this emerging interdisciplinary research area. Game theoretic modeling for multimedia social networks is a new emerging research field and is still in an infant stage. There are a lot of exciting problems to be investigated and addressed, which I will continue to devote my efforts to.

Security in multimedia social networks is one of these problems. Besides selfish users, there are a group of users, called malicious users, in multimedia social networks. Unlike selfish users whose aims are to maximize their own payoffs, the objective of malicious users is to damage or even break down the system. Therefore, to successfully deploy the multimedia social networks, we need to study and analyze the malicious users attack strategies and develop the corresponding attack-resistant strategies.

With my previous works, we can see that understanding the behavior dynamics among users can ultimately offer better system performance. Such analysis and modeling can be applied to other social networks. Therefore, in the future, I

would like to investigate the possibility of using such analysis and modeling in other networks such as online social networks, smart grid networks and camera networks.

In my previous works, I have successfully used the concept of multimedia social networks to reformulate image denoising and image interpolation problems. In the future, I would like to apply the concept to more classical problems such as image/video compression problems, estimation and detection problems, pattern recognition and classification problems, adaptive signal processing problems, and information theory related problems. It is our belief that from the multimedia social networks point of view, we can make the ill-posed problems well defined and are able to construct generalized and unified frameworks for the classical problems. We hope that we can introduce a new paradigm not only in the field of signal and image processing but also in many other fields such as computer vision and information theory.

## Bibliography

- [1] *Accustream iMedia Research Homepage*: <http://www.accustreamresearch.com>.
- [2] *Flickr*: <http://www.flickr.com>.
- [3] *Napster*: <http://www.napster.com>.
- [4] *PPLive*: <http://www.pplive.com>.
- [5] *PPStream*: <http://www.ppstream.com>.
- [6] *H.264/AVC Software*: <http://iphome.hhi.de/suehring/tml/>.
- [7] *Sopcast*: <http://www.sopcast.com>.
- [8] *UUSee*: <http://www.uusee.com>.
- [9] *Youtube*: <http://www.youtube.com>.
- [10] A. M. Frieze A. D. Flaxman and J. Vera. A geometric preferential attachment model of networks ii. In *Proceedings of the 5th Workshop On Algorithms And Models For The Web-Graph*, page 4155, 2007.
- [11] F. Abramovich and Y. Benjamini. Adaptive thresholding of wavelet coefficients. *Computational Statistics and Data Analysis*, 22:351–361, 1996.
- [12] Ishfaq Ahmad and Jiancong Luo. On using game theory for perceptually tuned rate control algorithm for video coding. *IEEE Trans. Circuits Syst. Video Technol.*, 16(2):202–208, Feb. 2006.
- [13] Allebach and P. W. Wong. Edge-directed interpolation. In *IEEE Int. Conf. Image Processing*, volume 3, pages 707–710, 1996.
- [14] L. Anderegg and S. Eidenbenz. Ad hoc-vcg: A truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In *Proc. ACM MobiCom*, 2003.
- [15] K. R. Apt and A. Witzel. A generic approach to coalition formation. In *Proc. Int. Workshop Computational Social Choice (COMSOC)*, 2006.
- [16] Lawrence M. Ausubel. An efficient ascending-bid auction for multiple objects. *American Economic Review*, 94:1452–1475, 2004.
- [17] A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.
- [18] B. Bollobas and O. Riordan. The diameter of a scale-free random graph. In *Combinatorica*, page 534, 2004.

- [19] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [20] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. *SIAM Multiscale Modeling and Simulation*, 4(2):490–530, July 2005.
- [21] A. Buades, B. Coll, and J. M. Morel. The staircasing effect in neighborhood filters and its solution. *IEEE Trans. Image Processing*, 15:1499–1505, 2006.
- [22] S. Buchegger and J. Y. Le Boudec. Performance analysis of the confidant protocol. In *Proc. ACM MobiHoc*, 2002.
- [23] L. Buttyan and J. P. Hubaux. Enforcing service availability in mobile ad-hoc networks. In *Proc. ACM MobiHoc*, 2000.
- [24] S. Carrato and L. Tenze. A high quality 2x image interpolator. *IEEE Signal Process. Lett.*, 7(6):132–135, Jun. 2000.
- [25] Y. Cha and S. Kim. The error-amended sharp edge (ease) scheme for image zooming. *IEEE Trans. Image Process.*, 16(6):1496–1505, Jun. 2007.
- [26] T. Chan and J. Shen. *Image processing and analysis: variational, PDE, wavelet, and stochastic methods*. SIAM, 2005.
- [27] Y. Chen, W. S. Lin, and K. J. R. Liu. Risk-distortion analysis for video collusion attacks: A mouse-and-cat game. *IEEE Trans. on Image Processing*, 19(7):1798–1807, July 2010.
- [28] Y. Chen and K. J. R. Liu. Indirect reciprocity game modelling for cooperation stimulation in cognitive networks. *IEEE Trans. Commun.*, 59(1):159–168, Jan. 2011.
- [29] Y. Chen, B. Wang, and K. J. R. Liu. Multi-user rate allocation games for multimedia communications. *IEEE Trans. Multimedia*, 11(6):1170–1181, Oct. 2009.
- [30] Y. Chen, Y. Wu, B. Wang, and K.J.R. Liu. Spectrum auction games for multimedia streaming over cognitive radio networks. *IEEE Trans. Commun.*, 58(8):2381–2390, Aug. 2010.
- [31] Yan Chen and Oscar C. Au. Simultaneous RD-Optimized rate control and video de-noising. In *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2008.
- [32] Yan Chen, Oscar C. Au, and Xiaopeng Fan. Simultaneous MAP-based video denoising and rate-distortion optimized video encoding. *IEEE Trans. on Circuits and Systems for Video Technology*, 19(1):15–26, Jan. 2009.

- [33] Yan Chen and K. J. Ray Liu. A game theoretical approach for image denoising. In *Proc. IEEE Int. Conf. Image Process.*, 2010.
- [34] Yan Chen, Beibei Wang, W. Sabrina Lin, Yongle Wu, and K. J. Ray Liu. Cooperative peer-to-peer streaming: An evolutionary game-theoretic approach. *IEEE Trans. on Circuits and Systems for Video Technology*, 20(10):1346–1357, Oct. 2010.
- [35] T. H. Chiang and Y. Q. Zhang. A new rate control scheme using quadratic rate distortion model. *IEEE Trans. Circuits Syst. Video Technol.*, 7:246–250, 1997.
- [36] Y. Chu, S. G. Rao, and H. Zhang. A case for end system multicast. In *ACM SIGMETRICS*, pages 1–12, 2000.
- [37] F. R. K. Chung and L. Lu. The diameter of sparse random graphs. In *Advances in Applied Mathematics*, page 257279, 2001.
- [38] C. Cooper and A. Frieze. A general model of web graphs. In *Random Structures and Algorithms*, page 311335, 2003.
- [39] J. R. Corbera and S. Lei. Rate control in DCT video coding for low-delay communications. *IEEE Trans. Circuits Syst. Video Technol.*, 9:172–185, 1999.
- [40] P. Cramton, Y. Shoham, and R. Steinberg. *Combinatorial Auctions*. MIT Press, 2006.
- [41] R. Cressman. *Evolutionary dynamics and extensive form games*. MIT Press, 2003.
- [42] J. Crowcroft, R. Gibbens, F. Kelly, and S. Ostring. Modeling incentives for collaboration in mobile ad hoc networks. In *Proc. Workshop Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOPT'03)*, 2003.
- [43] Y. Zhan D. Chakrabarti and C. Faloutsos. R-mat: A recursive model for graph mining. In *SIAM Conference on Data Mining*, 2004.
- [44] S. Lawrence E. J. Glover D. M. Pennock, G. W. Flake and C. L. Giles. Winners don't take all: Characterizing the competition for links on the web. In *Proceedings of the National Academy of Sciences*, page 52075211, 2002.
- [45] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3d transform-domain collaborative filtering. *IEEE Trans. Image Processing*, 16(8):2080–2095, 2007.
- [46] M. Dai, D. Loguinov, and H. Radha. Rate-distortion modeling of scalable video coders. In *Proc. IEEE Int. Conf. Image Processing*, 2004.

- [47] S. Deering and D. Cheriton. Multicast routing in datagram inter-networks and extended lans. *ACM Transactions on Computer Systems*, 8(2):85–111, 1990.
- [48] W. Ding and B. Liu. Rate control of MPEG video coding and recording by rate-quantization modeling. *IEEE Trans. Circuits Syst. Video Technol.*, 6:12–20, 1996.
- [49] P. Domingos and M. Richardson. Mining the network value of customers. In *Proceedings of the 7th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2001.
- [50] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. Inform. Theory*, 41:613–627, May 1995.
- [51] M. Elad. On the origin of the bilateral filter and ways to improve it. *IEEE Trans. Image Processing*, 10(10):1141–1151, 2002.
- [52] M. Elad. Why shrinkage is still relevant for redundant representations? *IEEE Trans. Inform. Theory*, 52(12):5559–5569, 2006.
- [53] M. Felegyhazi, J. P. Hubaux, and L. Buttyan. Nash equilibria of packet forwarding strategies in wireless ad hoc networks. *IEEE Trans. Mobile Comput.*, 5:463–476, May 2006.
- [54] R. Fisher. *The genetical theory of natural selection*. Clarendon Press, Oxford, 1930.
- [55] G. Gilboa, N. Sochen, and Y. Y. Zeevi. Image enhancement and denoising by complex diffusion processes. *IEEE Trans. Pattern Anal. Machine Intell.*, 26:1020–1036, Aug. 2004.
- [56] G. Gilboa, N. Sochen, and Y. Y. Zeevi. Variational denoising of partl textured images by spatially varying constraints. *IEEE Trans. Image Processing*, 15:2281–2289, Aug. 2006.
- [57] P. Golle, K. Leyton-Brown, and I. Mironov. Incentive for sharing in peer-to-peer networks. In *Proc. of the ACM Conference on Electronic Commerce*, 2001.
- [58] O. Guleryuz. Weighted averaging for denoising with overcomplete dictionaries. *IEEE Trans. Image Processing*, 16(12):3020–3034, 2007.
- [59] M. Gupta, P. Judge, and M. Ammar. A reputation system for peer-to-peer networks. In *Proc. of the ACM 13th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, 2003.
- [60] Ahsan Habib and John Chuang. Service differentiated peer selection: An incentive mechanism for peer-to-peer media streaming. *IEEE Trans. Multimedia*, 8(3):610–621, 2006.

- [61] Zhu Han, Zhu Ji, and K. J. Ray Liu. Fair multiuser channel allocation for OFDMA networks using Nash bargaining solutions and coalitions. *IEEE Trans. Commun.*, 53:1366–1376, 2005.
- [62] Zhu Han, Zhu Ji, and K. J. Ray Liu. A cartel maintenance framework to enforce cooperation in wireless networks with selfish users. *IEEE Trans. Wireless Commun.*, 7:1889–1899, 2008.
- [63] H. M. Hang and J. J. Chen. Source model for transform video coder and its application - part i: Fundamental theory. *IEEE Trans. Circuits Syst. Video Technol.*, 7:287–298, 1997.
- [64] Zhihai He and S. K. Mitra. A unified rate-distortion analysis framework for transform coding. *IEEE Trans. Circuits Syst. Video Technol.*, 11:1221–1236, 2001.
- [65] X. Hei, C. Liang, J. Liang, Y. Liu, and K. Ross. A measurement study of a large-scale p2p iptv system. *IEEE Trans. Multimedia*, 9(8):1672–1687, 2007.
- [66] Y. Hel-Or and D. Shaked. A discriminative approach for wavelet shrinkage denoising. *IEEE Trans. Image Processing*, 17(4):443–457, 2008.
- [67] K. Hirakawa and T. W. Parks. Joint demosaicking and denoising. *IEEE Trans. on Image Processing*, 15(8):2146–2157, Aug. 2006.
- [68] Cheng Huang, Jin Li, and Keith W. Ross. Can internet video-on-demand be profitable? In *ACM SIGCOMM*, pages 133–144, 2007.
- [69] K. Jensen and D. Anastassiou. Subpixel edge localization and the interpolation of still images. *IEEE Trans. Image Process.*, 4(3):285–295, Mar. 1995.
- [70] F. Jin, P. Fieguth, L. Winger, and E. Jernigan. Adaptive wiener filtering of noisy images and image sequences. In *Proc. IEEE Int. Conf. Image Process.*, pages 349–352, 2003.
- [71] T. Karagiannis, P. Rodriguez, and K. Papagiannaki. Should internet service providers fear peer-assisted content distribution? In *Proceedings of the Internet Measurement Conference*, 2005.
- [72] F. Kelly. Charging and rate control for elastic traffic. *Eur. Trans. Telecommun.*, 28:33–37, 1997.
- [73] C. Kervrann and J. Boulanger. Optimal spatial adaptation for patch-based image denoising. *IEEE Trans. Image Processing*, 15(10):2866–2878, 2006.
- [74] C. Kervrann and J. Boulanger. Local adaptivity to variable smoothness for exemplar-based image regularization and representation. *International Journal of Computer Vision*, 79:45–69, 2008.

- [75] R. G. Keys. Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoust., Speech, Signal Process.*, 29(6):1153–1160, Dec. 1981.
- [76] L. Kontothanassis, R. Sitaraman, J. Wein, D. Hong, R. Kleinberg, B. Mancuso, D. Shaw, and D. Stodolsky. A transport layer for live streaming in a content delivery network. *IEEE Proceedings*, 92:1408–1419, 2004.
- [77] V. Krishna. *Auction Theory*. Academic Press, 2002.
- [78] T. V. Lakshman, A. Ortega, and A. R. Reibman. VBR video: Trade-offs and potentials. In *Proc. IEEE*, 1998.
- [79] X. Wu Lei Zhang and D. Zhang. Color reproduction from noisy cfa data of single sensor digital cameras. *IEEE Trans. on Image Processing*, 16(9):2184–2197, Sept. 2007.
- [80] J. Leskovec. *Dynamics of large networks*. PhD Dissertation, Machine Learning Department, Carnegie Mellon University, Technical report CMU-ML-08-111, 2008.
- [81] M. Li and T. Q. Nguyen. Markov random field model-based edge-directed image interpolation. *IEEE Trans. Image Process.*, 17(7):1121–1128, Jul. 2008.
- [82] X. Li and M. T. Orchard. New edge-directed interpolation. *IEEE Trans. Image Process.*, 10(10):1521–1527, Oct. 2001.
- [83] Z. G. Li, F. Pan, K. P. Lim, X. Lin, and S. Rahardja. Adaptive rate control for H.264. In *Proc. IEEE Int. Conf. Image Processing*, 2004.
- [84] W. S. Lin, H. V. Zhao, and K. J. R. Liu. Incentive cooperation strategies for peer-to-peer live multimedia streaming social networks. *IEEE Trans. on Multimedia*, 11(3):396–412, 2009.
- [85] M. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proc. International Conference on Machine Learning*, 1994.
- [86] Yong Liu, Yang Guo, and Chao Liang. A survey on peer-to-peer video streaming systems. *Journal of Peer-to-Peer Networking and Applications*, 1:18–28, 2008.
- [87] H. V. Madhyastha, T. Isdal, M. Piatek, C. Dixon, T. Anderson, A. Krishnamurthy, and A. Venkataramani. iplane: An information plane for distributed services. In *7th Symposium on Operating Systems Design and Implementation (OSDI)*, 2006.
- [88] S. Marti and H. Garcia-Molina. Limited reputation sharing in p2p systems. In *Proc. of the 5th ACM Conference on Electronic Commerce*, 2004.
- [89] S. Marti, T. J. Giuli, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *Proc. ACM MobiCom*, 2000.

- [90] P. Michiardi and R. Molva. Core: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Proc. IFIP-Comm. and Multimedia Security Conf*, 2002.
- [91] P. Michiardi and R. Molva. A game theoretical approach to evaluate cooperation enforcement mechanisms in mobile ad hoc networks. In *Proc. Workshop Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOPT'03)*, 2003.
- [92] M. E. J. Newman. The spread of epidemic disease on networks. In *Physical Review E*, 2002.
- [93] Martin A. Nowak and Karl Sigmund. Evolution of indirect reciprocity. *Nature*, 437:1291–1298, 2005.
- [94] Hisashi Ohtsuki, Yoh Iwasa, and Martin A. Nowak. Indirect reciprocity provides only a narrow margin for efficiency for costly punishment. *Nature*, 457:79–82, 2009.
- [95] A. Ortega and K. Ramchandran. Rate-distortion techniques in image and video compression. *IEEE Signal Process. Mag.*, 15:25–50, 1998.
- [96] M. J. Osborne and A. Rubinste. *A Course in Game Theory*. The MIT Press, 1994.
- [97] Martin J. Osborne. *An introduction to game theory*. Oxford University Press, 2004.
- [98] H. Park and M. van der Schaar. Bargaining strategies for networked multimedia resource management. *IEEE Trans. Signal Process.*, 55:3496–3511, 2007.
- [99] A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy. A joint inter and intrascale statistical model for bayesian wavelet based image denoising. *IEEE Trans. Image Processing*, 11(5):545–557, 2002.
- [100] Darshan Purandare and Ratan Guha. An alliance based peering scheme for p2p live media streaming. *IEEE Trans. Multimedia*, 9(8):1633–1644, 2007.
- [101] M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley and Sons, 1994.
- [102] P. Raghavan R. Kumar, J. Novak and A. Tomkins. On the bursty evolution of blogspace. In *Proceedings of the 11th international conference on World Wide Web*, page 568576, 2003.
- [103] S. Rajagopalan D. Sivakumar A. Tomkins R. Kumar, P. Raghavan and E. Ufal. Stochastic models for the web graph. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, 2000.

- [104] D. Ray. *A Game-Theoretic Perspective on Coalition Formation*. Oxford University Press, 2007.
- [105] W. Rhee and J. M. Cioffi. Increase in capacity of multiuser OFDM system using dynamic subchannel allocation. In *Proc. IEEE Veh. Technol. Conf.*, 2000.
- [106] Cem U. Saraydar, Narayan B. Mandayam, and David J. Goodman. Efficient power control via pricing in wireless data networks. *IEEE Trans. Commun.*, 50:291–303, 2002.
- [107] S. Seetharaman and M. Ammar. Characterizing and mitigating inter-domain policy violations in overlay routes. In *Proc. of IEEE International Conference on Network Protocols*, 2006.
- [108] L. Shapley. Stochastic games. In *Proceedings of the National Academy of Sciences of USA*, page 10951100, 1953.
- [109] C. Shen and M. van der Schaar. Optimal resource allocation for multimedia applications over multiaccess fading channels. *IEEE Trans. Wireless Commun.*, 7:3546–3557, 2008.
- [110] J. Maynard Smith. *Evolutionary and the theory of games*. Cambridge University Press, 1982.
- [111] S. M. Smith and J. M. Brady. Susan - a new approach to low level image processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.
- [112] V. Srinivasan, P. Nuggehalli, C. F. Chiasserini, and R. R. Rao. Cooperation in wireless ad hoc networks. In *Proc. IEEE INFOCOM*, 2003.
- [113] C. Stein. Estimation of the mean of a multivariate normal distribution. In *Ann. Statist.*, pages 1135–1151, 1981.
- [114] K. Stuhlmuller, N. Farber, M. Link, and B. Girod. Analysis of video transmission over lossy channels. *IEEE J. Sel. Areas Commun.*, 18:1012–1032, 2000.
- [115] Y. Sun, W. Yu, Z. Han, and K. J. R. Liu. Information theoretic framework of trust modelling and evaluation for ad hoc networks. *IEEE Journal of Selected Areas in Communications, special issue on Security in Wireless Ad Hoc Networks*, 24:305–317, 2006.
- [116] Ryan W. Thomas, Daniel H. Friend, Luiz A. DaSilva, and Allen B. MacKenzie. Cognitive networks: Adaptation and learning to achieve end-to-end performance objectives. *IEEE Commun. Mag.*, 44:51–57, December 2006.
- [117] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proc. of the Sixth Int. Conf. on Computer vision*, pages 839–846, 1998.

- [118] A. Urpi, M. Bonuccelli, and S. Giordano. Modeling cooperation in mobile ad hoc networks: A formal description of selfishness. In *Proc. Workshop Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOPT'03)*, 2003.
- [119] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37, 1961.
- [120] D.V.D. Ville, M. Nachtgeael, D.V.D. Weken, E. E. Kerre, W. Philips, and I. Lemahieu. Noise reduction by fuzzy image filtering. Aug. 2004.
- [121] V. Vishumurthy, S. Chandrakumar, and E.G. Siner. Karma: A secure economic framework for peer-to-peer resource sharing. In *Proc. of the 2003 Workshop on Economics of Peer-to-Peer Systems*, 2003.
- [122] B. Wang, K. J. R. Liu, and T. C. Clancy. Evolutionary cooperative spectrum sensing game: How to collaborate? *IEEE Trans. Commun.*, 58(3):890–900, 2010.
- [123] B. Wang, Y. Wu, and K. J. R. Liu. Game theory for cognitive radio networks: An overview. *Computer Networks*, 54(14):2537–2561, Oct. 2010.
- [124] M. Wang and M. van der Schaar. Rate-distortion modeling for Wavelet video coders. In *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2005.
- [125] D. J. Watts and S. H. Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393:440–442, 1998.
- [126] J. Woods and C. Radewan. Kalman filtering in two dimensions. *IEEE Trans. Inform. Theory*, 23:473–482, Jul. 1977.
- [127] Di Wu, Chao Liang, Yong Liu, and Keith W. Ross. View-upload decoupling: A redesign of multi-channel p2p video systems. In *IEEE Conference on Computer Communications (IEEE INFOCOM '09), Mini-Conference*, 2009.
- [128] Min Xiao and Debao Xiao. Understanding peer behavior and designing incentive mechanism in peer-to-peer networks: An analytical model based on game theory. In *Proc. of International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP)*, 2007.
- [129] Haiyong Xie, Y. Richard Yang, Arvind Krishnamurthy, Yanbin Liu, and Avi Siberschatz. P4p: Provider portal for applications. In *ACM SIGCOMM*, pages 351–362, 2008.
- [130] W. Yu and K. J. Ray Liu. Attack-resistant cooperation stimulation in autonomous ad hoc networks. *IEEE J. Sel. Areas Commun.*, 23:2260–2271, Dec. 2005.

- [131] W. Yu and K. J. Ray Liu. Game theoretic analysis of cooperation stimulation and security in autonomous mobile ad hoc networks. *IEEE Trans. Mobile Comput.*, 6:459–473, May 2007.
- [132] L. Zhang and X. Wu. Image interpolation via directional filtering and data fusion. *IEEE Trans. Image Process.*, 15(8):2226–2238, Aug. 2006.
- [133] X. Zhang, J. Liu, B. Li, and T-S. P. Yum. Coolstreaming: A data-driven overlay network for peer-to-peer live media streaming. In *IEEE Conference on Computer Communications (INFOCOM)*, pages 2102–2111, 2005.
- [134] X. Zhang, A. Vetro, Y. Q. Shi, and H. Sun. Constant quality constrained rate allocation for FGS video coded bitstreams. In *Proc. SPIE Conf. on Visual Comm. and Image Processing (VCIP)*, 2002.
- [135] X. Zhang and X. Wu. Image interpolation by adaptive 2-d autoregressive modeling and soft-decision estimation. *IEEE Trans. Image Process.*, 17(6):887–896, June 2008.
- [136] H. Zhao, W.S. Lin, and K. J. R. Liu. Behavior modeling and forensics for multimedia social networks: A case study in multimedia fingerprinting. *IEEE Signal Process. Mag.*, 26(1):118–139, Jan. 2009.
- [137] X. J. Zhao, Y. W. He, S. Q. Yang, and Y. Z. Zhong. Rate allocation of equal image quality for MPEG-4 FGS video streaming. In *Proc. Packet Video (PV)*, 2002.
- [138] S. Zhong, J. Chen, and Y. R. Yang. Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks. In *Proc. IEEE INFOCOM*, 2003.
- [139] S. Zhong, L. Li, Y. G. Liu, and Y. R. Yang. On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks - an integrated approach using game theoretical and cryptographic techniques. In *Proc. ACM MobiCom*, 2005.