MEAN FIELD GAME THEORY FOR FUTURE HETEROGENEOUS AND HIERARCHICAL COMMUNICATION NETWORKS

by

Reginald A. Banez

A dissertation submitted to the Department of Electrical and Computer Engineering

Cullen College of Engineering

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in Electrical Engineering

Chair of Committee: Dr. Zhu Han

Committee Member: Dr. Miao Pan

Committee Member: Dr. Saurabh Prasad

Committee Member: Dr. Ekram Hossain

Committee Member: Dr. Wotao Yin

University of Houston

May 2020

Acknowledgements

I thank each and every one who has helped me during my Ph.D. study. My life as a graduate student has been extraordinary because of your presence and support.

First, I would like to express my sincerest gratitude and appreciation to my advisor, Dr. Zhu Han, for his outstanding guidance, unwavering support, and endless compassion during my graduate studies. His excellent academic background and smart intuitions have helped me immensely in my research studies. I have learned a lot from him, and I am extremely proud to have been one of his students.

I would also like to acknowledge my dissertation committee members, Dr. Miao Pan, Dr. Saurabh Prasad, Dr. Ekram Hossain, and Dr. Wotao Yin, who provided continuous encouragement and valuable advice in my research. Furthermore, I would like to thank all other scholars I have collaborated with, Dr. H. Vincent Poor, Dr. Hamidou Tembine, Dr. Lixin Li, Dr. Chungang Yang, Dr. Lingyang Song, Dr. Haitao Xu, Dr. Nguyen H. Tran, Dr. Ju Bin Song, and Dr. Choong Seon Hong. I appreciate their time and effort in helping me with my research works.

My gratitude also goes to my dear friends and colleagues at Wireless Networking, Signal Processing and Security Lab, Dr. Huaqing Zhang, Dr. Mounika Sai Errapotu, Dr. Neetu Raveendran, Dr. Jingyi Wang, Dr. Hongliang Zhang, Dr. Kevin Tsai, Xunsheng Du, Mohamed Ali ElMossallamy, Hao Gao, Yuhan Kang, and many others for creating a pleasant research environment.

Finally, I would like to express my sincerest gratitude and appreciation to my parents, Rolando and Regina, and my siblings Rolando Jr., Reena Kaye, and Ranier, for their everlasting love and encouragement. Thank you for motivating me to always do my best and making my life special.

Abstract

According to the latest Cisco Annual Internet Report, the current communication network infrastructures are approaching their limits caused by the increasing data traffic, more frequent network usage, and rising number of connected devices. In order to overcome these limitations, enabling technologies such as ultra-dense networks, multi-access edge networks, and massive antenna arrays are proposed as part of the future generation of communication networks. However, in order to analyze, model, and simulate these technologies, an appropriate mathematical framework that can handle large number of interacting entities is necessary. Hence, the application of mean field games (MFGs) to future communication networks is proposed in this dissertation.

The first work of this dissertation deals with the modeling of user behavior through belief and opinion evolution in social networks, which is essential in improving the services provided by a network provider. A multiple-population MFG approach is applied to depict the behavior of social network users in a multiple-group setting. Theoretical and experimental simulations using a social evolution dataset suggest the effectiveness of the MFG approach in estimating and predicting the distribution of belief and opinion in social networks. The second work of this dissertation investigates an effective and efficient method for computation offloading in multi-access edge computing networks (MECN). A mean-field-type game (MFTG) approach is utilized to design non-cooperative and cooperative computation offloading algorithms to decrease latency and energy consumption. The results indicate that the proposed MFTG-based algorithms can optimize energy consumption and latency associated with computation offloading. Then, the third work of this dissertation presents a proposed dynamic hierarchical framework for resource allocation in network virtualization.

Lastly, this dissertation is concluded with a summary of important results and remarks. Furthermore, future works integrating MFG in unmanned aerial vehicle (UAV) networks, network virtualization, and Internet-of-Things (IoT) are proposed.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 The Future of Wireless Communication Networks

The current generation of wireless networks allows users to access a variety of services such as data, voice, and multimedia. However, according to the latest Cisco Annual Internet Report [1], these networks are reaching their full potential caused by the growth of data traffic, more frequent network usage, and rise in number of devices and connections. Consequently, these circumstances have inspired interest and necessity for a new generation of wireless networks that can support the growing number of devices and connections as well as the volume of data they generate.

The main goal of the next generation of wireless networks is to improve the implementation of current wireless networks as well as the services provided to the network users. In 5G, the networks are expected to handle high capacity, high data rate, low latency, massive connectivity, increased network reliability, and improved energy efficiency [2]. In 6G, the networks are expected to improve over the 5G networks and support new applications such as extended reality (XR), wireless brain-computer interactions, and connected robotics and autonomous systems [3]. Researchers have proposed technologies that can enable the proposed enhancements. For 5G, some of these technologies include massive MIMO, millimeter wave communications, ultra-dense networks, spectrum sharing, device-to-device communications, full-duplex communications, and cloud technologies [4]. For 6G, the proposed technologies are communication with large intelligent surfaces, integrated terrestrial, aerial, and satellite networks, and energy transfer and harvesting [3].

This dissertation addresses some of these issues and technologies such as the analysis and modeling of user behavior for increased network reliability in multiple-population social networks, and the energy-efficient and low-latency computation offloading in multi-access edge computing networks. To be able do so, the applications of mean field games (MFGs) in analyzing, modeling, and simulation of future heterogeneous and hierarchical communication networks have been the main focus of this dissertation. Proposed future works including network virtualization for cloud and/or core networks and pursuit and evasion games involving multiple unmanned aerial vehicles are also discussed.

1.1.1 Social Networks

Social networks (SNs) are sets or groups of people with some patterns of contacts or interactions between them, forming meaningful social relationships [5]. These networks have evolved from simple web-based forums to ubiquitous mobile networks. As outlined in [6], the advancement of the internet has given rise to web-based social networks such as forums and chat rooms. Meanwhile, the advent of peer-to-peer networks and dedicated social websites has caused a huge number of people to use social networking through online social networks (OSNs) such as Facebook, Twitter, WeChat, and LinkedIn. OSNs allow users sharing common interests to form online connections and social communities. Moreover, the increasing capability of mobile devices has made way for the integration of social networks in mobile environments. Such networks, called mobile social networks (MSNs), allow users to access, share, and distribute data by exploiting social relations [7]. These relationships between social network users include physical contacts, financial exchanges, group participations, among others [8].

One important aspect of social networks is understanding user behavior that is essential to social network entities such as the service providers and network users. In the case of OSNs, the study of user behavior is crucial to different internet entities [9]. It helps internet service providers (ISPs) learn the evolution of traffic patterns in OSNs. Meanwhile, it provides OSN service providers knowledge about their users' behavior toward different situations. Finally, it enhances the experience of social network users. One way of understanding the behavior of social network users is to study the evolution of belief and opinion of these users regarding a social topic or issue. Hence, many researchers have studied and developed mathematical models for belief and opinion dynamics in social networks. These models attempt to capture emerging trends caused by the interactions and decisions of the users [10].

Numerous works in the literature have focused on the application of statistical methods, optimization techniques, and game theory to belief and opinion evolution in social networks. Specifically, game theory models have been an attractive approach since they deal with optimization involving several decision makers with conflicting interests. Game theory has also been an effective tool in modeling and optimizing physical networks such as wireless and communication networks with competing and/or cooperating network entities [11]. Furthermore, game theory has been used to improve the performance of wireless networks through efficient and effective allocation of network resources [12].

However, as the number of decision makers grows large, the analysis of a problem becomes more complicated due to the increasing number of interactions among the decision makers. Thus, an appropriate mathematical framework is necessary to ease the analysis and the calculation of the solution to the problem. MFG, a subclass of game theory introduced by Lasry and Lions [13], was proposed to emulate scenarios involving a large number of decision makers. MFG models the interaction of a decision maker with the collective behavior of other decision makers in the game rather than with the individual behavior of each and every decision maker in the game. Meanwhile, a new MFG framework proposed in [14] concentrates on multiple-population settings where the populations are competing and each player in a population are acting non-cooperatively. This kind of scenarios can be found in social networks. For instance, social network users can be divided according to political orientation, gender, age, and many other categories. Since users can interact with anyone belonging to any group, it would be beneficial to study how these interactions affect the belief and opinion of these users.

Inspired by this new research in MFG, Chapter 2 introduces a multiple-population MFG to model the belief and opinion evolution in social networks [15]. The proposed model aims to gain information on the behavior of social network users belonging to different groups. Specifically, the proposed model can be utilized to estimate and predict how a social network group affects the belief and opinion of other groups.

1.1.2 Multi-Access Edge Computing Networks

Mobile cloud computing (MCC) offers cloud services such as computing, caching, and communications to mobile end users. These services are processed in the cloud that might be geographically located far from the end users. Consequently, MCC suffers from high latency that is not acceptable in some applications. To alleviate this problem, multi-access edge computing (MEC) has been proposed in which the cloud services are provided at the edge network located in proximity with the end users. Therefore, the move of computing services from the cloud to the edge network effectively reduces latency. Aside from low latency, other benefits of MEC include proximity, high bandwidth, real-time radio network information, and location awareness [16]. MEC can be implemented through a network of computing nodes distributed over a geographic area. These computing nodes form the multi-access edge computing network (MECN) that provides computing services to the end users.

Computation offloading is one of the main services provided by an MECN where an end user equipment, such as a smartphone, can offload computation-intensive tasks, or portions of it, to the MECN instead of performing the task locally. The decision to offload depends on factors such as latency, bandwidth, and energy consumption of the equipment. In the literature, computation offloading has been formulated as a game theoretic problem or an optimization problem with the goal of minimizing the cost incurred by a mobile device or the network subject to constraints such as computing power, latency, and bandwidth.

In Chapter 3, the idea of computation offloading is extended to offloading among computing nodes [17]. Specifically, an MECN aggregates the computation tasks from the end users and then it offloads portions of the aggregated tasks to the computing nodes. Since an MECN can be implemented through finite number of computing nodes where a computing node can have a significant effect on the utility of the network, the mean-field-type game (MFTG), a relaxed version of MFG, is utilized to analyze and model the computation offloading problem. Specifically, non-cooperative and cooperative MFTG approaches are applied to computation offloading in MECN. In these approaches, the goal of each computing node is to minimize cost by controlling its own offloading strategy subject to the state dynamics of the MECN. Each computing node does not need to know the offloading strategy of other computing nodes in order to determine its own offloading strategy. Instead, a computing node only needs to know the mean field terms that correspond to the aggregate effect of other computing nodes to the network.

1.1.3 Network Virtualization

The internet has been successful in providing users access to information and communication through an assortment of applications. These applications include web browsing, voice and data communication, video streaming, among others. The service providers (SPs) offer these applications through networks that vary in topology and architecture. The infrastructure providers (InPs), entities that manage a network of physical infrastructures, can support these multiple diverse networks through network virtualization. Network virtualization (NV) allows multiple heterogeneous virtual networks (VNs) to share common physical network (PN) infrastructures. In the case of the internet, it will enable the support of multiple independent VNs which in turn offer various applications to the end users (EUs).

There are many advantages to implementing NV including flexibility, diversity, security, and manageability [18]. However, NV faces some challenges such as the allocation of network resources and the economic relationship between the SPs and the InPs. Hence, researchers have developed algorithms and models that solve NV resource allocation problems. Resource allocation in NV is performed by the InPs after receiving resource requests from the SPs [19]. Thus, the main focus of many works has been the dynamics between the SPs and the InPs. The lack of a unified framework for NV that involves all levels of abstraction of NV (e.g., the EUs, the SPs, and the InPs) has inspired us to develop an economic model that represents the dynamics within the entire NV system. The dynamics in an NV system include the failure, payoff, competition, and service rates at each level of abstraction. This model depicts not only the economic relationship among the InPs and the SPs but also the economic relationship among the SPs and the EUs.

In Chapter 4, a unique approach to NV resource allocation and economics is presented. NV is a dynamic system where the quantities of resource trading between the consumers and the producers are constantly changing with time. In addition, NV has multiple levels of interactions that are difficult to analyze and model using a conventional game theory approach. Thus, a dynamic hierarchical model, called the prey-predator food chain model, is used to analyze and model the dynamic resource allocation and economics in a multiple-level NV system. Consider an ecosystem with different species living together where a species is the only food source of one other species which in turn is the only food source of one other species and so on (i.e., a food chain). The interactions among many agents of the same species (i.e., competition) and the interactions between two agents of different species (i.e., consumption) are modeled mathematically in the prey-predator food chain model. In NV, the EUs, SPs, and InPs are the agents or decision makers while the gain or loss of money depicts the interaction between any two decision makers. When the prey-predator food chain model is applied to NV, the resulting model represents the dynamic resource allocation and economics in an NV system.

1.2 Mean Field Games

1.2.1 Background

Mean field game (MFG) theory is introduced by Lasry and Lions in [13]. It deals with the study and analysis of differential games with infinitely many decision makers or players. It enables the study of the solution concept of Nash equilibrium of games with very large number of indistinguishable players. The term *indistinguishable* refers to a scenario where players share common structures of the model and are allowed to have heterogeneous states [20]. An MFG is typically formulated as a couple of differential equations, a Fokker-Planck-Kolmogorov (FPK) equation for the evolution of state distribution of the players and a Hamilton-Jacobi-Bellman (HJB) equation for the evolution of a player's value function given the state distribution of all other players. Since the number of players in an MFG is large and the players are indistinguishable, the game is taken from the point of view of a *representative* or *reference* player playing against the aggregate or collective behavior of other players in the game.

Consider a non-cooperative game with large number of indistinguishable players. Let x be the state of a player and $m = m(x,t)$ be the *mean field* or distribution of players over state x at time t. Assume that a representative player tries to minimize the cost $J(x,m,u)$ subject to the state dynamics $\dot{x}$ through the input or control $u = u(x,t)$. That is,

$$\begin{aligned} \min_{u \in \mathcal{U}} \qquad & J(x,m,u) = \mathbb{E}\left[\int_0^T L(x,m,u)\,dt + \Psi(x,T) \right], \\ \text{subject to} \quad & dx = f(x,m,u)\,dt + \sigma\,dw(t), \end{aligned} \tag{1.1}$$

where $L(x,m,u)$ corresponds to the running cost, $\Psi(x,T)$ refers to the terminal cost or the cost at terminal time T, $f(x,m,u)$ denotes the average rate of change of the state, σ as the diffusion constant, and $w(t)$ as a standard Wiener process. Also, define the value function $v(x,t) = \min_{u\in\mathcal{U}} J(x,m,u)$. Then, an MFG can be expressed as a pair of HJB and FPK equations,

$$
\begin{aligned}
-\frac{\partial v}{\partial t}(x,t) - H(x,m,p) &= \frac{\sigma^2}{2} D^2 v(x,t), \\
\frac{\partial m}{\partial t}(x,t) + \operatorname{div}(f(x,m,u)m(x,t)) &= \frac{\sigma^2}{2} D^2 m(x,t),
\end{aligned}
\tag{1.2}
$$

with boundary conditions $v(x,T) = \Psi(x,T)$ and $m(x,0) = m_0(x)$. The first equation in (1.2) is the HJB equation that characterizes the optimized reaction of a representative player against the mean field while the second equation in (1.2) is the FPK equation that describes the evolution of the mean field of the players that behave optimally [20]. Meanwhile, $H(x,m,p)$ is called the Hamiltonian and is defined mathematically as

$$
H(x,m,p) = \min_{u \in \mathcal{U}} L(x,m,u) + pf(x,m,u), \tag{1.3}
$$

where $p = \frac{\partial v}{\partial x}(x,t)$.

The optimal control $u^* = u(x^*,t)$ is the input or control that achieves the minimum cost $J(x,m,u^*)$. It satisfies the equation

$$
J(x,m,u^*) = \mathbb{E}\left[\int_0^T L(x,m,u^*)\,dt + \Psi(x,T)\right] = v(x,t),
$$

$\forall t \in [0,T]$, with x^* as the solution to the state dynamics $\dot{x}$. Furthermore, the optimal control u^* is the solution to the partial differential equation

$$
f(x,m,u^*) = \frac{\partial H}{\partial p}(x,m,p). \tag{1.4}
$$

A related discussion of MFG utilized in this work can be found in Chapter 2.

1.2.2 Applications

MFGs have been applied in many applications in economics and engineering. Recently, the applications of MFGs in engineering include power control in device-to-device (D2D) networks [21], wireless communications in multiple unmanned aerial vehicle (UAV) networks [22], real-time D2D streaming networks [23], energy harvesting in ultra-dense small cell networks [24], power supply and interference mitigation in wireless-powered Internet-of Things (WP-IoT) [25], future ultra-dense wireless networks [26], electric vehicle competition in smart grids [27], security enhancements in mobile ad-hoc networks [28], and power allocation in full-duplex ultra-dense cellular networks [29].

1.3 Dissertation Contributions and Organization

This dissertation consists of three main works. The first work in Chapter 2 focuses on the MFG modeling of belief and opinion evolution in social networks. The main contributions of this work are summarized as follows.

- The belief and opinion evolution in a social network with a large number of users is formulated as a single population MFG. In this model, the users are assumed to share a common characteristic, and hence can be grouped into a single population. The proposed MFG model aims to provide the evolution with time of the distribution of opinions within the population.

- For social networks that can be divided into several populations or groups where users in a group share a common characteristic, the belief and opinion evolution is modeled using a multiple-population MFG framework. The goal of the proposed model is to capture the evolution of the distributions of opinions within the populations. Moreover, the proposed model aims to show the influence of the populations with each other.

- The existence of stubborn users in a social network is integrated in the proposed MFG models. Stubborn users refer to social network users that are less susceptible to change their initial opinions on a social topic and issue. Consequently, the resulting models indicate how these users affect the behavior of the populations.

- Simulations are provided to demonstrate the purpose and benefits of the proposed MFG models in estimating and predicting the distribution of opinions of social network users. First, theoretical results from implementing the proposed models are shown. These results are important in showcasing the theoretical aspect of the MFG-based belief and opinion evolution. Then, the proposed models are tested using a social evolution dataset [30]. The results provide insights on the effectiveness of the proposed MFG-based models.

The second work in Chapter 3 concentrates on the MFTG approach to computation offloading in MECN. The main contributions of this work are summarized as follows.

- Computation offloading is proposed and formulated as a non-cooperative MFTG problem where each computing node minimizes its own cost function subject to the state dynamics of the network. In the non-cooperative approach, the computing nodes operate in a decentralized manner where each computing node can compute its own computation offloading strategy without full knowledge of the strategies of other computing nodes.

- A cooperative MFTG problem of computation offloading is proposed and formulated where the computing nodes jointly minimize a global cost function subject to the state dynamics of the network. In the cooperative approach, the computing nodes operate in a centralized manner where the network determines the offloading strategy of each computing node that minimizes the cost incurred by the edge network.

- The optimal computation offloading control profile that minimizes the cost in each case is solved using a direct approach proposed in the literature. This approach does not require solving coupled partial differential equations that complicate the problem. Instead, it involves calculation of mean-field terms that represent the behavior of the entire network.

- The non-cooperative and cooperative MFTG computation offloading algorithms are designed based on the direct approach of solving MFTG computation offloading problems. The non-cooperative algorithm is implemented in a decentralized manner while the cooperative algorithm is implemented in a centralized manner.

- Simulations are presented to demonstrate the effectiveness of the proposed MFTG-based algorithms as well as the computation offloading behavior of a computing node under varying conditions. Moreover, the proposed non-cooperative and cooperative MFTG computation offloading algorithms are compared with typical computation offloading algorithms.

The third work in Chapter 4 emphasizes a dynamic hierarchical modeling of resource allocation and economics in network virtualization. The main contributions of this work are summarized as follows.

- A unified mathematical framework based on the prey-predator food chain model is proposed to represent the economics in an NV system. An analysis of this model is provided, as well as its properties and equilibrium point. This equilibrium point stands as the resource strategies at each level of the NV system given the economic dynamics of the system.

- Economic models are formulated based on the dynamics in the NV system to emulate the relationship between the total demand for the network resources and the total supply of the network resources.

- Simulations are provided to show the variations in the behavior of the NV system when the dynamics within the system have changed. Furthermore, results are presented to illustrate the relationship of the demand and supply of network resources between the EUs and the SPs and between the SPs and the InPs.

The organization of this dissertation is as follows. Chapter 2 presents the MFG-based modeling of belief and opinion evolution in multiple-population social networks. Then, Chapter 3 discusses the MFTG-based approaches to computation offloading in MECN. Meanwhile, Chapter 4 provides an introduction to a dynamic hierarchical modeling of resource allocation and economics in NV. Finally, Chapter 5 offers the conclusions and proposed future works.

Chapter 2

Multiple-Population Mean Field Belief and Opinion Evolution in Social Networks

The number of users engaged in social media through social networks continues to grow as people become more passionate on current social topics and issues. When social network users share similar characteristics such political orientation, age, and gender, they can be grouped and analyzed according to these similarities. Meanwhile, the behavior of users in a social network setting can be extracted from their belief and opinions regarding social topics and issues. Understanding user behavior has been a priority of network service providers in order to learn the evolution of traffic patterns, estimate or predict the users behavior, and enhance the experience of network users. Integrating the idea of user behavior modeling to a multiple-group social network, a multiple-population mean field game (MFG) approach is proposed to analyze and model the belief and opinion evolution of users in a social network. Through the proposed model, information can be gained on the behavior of social network users belonging to different groups. Specifically, the proposed model can be utilized to estimate and predict how a social network group affects the belief and opinion of other groups. Simulations are provided to show the belief and opinion evolution of users in a multiple-population setting. Theoretical results and experimental results using a social evolution dataset are presented to demonstrate the effectiveness of the proposed MFG approach.

2.1 System Model

In this section, the opinion dynamics equation that characterizes the change or update of opinion of a user in a single-population social network is derived. By single population, it is assumed that the users share a common characteristic (e.g., age, gender, and political orientation). Then, the cost function that each user is facing in a social network is described.

In the following derivations, consider a social network with n users denoted by graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of all social network users and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of all links between two distinct users in $\mathcal{V}$. Two distinct users that are linked are neighbors, and hence the neighborhood of a user refers to the set of all its neighbors. Also, denote $x_i, i = 1, \ldots, n$ as the state or scalar opinion of user i in a social network.

2.1.1 Opinion Dynamics Equation

The state or opinion x_i of user i evolves with time k depending on the opinions of its neighbors. In this work, the following opinion dynamics equation is adapted where at each stage $k = 0, 1, \ldots,$ the opinion of user i evolves according to

$$x_i(k+1) = \frac{\sum_{j=1}^{n} \alpha_{ij} x_j(k)}{\sum_{j=1}^{n} \alpha_{ij}} + \beta_i u_i. \tag{2.1}$$

The first term refers to the Hegselmann-Krause (HK) model of opinion evolution [31] where α_{ij} denotes the adjacency between users i and j, with $\alpha_{ij} = 1$ if users i and j are connected and $\alpha_{ij} = 0$ if they are not connected. Therefore, α_{ij} determines the neighbors of user i. The second term has been added to reflect the change in opinion based on the control effort $u_i = u_i(x_i, k)$ with β_i as a constant [32]. Fig. 2.1a shows the system model for a single-population social network. The blue-colored users belong to the neighborhood of user i while the gray-colored users are outside its neighborhood.

Social network users can be stubborn which means they consider their initial belief or opinion together with the belief or opinion of their neighbors. A popular social network

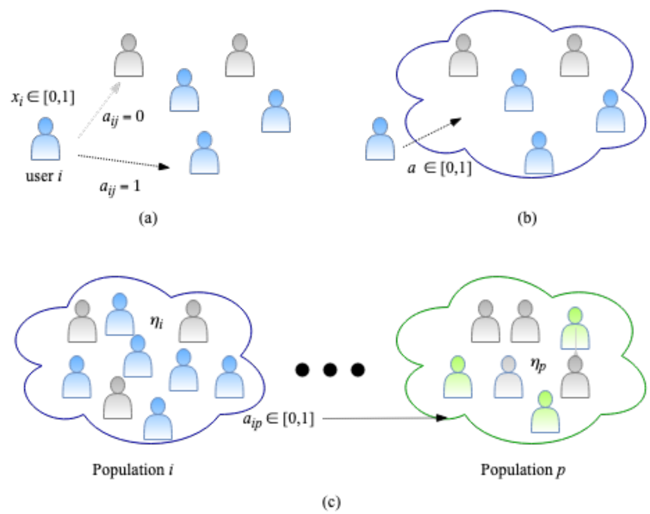


Figure 2.1: An illustration of a social network with single and multiple populations.

dynamic model that considers stubborn agents is the FJ model [33]. In the FJ model, the influence of neighbor j to user i is quantified by the interpersonal influence w_{ij}, which refers to the strength of influence of between social network users i and j. Moreover, the willingness of user i to external influence is denoted by λ_i, which refers to the likeliness of user i to change from its prejudice or initial opinion $x_{i,0}$. The evolution of opinion of user i according to the FJ model is given by

$$x_i(k+1) = \lambda_i \sum_{j=1}^{n} w_{ij} x_j(k) + (1 - \lambda_i) x_{i,0}. \tag{2.2}$$

In this work, the HK model in (2.1) is integrated to (2.2) such that the opinion of user i depends on the average of the opinion of its neighbors. Thus, the opinion dynamics equation for social networks with stubborn agents in (2.1) is equivalent to

$$x_i(k+1) = \lambda_i \frac{\sum_{j=1}^{n} \alpha_{ij} x_j(k)}{\sum_{j=1}^{n} \alpha_{ij}} + (1 - \lambda_i) x_{i,0} + \beta_i u_i, \tag{2.3}$$

where the interpersonal influence w_{ij} in (2.2) has been expressed as $w_{ij} = \alpha_{ij} / \sum \alpha_{ij}$.

In continuous time, the corresponding opinion dynamics equation for social networks without stubborn agent and with stubborn agents are, respectively,

$$\frac{x_i(t+h)-x_i(t)}{h} = \frac{1}{h}\left(\frac{\sum_{j=1}^{n}\alpha_{ij}x_j(t)}{\sum_{j=1}^{n}\alpha_{ij}} - x_i(t) + \beta_i u_i\right), \tag{2.4}$$

and
$$\frac{x_i(t+h)-x_i(t)}{h} = \frac{1}{h}\left(\lambda_i\frac{\sum_{j=1}^{n}\alpha_{ij}x_j(t)}{\sum_{j=1}^{n}\alpha_{ij}} + (1-\lambda_i)x_{i,0} - x_i(t) + \beta_i u_i\right). \tag{2.5}$$

Getting the limit of (2.4) and (2.5) as $h \to 0$ yields the continuous-time opinion dynamics,

$$dx_i = \frac{1}{h}\left(\frac{\sum_{j=1}^{n}\alpha_{ij}x_j}{\sum_{j=1}^{n}\alpha_{ij}} - x_i + \beta_i u_i\right)dt = f_i(x_i,u_i)\,dt, \tag{2.6}$$

and
$$dx_i = \frac{1}{h}\left(\lambda_i\frac{\sum_{j=1}^{n}\alpha_{ij}x_j}{\sum_{j=1}^{n}\alpha_{ij}} + (1-\lambda_i)x_{i,0} - x_i + \beta_i u_i\right)dt = f_i(x_i,u_i)\,dt. \tag{2.7}$$

2.1.2 Cost Function

Let $u_i = u_i(x_i, t)$ be the control or influence effort of user i with opinion x_i at time t. In a social network, a user can influence anybody in the network through various ways such as friendship links, advertisements, and social network posts. By controlling u_i, user i can minimize its own local cost function $J_i(x_i, u_i)$ that refers to the accumulation of the running cost $L_i(x_i, u_i)$ from $t = 0$ to T,

$$J_i(x_i,u_i) = \int_0^T L_i(x_i,u_i)\,dt. \tag{2.8}$$

The running cost $L_i(x_i, u_i)$ consists of the following: the control effort cost associated with the influence $u_i(x_i, t)$; the popularity cost corresponding to the cost or reward when user i has different or the same opinion as other users; and the opinion cost that depends on how far opinion x_i is from the desired or ideal opinion x_d of the population. Therefore,

$$L_i(x_i,u_i) = c_1u_i^2 + c_2\left(\frac{1}{n}\sum_{j=1}^{n}\delta_{x_j=x_i}\right) + c_3\left\|x_i - x_d\right\|, \tag{2.9}$$

where $\delta = 1$ if $x_j = x_i$ and $\delta = 0$ if $x_j \neq x_i$, and c_1, c_2, and c_3 are constants. When stubborn agents are considered, the cost function will include a trade-off between the distance of opinion x_i to the ideal or target opinion x_d and to the initial opinion $x_{i,0}$. Thus,

$$L_i(x_i, u_i) = c_1 u_i^2 + c_2 \left(\frac{1}{n} \sum_{j=1}^{n} \delta_{x_j = x_i} \right) + c_3 \lambda_i \left\| x_i - x_d \right\| + c_4 (1 - \lambda_i) \left\| x_i - x_{i,0} \right\|, \tag{2.10}$$

where c_4 is a constant.

The goal of each social network user i is to vary its control u_i to minimize its cost $J_i(x_i, u_i)$ subject to its opinion dynamics dx_i,

$$\begin{aligned} \min_{u_i \in \mathcal{U}_i} \quad & J_i(x_i, u_i), \\ \text{subject to} \quad & dx_i = f_i(x_i, u_i)\, dt. \end{aligned} \tag{2.11}$$

In the following section, the concepts and motivation behind MFGs are introduced. Then, the opinion dynamics and cost functions are reformulated using an MFG framework.

2.2 Mean Field Game with Single Population

2.2.1 Background and Motivation

Consider a non-cooperative game with n players. The goal of each player i is to find the control $u_i = u_i(x_i, t)$ that minimizes its cost $J_i(x_i, u_i)$ subject to the state dynamics equation dx_i,

$$\begin{aligned} \min_{u_i \in \mathcal{U}_i} \quad & J_i(x_i, u_i) = \mathbb{E}\left[\int_0^T L(x_i, u_i)\, dt + \Psi_i(x_i, T) \right], \\ \text{subject to} \quad & dx_i = f_i(x_i, u_i)\, dt + \sigma_i\, dw_i(t). \end{aligned} \tag{2.12}$$

Any control $u_i^* = u_i(x_i^*, t)$ that solves the optimization problem in (2.12) is called an optimal control of player i where x_i^* is a solution to dx_i. An optimal control u_i^* is a solution to (2.12) that minimizes the cost function subject to the state dynamics constraint. The formal definition of optimal control is stated as follows.

Definition 1. *Let the value function $v_i(x_i, t)$ be defined mathematically as*

$$v_i(x_i, t) = \min_{u_i \in \mathcal{U}_i} J_i(x_i, u_i),$$

$\forall t \in [0, T]$. *Then, the optimal control u_i^* satisfies*

$$J_i(x_i, u_i^*) = \mathbb{E}\left[\int_0^T L(x_i, u_i^*)\, dt + \Psi_i(x_i, T) \right] = v_i(x_i, t),$$

$\forall t \in [0, T]$.

Meanwhile, if the value function $v_i(x_i, t)$ associated with the optimization problem in (2.12) is smooth enough (i.e., has continuous first-order derivative), then it is a viscosity solution of the Hamilton-Jacobi-Bellman (HJB) equation [34],

$$\frac{\partial v_i}{\partial t}(x_i, t) + H(x_i, p_i) = \frac{\sigma_i^2}{2} D^2 v_i(x, t), \tag{2.13}$$

where $H_i(x_i, p_i)$ is called the Hamiltonian and is defined mathematically as

$$H(x_i, p_i) = \min_{u_i \in \mathcal{U}_i} L_i(x_i, u_i) + p_i f_i(x_i, u_i), \tag{2.14}$$

and $p_i = \frac{\partial v_i}{\partial x_i}(x_i, t)$.

When every player i faces the optimization problem in (2.12) and hence the corresponding HJB equation in (2.13), the resulting game is called a differential game [11]. Furthermore, when every player i solves its optimal control u_i^*, then it yields a solution to the game called Nash equilibrium. A Nash equilibrium is a solution concept to a game where no player can achieve a lesser cost by deviating from its optimal control given that other players maintain their optimal controls. A formal definition is given as follows.

Definition 2. *Let (u_i^*, u_{-i}^*) be the control profile consisting of the control of player i and the controls u_{-i} of other users. Then, the control profile (u_i^*, u_{-i}^*) is a Nash equilibrium if* *and only if*

$$J_i(u_i^*, u_{-i}^*) \leq J_i(u_i, u_{-i}^*), \forall u_i \in \mathcal{U},$$

for every player i in the game.

The calculation of the Nash equilibrium of a differential game with n players involves solving n coupled HJB equations in (2.13). Hence, it becomes more complicated due to increased interactions and coupling between the players. Consequently, mean field game (MFG) is proposed to reformulate the game problem. MFG is introduced by Lasry and Lions in [13] and has been applied in many applications in economics and engineering. MFG can be utilized when the number of players is large and when the players are indistinguishable yet can have heterogeneous states. In MFG, the aggregate effect of all the players is considered rather than the individual effect of each player.

A MFG can be characterized by a pair of partial differential equations, a HJB equation and a Fokker-Planck-Kolmogorov (FPK) equation. This is in contrast with a differential game that is characterized by n coupled HJB equations. Since the number of players in a MFG is large and the players are indistinguishable, the game is taken from the point of view of a representative or reference player playing against the aggregate or collective behavior of other players. Thus, MFG can be expressed as a pair of HJB and FPK equations,

$$\begin{aligned} -\frac{\partial v}{\partial t}(x,t) - H(x,m,p) &= \frac{\sigma^2}{2} D^2 v(x,t), \\ \frac{\partial m}{\partial t}(x,t) + \operatorname{div}(f(x,m,u)m(x,t)) &= \frac{\sigma^2}{2} D^2 m(x,t), \end{aligned} \tag{2.15}$$

where $H(x,m,p)$ is called the Hamiltonian and is defined mathematically as

$$H(x,m,p) = \min_{u \in \mathcal{U}} L(x,m,u) + pf(x,m,u), \tag{2.16}$$

and $p = \frac{\partial v}{\partial x}(x,t)$, with boundary conditions $v(x,T) = \Psi(x,T)$ and $m(x,0) = m_0(x)$. The first equation in (2.15) is the HJB equation that characterizes the optimized reaction of a player with the mean field while the second equation in (2.15) is the FPK equation that describes the evolution of the population that behaves optimally [20]. Note that in single-population MFG, the subscripts i have been dropped to imply that the players are indistinguishable from each other.

Meanwhile, the optimal control u^* is the solution to the partial differential equation

$$f(x, m, u^*) = \frac{\partial H}{\partial p}(x, m, p). \tag{2.17}$$

The function $m = m(x, t)$ refers to the mean field and it corresponds to the distribution of the states of the players with respect to time. The formal definition of $m(x, t)$ is given as follows.

Definition 3. *The mean field $m(x, t)$ denotes the probability distribution of the players with state x at time t. Mathematically,*

$$m(x, t) = \lim_{n \to \infty} \frac{1}{n} \left(\sum_{i=1}^{n} \delta_{x_i = x} \right), \tag{2.18}$$

where $\delta = 1$ if $x_i = x$ and $\delta = 0$ if $x_i \neq x$.

The following theorem summarizes the mathematical framework of a MFG. A brief proof is provided as well. For more comprehensive proof, the reader may refer to [20] and [35].

Theorem 1. *Consider a non-cooperative game among large number of indistinguishable players. If every player faces the optimization problem*

$$\min_{u \in \mathcal{U}} \qquad J(x, m, u) = \mathbb{E}\left[\int_0^T L(x, m, u)\, dt + \Psi(x, T) \right],$$

$$\textit{subject to} \quad dx = f(x, m, u)\, dt + \sigma\, dw(t),$$

then the equivalent non-cooperative mean field game is the pair of HJB and FPK equations,

$$\begin{aligned} -\frac{\partial v}{\partial t}(x, t) - H(x, m, p) &= \frac{\sigma^2}{2} D^2 v(x, t), \\ \frac{\partial m}{\partial t}(x, t) + div(f(x, m, u) m(x, t)) &= \frac{\sigma^2}{2} D^2 m(x, t), \end{aligned} \tag{2.19}$$

where the Hamiltonian $H(x, m, p) = \min_{u\in\mathcal{U}} L(x, m, u) + pf(x, m, u)$ *and boundary conditions* $v(x, T) = \Psi(x, T)$ *and* $m(x, 0) = m_0(x)$*. In addition, the optimal control* $u^* = u(x^*, t)$ *is the solution to the equation*

$$f(x, m, u^*) = \frac{\partial H}{\partial p}(x, m, p),$$

where x^* *is a solution to state dynamic equation* $dx = f(x, m, u)\, dt + \sigma\, dw(t)$*.*

Proof. Let the value function $v(x, t) = \min_{u\in\mathcal{U}} J(x, m, u)$. Then, applying the dynamic programming principle for stochastic optimal control,

$$v(x, t) = \min_{u\in\mathcal{U}} \mathbb{E}\bigg[\int_t^{t+h} L(x, m(x, s), u(x, s))\, ds + v(x, t + h)\bigg],$$

dividing the result by h and getting the limit as $h \to 0$ yields

$$-\frac{\partial v}{\partial t}(x, t) - L(x, m, u) - f(x, m, u)\frac{\partial v}{\partial x}(x, t) = \frac{\sigma^2}{2} D^2 v(x, t),$$

which is equivalent to the HJB equation in (2.19). Meanwhile, let $\phi(x)$ be twice differentiable with compact support. Then,

$$d\phi(x) = \bigg(f(x, m, u)D\phi(x) + \frac{\sigma^2}{2} D^2\phi(x)\bigg)\, dt + D\phi(x)\sigma\, dw(t).$$

Dividing by dt and taking the expectation yields

$$\mathbb{E}\bigg[\frac{d\phi(x)}{dt}\bigg] = \mathbb{E}\bigg[f(x, m, u)D\phi(x) + \frac{\sigma^2}{2} D^2\phi(x)\bigg].$$

Rewriting in terms of $m(x, t)$ and using integration by parts results to the FPK equation in (2.19). ■

The following subsection is devoted to applying the single-population MFG to social network belief and opinion evolution. Through MFG, the evolution with time of the opinions of the users in a huge social network can be characterized by a pair of HJB and FPK equations.

2.2.2 Mean Field Game Problem for Single-Population Social Network

Given the definition of the mean field $m(x,t)$ in (2.18), the running cost functions (2.9) and (2.10) become

$$L(x,m,u) = c_1 u^2 + c_2 m + c_3 \left\| x - x_d \right\|, \tag{2.20}$$

and
$$L(x,m,u) = c_1 u^2 + c_2 m + c_3 \lambda \left\| x - x_d \right\| + c_4 (1-\lambda) \left\| x - \bar{x}_0 \right\|, \tag{2.21}$$

where $\bar{x}_0$ denotes the average initial opinion. The subscript i has been dropped to imply that the users are indistinguishable from each other.

To be able to derive the opinion dynamics in terms of the mean field $m(x,t)$, an important concept in probability theory known as the law of large numbers [36] is necessary.

Theorem 2. *Let $X_1, X_2, \ldots$ be a sequence of independent and identically distributed random variables, each having the same finite mean μ. Then,*

$$\mathbb{P}\left(\lim_{n \to \infty} \frac{1}{n} (X_1 + X_2 + \cdots + X_n) = \mu \right) = 1.$$

In words, the partial averages $\frac{1}{n}(X_1 + X_2 + \cdots X_n)$ converges almost surely to μ.

Consequently, based on Theorem 2, the opinion dynamics in (2.6) can be written as

$$\begin{aligned} dx &= \left(\bar{a} \int_{\mathcal{X}} x m(x,t)\, dx + ax + bu \right) dt + \sigma\, dw(t), \\ &= \left(\bar{a}\bar{x} + ax + bu \right) dt + \sigma\, dw(t) = f(x,m,u)\, dt + \sigma\, dw(t), \end{aligned} \tag{2.22}$$

where $\bar{x} = \bar{x}_{\mathcal{N}} / \alpha$ is the average population opinion, $\bar{x}_{\mathcal{N}} = \sum_{j=1}^{n} \alpha_{ij} x_j(t) / \sum_{j=1}^{n} \alpha_{ij}$ is the average neighborhood opinion, α is an adjacency parameter, and $\bar{a}$, a, and b are constants. The third term in (2.22) is added to capture the opinion change due to uncertainty or random phenomenon with σ as the diffusion constant and $w(t)$ as a standard Wiener process.

Similarly, for social networks with stubborn agents, the opinion dynamics in (2.7) becomes

$$
\begin{aligned}
dx &= \bar{a}\Big(\lambda \int_{\mathcal{X}} xm(x,t)\,dx + (1-\lambda)\int_{\mathcal{X}} xm_0(x)\,dx\Big)\,dt + (ax+bu)\,dt + \sigma\,dw(t), \\
&= \big(\bar{a}(\lambda\bar{x} + (1-\lambda)\bar{x}_0) + ax + bu\big)\,dt + \sigma\,dw(t) = f(x,m,u)\,dt + \sigma\,dw(t).
\end{aligned} \tag{2.23}
$$

Fig. 2.1b shows an illustration of a single-population MFG social network where the aggregate effect of other users to a representative user is taken into account rather than the individual effect of each other user. The social network problem solved through MFG is formally stated by following.

Problem 1. *Consider a non-cooperative social network game with large number of indistinguishable users. In this game, a user tries to minimize its cost $J(x,m,u)$ subject to the opinion evolution dx. Denote $v(x,t) = \min_{u\in\mathcal{U}} J(x,m,u)$ and $u = u(x,t)$ as the value and control functions of a user, respectively. Let $m = m(x,t)$ correspond to the distribution of users according to opinion x at time t. Then, a social network user views the game as a mean field game where the evolution of $v(x,t)$ and $m(x,t)$ with time are represented by the pair of HJB and FPK equations,*

$$
\begin{aligned}
-\frac{\partial v}{\partial t}(x,t) - L(x,m,u^*) - f(x,m,u^*)\frac{\partial v}{\partial t}(x,t) &= \frac{\sigma^2}{2}D^2 v(x,t), \\
\frac{\partial m}{\partial t}(x,t) + div(f(x,m,u)m(x,t)) &= \frac{\sigma^2}{2}D^2 m(x,t),
\end{aligned} \tag{2.24}
$$

with boundary conditions $v(x,T) = 0$ and $m(x,0) = m_0(x)$. The drift $f(x,m,u)$ is given by

$$f(x,m,u) = \bar{a}\bar{x} + ax + bu,$$

and

$$f(x,m,u) = \bar{a}(\lambda\bar{x} + (1-\lambda)\bar{x}_0) + ax + bu,$$

when there is absence and presence of stubborn agents, respectively. Meanwhile, the optimal control $u^* = u(x^*,t)$ is the solution to the equation

$$f(x,m,u^*) = \frac{\partial H}{\partial p}(x,m,p),$$

where the Hamiltonian is defined as $H(x, m, p) = \min_{u \in \mathcal{U}} L(x, m, u) + pf(x, m, u)$ *and* $p = \frac{\partial v}{\partial t}(x, t)$.

In a single population, MFG focuses on the Nash equilibrium (i.e., a solution in which no player gains anything by changing only its own strategy) among the players in the group. The same point of view can be extended to scenarios with several populations. In the following section, an extension of MFG to scenarios involving several populations is discussed. Then, the methodology is applied to social networks when users can be grouped into several populations.

2.3 Mean Field Game with Several Populations

In this section, the multiple-population framework of MFG presented in [14] is introduced. This MFG framework will be used in the next section to model the belief and opinion dynamics in social networks.

2.3.1 Background and Motivation

Consider a non-cooperative game among a large number of players that can be divided into P populations. Let the state $x \in \mathbb{R}^d$. Denote mean field vector $m = (m_i)^{\top}_{i=1,\dots,P}$ and control vector $u = (u_i)^{\top}_{i=1,\dots,P}$. Assume that each population considers that the distribution of all populations are fixed or predictable and tries to minimize its own individual cost. The goal of a player in population i is to minimize the population cost $J_i(x, m, u_i)$ subject to the population state dynamics dx.

A MFG problem with P populations is to find (m^*, u^*) where $u^* = (u_i^*)^{\top}_{i=1,\dots,P}$ is a Nash equilibrium for the cost functions

$$J_i(x, m, u_i) = \mathbb{E}\left[\int_0^T L_i(x, m^*, u_i)\, dt + \Psi_i(x, m^*(x, T))\, dx\right], \tag{2.25}$$

and $m^* = (m_i^*)_{i=1,\dots,P}^\top$ satisfies

$$\frac{\partial m_i}{\partial t}(x,t) + \operatorname{div}\big(f_i(x,m,u_i^*)m_i(x,t)\big) = \frac{\sigma_i^2}{2}D^2 m_i(x,t). \tag{2.26}$$

Thus, population i faces the problem of finding u_i that solves

$$\begin{aligned}
\min_{u_i \in \mathcal{U}_i} \qquad & J_i(x,m,u_i) = \mathbb{E}\bigg[\int_0^T L_i(x,m^*,u_i)\,dt + \Psi_i(x,m^*(x,T))\bigg], \\
\text{subject to} \quad & \frac{\partial m_i}{\partial t}(x,t) + \operatorname{div}\big(f_i(x,m^*,u_i)m_i(x,t)\big) = \frac{\sigma_i^2}{2}D^2 m_i(x,t), \\
& \frac{\partial m_p}{\partial t}(x,t) + \operatorname{div}\big(f_p(x,m^*,u_p^*)m_p(x,t)\big) = \frac{\sigma_p^2}{2}D^2 m_p(x,t),
\end{aligned} \tag{2.27}$$

for $p \neq i$, with initial conditions $m_i(x,0) = m_{i,0}(x), i = 1,\dots,P$.

The pair (m_i, u_i) that solves (2.27) is also a solution of the system of partial differential equations

$$\begin{aligned}
-\frac{\partial v_i}{\partial t}(x,t) - H_i(x,m,Dv_i) &= \frac{\sigma_i^2}{2}D^2 v_i(x,t), \\
\frac{\partial m_i}{\partial t}(x,t) + \operatorname{div}\bigg(\frac{\partial H_i}{\partial p_i}(x,m,Du_i)m_i(x,t)\bigg) &= \frac{\sigma_i^2}{2}D^2 m_i(x,t),
\end{aligned} \tag{2.28}$$

with boundary conditions $v_i(x,T) = \Psi_i(x,m(x,T))$ and $m_i(x,0) = m_{i,0}(x)$. Also, the Hamiltonian is defined as

$$H_i(x,m,Dv_i) = \min_{u_i \in \mathcal{U}_i} L_i(x,m,u_i) + p_i f_i(x,m,u_i), \tag{2.29}$$

where $p_i = \frac{\partial v_i}{\partial t}(x,t)$.

The following theorem is the extension of Theorem 1 to MFG with several populations. For more detailed proof, please refer to [14] and the references therein.

Theorem 3. *Consider a non-cooperative game among large a number of indistinguishable* *players. If population i faces the optimization problem*

$$\min_{u_i \in \mathcal{U}_i} \qquad J_i(x, m, u_i) = \mathbb{E}\Big[\int_0^T L_i(x, m^*, u_i)\, dt + \Psi_i(x, m^*(x,T))\Big],$$

$$\textit{subject to} \quad \frac{\partial m_p}{\partial t}(x,t) + \mathit{div}\big(f_p(x, m, u_p^*) m_p(x,t)\big) = \frac{\sigma_p^2}{2} D^2 m_p(x,t),$$

$p = 1, \ldots, P$, with boundary conditions $v_i(x,T) = \Psi_i(x, m(x,T))$ and $m_i(x,0) = m_{i,0}(x)$, then the equivalent non-cooperative mean field game consists of the HJB and FPK equation,

$$\begin{aligned} -\frac{\partial v_i}{\partial t}(x,t) - H_i(x, m, Dv_i) &= \frac{\sigma_i^2}{2} D^2 v_i(x,t), \\ \frac{\partial m_i}{\partial t}(x,t) + \mathit{div}\bigg(\frac{\partial H_i}{\partial p_i}(x, m, Du_i) m_i(x,t)\bigg) &= \frac{\sigma_i^2}{2} D^2 m_i(x,t). \end{aligned} \tag{2.30}$$

Meanwhile, the optimal control $u_i^* = u_i(x^*, t)$ is the solution to the equation

$$f_i(x, m, u_i^*) = \frac{\partial H_i}{\partial p_i}(x, m, p_i),$$

where x^* is a solution to state dynamic equation $dx = f_i(x, m, u_i)\, dt + \sigma_i\, dw_i(t)$.

Proof. Let the value function $v_i(x,t) = \min_{u_i \in \mathcal{U}_i} J_i(x, m, u_i)$. Then, applying the dynamic programming principle for stochastic optimal control,

$$v_i(x,t) = \min_{u_i \in \mathcal{U}_i} \mathbb{E}\Big[\int_t^{t+h} L_i(x, m(x,s), u_i(x,s))\, ds + v_i(x, t+h)\Big],$$

dividing the result by h and getting the limit as $h \to 0$ yields

$$-\frac{\partial v_i}{\partial t}(x,t) - L_i(x, m, u_i) - f_i(x, m, u_i)\frac{\partial v_i}{\partial x}(x,t) = \frac{\sigma_i^2}{2} D^2 v_i(x,t),$$

which is equivalent to the HJB equation in (2.30). Meanwhile, let $\phi_i(x)$ be twice differentiable with compact support. Then,

$$d\phi_i(x) = \bigg(f_i(x, m, u_i) D\phi_i(x) + \frac{\sigma_i^2}{2} D^2 \phi_i(x)\bigg)\, dt + D\phi_i(x)\sigma_i\, dw_i(t).$$

Dividing by dt and taking the expectation yields

$$\mathbb{E}\left[\frac{d\phi_i(x)}{dt}\right] = \mathbb{E}\left[f_i(x, m, u_i)D\phi_i(x) + \frac{\sigma_i^2}{2}D^2\phi_i(x)\right].$$

Rewriting in terms of $m_i(x,t)$ and using integration by parts results to the FPK equation in (2.30). ■

2.3.2 Mean Field Game Problem for Multiple-Population Social Network

In this section, the MFG model in the previous section is applied to a social network consisting of multiple populations where the users can be divided into several populations and the users in the same population or group share a common characteristic.

The running cost function $L_i(x, m, u_i)$ of a social network user in population i is proportional to the influence effort $u_i = u_i(x,t)$ it exerts to minimize its own cost. In addition, a user is rewarded (or penalized) if it influences users from any population. In other words, the cost decreases (or increases) if the mean fields of the populations $m_p = m_p(x,t), p = 1, \ldots, P$ are higher (or lower). If $x_{i,d}$ denotes the ideal or desired opinion of population i, then each user must minimize the distance of its opinion to the ideal opinion, $\|x - x_{i,d}\|$. In addition, when the social network users have a degree of stubbornness, the distance to the average initial opinion is also affects the cost, $\|x - \bar{x}_{i,0}\|$. Therefore, the running cost functions $L_i(x, m, u_i)$ in (2.20) and (2.21) can be written mathematically as

$$L_i(x, m, u_i) = c_1 u_i^2 + \sum_{p=1}^{P} c_{2,p} m_p + c_3 \|x - x_{i,d}\|, \tag{2.31}$$

and
$$L_i(x, m, u_i) = c_1 u_i^2 + \sum_{p=1}^{P} c_{2,p} m_p + c_3 \lambda_i \|x - x_{i,d}\| + c_4(1 - \lambda_i)\|x - \bar{x}_{i,0}\|. \tag{2.32}$$

Consequently, the average running cost of a user in population i is given by

$$J_i(x, m, u_i) = \mathbb{E}\left[\int_0^T L_i(x, m, u_i)\, dt\right]. \tag{2.33}$$

Since the users can be divided into P distinct populations, the opinion dynamics equation in (2.22) for population i becomes

$$dx = \left(\sum_{p=1}^{P} \eta_p \bar{a}_{i,p} \bar{x}_p + a_i x + b_i u_i \right) dt + \sigma_i \, dw_i(t) = f_i(x, m, u_i) \, dt + \sigma_i \, dw_i(t), \tag{2.34}$$

where η_p refers to the ratio of the size of population p and the total population, $\bar{x}_p = \bar{x}_{\mathcal{N},p}/\alpha_{i,p}$ is the average opinion at population p, $\alpha_{i,p}$ is the average adjacency to population p in population i, $\bar{a}_{i,p} = \bar{a}/\alpha_{i,p}$, b_i is a control constant, σ_i is the diffusion constant, and $w_i(t)$ is a standard Wiener process.

When there are stubborn agents, the opinion dynamics equation in (2.23) for population i becomes

$$\begin{aligned} dx &= \left(\lambda_i \sum_{p=1}^{P} \eta_p \bar{a}_{i,p} \bar{x}_p + (1 - \lambda_i) \bar{a}_{i,i} \bar{x}_{i,0} + a_i x + b_i u_i \right) dt + \sigma_i \, dw_i(t), \\ &= f_i(x, m, u_i) \, dt + \sigma_i \, dw_i(t), \end{aligned} \tag{2.35}$$

where λ_i is the average susceptibility of a user in population i. The following problem states the multiple-population opinion evolution on social network modeled using MFG.

Problem 2. *Consider a non-cooperative game among P distinct populations in a social network. Assume that each population consists of large number of indistinguishable users. In this game, a user in population i tries to minimize the population cost $J_i(x, m, u_i)$ subject to the opinion evolution dx. Denote $v_i(x, t) = \min_{u_i \in \mathcal{U}_i} J_i(x, m, u_i)$ and $u_i = u_i(x, t)$ as the value and control functions of a user in population i, respectively. Let $m_i = m_i(x, t)$ correspond to the distribution of users at population i according to opinion x at time t. Then, a social network user at population i views the game as a mean field game where the evolution of $v_i(x, t)$ and $m_i(x, t)$ with time are represented by the pair of HJB and FPK* *equations,*

$$
\begin{aligned}
-\frac{\partial v_i}{\partial t}(x,t) - H_i(x,m,Dv_i) &= \frac{\sigma_i^2}{2} D^2 v_i(x,t), \\
\frac{\partial m_i}{\partial t}(x,t) + div\bigg(\frac{\partial H_i}{\partial p_i}(x,m,Du_i) m_i(x,t) \bigg) &= \frac{\sigma_i^2}{2} D^2 m_i(x,t),
\end{aligned}
\tag{2.36}
$$

with boundary conditions $v_i(x,T) = 0$ *and* $m_i(x,0) = m_{i,0}(x)$. *The drift* $f_i(x,m,u_i)$ *of population* i *is given by*

$$f_i(x,m,u_i) = \sum_{p=1}^{P} \eta_p \bar{a}_{i,p} \bar{x}_p + a_i x + b_i u_i,$$

and

$$f_i(x,m,u_i) = \lambda_i \sum_{p=1}^{P} \eta_p \bar{a}_{i,p} \bar{x}_p + (1-\lambda_i) \bar{a}_{i,i} \bar{x}_{i,0} + a_i x + b_i u_i,$$

when there is absence and presence of stubborn agents, respectively. Meanwhile, the optimal control $u_i^* = u_i(x^*,t)$ *of a user in population* i *is the solution to the equation*

$$f_i(x,m,u_i^*) = \frac{\partial H_i}{\partial p_i}(x,m,p_i),$$

where the Hamiltonian is defined as $H_i(x,m,p_i) = \min_{u_i \in \mathcal{U}_i} L_i(x,m,u_i) + p_i f_i(x,m,u_i)$ *and* $p_i = \frac{\partial v_i}{\partial t}(x,t)$.

In the next section, an analytical method to solve the pair of partial differential equations in (2.30) is presented. Then, this method is applied to the proposed social network problem.

2.4 Adjoint Method for Mean Field Games with Several Populations

An adjoint method was proposed in [35] to solve the HJB-FPK pair associated with a single-population MFG. In this section, this methodology is extended to multiple-population MFG and solve the HJB-FPK pair (2.30) for population i. Afterwards, the result of applying the methodology to social network belief and opinion evolution is presented.

2.4.1 Background

The method starts with introducing an adjoint variable, $v_i(x,t)$ that also corresponds to the value function

$$v_i(x,t) = \min_{u_i \in \mathcal{U}_i} J_i(x,m,u_i). \tag{2.37}$$

Using this adjoint variable, the FPK equation is appended to the original cost function $J_i(x,m,u_i)$. Thus, the resulting MFG problem optimizes the extended cost function

$$\begin{aligned}\mathcal{J}_i(u_i,m_i,v_i) = J_i(x,m,u_i) + v_i(x,t)\bigg[- \frac{\partial m_i}{\partial t}(x,t) - \operatorname{div}\bigg(\frac{\partial H_i}{\partial p_i}(x,m,Du_i)m_i(x,t)\bigg) \\ + \frac{\sigma_i^2}{2}D^2 m_i(x,t)\bigg].\end{aligned} \tag{2.38}$$

There exists a pair (m_i,u_i) that minimizes the extended cost function in (2.38) if there is a v_i such that (m_i,u_i,v_i) is a stationary solution. Thus, the following optimality conditions are necessary:

$$\frac{\partial \mathcal{J}_i}{\partial u_i} = 0, \quad \frac{\partial \mathcal{J}_i}{\partial m_i} = 0, \quad \text{and} \quad \frac{\partial \mathcal{J}_i}{\partial v_i} = 0. \tag{2.39}$$

Now, consider the optimal control problem

$$\begin{aligned} \min_{m_i,u_i} \quad & J_i(x,m,u_i) = \mathbb{E}\bigg[\int_0^T L_i(x,m^*,u_i)\,dt + \Psi_i(x,m^*(x,T))\bigg], \\ \text{subject to} \quad & \frac{\partial m_p}{\partial t}(x,t) + \operatorname{div}\big(f_p(x,m,u_p)m_p(x,t)\big) = \frac{\sigma_p^2}{2}D^2 m_p(x,t),\end{aligned} \tag{2.40}$$

for all $p = 1,\ldots,P$. The corresponding extended cost function is

$$\begin{aligned}\mathcal{J}_i(m_i,u_i,v_i) = \int_0^T \int_{\mathbb{R}^n} L_i(x,m,u_i)m_i(x,t)\,dxdt + \int_{\mathbb{R}^n} \Psi_i(x,m(x,T))m_i(x,T)\,dx \\ + \sum_{p=1}^{P} \int_0^T \int_{\mathbb{R}^n} v_p(x,t)\bigg(- \frac{\partial m_p}{\partial t}(x,t) - \operatorname{div}\big(f_p(x,m,u_p)m_p(x,t)\big) \\ + \frac{\sigma_p^2}{2}D^2 m_p(x,t)\bigg)\,dxdt.\end{aligned} \tag{2.41}$$

Solving $\frac{\partial \mathcal{J}_i}{\partial u_i} = 0$ yields

$$m_i(x,t)\frac{\partial L_i}{\partial u_i}(x,m,u_i) + m_i(x,t)Dv_i(x,t)\frac{\partial f_i}{\partial u_i}(x,m,u_i) + m_i(x,t)\frac{\partial}{\partial u_i}\left(\frac{\sigma_i^2}{2}D^2v_i(x,t)\right) = 0. \tag{2.42}$$

Meanwhile, $\frac{\partial \mathcal{J}_i}{\partial m_i} = 0$ results to

$$\begin{aligned} L_i(x,m,u_i) + m_i\frac{\partial L_i}{\partial m_i}(x,m,u_i) + \frac{\partial v_i}{\partial t}(x,t) + f_i(x,m,u_i)Dv_i(x,t) \\ + \sum_{p=1}^{P}\left(m_p\frac{\partial f_p}{\partial m_i}(x,m,u_p)Dv_p(x,t)\right) + \frac{\sigma_i^2}{2}D^2v_i(x,t) = 0, \end{aligned} \tag{2.43}$$

which can be written as

$$\begin{aligned} -\frac{\partial v_i}{\partial t}(x,t) - H_i(x,m,Dv_i) = \frac{\sigma_i^2}{2}D^2v_i(x,t) + m_i\frac{\partial L_i}{\partial m_i}(x,m,u_i) \\ + \sum_{p=1}^{P}\left(m_p\frac{\partial f_p}{\partial m_i}(x,m,u_p)Dv_p(x,t)\right). \end{aligned} \tag{2.44}$$

Finally, $\frac{\partial \mathcal{J}_i}{\partial v_i} = 0$ yields

$$\frac{\partial m_i}{\partial t}(x,t) + \operatorname{div}\big(f_i(x,m,u_i)m_i(x,t)\big) = \frac{\sigma_i^2}{2}D^2m_i(x,t). \tag{2.45}$$

In summary, (2.42), (2.44), and (2.45) are solved iteratively to find the stationary solution (m_i, u_i, v_i).

2.4.2 Adjoint Method for the Social Network Problems

For a social network without stubborn agents, the extended cost function can be written as

$$\begin{aligned}\mathcal{J}_i(m_i,u_i,v_i) &= \int_0^T\int_{\mathbb{R}^n} L_i(x,m,u_i)m_i(x,t)\,dxdt \\ &\quad + \sum_{p=1}^{P}\int_0^T\int_{\mathbb{R}^n} v_p(x,t)\Big(-\frac{\partial m_p}{\partial t}(x,t) - \operatorname{div}\Big(\Big(\sum_{q=1}^{Q}\eta_q\bar{a}_{p,q}\bar{x}_q + a_p x \\ &\quad + b_p u_p\Big)m_p(x,t)\Big) + \frac{\sigma_p^2}{2}D^2 m_p(x,t)\Big)\,dxdt.\end{aligned} \tag{2.46}$$

Applying the optimality conditions in (2.39) to (2.46) result to the following equations:

$$\frac{\partial \mathcal{J}_i}{\partial u_i} = m_i(x,t)\frac{\partial L_i}{\partial u_i}(x,m(x,t),u_i(x,t)) + b_i m_i(x,t)Dv_i(x,t), \tag{2.47}$$

$$\begin{aligned}\frac{\partial \mathcal{J}_i}{\partial m_i} &= \frac{\partial v_i}{\partial t}(x,t) + H_i(x,m,Dv_i) + \frac{\sigma_i^2}{2}D^2 v_i(x,t) \\ &\quad + c_{2,i}m_i + \frac{1}{2}\eta_i\bar{a}_{i,i}x^2\sum_{p=1}^{P} m_p Dv_p(x,t),\end{aligned} \tag{2.48}$$

and

$$\frac{\partial \mathcal{J}_i}{\partial v_i} = \frac{\partial m_i}{\partial t}(x,t) + \operatorname{div}\big(f_i(x,m,u_i)m_i(x,t)\big) - \frac{\sigma_i^2}{2}D^2 m_i(x,t). \tag{2.49}$$

Meanwhile, for a social network with stubborn agents, the extended cost function can be written as

$$\begin{aligned}\mathcal{J}_i(m_i,u_i,v_i) &= \int_0^T\int_{\mathbb{R}^n} L_i(x,m,u_i)m_i(x,t)\,dxdt + \sum_{p=1}^{P}\int_0^T\int_{\mathbb{R}^n} v_p(x,t)\Big(-\frac{\partial m_p}{\partial t}(x,t) \\ &\quad - \operatorname{div}\Big(\Big(\lambda_p\sum_{q=1}^{Q}\eta_q\bar{a}_{p,q}\bar{x}_q + (1-\lambda_p)\bar{a}_{p,p}\bar{x}_{p,0} + a_p x + b_p u_p\Big)m_p(x,t)\Big) \\ &\quad + \frac{\sigma_p^2}{2}D^2 m_p(x,t)\Big)\,dxdt.\end{aligned} \tag{2.50}$$

Applying the optimality conditions in (2.39) to (2.50) result to the following equations:

$$\frac{\partial \mathcal{J}_i}{\partial u_i} = m_i(x,t)\frac{\partial L_i}{\partial u_i}(x,m,u_i) + b_i m_i(x,t) D v_i(x,t), \tag{2.51}$$

$$\begin{aligned}\frac{\partial \mathcal{J}_i}{\partial m_i} = &\frac{\partial v_i}{\partial t}(x,t) + H_i(x,m,Dv_i) + \frac{\sigma_i^2}{2} D^2 v_i(x,t) + c_{2,i} m_i \\ &+ \frac{1}{2}\big(\lambda_i \eta_i \bar{a}_{i,i} x^2 + (1-\lambda_i)\bar{a}_{i,i} x^2\big)\sum_{p=1}^{P} m_p D v_p(x,t),\end{aligned} \tag{2.52}$$

and

$$\frac{\partial \mathcal{J}_i}{\partial v_i} = \frac{\partial m_i}{\partial t}(x,t) + \operatorname{div}\big(f_i(x,m,u_i)m_i(x,t)\big) - \frac{\sigma_i^2}{2} D^2 m_i(x,t). \tag{2.53}$$

In the next section, a numerical method to solve the optimality conditions in (2.42), (2.44), and (2.45) is presented.

2.5 Numerical Method for Mean Field Games with Several Populations

To calculate the stationary solution (m_i, u_i, v_i) corresponding to the optimality conditions in (2.42), (2.44), and (2.45), a well-known numerical method called finite difference method is implemented. In this method, the partial differential equations (PDEs) are converted into their equivalent discrete-time difference equations. In addition to finite difference method, a numerical scheme called the Lax-Friedrichs scheme is discussed. Afterwards, these numerical techniques are applied to solve the MFG problems presented in the previous sections.

2.5.1 Background

Consider a bounded region $[0, X_{\max}] \times [0, T_{\max}]$ over which independent variables x and t of the PDEs are defined. In order to implement a numerical method to solve the PDEs, the region is converted into a finite grid of points. Given positive integers M and N, a space step is defined as $\Delta x = \frac{X_{\max}}{M}$ and a time step is defined as $\Delta t = \frac{T_{\max}}{N}$. Hence, the grid of points are defined by

$$x_j = j\Delta x, j = 0, ..., M \quad \text{and} \quad t_k = k\Delta t, k = 0, ..., N. \tag{2.54}$$

Furthermore, for any function ρ defined over the space-time region, $\rho_j^k = \rho(x_j, t_k)$.

The finite difference (FD) method is a numerical method to approximate and solve PDEs. This method uses finite differences to approximate derivatives. For the first-order derivatives in x, the forward, backward, and central differences are defined as

$$\frac{\partial \rho}{\partial x} = \frac{\rho_j^{k+1} - \rho_j^k}{\Delta x}, \quad \frac{\partial \rho}{\partial x} = \frac{\rho_j^k - \rho_j^{k-1}}{\Delta x}, \quad \text{and} \quad \frac{\partial \rho}{\partial x} = \frac{\rho_j^{k+1} - \rho_j^{k-1}}{2\Delta x}, \tag{2.55}$$

where $\Delta x = x_{j+1} - x_j = x_j - x_{j-1}$. For the second-order derivatives in x, the symmetric difference approximation is defined as

$$\frac{\partial^2 \rho}{\partial x^2} = \frac{\rho_{j+1}^k - 2\rho_j^k + \rho_{j-1}^k}{(\Delta x)^2}. \tag{2.56}$$

These definitions apply in a similar fashion to derivatives in t.

In order to discretize the extended Lagrangian in (2.41), consider a one-dimensional Fokker-Planck equation

$$\frac{\partial m_i}{\partial t}(x,t) + \frac{\partial \phi_i}{\partial x}(m_i(x,t)) = \varepsilon_i \frac{\partial^2 m_i}{\partial x \partial x}(x,t), \tag{2.57}$$

where $\varepsilon_i = \sigma_i^2/2$. This FPK equation can be approximated as

$$\frac{m_{i,j}^{k+1} - m_{i,j}^k}{\Delta t} + \frac{\rho_i(m_{i,j}^k, m_{i,j+1}^k) - \rho_i(m_{i,j-1}^k, m_{i,j}^k)}{\Delta x} = \varepsilon_i \frac{m_{i,j+1}^k - 2m_{i,j}^k + m_{i,j-1}^k}{(\Delta x)^2}, \tag{2.58}$$

where $\rho_i(m, m)$ is a consistent numerical flow for $\phi_i(m)$, $\rho_i(m, m) = \phi_i(m), \forall m \in \mathbb{R}$.

Using the FD method, the extended Lagrangian in (2.41) becomes

$$\mathcal{J}_i(m_{i,j}^k, u_{i,j}^k, v_{i,j}^k) = \Delta x \Delta t \sum_{j,k}^{M,N} L_{i,j}^k m_{i,j}^k + \Delta x \sum_{j}^{M} \Psi_{i,j}^N + \Delta x \Delta t \sum_{p=1}^{P} \sum_{j,k}^{M,N} -v_{p,j}^k \frac{m_{p,j}^{k+1} - m_{p,j}^k}{\Delta t}$$
$$- v_{p,j}^k \frac{\rho_p(m_{p,j}^k, m_{p,j+1}^k) - \rho_p(m_{p,j-1}^k, m_{p,j}^k)}{2\Delta x}$$
$$+ \varepsilon_p v_{p,j}^k \frac{m_{p,j+1}^k - 2m_{p,j}^k + m_{p,j-1}^k}{(\Delta x)^2}, \tag{2.59}$$

where the numerical flow $\rho_p(m_{p,j}^k, m_{p,j+1}^k) = \rho_{p,j+\frac{1}{2}}^k$ and $\rho_p(m_{p,j-1}^k, m_{p,j}^k) = \rho_{p,j-\frac{1}{2}}^k$.

To preserve the sign of $m_{i,j}^k$ in (2.58), a monotone numerical scheme such as the Lax-Friedrichs scheme should be utilized. Given the Lax-Friedrichs flux

$$\rho(m, n) = \frac{1}{2}(\phi(m) + \phi(n)) + \frac{\Delta t}{2\Delta x}(m - n), \tag{2.60}$$

the Lax-Friedrichs scheme for (2.58) is

$$m_{i,j}^{k+1} = \frac{1}{2}(m_{i,j+1}^k + m_{i,j-1}^k) - \frac{\Delta t}{2\Delta x}\left(\phi_{i,j+1}^k - \phi_{i,j-1}^k\right) + \varepsilon_i \frac{\Delta t}{(\Delta x)^2}(m_{i,j+1}^k - 2m_{i,j}^k + m_{i,j-1}^k). \tag{2.61}$$

Consequently, the discrete extended Lagrangian in (2.59) becomes

$$\mathcal{J}_i(m_{i,j}^k, u_{i,j}^k, v_{i,j}^k) = \Delta x \Delta t \sum_{j,k}^{M,N} L_{i,j}^k m_{i,j}^k + \Delta x \sum_{j}^{M} \Psi_j^N$$
$$+ \Delta x \Delta t \sum_{p=1}^{P} \sum_{j,k}^{M,N} -v_{p,j}^k \frac{m_{p,j}^{k+1} - 0.5(m_{p,j+1}^k + m_{p,j-1}^k)}{\Delta t} \tag{2.62}$$
$$- v_{p,j}^k \frac{\phi_{p,j+1}^k - \phi_{p,j-1}^k}{2\Delta x} + \varepsilon_p v_{p,j}^k \frac{m_{p,j+1}^k - 2m_{p,j}^k + m_{p,j-1}^k}{(\Delta x)^2}.$$

Applying the optimality conditions in (2.39) to the discrete extended Lagrangian in (2.62) yields

$$\frac{\partial \mathcal{J}_i}{\partial u_{i,j}^k} = m_{i,j}^k \frac{\partial L_{i,j}^k}{\partial u_{i,j}^k} - \frac{\partial \phi_{i,j}^k}{\partial u_{i,j}^k} \frac{v_{i,j-1}^k - v_{i,j+1}^k}{2\Delta x}, \tag{2.63}$$

$$
\begin{aligned}
\frac{\partial \mathcal{J}_i}{\partial m_{i,j}^k} = L_{i,j}^k + m_{i,j}^k \frac{\partial L_{i,j}^k}{\partial m_{i,j}^k} - \frac{v_{i,j}^{k-1} - 0.5(v_{i,j-1}^k + v_{i,j+1}^k)}{\Delta t} - f_{i,j}^k \frac{v_{i,j-1}^k - v_{i,j+1}^k}{2\Delta x} \\
- \sum_{p=1}^{P} m_{p,j}^k \frac{\partial f_p}{\partial m_{i,j}^k} \frac{v_{p,j-1}^k - v_{p,j+1}^k}{2\Delta x} + \varepsilon_i \frac{v_{i,j-1}^k - 2v_{i,j}^k + v_{i,j+1}^k}{(\Delta x)^2},
\end{aligned}
\tag{2.64}
$$

and
$$
\frac{\partial \mathcal{J}_i}{\partial v_{i,j}^k} = \frac{m_{i,j}^{k+1} - 0.5(m_{i,j+1}^k + m_{i,j-1}^k)}{\Delta t} + \frac{\phi_{i,j+1}^k - \phi_{i,j-1}^k}{2\Delta x} - \varepsilon_i \frac{m_{i,j+1}^k - 2m_{i,j}^k + m_{i,j-1}^k}{(\Delta x)^2}.
\tag{2.65}
$$

2.5.2 Numerical Method for the Social Network Problems

From (2.65), the update rule for the mean field $m_{i,j}^{k+1}$ for the social network problems with and without stubborn agents is

$$
m_{i,j}^{k+1} = 0.5(m_{i,j+1}^k + m_{i,j-1}^k) + \Delta t\left(- \frac{\phi_{i,j+1}^k - \phi_{i,j-1}^k}{2\Delta x} + \varepsilon_i \frac{m_{i,j+1}^k - 2m_{i,j}^k + m_{i,j-1}^k}{(\Delta x)^2} \right).
\tag{2.66}
$$

Furthermore, from (2.64), the update rule for the value function is

$$
\begin{aligned}
v_{i,j}^{k-1} = 0.5(v_{i,j-1}^k + v_{i,j+1}^k) + \Delta t\Bigg(L_{i,j}^k + m_{i,j}^k \frac{\partial L_{i,j}^k}{\partial m_{i,j}^k} - f_{i,j}^k \frac{v_{i,j-1}^k - v_{i,j+1}^k}{2\Delta x} \\
- \frac{\partial f_i}{\partial m_{i,j}^k} \sum_{p=1}^{P} m_{p,j}^k \frac{v_{p,j-1}^k - v_{p,j+1}^k}{2\Delta x} + \varepsilon_i \frac{v_{i,j-1}^k - 2v_{i,j}^k + v_{i,j+1}^k}{(\Delta x)^2} \Bigg).
\end{aligned}
\tag{2.67}
$$

Lastly, the following update rule is implemented for the control function $u_{i,j}^k$ for both social network problems,

$$
u_{i,j}^k \leftarrow \frac{w}{1+w} u_{i,j}^k - \frac{1}{1+w} \frac{\partial L_i}{\partial u_{i,j}^k},
\tag{2.68}
$$

where w is a constant.

Algorithm 1 provides the procedure of solving (2.47), (2.48), and (2.49) for the multiple-population social network belief and opinion evolution. In the algorithm, K is the maximum number of iterations and w is the control update parameter.

Algorithm 1 Solving the Mean Field Game Problem for Several Populations

Set: $c_1, c_2, c_3, w, \epsilon, \sigma_i, \eta_i, \alpha_i, a_i \forall i \in \{1, \ldots, P\}$

Initialize: $m^0_{i,j}, v^N_{i,j}, u^0_{i,j}, \forall i \in \{1, \ldots, P\}, \forall j \in \{1, \ldots, N\}$

while $n \leq n_{\max}$ or error $\leq \epsilon$ **do**

  for $i = 1, \ldots, P$ **do**

    for $k = 1, \ldots, N$ **do**

      Solve the mean field $m^{k+1}_{i,j}$ using (2.66).

      Solve the value $v^{k-1}_{i,j}$ using (2.67).

      Update $u^k_{i,j}$ using (2.68).

    end for

  end for

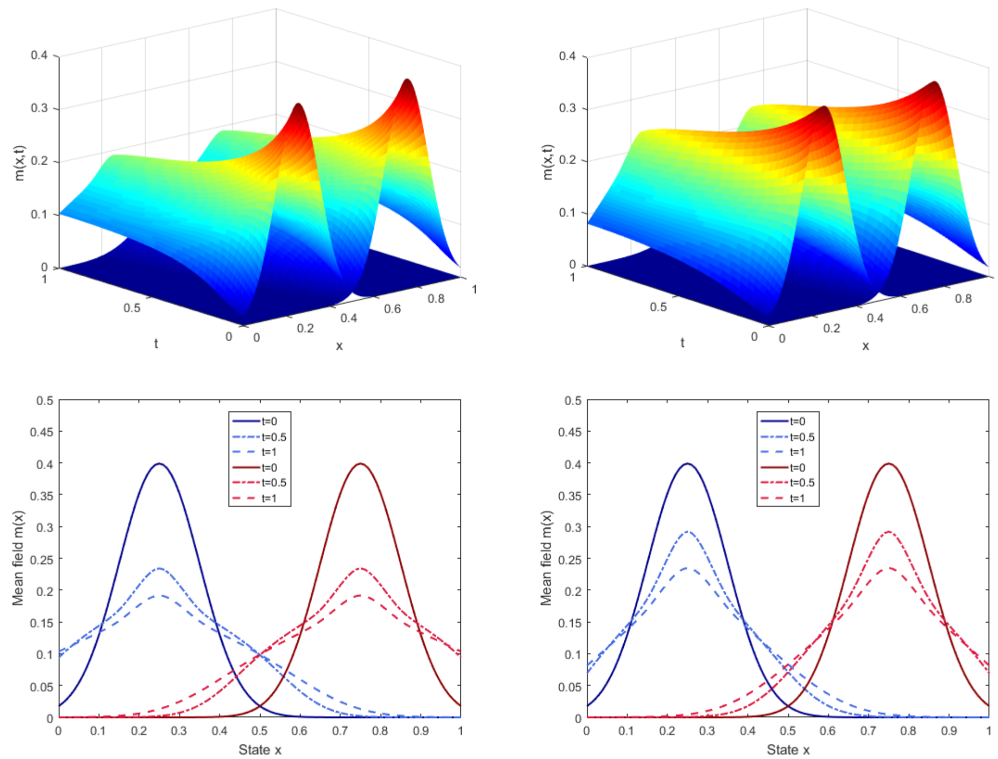
end while

2.6 Simulation Results

2.6.1 Theoretical Results

The following simulations demonstrate the theoretical aspects of the evolution of opinion in a social network consisting of single and multiple groups of users. The mean field $m_i(x, t)$ refers to the distribution of opinion x at population i at time t. To calculate the solution (m_i, v_i, u_i) of the MFG-based belief and opinion evolution models, Algorithm 1 is implemented using MATLAB.

Consider a social network consisting of users that can be divided into two separate populations, $i = 1, 2$. Let the state or opinion of a user be $x \in [0, 1]$ and time be $t \in [0, 1]$. Let the initial mean fields $m_i(x, 0)$ be $\mathcal{N}(0.25, 0.1)$ and $\mathcal{N}(0.75, 0.1)$ for populations 1 and 2, respectively. Also, assume a 50%-50% ratio between the two populations (i.e., $\eta_1 = \eta_2 = 0.5$). The opinion dynamics equation parameters are set at $\bar{a}_i = 0.001$, $a_i = -0.001$, and



(a) Two independent populations

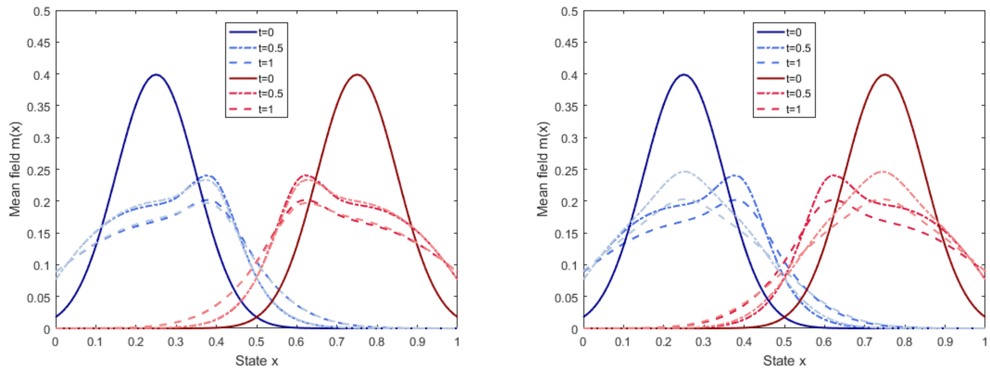
(b) Two dependent populations

Figure 2.2: The mean field at specific times.

$\sigma_i = 0.01$. Meanwhile, the cost function parameters are $c_1 = 0.5, c_2 = 1, c_3 = 2, c_4 = 1, x_{0,1} = 0.25, x_{0,2} = 0.75$. Finally, $w = 1,000$, $K = 2,000$ and $\epsilon = 1 \times 10^{-6}$.

Fig. 2.2 shows the mean field $m_i(x,t)$ at $t = 0, 0.5, 1$ for populations $i = 1, 2$ in blue and red, respectively: (a) independent and formulated as two single-population MFG; and (b) dependent and formulated as a two-population MFG. Based on the figure, the presence of another population affects the distribution of opinion of the other population. Specifically, the populations hold on to the their initial distributions with much ease in the presence of a competing population.

The effect of social network user stubbornness is presented in Fig. 2.3. When users are more susceptible to neighbor opinions, or have low stubbornness, more users are likely to achieve the population target or ideal opinion, as shown in Fig. 2.3a with $\lambda_i = 0.9$.



(a) $\lambda_i = 0.9$

(b) $\lambda_i = 0.1$

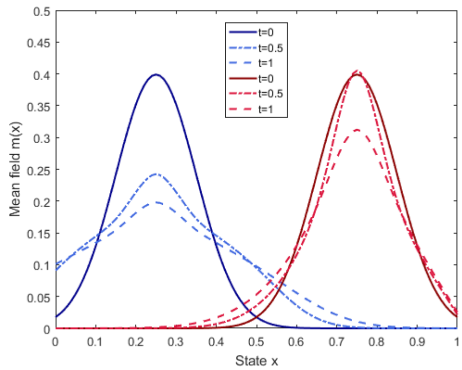
Figure 2.3: The effect of stubbornness to the mean field.

However, as users become less susceptible to neighbor opinions, or have high stubbornness, more users are likely to stay with the initial population opinion, as shown in Fig. 2.3b with $\lambda_i = 0.1$.

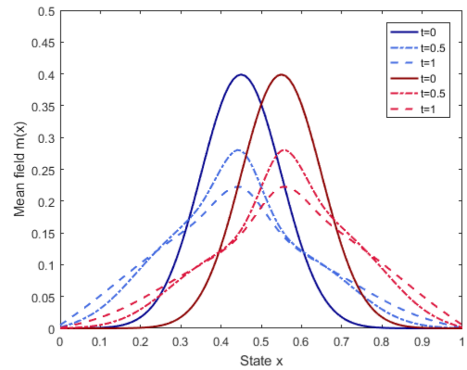
The effect of other parameters to the mean field $m_i(x, t)$ of population i are shown in Fig. 2.4. In Fig. 2.4a, population $i = 1$ with a higher percentage (i.e., 90% of total population) achieves a larger variance than population $i = 2$ with a lower percentage (i.e., 10% of total population). Meanwhile, in Fig. 2.4b, the distance between the means μ_i introduces variations in the mean field $m_i(x, t)$, as shown by the asymmetrical plots of $m_i(x, t)$.

2.6.2 Experimental Results Using a Social Evolution Dataset

The following simulations test the effectiveness of the proposed MFG-based algorithm in modeling the opinion evolution of a social network with multiple populations. The experiments use the social evolution dataset published in [30].



(a) $\eta_1 = 0.9, \eta_2 = 0.1$



(b) $\mu_1 = 0.45, \mu_2 = 0.55$

Figure 2.4: The effect of population fraction and mean state to the mean field.

Description of the dataset

The social evolution dataset consists of the political parties and opinions of 80 subjects as well as the relationship among them. The subjects were college students living in the same housing facility at MIT. One of the attributes of the subjects was their political party or affiliation (i.e., *Democratic Party* or *Republican Party*). The students were surveyed three times, March, April, and June 2009, about their opinion (i.e., approval or satisfaction level) of U.S. President Obama as the head of the state and the government. In each survey, they were asked whether they "*strongly disapprove*," "*disapprove*," "*don't know*," "*approve*," or "*strongly approve*," of President Obama. Moreover, they were asked about the extent of their relationship with other students involved in the survey. A relationship between two students exists if they are close friends, have had political discussions, and are involved together in social activities as well as social media networks such as the Facebook, Twitter, blog and live journal posts.

Processing of the dataset

To be able to utilize the dataset for the proposed MFG-based method, the opinions and relationship links are converted into their numerical equivalent. For the state or opinion x of a student, "*strongly disapprove*" $= 0.1$, "*disapprove*" $= 0.3$, "*don't know*" $= 0.5$, "*approve*" $= 0.7$, and "*strongly approve*" $= 0.9$. Meanwhile, the control or number of interactions u of a student refers to the total number of relationships it has had during the month. Then, the number of relationships are normalized between $[u_{\min}, u_{\max}]$ and centered at $u = 0$.

The state dynamics and cost function parameters are calculated and estimated from the dataset. The expectation of the state dynamics for political party i in (2.23) can be rewritten as

$$\mathbb{E}\left[\frac{dx}{dt}\right] = \mathbb{E}[\dot{x}] = \mathbb{E}\big[\bar{a}_i(\lambda_i \bar{x} + (1-\lambda_i)\bar{x}_{i,0}) + a_i x + b_i u_i\big] = \mathbb{E}\big[\omega_s \dot{x} + \omega_c \dot{x}\big], \tag{2.69}$$

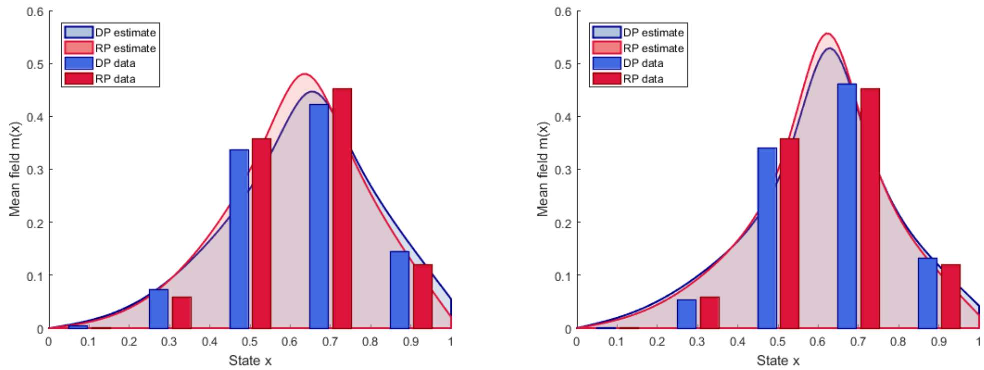
where $\bar{x}$ is the average opinion of all the students, $\bar{x}_{i,0}$ is the average opinion of students at $t = 0$, ω_s is the portion of $\dot{x}$ due to state x and ω_c is the portion of $\dot{x}$ due to control u_i. Note that, from the definition of a standard Wiener process, $\mathbb{E}[dw_i(t)] = 0$. Thus, the parameters of the state dynamics in (2.23) can be calculated from the dataset as

$$\bar{a}_i = \frac{\omega_s \mathbb{E}[\dot{x}]}{\mathbb{E}[\lambda_i \bar{x} + (1-\lambda_i)\bar{x}_{i,0} - x]} \quad \text{and} \quad b_i = \frac{\omega_c \mathbb{E}[\dot{x}]}{\mathbb{E}[u_i]}, \tag{2.70}$$

where the expectation is taken with respect to all the students in the survey. Since the dataset contains only the political opinions of the students, the values of the cost function parameters in (2.31) and (2.32) are estimated until the theoretical mean field $m_i(x, 1)$ fits the calculated mean field $M_i(x, t)$ from the dataset.

Procedure of the experiment

In the following simulations, $i = 1$ for the Democratic Party and $i = 2$ for the Republican Party while $t = 0$ for March 2009, $t = 1$ for April 2009, and $t = 2$ for June 2009. Given the political opinions for March, April, and June 2009, the distribution $M_i(x, t)$ of opinions of political party i is calculated for each month. The goal of the simulations is to use the MFG-based algorithm to model and predict the evolution of distribution of opinions of the students regarding President Obama. Hence, given $M_i(x, 0)$, the mean field



(a) April 2009 (b) June 2009

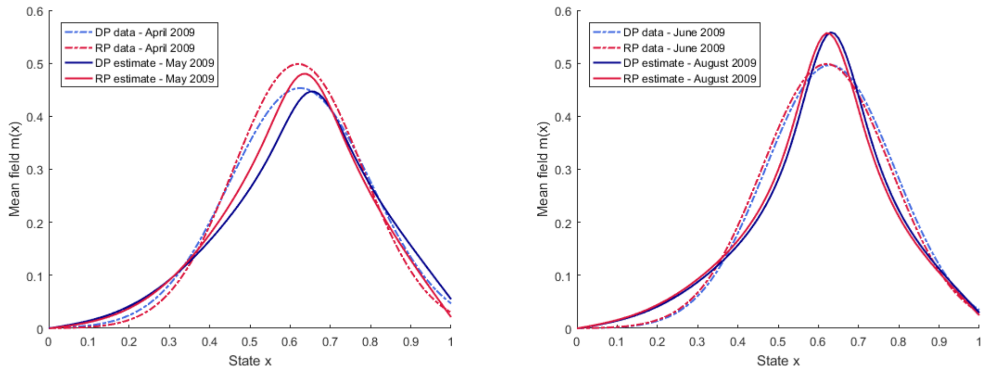
Figure 2.5: Estimated distribution of student opinions regarding U.S. President Obama.

$m_i(x, 1)$ is predicted using the proposed MFG-based model and compared with the calculated $M_i(x, 1)$. Similarly, given $M_i(x, 1)$, the mean field $m_i(x, 2)$ is predicted and compared with the calculated $M_i(x, 2)$.

Performance analysis

Fig. 2.5 shows the results of estimating the opinion distribution for the Democratic Party (DP) and Republican Party (RP) from the dataset. The performance is evaluated by calculating the average absolute error between the estimated mean field $m_i(x, t)$ from the proposed MFG-based algorithm and the calculated mean field $M_i(x, t)$ from the dataset. In Fig. 2.5a, the data for March 2009 is used to estimate the distribution of opinions for April 2009. The accuracy of the estimation is 97.25%. Meanwhile, in Fig. 2.5b, the data for April 2009 is used to estimate the distribution of opinions for June 2009. The accuracy of the estimation is 96.93%.

Finally, Fig. 2.6 presents the prediction of opinion distribution for May and August 2009. The prediction for May 2009 uses the data collected during April 2009, as shown in Fig. 2.6a, while the prediction for August 2009 uses the data collected during June 2009, as shown in Fig. 2.6b. It can be concluded from these results that the distribution of



(a) May 2009

(b) August 2009

Figure 2.6: Predicted distribution of student opinions regarding U.S. President Obama.

opinions are affected by the social dynamics (i.e., opinion change and relationships) of the students. Furthermore, the MFG-based algorithm can estimate and predict the distribution of opinions of the students.

2.7 Related Works

Belief and opinion dynamics determine how social network users influence each other. Specifically, these models dictate how a user changes or updates its opinion based not only on its initial opinion but also on the opinions of the neighboring users. Well-known models of belief and opinion evolution are the DeGroot (DG) model [37], the Friedkin-Johnsen (FJ) model [33], and the Hegselmann-Krause (HK) model [31]. The DG model states that the current opinion of a user is the weighted sum of the opinions of the neighboring users, whereas the HK model states that the current opinion of a user is based on the average opinion of the neighboring users. Meanwhile, the FJ model builds on the DG model where the current opinion of a user is calculated by imposing a trade-off between the weighted sum of the neighboring opinions and the user's initial opinion. Variations of these models as well as other opinion dynamics models can be found in [10], [38], [39], and [40].

Since social networks are populated with a large number of interacting users, mean field analysis has been utilized in a few works. In [41], the authors developed a model of social belief and opinion evolution. They derived a Fokker-Planck (FP) equation representing how the probability density of the users in the belief-personality space varies with time. Meanwhile, the authors of [42] proposed an MFG approach to model emulation, mimicry, and herding behaviors that can be observed when a large number of social groups interact. Also, this work considered social networks with a large number of groups. MFG was utilized in [43] to describe the propagation of opinion in social network according to a stochastic averaging process in the presence of an adversarial disturbance.

Other mathematical frameworks have been proposed as tools to model opinion dynamics in social networks. The authors of [44] proposed the evolution of opinions as well as their uncertainties in social networks as a fuzzy opinion network. The evolution of opinions on several interdependent topics as well as the convergence of mutually dependent opinions were investigated in [45]. Community detection, or finding connected users with similar opinion, was formulated as a multi-objective optimization problem through a graph K-means framework in [46]. A measurement analysis of user behavior in popular OSNs was implemented in [47]. The authors addressed two issues in OSNs, the characterization of user activities and usage patterns and found out that these issues can be represented by well-known probability distributions.

2.8 Conclusion

Social networks have been an important platform for social media users to express their opinion on social topics and events. Meanwhile, social network service providers aim to improve their service despite the growing number of demand from the network users. One way of improving their service is to gain information and predict the behavior of the network users. Thus, belief and opinion evolution in social networks has gained the interest of researchers. Since the number of social network users has grown recently, an appropriate mathematical framework is necessary in order to analyze and model the behavior of social network users. Consequently, this work has proposed the application of a multiple-population mean field game (MFG) to gain knowledge about the belief and opinion evolution of social network users. Based on the simulations, the proposed MFG-based framework provides insights on how users and populations behave on a multiple-population social network. Using a social evolution dataset as a benchmark, the proposed MFG-based method allows for an effective approach to estimate and predict the distribution of opinions of social network users on a social topic.

Chapter 3

Mean-Field-Type Game Approach to Computation Offloading in Multi-Access Edge Computing Networks

Multi-access edge computing (MEC) has been proposed to reduce the latency inherent in traditional cloud computing. One of the services offered in an MEC network (MECN) is computation offloading in which computing nodes, with limited capabilities and performance, can offload computation-intensive tasks to other computing nodes in the network. Meanwhile, mean-field-type game (MFTG) has been recently applied in engineering applications where the number of decision makers is finite and a decision maker can be distinguishable from other decision makers and have a non-negligible effect on the total utility of the network. Since MECNs are implemented through a finite number of computing nodes and the computing capability of a computing node can affect the state (i.e., the number of computation tasks) of the network, non-cooperative and cooperative MFTG approaches are proposed in this work to formulate the computation offloading problems in an MECN. In these scenarios, the goal of each computing node is to offload a portion of the aggregate computation tasks from the network that minimizes a specific cost. Then, a direct approach is utilized to calculate the optimal computation offloading solution (i.e., optimal portion of the aggregate computation tasks) of a computing node. Furthermore, non-cooperative and cooperative MFTG-based algorithms are proposed to implement computation offloading in an MECN. Finally, simulations are presented to show the significance and advantage of the proposed MFTG-based computation offloading algorithms over the traditional methods.

3.1 System Model

Fig. 3.1 shows the system model proposed in this work. The end user (EU) devices offload computation-intensive tasks that cannot be performed locally to the task aggregator (TA) in the area or cell the EU is located. Each EU decides to offload based on algorithms presented in the literature such as in [48]. Then, the TA combines all the computation tasks submitted by the EUs in the area. It organizes the computation tasks to reduce redundancy and overloading of computation tasks. Moreover, it performs a portion of the tasks and directly sends the results to the corresponding EUs. Afterwards, the TA offloads parts of the remaining aggregate computation tasks to the edge computing nodes (ECNs) in the cell. Each ECN is capable of performing computation-intensive task and is more powerful than a typical mobile EU equipment. For instance, a typical ECN has a computing power of about 10,000 to 100,000 times that of a mobile phone [49]. After the ECNs perform their respective offloaded computation tasks, the results are transmitted to the requesting EUs.

Consider a cell of an MECN consisting of one TA and a finite set $\mathcal{N}$ of ECNs with $|\mathcal{N}| = N$. The time horizon defined as $t \in [0, T]$ is finite, where $T > 0$ is the terminal time. Let the network state $x(t)$ be the number of aggregate computation tasks to be offloaded by the TA to the ECNs at time t. In addition, denote the network state dynamics $\dot{x}(t) = dx(t)/dt$ as the change or evolution of the number of aggregate computation tasks with respect to time. Also, let the admissible computation offloading control $u_i(t)$ be the portion of $x(t)$ offloaded by ECN i from the TA at time t. The goal of each ECN $i \in \mathcal{N}$ is to determine its optimal control $u_i^*(t)$ that minimizes its cost, defined by a cost function J_i, subject to the network state dynamics $\dot{x}(t)$.

In the following subsections, the cost function and network state dynamics equation are discussed. Important parameters that influence the optimal control of an ECN are also introduced.

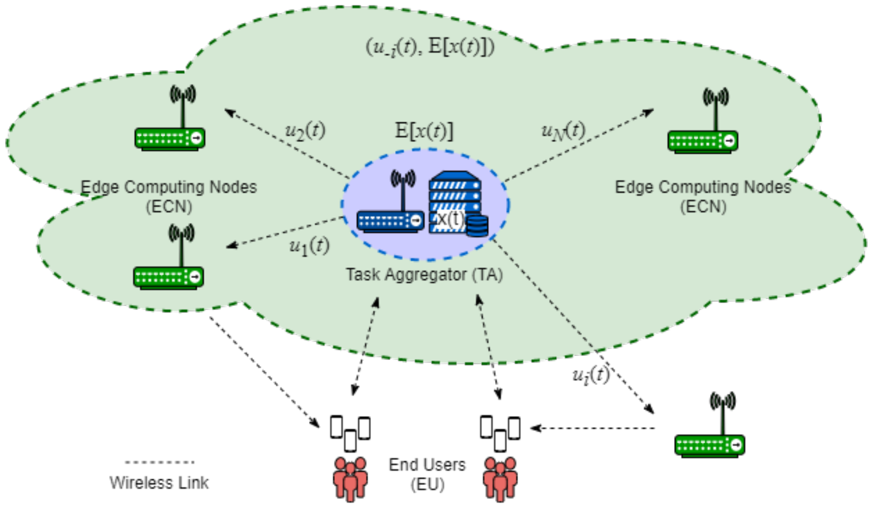


Figure 3.1: The proposed system model for computation offloading in MECNs.

3.1.1 Cost Functions

In this work, the ECNs are assumed to follow a quadratic cost function because of its desirable economic properties such as monotonicity, concavity, non-decreasing [50].

Let the consumed energy of ECN i per CPU cycle be $\varepsilon_i = \kappa_{e,i} f_i^2$ [51], where $\kappa_{e,i}$ is a constant depending on the architecture of the CPU of ECN i and f_i is the computing capability (i.e., the number of CPU cycles per unit time) of ECN i. To calculate the cost associated with the energy consumed by a certain number of computation tasks, define the energy cost coefficient e_i as the cost (per unit time) of energy spent by computing node i per squared number of CPU cycles,

$$e_i = w_{e,i}\varepsilon_i^2 = w_{e,i}(\kappa_{e,i} f_i^2)^2, \tag{3.1}$$

where the constant $w_{e,i}$ is the weight or significance assigned to energy consumption cost. A higher value of $w_{e,i}$ means that an ECN prioritizes minimizing its energy consumption.

Meanwhile, to calculate the cost corresponding to the execution or computation time of a certain number of computation task, define the computation time cost coefficient τ_i as the cost (per unit time) associated with execution time spent by ECN i per squared number of

CPU cycles,

$$\tau_i = \frac{w_{d,i}}{f_i^2}, \tag{3.2}$$

where the constant $w_{d,i}$ is the weight assigned by ECN i to the computation time cost. A higher value of $w_{d,i}$ means that ECN i prioritizes minimizing the cost from computation time.

Lastly, to quantify the cost earned by the TA from offloading computation task to ECN i, define the offloading cost coefficient ρ_i as the cost (per unit time) incurred by the TA per squared number of aggregate computation task. It takes into account the cost not associated with computation by an ECN i such as processing and transmission from the TA to ECN i.

Combining these cost coefficients with the network state $x(t)$ and the control $u_i(t)$ yields the running cost function that tells how much the cost increases or decreases with time,

$$L_i(x(t), u_i(t), t) = \frac{1}{2}\big[\rho_i x^2(t) + (\tau_i + e_i)u_i^2(t)\big]. \tag{3.3}$$

Since the goal of computation offloading is to offload tasks from the TA to the ECNs, the number of computation tasks $x(t)$ that remains at the TA at terminal time T is penalized. That is, $x(T)$ is considered as a part of the cost to be minimized. Since the cost at terminal time T is proportional to the number of computation tasks $x(T)$, the terminal cost function for ECN i is stated as

$$\Phi_i(x(T), T) = \frac{1}{2}\rho_i x^2(T). \tag{3.4}$$

In other words, it computes the cost incurred by ECN i based on the network state $x(t)$ at $t = T$.

3.1.2 Network State Dynamics Equation

The network state dynamics $\dot{x}(t)$ refers to the evolution of the network state $x(t)$ with respect to time t. In our computation offloading system model, $\dot{x}(t)$ refers to the dynamics or change in the number of aggregate computation tasks at the TA with time. Let $q_{\text{in}}(t)$ be the incoming rate of the computation tasks to the TA. The number of computation tasks $x(t)$ at the TA is related according to $q_{\text{in}}(t) = r_0 x(t)$, where r_0 is the frequency at which computation tasks arrive at the TA and is defined as

$$r_0 = \frac{R_0}{C_0} = \frac{1}{C_0}\sum_{j=1}^{M} B_j \log_2(1+\gamma_j) = \frac{1}{C_0}\sum_{j=1}^{M} B_j \left(1 + \frac{P_j g_{0,j}}{N_0 + I_j}\right), \tag{3.5}$$

with R_0 as the maximum incoming rate of computation tasks the TA handle and C_0 as the capacity or the maximum number of computation tasks the TA can store. Furthermore, R_0 is the sum of the rates the TA receives from M EUs, B_j is the channel bandwidth for EU j, and γ_j is the signal-to-interference-plus-noise (SINR) ratio between the TA and EU j, with P_j as the EU j transmit power, $g_{j,i}$ as the channel gain between the TA and EU j, N_0 as the background noise power, and I_j as the interference noise power experience by EU j.

On the other hand, the outgoing rate $q_{\text{out}}(t)$ of computation task from the TA is affected by the computation offloading control $u_i(t)$ of ECN $i \in \mathcal{N}$. Hence, $q_{\text{out}}(t) = \sum_{i=1}^{N} r_i u_i(t)$, where r_i is the frequency at which computation tasks arrive at ECN i and is defined as

$$r_i = \frac{R_i}{C_i} = \frac{B_i \log_2(1+\gamma_i)}{C_i} = \frac{B_i}{C_i}\log_2\left(1 + \frac{P_i g_{i,0}}{N_0 + I_i}\right), \tag{3.6}$$

with R_i as the maximum outgoing rate of computation tasks to ECN i and C_i as the capacity or the maximum number of computation tasks ECN i can handle. In addition, B_i is the channel bandwidth of ECN i and γ_i is the SINR between the TA and ECN i, with P_i as the transmit power of ECN i, $g_{i,0}$ as the channel gain between the TA and ECN i, N_0 as the background noise power, and I_i as the interference power experienced by ECN i.

Since the total rate of computation task $\dot{x}(t) = q_{\text{in}}(t) - q_{\text{out}}(t)$, then the network state dynamics equation can be written as

$$dx(t) = \left(r_0 x(t) - \sum_{i=1}^{N} r_i u_i(t)\right) dt, \tag{3.7}$$

which is similar in form to the state dynamics equation used in [52].

To summarize, the cost function is affected by $u_i(t)$ since the the cost depends on the number of tasks ECN i offloads from the TA. On the other hand, the state dynamic equation is affected by $r_i u_i(t)$ since the change in the number of $x(t)$ depends on the rate $r_i u_i(t)$ at which tasks are offloaded to ECN i.

In the next section, these formulations are extended to be able to adapt an MFTG approach. The main feature of this method is the addition of mean field terms $\bar{x}(t)$ and $\bar{u}(t)$ in the cost functions and the state dynamics equation. Consequently, each ECN now aims to compute its optimal control $u_i^*(t)$ given the mean field terms $\bar{x}(t)$ and $\bar{u}(t)$ and the difference $x(t) - \bar{x}(t)$ and $u(t) - \bar{u}(t)$.

3.2 Mean-Field-Type Game Problem Formulation

The theory of mean field game (MFG), introduced in [13], [53], [54], and [55], has been used in a variety of applications formulated as games among a large number of decision makers that aim to optimize their own payoffs or cost functions subject to a state dynamic equation. The main concept behind MFG is that each decision maker determines its optimal strategy (i.e., the strategy or action that optimizes its payoff or cost function) based on an aggregate information about the states of other decision makers. In other words, a decision maker computes its optimal strategy based on a statistical distribution of the states of other decision makers (i.e., a mean-field term) instead on a full knowledge of the states of other decision makers.

According to [56], most MFG models share the following assumptions: (i) there are infinitely many decision makers, (ii) the decision makers are indistinguishable, and (iii) a decision maker has negligible effect on the global utility. However, in engineering applications, these assumptions may be difficult to prove. Consequently, a more relaxed mean-field-type game (MFTG) has been proposed in the literature. In MFTG, the number of decision makers may be infinite or finite, the decision makers may not be indistinguishable, and finally, a decision maker may have a significant effect on the global utility. Applications of MFTG include distributed power networks [57], network security [58], and multilevel building evacuation [59].

In this section, computation offloading in MECN is formulated as an MFTG. First, the cost functions to be minimized by an ECN are derived. Then, the state dynamics, the differential equation constraint of the minimization problem, is formulated. The MFTG cost functions and state dynamic equation contain mean field terms that quantify the behavior or strategy of all the computing nodes. Lastly, the two MFTG computation offloading problems are stated, a non-cooperative MFTG problem where the ECNs minimize their own cost function independently and a cooperative MFTG problem where the ECNs minimize a single global cost function.

3.2.1 Preliminaries

The network state or state of the TA $x(t)$ refers to the number of aggregate computation tasks to be offloaded to the ECNs. An admissible computation offloading control or strategy $u_i(t)$ of ECN i refers to a portion of $x(t)$ it can offload from the TA, while the set $\mathcal{U}_i$ denotes the set of all admissible controls of ECN i. Vector $u(t) = [u_i(t)]_{i\in\mathcal{N}}$ contains the control of all the ECNs in the cell, while the vector $u_{-i}(t) = [u_i(t)]_{i\in\mathcal{N}\setminus i}$ contains the control of all ECNs in the cell except ECN i.

The following subsections present the cost functions and state dynamic equation in an MFTG setting. The main difference in the formulations to follow is the inclusion of mean field terms $\bar{x}(t) = \mathbb{E}[x(t)]$ and $\bar{u}(t) = \mathbb{E}[u(t)]$. Consequently, a tilde $\sim$ is put on top of the MFTG cost function $\tilde{J}_i$ and state dynamic function $\tilde{f}$ to differentiate them from their mean-field-free counterparts. Afterwards, the resulting MFTG-based computation offloading problems are stated.

3.2.2 Cost Functions

The total cost function $\tilde{J}_i(u)$ of ECN i consists of the running cost function $\tilde{L}_i(x, u, \bar{x}, \bar{u}, t)$ that corresponds to the accumulated cost of ECN i for performing a portion $u_i(t)$ of $x(t)$ and the terminal cost function $\tilde{\Phi}_i(x, \bar{x}, T)$ that penalizes the computing node at terminal time $t = T$ depending on how far the network state $x(t)$ is from a target state (e.g., $x(t) = 0$, when all of the aggregate computation task are offloaded). Mathematically,

$$\tilde{J}_i(u) = \mathbb{E}\bigg[\int_0^T \tilde{L}_i(x, u, \bar{x}, \bar{u}, t)\, dt + \tilde{\Phi}_i(x, \bar{x}, T)\bigg]. \tag{3.8}$$

However, the running cost $\tilde{L}_i(x, u, \bar{x}, \bar{u}, t)$ for MFTG differs from that in (3.3) since $\tilde{L}_i$ depends on the expected values of the network state $\bar{x}(t)$ and the control $\bar{u}$. The expected values have been included in the cost function because these values are assumed to be known (i.e., the MECN keeps track of them), and consequently, the difference to these expected values, $x(t) - \bar{x}(t)$ and $u_i(t) - \bar{u}_i(t)$. Hence,

$$\begin{aligned}
\mathbb{E}\big[\tilde{L}_i(x, u, \bar{x}, \bar{u}, t)\big] &= \frac{1}{2}\mathbb{E}\big[\rho_i x^2(t) + \bar{\rho}_i \bar{x}^2(t) + (\tau_i + e_i)u_i^2(t) + (\bar{\tau}_i + \bar{e}_i)\bar{u}_i^2(t)\big], \\
&= \frac{1}{2}\mathbb{E}\big[\rho_i (x(t) - \bar{x}(t))^2 + (\rho_i + \bar{\rho}_i)\bar{x}^2(t) + (\tau_i + e_i)(u_i(t) - \bar{u}_i(t))^2 \\
&\quad + (\tau_i + \bar{\tau}_i + e_i + \bar{e}_i)\bar{u}_i^2(t)\big], \\
&= \frac{1}{2}\big(\rho_i \mathrm{var}[x(t)] + (\rho_i + \bar{\rho}_i)\bar{x}^2(t) + (\tau_i + e_i)\mathrm{var}[u_i(t)] \\
&\quad + (\tau_i + \bar{\tau}_i + e_i + \bar{e}_i)\bar{u}_i^2(t)\big),
\end{aligned} \tag{3.9}$$

where $\bar{\rho}_i$, $\bar{\tau}_i$ and $\bar{e}_i$ refer to the mean of cost coefficients defined in Section 3.1.1.

Similarly, the terminal cost $\tilde{\Phi}_i(x, \bar{x}, T)$ depends as well on the expected value of the network state at time T so that the goal of an ECN i is to minimize

$$\mathbb{E}\big[\tilde{\Phi}_i(x,\bar{x},T)\big] = \frac{1}{2}\mathbb{E}\big[\rho_i x^2(T) + \bar{\rho}_i \bar{x}^2(T)\big] = \frac{1}{2}\mathbb{E}\big[\rho_i (x(T)-\bar{x}(T))^2 + (\rho_i+\bar{\rho}_i)\bar{x}^2(T)\big],$$
$$= \frac{1}{2}\big(\rho_i \mathrm{var}[x(T)] + (\rho_i+\bar{\rho}_i)\bar{x}^2(T)\big). \tag{3.10}$$

The quadratic cost functions presented in this work refer to the penalty incurred by the network through the ECNs for executing a specific number of computation tasks. While the ECNs are constrained in terms of their local energy consumption and execution time through e_i and τ_i, respectively, the penalty allows the ECNs follow a network-wide algorithm that optimizes the network performance. Hence, the physical meaning of the costs refers to the penalty set by the network to the ECNs. These penalties are based on physical quantities the computing nodes spend when performing computation tasks. The number of computation tasks offloaded by each ECN is limited by these penalties. Moreover, the costs are also indicators of network performance since a low cost may indicate a low terminal cost $\tilde{\Phi}_i(x,\bar{x},T)$ which means a low number of un-offloaded task remains in the TA; also, a low cost may indicate a low running cost $\tilde{L}_i(x,u,\bar{x},\bar{u},t)$ which means the ECNs execute the number of computation tasks that satisfy their local energy and time constraints.

3.2.3 Network State Dynamics Equation

The network state dynamics $\dot{x}(t)$ of refers to the evolution of the state $x(t)$ of the TA with respect to time t. In (3.7), the network state dynamics is affected by the current network state $x(t)$ and the controls $u_i(t), \forall i \in \mathcal{N}$. Like in the MFTG cost functions, the expected values $\bar{x}$ and $\bar{u}$ are included in the MFTG state dynamic equation $\dot{x}(t)$ so that the evolution of the state can be formulated in terms of the deviation of $x(t)$ and $u(t)$ from their respective expected values. Consequently, the updated network state dynamics of a computation offloading in an MFTG setting is

$$dx(t) = \tilde{f}(x, u, \bar{x}, \bar{u})\, dt + \sigma\, dw(t), \tag{3.11}$$

where $w(t)$ is a standard Wiener process, σ is a coefficient that captures the randomness in the state dynamics, the drift term $\tilde{f}(x, u, \bar{x}, \bar{u}, t)$ is given by

$$\tilde{f}(x, u, \bar{x}, \bar{u}, t) = r_0 x(t) + \bar{r}_0 \bar{x}(t) - \left(\sum_{i=1}^{N} r_i u_i(t) + \sum_{i=1}^{N} \bar{r}_i \bar{u}_i(t) \right), \tag{3.12}$$

and the coefficients are defined as $\bar{r}_0 = \mathbb{E}[\frac{R_0}{C_0}]$ and $\bar{r}_i = \mathbb{E}[\frac{R_i}{C_i}] = \mathrm{E}[\frac{B_i \log_2(1+\gamma_i)}{C_i}]$. The drift term can be written in an equivalent form

$$\begin{aligned} \tilde{f}(x, u, \bar{x}, \bar{u}, t) &= r_0(x(t) - \bar{x}(t)) + (r_0 + \bar{r}_0)\bar{x}(t) \\ &\quad - \left(\sum_{i=1}^{N} r_i(u_i(t) - \bar{u}_i(t)) + \sum_{i=1}^{N} (r_i + \bar{r}_i)\bar{u}_i(t) \right), \end{aligned} \tag{3.13}$$

that expresses the network state dynamics as the sum of the mean fields $\bar{x}(t)$ and $\bar{u}_i(t)$ and the terms $x(t) - \bar{x}(t)$ and $u_i(t) - \bar{u}_i(t)$. In the following computation offloading scenarios, since the mean number of computation tasks $x(t)$ is tracked, the difference $x(t) - \bar{x}(t)$ of the current state $x(t)$ from mean state $\bar{x}(t)$ can be easily known. The same principle is applied with $u_i(t)$ and $\bar{u}_i(t)$.

3.2.4 Non-Cooperative Problem

Consider a MECN consisting of $N \geq 2$ ECNs. Each ECN is capable of offloading and performing computation tasks from the TA. In addition, suppose the MECN implements a non-cooperative scenario where each ECN computes its offloading strategy by minimizing its own cost function. If the cost function of ECN i is defined by (3.8), then in a non-cooperative setting each ECN i tries to solve the MFTG problem

$$\begin{aligned}
\inf_{u_i \in \mathcal{U}_i} \quad & \tilde{J}_i(u) = \frac{1}{2}\mathbb{E}\bigg[\int_0^T \big[\rho_i(x(t)-\bar{x}(t))^2 + (\rho_i+\bar{\rho}_i)\bar{x}^2(t) + (\tau_i+e_i)(u_i(t)-\bar{u}_i(t))^2 \\
& \qquad + (\tau_i+\bar{\tau}_i+e_i+\bar{e}_i)\bar{u}_i^2(t)\big]\, dt \\
& \qquad + \rho_i(x(T)-\bar{x}(T))^2 + (\rho_i+\bar{\rho}_i)\bar{x}^2(T)\bigg], \\
\text{subject to} \quad & dx(t) = \bigg[r_0(x(t)-\bar{x}(t)) + (r_0+\bar{r}_0)\bar{x}(t) - \bigg(\sum_{i=1}^N r_i(u_i(t)-\bar{u}_i(t)) \\
& \qquad + \sum_{i=1}^N (r_i+\bar{r}_i)\bar{u}_i(t)\bigg)\bigg]\, dt + \sigma(t)\, dW(t), \\
& x(0) = x_0,
\end{aligned} \tag{3.14}$$

where $\bar{x}(t) < +\infty$. Any control $u_i^*(t)$ that satisfies (3.14) is the best-response of computing node i to $(u_{-i}, \mathbb{E}[x(t)])$.

Definition 4. *Any control $u_i^*(t) \in \mathcal{U}_i$ satisfying* (3.14) *is called a risk-neutral best-response control of computing node i to the control $u_{-i} \in \Pi_{j\in\mathcal{N}}\mathcal{U}_j$ of the other computing nodes $j \neq i$.*

The set of best-response controls of computing node i is defined by $\mathcal{BR}_i : \Pi_{j\in\mathcal{N}}\mathcal{U}_j \to 2^{\mathcal{U}_i}$, where $2^{\mathcal{U}_i}$ is the set of subsets of $\mathcal{U}_i$. Using the concept of best-response control strategy, a Nash equilibrium of (3.14) is (u_i^*, u_{-i}^*), where every ECN i solves their best-response control u_i^*.

Definition 5. *A Nash equilibrium of the mean-field-type game in* (3.14) *is a control profile (u_i^*, u_{-i}^*) such that for every computing node i,*

$$\tilde{J}_i(u_i^*, u_{-i}^*) \leq \tilde{J}_i(u_i, u_{-i}^*), \forall u_i \in \mathcal{U}_i. \tag{3.15}$$

3.2.5 Cooperative Problem

Suppose the ECNs try to jointly minimize a global cost function $\tilde{J}_0(u) = \mathbb{E}[\sum_{i=1}^{N} \tilde{J}_i(u)]$ where $u = (u_1, \ldots, u_N)$ is the computation offloading control profile in a cooperative setting. Then, the corresponding cooperative MFTG problem is given by

$$
\begin{aligned}
\inf_{u_i \in \mathcal{U}_i} \quad & \tilde{J}_0(u) = \frac{1}{2}\mathbb{E}\Big[\sum_{i=1}^{N}\int_0^T \big[\rho_i(x(t)-\bar{x}(t))^2 + (\rho_i+\bar{\rho}_i)\bar{x}^2(t) + (\tau_i+e_i)(u_i(t)-\bar{u}_i(t))^2 \\
& \qquad\qquad + (\tau_i+\bar{\tau}_i+e_i+\bar{e}_i)\bar{u}_i^2(t)\big]\,dt \\
& \qquad + \rho_i(x(T)-\bar{x}(T))^2 + (\rho_i+\bar{\rho}_i)\bar{x}^2(T)\Big], \\
\text{subject to} \quad & dx(t) = \Big[r_0(x(t)-\bar{x}(t)) + (r_0+\bar{r}_0)\bar{x}(t) - \Big(\sum_{i=1}^{N} r_i(u_i(t)-\bar{u}_i(t)) \\
& \qquad + \sum_{i=1}^{N}(r_i+\bar{r}_i)\bar{u}_i(t)\Big)\Big]\,dt + \sigma(t)\,dW(t), \\
& x(0) = x_0.
\end{aligned}
\tag{3.16}
$$

Any control profile $u^* = (u_1^*, \ldots, u_N^*)$ that satisfies (3.16) is a global optimum control profile that minimizes the global cost function $\tilde{J}_0$.

The next section provides the method proposed in [60] to solve for a solution of linear-quadratic MFTGs such as (3.14) and (3.16).

3.3 Linear-Quadratic Mean-Field-Type Game Solution Using A Direct Method

The MFTG problems defined in (3.14) and (3.16) are called linear-quadratic MFTGs (LQ-MFTG) since the cost functional is quadratic and the state dynamics is linear with respect to the state and control. Because of their special form, the authors in [60] proposed a direct approach in computing the optimal control $u_i^*(t)$ of LQ-MFTG. The proposed method can solve a LQ-MFTG without dealing with coupled partial differential equations. The authors proved that the proposed direct approach to LQ-MFTG yields the same solution as an analytical approach. Based on this method, the main concepts in deriving the solution for the non-cooperative and cooperative MFTG problems are presented. The solution $u_i^*(t)$ to each problem refers to the computation offloading control or the number of computation tasks $u_i^*(t)$ ECN i must offload from the TA in order to minimize the corresponding cost of the problem. In other words, the optimal control is the optimal number of computation tasks to be offloaded by an ECN such that the penalties incurred by the network due to the number of executed computation tasks by the ECN and the remaining tasks at the TA are minimized.

3.3.1 Non-Cooperative Solution

The direct method for the LQ-MFTG problem starts with choosing a guess cost functional $\phi_i(x,t)$. Since the cost functional J_i is quadratic, the corresponding $\phi_i(x,t)$ is quadratic as well,

$$\phi_i(x,t) = \frac{1}{2}\alpha_i(x-\bar{x})^2 + \frac{1}{2}\beta_i\bar{x}^2 + \gamma_i\bar{x} + \delta_i,$$

where α_i, β_i, γ_i, and δ_i are restricted to time-invariant coefficients for $[0,T]$.

Then, apply the Ito's formula for a drift-diffusion process to $\phi_i(x,t)$ with $t=T$,

$$\begin{aligned}\phi_i(x(T),T) = \phi_i(x(0),0) &+ \int_0^T \left(\partial_t\phi_i + \tilde{f}(x,u,\bar{x},\bar{u},t)\partial_x\phi_i + \frac{\sigma^2}{2}\partial_{xx}\phi_i\right)dt \\ &+ \int_0^T \sigma(t)\partial_x\phi_i\, dW(t).\end{aligned} \tag{3.17}$$

The next step is to compute and substitute the partial derivatives $\partial_t\phi_i, \partial_x\phi_i$, and $\partial_{xx}\phi_i$ to $\phi_i(x(T),T)$ and take its expectation, $\mathbb{E}[\phi_i(x(T),T) - \phi_i(x(0),0)]$. Afterwards, the gap $\tilde{J}_i - \mathbb{E}[\phi_i(x(0),0)]$ is calculated.

Finally, the optimal control u_i^* is derived from $\min_{u_i \in \mathcal{U}_i} \tilde{J}_i(u)$ using the appropriate optimality principles. A control u_i is called a feedback control if it is a function of time t and the state $x(t)$. To compute the best-response control u_i^* of computing node i to feedback strategies $u_j, j \neq i$, complete the square of the gap $\tilde{J}_i - \mathbb{E}[\phi_i(x(0), 0)]$ to get

$$\begin{aligned}\tilde{J}_i - \mathbb{E}[\phi_i(x(0), 0)] = \frac{1}{2}\mathbb{E}\Big[\int_0^T (\tau_i + e_i)\Big(u_i - \bar{u}_i - \frac{r_i}{\tau_i + e_i}\alpha_i(x - \bar{x})\Big)^2 dt\Big] \\ + \frac{1}{2}\mathbb{E}\Big[\int_0^T (\tau_i + \bar{\tau}_i + e_i + \bar{e}_i)\Big(\bar{u}_i - \beta_i \frac{r_i + \bar{r}_i}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i}\bar{x}\Big)^2 dt\Big] \\ + \frac{1}{2}\mathbb{E}\Big[\int_0^T \sigma^2 \alpha_i \, dt\Big].\end{aligned} \tag{3.18}$$

Consequently, the equivalent objective functional becomes

$$\inf_{u_i \in \mathcal{U}_i} \tilde{J}_i = \frac{1}{2}\alpha_i(0)\mathrm{var}[x(0)] + \frac{1}{2}\beta_i(0)(\mathbb{E}[x(0)])^2 + \frac{1}{2}\mathbb{E}\Big[\int_0^T \sigma^2(t)\alpha_i(t)\, dt\Big]. \tag{3.19}$$

Using this equivalent objective functional, the following theorem holds for the optimal control u_i^*.

Theorem 4. *Let the cost functional $\tilde{J}_i(u)$ of a LQ-MFTG problem take the form $\phi_i(x, t) = \frac{1}{2}\alpha_i(x - \bar{x})^2 + \frac{1}{2}\beta_i\bar{x}^2 + \gamma_i\bar{x} + \delta_i$, where α_i, β_i, γ_i, and δ_i are constants. Then, the optimal control $u_i^*(t)$ associated with the problem is given by*

$$u_i^*(t) = \frac{r_i}{\tau_i + e_i}\alpha_i(x - \bar{x}) + \frac{r_i + \bar{r}_i}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i}\beta_i\bar{x}, \tag{3.20}$$

where the constants α_i and β_i solve the following equations, respectively,

$$\frac{r_i^2}{\tau_i + e_i}\alpha_i^2 + 2\Big(\sum_{j=1, j\neq i}^{N} \frac{r_j^2}{\tau_j + e_j}\alpha_j - r_0\Big)\alpha_i - \rho_i = 0, \tag{3.21}$$

and

$$\frac{(r_i + \bar{r}_i)^2}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i}\beta_i^2 + 2\Big(\sum_{j=1, j\neq i}^{N} \frac{(r_j + \bar{r}_j)^2}{\tau_j + \bar{\tau}_j + e_j + \bar{e}_j}\beta_j - (r_0 + \bar{r}_0)\Big)\beta_i - (\rho_i + \bar{\rho}_i) = 0. \tag{3.22}$$

The mean field term $\bar{x}(t)$ is given by

$$\bar{x}(t) = \bar{x}(0) \exp\left(\int_0^t \left((r_0 + \bar{r}_0) - \sum_{i=1}^{N} \frac{\beta_i (r_i + \bar{r}_i)^2}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i} \right) ds \right), \tag{3.23}$$

and $\bar{u}_i$ is expressed as $\beta_i(r_i + \bar{r}_i)/(\tau_i + \bar{\tau}_i + e_i + \bar{e}_i)\bar{x}$.

Proof. The optimal control u_i^* is obtained by minimizing the following terms with respect to control u_i and $\bar{u}_i$,

$$\begin{aligned} \frac{\partial}{\partial u_i} \Bigg[& (\tau_i + e_i) \left(u_i - \bar{u}_i - \frac{r_i}{\tau_i + e_i} \alpha_i (x - \bar{x}) \right)^2 \\ & + (\tau_i + \bar{\tau}_i + e_i + \bar{e}_i) \left(\bar{u}_i - \beta_i \frac{r_i + \bar{r}_i}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i} \bar{x} \right)^2 \Bigg] \quad = 0, \end{aligned}$$

which yields $u_i = \frac{r_i}{\tau_i + e_i} \alpha_i (x - \bar{x}) + \bar{u}_i$, where $\bar{u}_i = \beta_i (r_i + \bar{r}_i)/(\tau_i + \bar{\tau}_i + e_i + \bar{e}_i)\bar{x}$. Meanwhile, the mean field $\bar{x}(t)$ is derived by taking the expectation of the state dynamic equation in (3.16) and then solving the resulting differential equation for $\bar{x}(t)$. ∎

Theorem 1 states that the optimal number of computation tasks ECN i must offload from the TA in a non-cooperative scenario is given in (3.20). This number minimizes the cost incurred by ECN i where α_i and β_i satisfy the conditions (3.21) and (3.22), respectively, and the mean field $\bar{x}(t)$ satisfies (3.23). It can be deduced that $u_i^*(t)$ not only depends on $x(t)$ but also on how much $x(t)$ exceeds $\bar{x}(t)$. Moreover, α_i and β_i reflect the weights of how much $u_i^*(t)$ depends on $x(t) - \bar{x}(t)$ and $\bar{x}(t)$, respectively.

3.3.2 Cooperative Solution

To obtain the global optimum solution to the cooperative LQ-MFTG problem in (3.16), the procedure stated in the previous subsection is followed. Hence, the LQ-MFTG problem in (3.16) is equivalent to

$$\inf_{u_1, \dots, u_N} \tilde{J}_0 = \frac{1}{2} \alpha_0(0) \mathrm{var}[x(0)] + \frac{1}{2} \beta_0(0) (\mathbb{E}[x(0)])^2 + \frac{1}{2} \mathbb{E}\Big[\int_0^T \sigma^2(t) \alpha_0(t) \, dt \Big]. \tag{3.24}$$

The corresponding optimal control $u_i^*(t)$ is given by the following theorem.

Theorem 5. *Let the cost functional $\tilde{J}_0(u)$ of a LQ-MFTG problem take the form $\phi_0(x,t) = \frac{1}{2}\alpha_0(x-\bar{x})^2 + \frac{1}{2}\beta_0\bar{x}^2$, where α_0 and β_0, are constants. Then, the optimal control $u_i^*(t)$ associated with the problem is given by*

$$u_i^*(t) = \frac{r_i}{\tau_i + e_i}\alpha_0(x-\bar{x}) + \frac{r_i + \bar{r}_i}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i}\beta_0\bar{x}, \tag{3.25}$$

where the constants α_0 and β_0 solve the following equations, respectively,

$$\left(\sum_{i=1}^{N}\frac{r_i^2}{\tau_i + e_i}\right)\alpha_0^2 - 2r_0\alpha_0 - \rho_0 = 0, \tag{3.26}$$

and

$$\left(\sum_{i=1}^{N}\frac{(r_i+\bar{r}_i)^2}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i}\right)\beta_0^2 - 2(r_0+\bar{r}_0)\beta_0 - (\rho_0 + \bar{\rho}_0) = 0. \tag{3.27}$$

The mean field term $\bar{x}(t)$ is given by

$$\bar{x}(t) = \bar{x}(0)\exp\left(\int_0^t \left((r_0+\bar{r}_0) - \beta_0\sum_{i=1}^{N}\frac{(r_i+\bar{r}_i)^2}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i}\right)ds\right), \tag{3.28}$$

and $\bar{u}_i$ is expressed as $\beta_0(r_i+\bar{r}_i)/(\tau_i + \bar{\tau}_i + e_i + \bar{e}_i)\bar{x}$.

Proof. The optimal control u_i^* is obtained by minimizing the following terms with respect to control u_i and $\bar{u}_i$,

$$\frac{\partial}{\partial u_i}\Bigg[(\tau_i + e_i)\left(u_i - \bar{u}_i - \frac{r_i}{\tau_i + e_i}\alpha_0(x-\bar{x})\right)^2 + (\tau_i + \bar{\tau}_i + e_i + \bar{e}_i)\left(\bar{u}_i - \beta_0\frac{r_i + \bar{r}_i}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i}\bar{x}\right)^2\Bigg] = 0,$$

which yields $u_i = \frac{r_i}{\tau_i + e_i}\alpha_0(x-\bar{x}) + \bar{u}_i$, where $\bar{u}_i = \beta_0(r_i+\bar{r}_i)/(\tau_i + \bar{\tau}_i + e_i + \bar{e}_i)\bar{x}$. Meanwhile, the mean field $\bar{x}(t)$ is derived by taking the expectation of the state dynamic equation in (3.14) and then solving the resulting differential equation for $\bar{x}(t)$. ∎

Theorem 2 states that the optimal number of computation tasks ECN i must offload from the TA in a cooperative scenario is given by (3.25). This number minimizes the cost incurred by ECN i where α_0 and β_0 satisfy (3.26) and (3.27), respectively, and the mean field $\bar{x}(t)$ satisfies (3.28). It can be deduced that $u_i^*(t)$ not only depends on $x(t)$ but also on how much $x(t)$ exceeds $\bar{x}(t)$. Moreover, α_0 and β_0 capture how dependent $u_i^*(t)$ is on $x(t) - \bar{x}(t)$ and $\bar{x}(t)$, respectively.

3.4 Mean-Field-Type-Game-Based Computation Offloading Algorithms

This section presents the algorithms that implement the proposed MFTG-based computation offloading developed in the previous sections. A non-cooperative algorithm based on Theorem 4 is designed to simulate a scenario when the ECNs decide to minimize their own cost function. The algorithm can be implemented in a decentralized manner where each ECN decides for itself the optimal number of task to offload from the TA. Meanwhile, a cooperative algorithm based on Theorem 5 is designed for situations when the ECNs decide to minimize a global cost function. The algorithm can be implemented in a centralized manner where the TA decides for every ECN the optimal number of task to offload to the ECN. The proposed MFTG-based algorithms calculate the optimal solution $u_i^*(t)$ that corresponds to the portion of computation tasks that each ECN must offload in order to optimize its cost. As illustrated later in Section 3.5, these algorithms improve the system cost and benefit-cost ratio of the local computing and dynamic greedy algorithms for computation offloading. Thus, the proposed MFTG-based algorithms can improve the targeted network performance.

Fig. 3.2 illustrates the general procedure involved in the proposed algorithms. First, each ECN i determines its own cost coefficients r_i, τ_i, and e_i. Then, in the non-cooperative

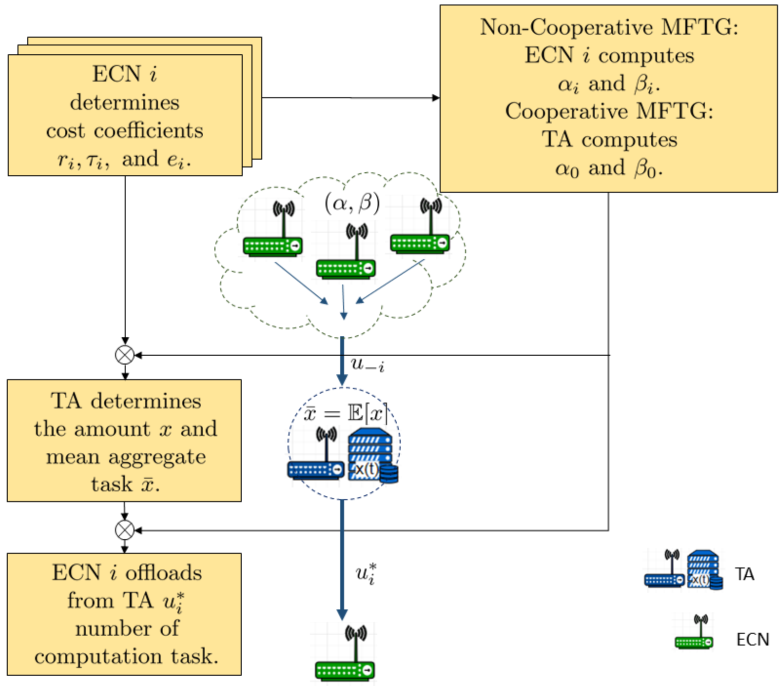


Figure 3.2: An illustration of the proposed MFTG computation offloading algorithms.

setting, ECN i computes the state and mean-state coupling coefficients α_i and β_i, whereas in the cooperative setting, the TA determines α_0 and β_0. These coefficients capture the effect of the state and mean-state to the optimal computation offloading control. At the same time, the TA determines the state $x(t)$ and mean-state $\bar{x}(t)$. Finally, the TA offloads the number of computation tasks to ECN i based on $u_i^*(t)$. The non-cooperative algorithm emulates a decentralized approach where each ECN determines its own $u_i^*(t)$, while the cooperative algorithm follows a centralized approach where the TA determines $u_i^*(t)$ of each ECN.

These algorithms require a sample period T_s and number of samples M, instead of a specified terminal time T. One main reason for this requirement is to avoid network parameter updates every time t, which can now be done every T_s. In addition, the cell dimension L and the number of ECNs N are required as well. The location of each node in the cell is limited within the area defined by $[0, L] \times [0, L]$. The TA is located at $z_0 = [L/2, L/2]$, whereas the location z_i of each ECN i is distributed randomly in the area. For each ECN i, its computing capability f_i and cost weights $w_{d,i}$ and $w_{e,i}$ are also defined.

3.4.1 Non-Cooperative Computation Offloading

Since the non-cooperative solution using the direct approach stated in Theorem 4 assumes that each ECN has knowledge about the other ECNs, it has to be simplified in order to be implemented more practically. Let

$$\bar{\lambda} = \frac{1}{N} \sum_{j \in \mathcal{N}} \frac{r_j^2}{\tau_j + e_j} \alpha_j = \frac{1}{N} \sum_{j \in \mathcal{N}} \lambda_j,$$

and

$$\bar{\mu} = \frac{1}{N} \sum_{j \in \mathcal{N}} \frac{(r_j + \bar{r}_j)^2}{\tau_j + \bar{\tau}_j + e_j + \bar{e}_j} \beta_j = \frac{1}{N} \sum_{j \in \mathcal{N}} \mu_j.$$

Then, it follows that

$$\sum_{j \in \mathcal{N} \setminus i} \lambda_j = N\bar{\lambda} - \lambda_i, \quad \text{and} \quad \sum_{j \in \mathcal{N} \setminus i} \mu_j = N\bar{\mu} - \mu_i. \tag{3.29}$$

Consequently, (3.21) and (3.22) can be rewritten as

$$\frac{r_i^2}{\tau_i + e_i} \alpha_i^2 + 2\big(N\bar{\lambda} - \lambda_i - r_0\big)\alpha_i - \rho_i = 0, \tag{3.30}$$

and

$$\frac{(r_i + \bar{r}_i)^2}{\tau_i + \bar{\tau}_i + e_i + \bar{e}_i} \beta_i^2 + 2\big(N\bar{\mu} - \mu_i - (r_0 + \bar{r}_0)\big)\beta_i - (\rho_i + \bar{\rho}_i) = 0. \tag{3.31}$$

Meanwhile, the mean values $\bar{r}_i$, $\bar{\tau}_i$, $\bar{e}_i$, and $\bar{\rho}_i$ can be found using the law of large numbers. It states that a sample average

$$\bar{S}_m = \frac{1}{m}(y_1 + \cdots + y_m),$$

converges to the expected value $\bar{y} = \mathbb{E}[y]$ as $m \to \infty$. Hence, the relationship between the parameters r_i, τ_i, e_i, and ρ_i and their respective expected values is given by

$$\lim_{m \to \infty} \frac{1}{m}(r_{i,1} + \cdots + r_{i,m}) = \bar{r}_i,$$

$$\lim_{m \to \infty} \frac{1}{m}(\tau_{i,1} + \cdots + \tau_{i,m}) = \bar{\tau}_i,$$

$$\lim_{m\to\infty} \frac{1}{m}(e_{i,1} + \cdots + e_{i,m}) = \bar{e}_i,$$

and

$$\lim_{m\to\infty} \frac{1}{m}(\rho_{i,1} + \cdots + \rho_{i,m}) = \bar{\rho}_i,$$

$\forall i \in \mathcal{N}$.

Algorithm 2 Non-Cooperative MFTG Computation Offloading

Set M, T_s, L, N, z_i, f_i, $w_{e,i}$, and $w_{d,i}$, $\forall i \in \mathcal{N}$.

Initialize $\bar{e}_i^{(0)}$, $\bar{\tau}_i^{(0)}$, $\bar{r}_i^{(0)}$, $\bar{\rho}_i^{(0)}$, $\bar{\lambda}$, and $\bar{\mu}$.

for $m = 1$ to M **do**

  for each ECN i in $\mathcal{N}$ **do**

    Compute e_i, τ_i, r_i using (3.1), (3.2), and (3.6), respectively. Compute α_i, β_i using (3.30) and (3.31).

    for each t in $0 \leq t \leq T_s$ **do**

      Observe and measure $x(t)$. Calculate $\bar{x}(t)$ using (3.23). Calculate $u_i^*(t)$ using (3.20).

    end for

    Update

      $\bar{e}_i^{(m)} = \frac{1}{m}(e_i + (m-1)\bar{e}_i^{(m-1)})$,

      $\bar{\tau}_i^{(m)} = \frac{1}{m}(\tau_i + (m-1)\bar{\tau}_i^{(m-1)})$,

      $\bar{r}_i^{(m)} = \frac{1}{m}(r_i + (m-1)\bar{r}_i^{(m-1)})$,

      $\bar{\rho}_i^{(m)} = \frac{1}{m}(\rho_i + (m-1)\bar{\rho}_i^{(m-1)})$.

    Update $\bar{\lambda}$ and $\bar{\mu}$ using (3.29).

  end for

end for

As a result, Algorithm 2 shows the non-cooperative computation offloading algorithm based on Theorem 4. After setting up some network parameters, each ECN i needs to initialize $\bar{r}_i$, $\bar{\tau}_i$, $\bar{e}_i$, $\bar{\lambda}$, and $\bar{\mu}$. Then, each ECN i determines r_i, τ_i, and e_i. Also, each ECN i estimates α_i and β_i using (3.21) and (3.22). Meanwhile, the TA broadcasts $x(t)$ and $\bar{x}(t)$ to the ECNs. Consequently, each ECN i can now calculate and offload from the TA the optimal offloading portion $u_i^*(t)$ that minimizes their own cost. Lastly, ECN i updates $\bar{r}_i$, $\bar{\tau}_i$, $\bar{e}_i$, $\bar{\lambda}$, and $\bar{\mu}$.

3.4.2 Cooperative Computation Offloading

Algorithm 3 implements the cooperative computation offloading based on Theorem 5. It starts with setting up some network parameters. Then, each ECN i initializes parameters such as $\bar{r}_i$, $\bar{\tau}_i$, and $\bar{e}_i$ and transmits them to the TA. Next, the TA computes α_0 and β_0 based on (3.26) and (3.27). Afterwards, the TA can now compute the optimal offloading control $u_i^*(t)$ of each ECN based on the values of $x(t)$ and $\bar{x}(t)$. Then, the TA offloads the corresponding number of computation task to each ECN i. Finally, the TA updates $\bar{r}_i$, $\bar{\tau}_i$, and $\bar{e}_i$.

3.5 Performance Evaluation

3.5.1 Baseline Approaches

To be able to evaluate the performance of the proposed MFTG computation offloading algorithms, they are compared with two typical algorithms in computation offloading. The first algorithm is the local computing based on [61]. It finds the number of computation tasks $x_0(t)$ that can be executed locally in the TA such that it satisfies the required deadline d_0, $x_0(t)/f_0 < d_0$. The cost function of the TA for local computing is defined by

$$J_{lo} = \mathbb{E}\left[\int_0^T (w_{d,0}\tau_{lo}x_0(t) + w_{e,0}e_{lo}x_0(t))\,dt\right], \tag{3.32}$$

Algorithm 3 Cooperative MFTG Computation Offloading

Set M, T_s, L, N, z_i, f_i, $w_{e,i}$, and $w_{d,i}$, $\forall i \in \mathcal{N}$.

Initialize $\bar{e}_i^{(0)}$, $\bar{\tau}_i^{(0)}$, and $\bar{r}_i^{(0)}$.

for $m = 1$ to M **do**

for each ECN i in $\mathcal{N}$ **do**

Compute e_i, τ_i, and r_i using (3.1), (3.2), and (3.6), respectively. Compute α_0 and β_0 using (3.26) and (3.27).

for each t in $0 \leq t \leq T_s$ **do**

Observe and measure $x(t)$. Calculate $\bar{x}(t)$ using (3.28). Calculate $u_i^*(t)$ using (3.25).

end for

Update

$\bar{e}_i^{(m)} = \frac{1}{m}(e_i + (m-1)\bar{e}_i^{(m-1)})$,

$\bar{\tau}_i^{(m)} = \frac{1}{m}(\tau_i + (m-1)\bar{\tau}_i^{(m-1)})$,

$\bar{r}_i^{(m)} = \frac{1}{m}(r_i + (m-1)\bar{r}_i^{(m-1)})$.

end for

end for

where $\tau_{lo} = 1/f_0$ refers to the amount of time to execute a unit of computation task, $e_{lo} = \kappa_e f_0^2$ refers to the energy consumption per unit of computation task, and the constants $w_{d,0}$ and $w_{e,0}$ refer to the weights given by the TA to energy- and time-efficient optimization, respectively.

Another baseline algorithm used in this work is the dynamic greedy algorithm based on [61]. This algorithm finds the number of computation task $x_i(t)$ to be offloaded to ECN i that satisfies $x_i(t)/f_i < d_i$ where d_i is the deadline associated with $x_i(t)$. The cost function of ECN i for dynamic greedy computing is defined by

$$J_{dg,i} = \mathbb{E}\left[\int_0^T (w_{d,i}\tau_{dg,i}x_i(t) + w_{e,i}e_{dg,i}x_i(t))\,dt\right], \tag{3.33}$$

where $\tau_{dg,i} = 1/f_i$ refers to the number of time to execute a unit of computation task, $e_{dg,i} = \kappa_e f_i^2$ refers to the energy consumption per unit of computation task, and the constants $w_{d,i}$ and $w_{e,i}$ refer to the weights given by ECN i to energy- and time-efficient optimization, respectively.

To bridge the gap between the baseline algorithms with linear cost functions and the proposed MFTG-based algorithms with quadratic cost functions, a quadratic term is added to the linear costs so that

$$J_{lo} = \mathbb{E}\left[\int_0^T (a_1\xi_0 x_0(t) + a_2\xi_0^2 x_0^2(t))\,dt\right], \tag{3.34}$$

and

$$J_{dg,i} = \mathbb{E}\left[\int_0^T (a_1\xi_i x_i(t) + a_2\xi_i^2 x_i^2(t))\,dt\right], \tag{3.35}$$

where $\xi_0 = w_{d,0}\tau_{lo} + w_{e,0}e_{lo}$, $\xi_i = w_{d,i}\tau_{dg,i} + w_{e,i}e_{dg,i}$, a_1 and a_2 as constants with $a_2 << a_1$.

3.5.2 Performance Metrics

The following metrics are calculated in order to compare the performance of the computation offloading approaches presented in this work.

An offloading control fraction $p_i(t)$ is the ratio between the offloading control $u_i(t)$ and the state $x(t)$ of the TA, $p_i(t) = u_i(t)/x(t)$. Consequently, an optimal offloading control fraction $p_i^*(t)$ is written mathematically as

$$p_i^*(t) = \frac{u_i^*(t)}{x(t)}, \tag{3.36}$$

where $u_i^*(t)$ is the optimal offloading control of ECN i.

The two main parameters in the MFTG formulation of computation offloading that limit the control of an ECN are energy consumption and computation or execution time. Consequently, the performance of the computation offloading methods are evaluated through energy efficiency and time efficiency. In this work, efficiency refers to how much computation task are executed per unit of network resource. Hence, energy efficiency is defined as the ratio between the number of computation tasks and the associated energy consumption. For an MECN with N ECNs, the network energy efficiency is written as

$$\eta_e = \frac{x(t)}{\sum_{i=1}^{N} \kappa_{e,i} f_i^2 u_i(t)}, \tag{3.37}$$

where the ratio is taken between the total number of tasks at the TA and the total energy consumed by all the ECNs.

Meanwhile, time efficiency refers to the ratio between the number of computation tasks and the corresponding computation or execution time spent. For an MECN with N ECNs, the network time efficiency is given by

$$\eta_d = \frac{x(t)}{\sum_{i=1}^{N} \frac{u_i(t)}{f_i}}, \tag{3.38}$$

where the ratio is taken between the total number of tasks at the TA and the cumulative computation time of the tasks through the ECNs.

System cost is another way of comparing the computation offloading methods. It consists of the computation offloading cost and the overhead cost associated with each computation offloading algorithm. For both MFTG approaches, overhead exists between an ECN and the TA. Thus, the system costs for the non-cooperative and cooperative MFTG methods are given by

$$C_{nc} = \sum_{i=1}^{N} \big(\tilde{J}_i + 2\delta_i \theta_{i,0}\big), \quad \text{and} \quad C_{co} = \tilde{J}_0 + \sum_{i=1}^{N} 2\delta_i \theta_{i,0}, \tag{3.39}$$

where δ_i is the cost associated per overhead while $\theta_{i,0}$ is the number of overhead between ECN i and the TA. For the local computing algorithm, since the TA does not collaborate with any computing nodes, the overhead is zero. In the dynamic greedy offloading, overhead exists not only between the TA and ECNs but also between any two ECNs. Thus, the system costs for these two baseline approaches are

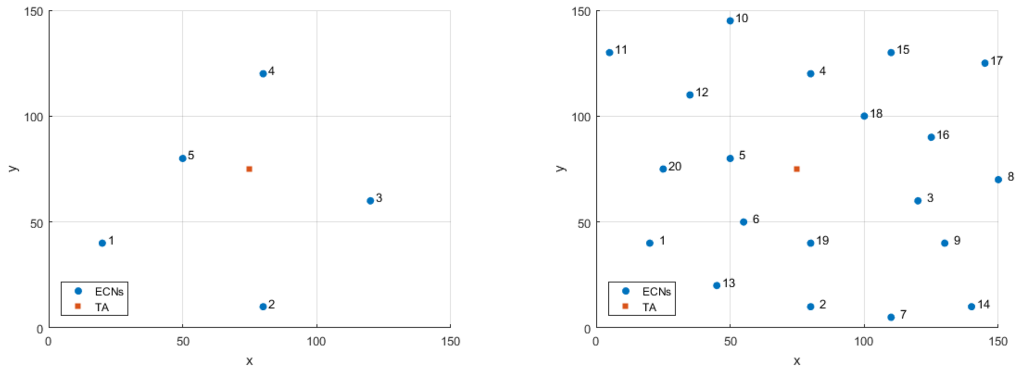
$$C_{lo} = J_{lo}, \quad \text{and} \quad C_{dg} = \sum_{i=1}^{N} \big(J_{dg,i} + N\delta_{dg,i}\theta_{dg,i}\big), \tag{3.40}$$

where $\delta_{dg,i}$ is the cost associated per overhead while $\theta_{dg,i,}$ is the number of overhead from ECN i to another computing node. In this work, overhead refers to the delay associated with the transmission time of overhead messages between any two computing nodes. For an overhead message of length b, the transmission time is b/r, where r is the rate at which the message is transmitted.

Lastly, to be able to compare the computational overhead and benefits of the proposed algorithms, a benefit-cost analysis is performed on the proposed algorithms as well as the typical algorithms in computation offloading. The metric used to compare the algorithms is called the benefit-cost ratio B/C. The benefit B of each algorithm is the weighted sum of the energy and time efficiencies,

$$B = w_d \eta_d + w_e \eta_e, \tag{3.41}$$

where the constants w_d and w_e denote the weights given to the efficiencies and $w_d + w_e = 1$. The cost C used for each algorithm is the system cost defined in (3.39) and (3.40).



(a) A sparse MECN with $N = 5$.

(b) A dense MECN with $N = 20$.

Figure 3.3: The location of the ECNs.

3.6 Simulation Results

3.6.1 Simulation Setup

The simulations in this work can be extended to networks containing multiple cells assuming that each cell operate independently of each other. That is, the TA of a cell can offload task only to ECNs located in its cell. Moreover, the interference between cells are minimized using techniques such as FDMA and SDMA. Each simulation has been performed over 100 iterations and the average of the results has been drawn in each figure.

Consider one network cell with an area of 150 $\times$ 150 m^2 containing one TA located at the center of the cell. The number of ECNs has been varied from 2 to 20. The location of each ECN is randomly distributed within the cell. Fig. 3.3 shows the locations of the ECNs for the sparse MECN with $N = 5$ and the dense MECN with $N = 20$ utilized in the following simulations. The end users are located randomly within the cell. The number of end users is set at 50. The computation tasks arrive at the TA randomly, and the users are assumed to submit an average of 5 Tcycles of computation tasks.

Assume that the TA has a transmit power of 100 mW, a maximum incoming rate of computation task $R_0 = 10$ Gbps, and a maximum capacity C_0 of 10 Tb worth of computation

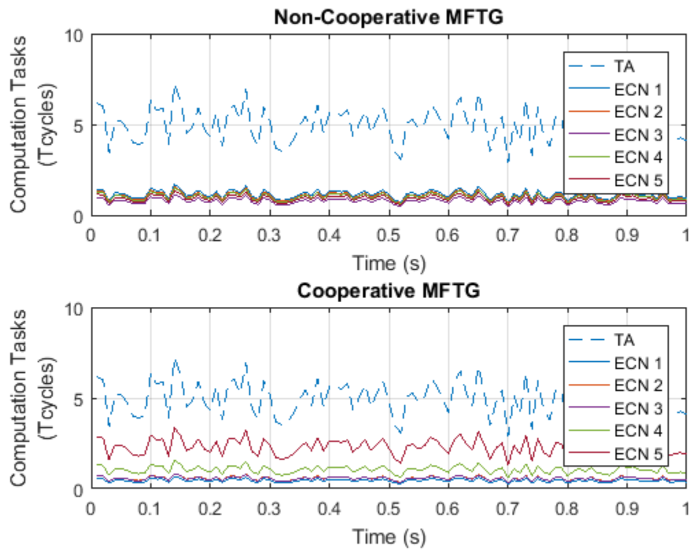
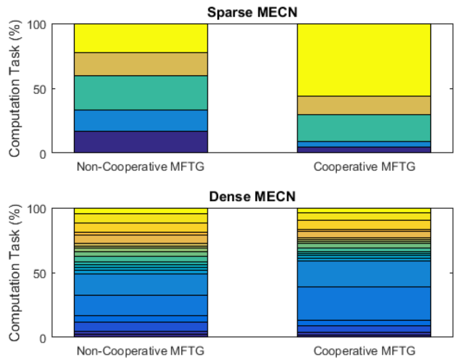


Figure 3.4: The optimal offloading control of the ECNs.

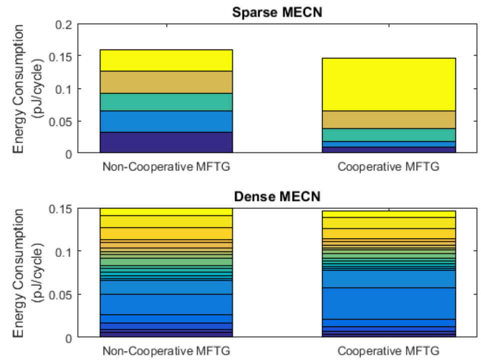
task. Meanwhile, the computing nodes have a transmit power p_i of 100 mW, capacity C_i of 100 Gb worth of computation task. The computing capability f_i of each computing node is randomly selected from 10, 12, and 14 Tcycles/s. The cost weights $w_{d,i}$ and $w_{e,i}$ for the computation time and energy consumption are both set to 0.5. For SINR γ_i computations, the channel gain model used between any two nodes i and j is $g_{i,j} = d_{i,j}^{-\alpha}$ where $d_{i,j}$ denotes the distance between the two nodes and the path loss exponent $\alpha = 4$. Meanwhile, the background noise N_0 is set at -100 dBm. The quadratic cost constants for the baseline algorithms are set at $a_1 = 0.9$ and $a_2 = 0.1$.

3.6.2 Optimal Offloading Control

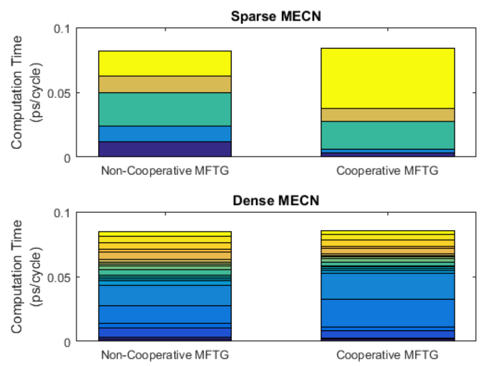
In the first of part the simulations, the optimal offloading control $u_i^*(t)$ of ECN i based on the feedback controls $u_{-i}^*(t)$ of other ECNs is computed using the MFTG computation offloading algorithms. Fig. 3.4 shows the plots of $u_i^*(t)$ for the sparse MECN in both non-cooperative and cooperative MFTG scenarios as well as the number of computation task $x(t)$ at the TA. It can be noted from this figure that the two MFTG algorithms divide the



(a) Average percentage of computation task.



(b) Average energy consumption of ECNs.

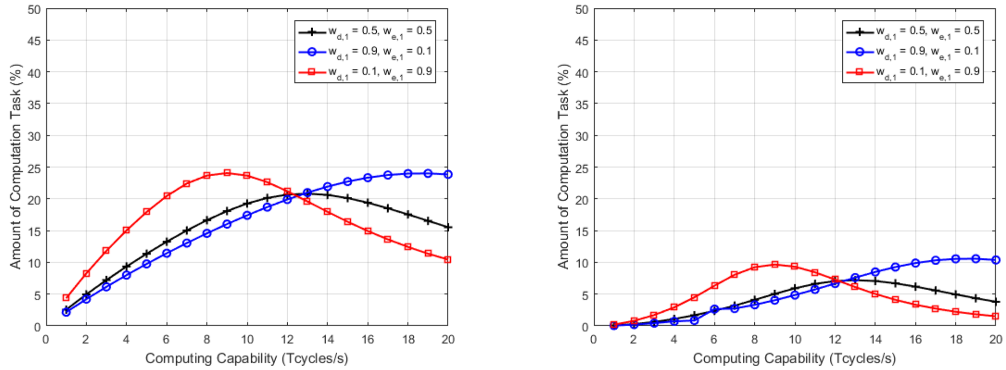


(c) Average computation time by ECNs.

Figure 3.5: The partition of aggregate computation task for each ECN.

computation tasks at the TA to the ECNs differently.

The partition of computation tasks among the ECNs is shown in Fig. 3.5 where each color denotes the particular share of an ECN. Fig. 3.5a shows the average percentage of offloaded computation task from the TA to each ECN i in the sparse and dense MECN. In the sparse MECN, the non-cooperative MFTG approach distributed the tasks more evenly than the cooperative approach. Meanwhile, in the dense MECN, the distribution of tasks is almost similar between the two MFTG offloading algorithms. The figure also implies that the offloading controls change accordingly when the number of ECNs is varied. Meanwhile, Fig. 3.5b presents the energy consumption per cycle of each ECN and Fig. 3.5c shows the computation time per cycle contributed by each ECN.

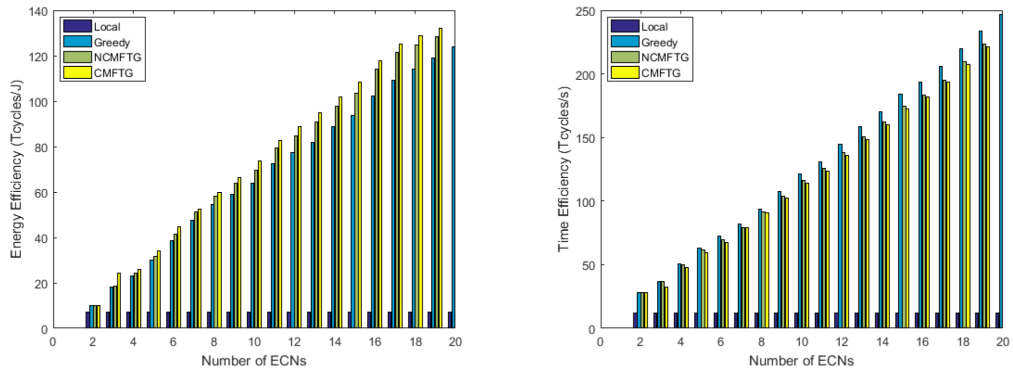


(a) Non-cooperative MFTG. (b) Cooperative MFTG.

Figure 3.6: The effect of computing capability and cost weights to the optimal control.

Next, the effects of computing capability f_i and the cost weights $w_{d,i}$ and $w_{e,i}$ of ECN i to its optimal control $u_i^*(t)$ are investigated. Even though Fig. 3.6 shows $u_i^*(t)$ as a fraction of $x(t)$ averaged over time for ECN 1, the analyses that follow can be generalized to any ECNs. From the figure, it can be noticed that as the computing capability f_i of an ECN increases, the average percentage of aggregate computation task it offloads rises up to a certain point and then decreases. The reason for this trend is the compromise between minimizing computation time and energy consumption. When f_i is low, the energy consumption of ECN i is also low; however, the computation time to execute the offloaded task is high. As f_i becomes higher, the energy consumption of an ECN increases while the computation time to execute the offloaded task becomes lesser.

Meanwhile, as computation time is given more weight by increasing its weight from 0.1 to 0.9, the curve shifts to the right. This means that as an ECN prioritizes minimizing computation time, the computing capability at which it can afford to offload the highest percentage of the aggregate computation task increases. However, as more weight is given to energy consumption from 0.1 to 0.9, the curve shifts to the left. That is, to lower the energy consumption of an ECN, the computing capability at which its offloading percentage is at the highest decreases.



(a) Average network energy efficiency.

(b) Average network time efficiency.

Figure 3.7: The average network efficiency of computation offloading approaches.

In summary, an ECN with lower computing capability offloads more from the TA if minimizing the energy consumption is more critical, as shown by the red curves. However, if the priority is to minimize computation time, then an ECN with higher computing capability offloads more from the TA, as shown by the blue curves.

3.6.3 Network Efficiency

The energy efficiency η_e using different computation offloading approaches are compared in Fig. 3.7a. From the figure, the cooperative MFTG approach (CMFTG) has better η_e than the non-cooperative MFTG (NCMFTG) approach. However, both MFTG algorithms have higher η_e than the local and dynamic greedy algorithms. This is one of the reasons that justifies the significance of computation offloading in MECN.

Meanwhile, the time efficiency η_d of the network under different computation offloading approaches is displayed in Fig. 3.7b. It can be concluded from the figure that MFTG computation offloading approaches maintain a competitive η_d against the dynamic greedy algorithm.

Hence, the MFTG offloading algorithms can be as efficient as the dynamic greedy algorithm which requires full knowledge of the characteristics of all the ECNs. In the following

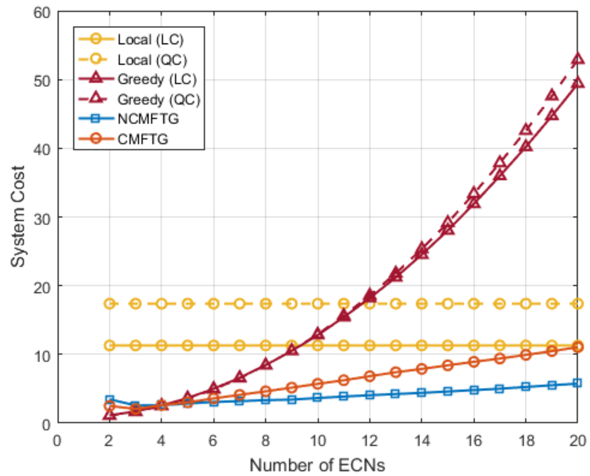


Figure 3.8: The average network cost caused by the computation offloading methods.

subsection, the system costs of the computation offloading algorithms are compared.

3.6.4 System Cost and Benefit-Cost Ratio

Fig. 3.8 presents the system cost sustained by each computation algorithm investigated in this work. The system cost of the dynamic greedy approach is higher than both the MFTG approaches because the overhead required to implement the greedy algorithm is larger than the overhead required by the MFTG approaches. The system cost of the local computing approach is shown for comparison purposes even though it does not require the use of ECNs. Between the two MFTG approaches, the cooperative approach has lower system cost than the non-cooperative approach when the number of ECNs is lower. However, as the number of ECNs increases, the system cost of the non-cooperative approach becomes lower than that of the cooperative approach.

Fig. 3.9 shows the benefit-cost ratio B/C for each computation offloading approaches. It is evident that the non-cooperative MFTG approach has the best B/C, followed by the cooperative MFTG approach. The benefit of the MFTG approaches are contributed by the energy- and time-efficient partition of computation tasks as well as the low number of

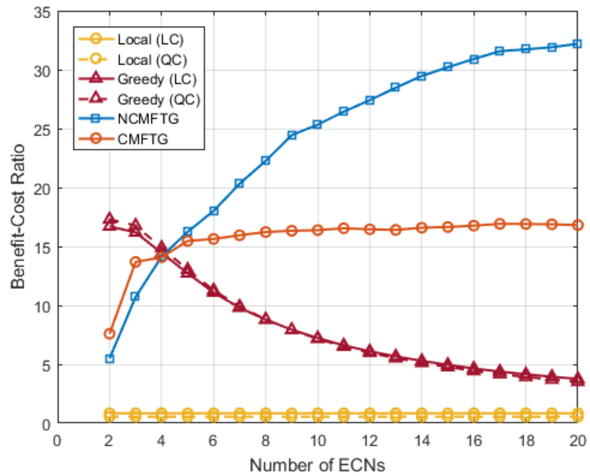


Figure 3.9: The benefit-cost ratio of the computation offloading approaches.

network overhead required to implement the offloading.

Moreover, the system cost and benefit-cost ratio of the local computing and dynamic greedy algorithms with quadratic cost (QC) are almost equivalent to the system cost and benefit-cost ratio of the original linear cost (LC). Therefore, the form of the cost function does not affect the performance of the algorithms significantly since the main difference between the cost functions of the proposed work and the baseline algorithms is the overhead cost.

3.7 Related Works

Offloading of computation-intensive tasks from mobile devices to MECNs has garnered a lot of interests in the research community. In this section, these computation offloading methods are briefly described.

Various game theoretic methods have been applied to model computation offloading among many computing units. In [62], the authors utilized a game theoretic approach to computation offloading problem among mobile users in a multi-user, multi-channel wireless mobile edge computing network. Meanwhile, in order to utilize the computation resources in the cloud, collaborative computation offloading between the centralized cloud server and the MEC servers was studied in [63].

Many research have jointly optimized computation offloading with other network technologies and issues. The authors of [64] formulated computation offloading among mobile devices as a joint optimization of the radio and computation resources that minimizes a user's energy consumption while satisfying latency requirements. An energy-efficient dynamic offloading and resource scheduling formulated as a minimization problem was investigated in [65]. In [66], interference management was integrated in computation offloading and formulated together as an optimization problem. To reduce execution delay, computation offloading was integrated with cache placement in MEC to store and share popular computation results to mobile users [67].

Several works have focused on integrating computation offloading feature in networks involving wireless power. Computation offloading in mobile cloud computing powered by wireless energy transfer was studied in [68]. The authors proposed the use of CPU-cycle statistics information and channel state information to enforce policies that maximize the probability of successful computation of data subject to the energy harvesting and latency constraints. In [69], the authors combined the concepts of MEC with wireless power transfer so that the MEC access point can transmit wireless power to mobile users which can be used for local computing. Then, the authors of [61] proposed a Lyapunov optimization-based dynamic algorithm for MEC with energy-harvesting devices that jointly decides on the offloading, CPU frequency, and transmit power.

Energy-efficient computation offloading algorithms have been the focus of several works as well. An energy-efficient computation offloading scheme was proposed in [70] where the energy consumption of the offloading system was minimized while still satisfying the latency requirements of the tasks. Meanwhile, energy-efficient task offloading in software defined ultra-dense network was investigated in [71].

Partial offloading in which only a part of an application is offloaded to computing entities has been studied by several works. The authors of [72] considered partial offloading due to limited bandwidth in wireless networks. Also, the authors of [73] proposed a cooperative partial computation offloading between cloud computing and MEC-enabled IoT.

Before concluding this section, some works involving MFTGs are worth mentioning. In [74], energy storage problem in a microgrid was formulated as an MFTG. The mean and variance of the energy level were added to the cost function and used MFTG to keep track and maintain the desired energy level in the microgrid. Meanwhile, MFTG was utilized as a particle filter for video-based vehicular tracking in Intelligent Traffic Systems (ITS) [75]. A mean field term was included in the formulation to provide accurate and robust state (i.e., vehicle position) prediction. In [76], MFTG was applied in blockchain token economics. This work introduced variance in the utility function to capture the risk of cryptographic tokens associated with the uncertainties of technology adoption, network security, regulatory legislation, and market volatility.

The main difference of this work is that computation offloading in MECN has been formulated as an MFTG in which each computing node has a desired level of computation tasks it can handle. This level is dictated by the energy consumption and computing capability of the computing node. Moreover, this work utilizes a direct approach that does not require solving coupled partial differential equations to calculate the optimal computation offloading strategy of each computing node. Lastly, non-cooperative and cooperative scenarios among the computing nodes are both considered and investigated.

3.8 Conclusion

Multi-access edge computing networks (MECN) reduce the latency inherent in cloud computing networks by performing the tasks in an edge network near the network users rather than in a cloud network. Computation offloading is one of the services in an MECN in which computation-intensive tasks in a computing node may be offloaded to other computing nodes in the network. In this work, computation offloading problem has been formulated using mean-field-type game (MFTG). Then, non-cooperative and cooperative computation offloading algorithms have been proposed. These algorithms search for the optimal computation offloading controls of each computing node in an MECN. The non-cooperative algorithm is a decentralized approach since each computing node determine its own offloading control. Nevertheless, the cooperative algorithm is a centralized approach in which the network determines the offloading control of each computing node. Lastly, the simulation results have indicated that MFTG is an effective way to model computation offloading in MECNs.

Chapter 4

Dynamic Hierarchical Game Model for Resource Allocation and Economics in Network Virtualization

Network virtualization (NV) allows multiple heterogeneous virtual networks (VNs) to coexist and operate over the same physical network (PN) infrastructures. Some of the benefits of this advancement include flexibility in VN topologies, heterogeneity in VN technologies, and modularity of network operations. However, there are a few areas such as resource allocation and economics that challenge the implementation of NV. In this work, the NV parameters that influence the resource allocation and economics of an NV system are introduced. Next, an economic model for NV is formulated using the prey-predator food chain model. This model takes into account the dynamics in an NV system such as the service, payoff, failure, and competition rates involving the VNs and the PNs. The solution point to this model represents the resource strategy of the service provider (SP) given the number of users trying to use its VN and the resource strategy of the infrastructure provider (InP) given the strategy of the VN leasing its PN. In addition, economic models that relate the capacities of the end users (EUs), the SPs, and the InPs are established. Finally, simulations are provided to illustrate effectiveness of the prey-predator food chain model in analyzing and modeling an NV system.

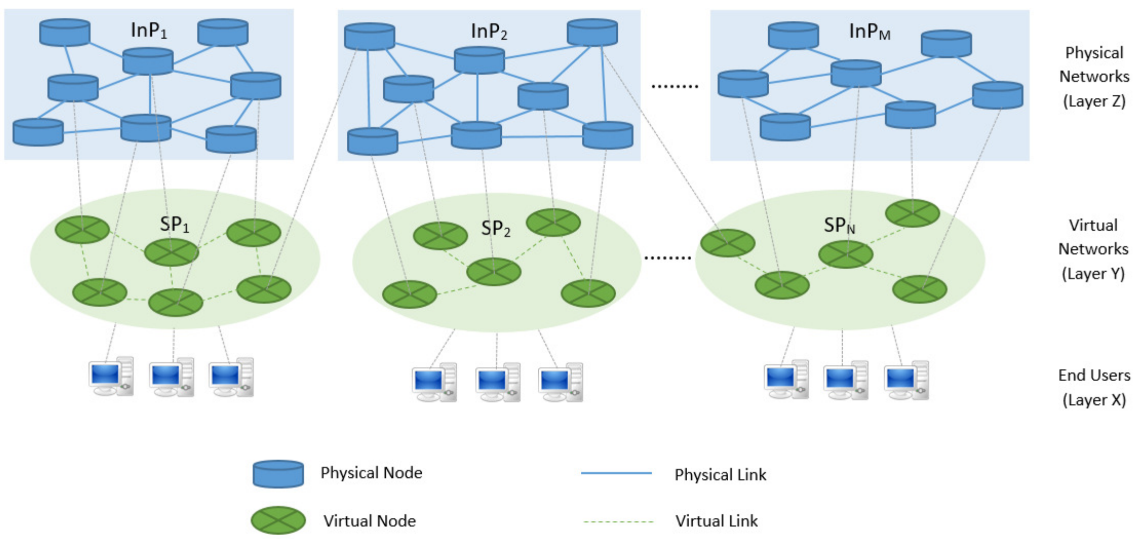


Figure 4.1: A network virtualization architecture.

4.1 Network Virtualization

4.1.1 System Architecture

The NV architecture is shown in Fig. 4.1. It has three layers of abstraction: the InPs, the SPs, and the EUs levels. An infrastructure provider (InP) provides and manages a physical network (PN) infrastructure consisting of physical resources (PRs) such as physical nodes and physical links. A service provider (SP) operates and manages a virtual network (VN) which is composed of virtual resources (VRs) such as virtual nodes and virtual links. In order to build and expand its VN, an SP needs to lease PRs from InPs. Through NV, one or more SPs can lease the same PRs at the same time. In other words, multiple VRs may reside at a single PR. Also, an SP may only lease a subset of the PN according to its VN needs. The end users (EUs) can communicate with the network or to each other, as well as use web applications, through the services provided by an SP.

4.1.2 System Parameters

The focus of the system model presented in the next section is the distribution of bandwidth in an NV system. Thus, the primary network resource of interest in this work is bandwidth, and the terms "network resource" and "bandwidth" are used interchangeably, unless otherwise stated. Other network resources such as CPU and memory are assumed to have satisfied their respective constraints, and therefore are not affecting the distribution of bandwidth in the system. As a consequence of these assumptions, the following system parameters are related to the amount of bandwidth demand by or supply from a particular entity (i.e., EUs, SPs, and InPs) in an NV system.

Next, the important frequencies or rates $r^{(l)}$ at each level l of the NV system are introduced. The main players at each level $l \in \{1, 2, 3\}$, respectively, are

- the EUs in the metropolitan area (MA),
- the SPs, each managing a VN, and
- the InPs, each operating a PN.

Consider a metropolitan area where an SP is overlaying its VN and an InP has already established its PN. Suppose the EUs who want to connect to and use the VN submit their requests, or user requests (URs), to the SP. For simplicity, assume that a single EU can only submit one UR at a time. If χ is the number of EUs trying to access the VN at time τ, then it is also the number of URs generated simultaneously in the area. Since bandwidth is the network resource of interest, χ also refers to the number of unit EU bandwidth requests generated in the area. For example, if $\chi = 10$ and a unit EU bandwidth is 10 Mbps, then there are 10 users requesting for 10 Mbps of bandwidth each. The quantity χ can also be referred to as the resource (i.e., bandwidth) demand at the EU level. The *arrival rate* $r_0^{(1)}$ of these URs is the theoretical maximum increase rate of the number of URs χ generated in the network coverage area per EU,

$$r_0^{(1)} = \frac{1}{\chi}\frac{d\chi}{d\tau}, \tag{4.1}$$

where τ denotes time. Assuming the number of URs approaches a maximum limit $\kappa^{(1)}$, the logistic model equation [77] can be used, then (4.1) becomes

$$r^{(1)} = \frac{\frac{1}{\chi}\frac{d\chi}{d\tau}}{1 - \frac{\chi}{\kappa^{(1)}}}, \tag{4.2}$$

where $\kappa^{(1)}$ is the maximum number of EUs in the area. The term $\frac{\chi}{\kappa_1}$ signifies the percentage of EUs trying to connect to and use the VN. The solution to (4.1) is called the logistic function,

$$\chi(\tau) = \frac{\kappa^{(1)}}{1 + (\frac{\kappa^{(1)}}{\chi_0} - 1)e^{-r^{(1)}\tau}}, \tag{4.3}$$

where χ_0 is the initial value of $\chi(\tau)$ at $\tau = \tau_0$, $\kappa^{(1)} > 0$ is the limiting value of $\chi(\tau)$, and $r^{(1)} > 0$ for a logistic growth model.

In order to accommodate the additional VN users and lease a PR, the SP has to submit a VR request (VRR) for a PR to the InP. If ψ is the number of VRRs generated simultaneously in the area at time τ, then ψ also refers to the number of unit VR bandwidth requests generated in the area. For example, if $\psi = 5$ and a unit VR bandwidth is 100 Mbps, then there are 5 requests for a VR of 100 Mbps each. Moreover, ψ can be regarded as the resource demand at the SP level. Also, let $\kappa^{(2)}$ as the total number of VRRs that can be generated in the area. Assume that the SP can submit only one VRR per VR at a time. Therefore, $\kappa^{(2)}$ also refers to the maximum number of VRs an SP can manage. Not all of these VRRs can successfully turn into a VR. Hence, define the *failure rate* $r^{(2)}$ of VRRs as the theoretical maximum VRR failures per unit time per VR. Mathematically, it is written as

$$r^{(2)} = -\frac{1}{\psi}\frac{d\psi}{d\tau}. \tag{4.4}$$

For the InP, if ω is the number of active or operational PRs at time τ, then ω also refers to the number of unit PR bandwidth requests. For instance, if $\omega = 2$ and a unit PR bandwidth is 1 Gbps, then there are 2 operational PR each with capacity of 1 Gbps. Also, let $\kappa^{(3)}$ be the total number of PRs owned by the InP. Some of these active PRs may be suddenly unavailable for lease. Thus, the *unavailability rate* $r^{(3)}$ of PRs is the number of PRs z unavailable per unit time per PR. It can be shown mathematically as

$$r^{(3)} = -\frac{1}{\omega}\frac{d\omega}{d\tau}. \tag{4.5}$$

Each UR, VRR, and PR corresponds to a request for a unit resource quantity at its respective level. That is, χ, ψ, and ω represent an amount of network resource (i.e., bandwidth). Thus, χ and ψ refer to the resource demand at the EUs and SP, respectively. On the other hand, ω represents the resource supply at the InP.

There are other important parameters in NV. The network *service availability* $s^{(l)}$ is the network user acceptance or service rate per VR (or PR). It measures how many users at level $l \in \{1, 2\}$ are being served by the network. The network *payoff rate* $p^{(l)}$ is regarded as the network-user-to-network-income conversion rate per network user. It quantifies the gain of network at level $l \in \{2, 3\}$ from rendering services. The parameters $s^{(l)}$ and $p^{(l)}$ are related. For the SP,

$$p^{(2)} = \frac{s^{(1)}}{n}, \tag{4.6}$$

where n is the maximum allowable number of EUs per VR. For the InP,

$$p^{(3)} = \frac{s^{(2)}}{m}, \tag{4.7}$$

where m is the maximum allowable number of VRs that can coexist per PR. Lastly, the network *congestion (or competition) rate* q is defined as the VR (or PR) competition rate per VR (or PR). Accordingly, it is a measure of how much the network traffic degrades the network resources.

If the service rate s is divided by the unit cost of service c_s, the result is the per unit cost service rate S. Likewise, dividing the payoff rate p by the unit income quantity c_p yields the per unit income payoff rate P. Also, getting the ratio of the congestion rate q to the unit penalty from congestion c_q gives the per unit penalty congestion rate Q. Mathematically, these quantities are expressed as

$$S = \frac{s}{c_s}, \quad P = \frac{p}{c_p}, \quad \text{and} \quad Q = \frac{q}{c_q}. \tag{4.8}$$

Table 4.1 summarizes the parameters discussed in this section.

4.2 System Model and Problem Formulation

The main goal of this work is to introduce a unified framework for the resource allocation and economics problem in NV. In an NV system that consists of many users, SPs, and InPs, two problems exist. One of the problems is how much resources an SP must own and allocate to the EUs given the dynamics that exist in the NV system such as the amount of connection requests it receives from the EUs, the intensity of competition it experiences from other SPs, and the amount of resources it can request to and can be granted by an InP. Similarly, an InP faces the same problem of how much resources it must operate and allocate to the SPs that are trying to lease resources from it depending on factors such as the number of requests it receives from the SPs, and the intensity of competition it experiences from other InPs. In this work, these two NV system problems are modeled together in order to provide a framework that covers the three levels of the NV system. The resulting system model, based on the prey-predator food chain model, takes as input the NV system parameters that characterize the dynamics in the system. Then, the resource strategies of the SPs and InPs are computed where a resource strategy refers to the bandwidth capacity that an SP or InP should own in order to support its users.

Three mathematical models, one for each configuration of a three-level NV system, are introduced and developed in this section. The first model is for a simple NV system that has one SP and one InP. The second model is for an NV system with two SPs and one InP. Then, a generalized model for NV systems with N-SPs and M-InPs is introduced. Each model has finite number of EUs. The EUs that are granted with network connection communicate with their respective SP through to its VN. Hence, an SP forms a star topology with its EUs. Since an SP requests only for virtual nodes that are needed to support the EUs, its VN capacity is assumed to be equal to the sum of the capacities of the virtual nodes. Meanwhile, the SPs that are granted with lease of physical resources communicate with their respective InP through its PN. Thus, an InP forms a star topology with the SPs that are leasing its PN. Since an InP operated and manages only physical nodes that are needed to support its client SPs, its PN capacity is assumed to be equal to the sum of the physical node capacities. In each model, the utilities of the SPs and InPs are first modeled as the net amount of cash per unit time they receive from their respective consumers. Since this work focuses on the distribution of network resources (i.e., bandwidth) within the NV system, the cash flows are converted to the amount of bandwidth the consumers are demanding from their respective producers. These models can be implemented by a trusted central authority that supervises and regulates the NV system in the coverage area.

The purpose of each model is to calculate the equilibrium point of the NV system which signifies the resource strategy of each entity in the NV system and to simulate the economics of the NV system. The results of each simulation demonstrate how the resource strategy at each level and the relationship between demand and supply change with the network parameters. The concept and analysis of equilibrium is discussed in the next section.

This section is organized as follows. First, models for a one-SP, one-InP NV system and a two-SP, one-InP NV system are formulated. Then, the extension of this formulation to an N-SP, M-InP NV system is developed in the following subsection. Finally, the key results are summarized.

4.2.1 One-SP, One-InP Network Virtualization System

Consider an NV system in a metropolitan area with multiple EUs, one SP, and one InP. Let χ be the number of EUs trying to access the VN of the SP by submitting URs to the SP. Assuming an EU submits only one UR at a time, the total number of URs submitted to the SP is also χ. Moreover, χ is also related to the demand at the EU level. Meanwhile, suppose ψ is the number of VRRs the SP submits to the InP. Hence, ψ also means the total number of VRRs needed by the SP to realize its VN and support the demand of the EUs. Lastly, let ω be the number of active PRs in the InP.

To formulate the mathematical model for the NV system, the flow of payment from the EUs to the SP, and from the SP to the InP is followed. Let $c^{(l)}$ be the unit cost of resources at each level $l \in \{1, 2, 3\}$ of the NV system. Assign constants $c^{(1)}$ as the cost per EU bandwidth, $c^{(2)}$ as the cost per VR bandwidth, and $c^{(3)}$ as the net cost incurred by the InP per PR. Each EU with a successful VN connection pays the SP; hence, the SP earns an amount proportional to the number of successful URs. Meanwhile, the SP has to pay the InP an amount proportional to the number of VRs in its VN. The InP incurs a cost proportional to the number of active PRs in its PN. Therefore, the EUs, the SP, and the InP interact through these transactions. Based on the transactions just described, the rate of change of the net income at each level can be formulated.

Suppose $X = c^{(1)}\chi$ is the potential "revenue" at the EU level coming from all the URs, $Y = c^{(2)}\psi$ is the potential revenue of the SP coming from all the VRRs, and $Z = c^{(3)}\omega$ is the potential revenue of the InP for leasing and operating its PN. Also, consider the quantities

$$K^{(1)} = c^{(1)}\kappa^{(1)}, \quad S^{(l)} = \left.\frac{s^{(l)}}{c^{(l+1)}}\right|_{l\in\{1,2\}}, \quad P^{(l)} = \left.\frac{p^{(l)}}{c^{(l-1)}}\right|_{l\in\{2,3\}},$$

and

$$Q^{(l)} = \left.\frac{q^{(l)}}{c^{(l)}}\right|_{l\in\{2,3\}}, \tag{4.9}$$

where s, p, q, S, P, and Q are the network parameters defined in the previous section, with indices $l \in \{1, 2, 3\}$ indicating the EUs, the SP, and the InP levels, respectively. Referring to interaction between the EUs and the SP described in the previous paragraph, the revenue rate at EU level $\frac{1}{X}\frac{dX}{d\tau}$ is reduced by the service cost $S^{(1)}Y$ charged by the SP. Thus, (4.2) becomes an equation for net income,

$$\frac{1}{X}\frac{dX}{d\tau} = r^{(1)}\left(1 - \frac{X}{K^{(1)}}\right) - S^{(1)}Y. \tag{4.10}$$

Similarly, the net income rate $\frac{1}{Y}\frac{dY}{d\tau}$ of the SP is the sum of the failed VRR rate $-r^{(2)}$, the payoff $+P^{(2)}X$ from the EUs, the cost of congestion $-Q^{(2)}Y$ among the VRRs in the VN, and the cost of service availability $-S^{(2)}Z$ of the InP. Therefore, the net income rate equation (4.4) for the SP becomes

$$\frac{1}{Y}\frac{dY}{d\tau} = -r^{(2)} + P^{(2)}X - Q^{(2)}Y - S^{(2)}Z. \tag{4.11}$$

Lastly, the net income rate $\frac{1}{Z}\frac{dZ}{d\tau}$ of the InP becomes the sum of the unavailable PR rate $-r^{(3)}$, the payoff $+P^{(3)}Y$ from the SP, and the cost of congestion $-Q^{(3)}Z$ among active PRs. Consequently, the net income equation in (4.5) for the InP becomes

$$\frac{1}{Z}\frac{dZ}{d\tau} = -r^{(3)} + P^{(3)}Y - Q^{(3)}Z. \tag{4.12}$$

Dividing both sides of (4.10), (4.11), and (4.12) by their respective unit costs $c^{(1)}$, $c^{(2)}$, and $c^{(3)}$, and applying the substitutions in (4.9) yield the following system of nonlinear differential equations

$$\begin{aligned}
\frac{d\chi}{d\tau} &= \chi\left(r^{(1)}\left(1 - \frac{\chi}{\kappa^{(1)}}\right) - s^{(1)}\psi\right), \\
\frac{d\psi}{d\tau} &= \psi(-r^{(2)} + p^{(2)}\chi - q^{(2)}\psi - s^{(2)}\omega), \\
\text{and} \qquad \frac{d\omega}{d\tau} &= \omega(-r^{(3)} + p^{(3)}\psi - q^{(3)}\omega),
\end{aligned} \tag{4.13}$$

where $s^{(1)}$ and $s^{(2)}$ are the SP and InP service availability rates, $p^{(2)} = \frac{s^{(1)}}{n}$ and $p^{(3)} = \frac{s^{(2)}}{m}$ are the SP and InP payoff rates, and $q^{(2)}$ and $q^{(3)}$ are the SP and InP congestion rates, respectively.

To make the analysis easier, reduce the number of parameters of system in (4.13) by introducing the following substitutions: $x = \frac{\chi}{\kappa^{(1)}}$, $y = \frac{\psi}{\kappa^{(2)}}$, $z = \frac{\omega}{\kappa^{(3)}}$, and $t = r^{(1)}\tau$. Thus, x, y, z and t represent the following quantities:

- x is the EU capacity demand;
- y is the VN capacity supply/demand;
- z is the PN capacity supply; and
- t is the dimensionless time.

The substitutions yield the following system of dimensionless equations representing the one-SP, one-InP NV system model

$$\frac{dx}{dt} = x(1 - x - w_1 y) = xf(x, y, z),$$

$$\frac{dy}{dt} = y(-w_2 + w_3 x - w_4 y - w_5 z) = yg(x, y, z), \tag{4.14}$$

and

$$\frac{dz}{dt} = z(-w_6 + w_7 y - w_8 z) = zh(x, y, z).$$

For convenience, it is assumed that the SPs and InPs approve and grant only the connection requests they receive, and operate and manage only the nodes that are active. Thus, their VNs and PNs capacities, y and z, respectively, are the same as the total virtual/physical node capacities. That is, a network capacity is equal the sum of its node capacities.

The parameters of (4.14) are listed in Table 4.2. Since it has been established in Section 4.1 that $p^{(2)} = \frac{s^{(1)}}{n}$ for the SP and that $p^{(3)} = \frac{s^{(2)}}{m}$ for the InP, then parameters w_3 and w_7 can also be expressed as

$$w_3 = \frac{w_1}{\eta^{(2)}} \quad \text{and} \quad w_7 = \frac{w_5}{\eta^{(3)}}, \tag{4.15}$$

where

$$\eta^{(2)} = \frac{\kappa^{(2)}}{\kappa^{(1)}/n} \quad \text{and} \quad \eta^{(3)} = \frac{\kappa^{(3)}}{\kappa^{(2)}/m}, \tag{4.16}$$

are the network coverage ratios of the SP and the InP, respectively. These two quantities represent the ratio of the maximum network resource supply to the maximum network resource demand.

The association between the demand x and supply y can be derived using the chain rule, $\frac{dv}{du} = \frac{dv}{dt} \cdot \frac{dt}{du}$. Hence, it follows that

$$\frac{dy}{dx} = \frac{y(-w_2 + w_3 x - w_4 y - w_5 z)}{x(1 - x - w_1 y)}. \tag{4.17}$$

Likewise, the relationship between demand y and supply z can be written as

$$\frac{dz}{dy} = \frac{z(-w_6 + w_7 y - w_8 z)}{y(-w_2 + w_3 x - w_4 y - w_5 z)}. \tag{4.18}$$

For the one-SP, one-InP system in (4.14), the resource strategy is the positive equilibrium point denoted by $E(x^*, y^*, z^*)$. Meanwhile, the demand-supply relationships can be found by solving for $y = f(x)$ and $z = g(y)$. The function $f(x)$ indicates the behavior of the VN capacity supply with respect to the EU capacity demand x, whereas the function $g(y)$ specifies the reaction of the PN capacity supply in terms of the VN capacity demand. Since (4.17) and (4.18) are both first-order nonlinear differential equations (NDE), a numerical method is used to graph $f(x)$ and $g(y)$.

4.2.2 Two-SP, One-InP Network Virtualization System

Based on the formulation in the previous subsection, the system model for an NV system serviced by two SPs ($N = 2$) and one InP ($M = 1$) is expressed by the system of equations

$$\begin{aligned}
\frac{d\chi}{d\tau} &= \chi\left(r^{(1)}\left(1 - \frac{\chi}{\kappa^{(1)}}\right) - s_1^{(1)}\psi_1 - s_2^{(1)}\psi_2\right),\\
\frac{d\psi_1}{d\tau} &= \psi_1(-r_1^{(2)} + p_1^{(2)}\chi - q_{11}^{(2)}\psi_1 - q_{21}^{(2)}\psi_2 - s^{(2)}\omega),\\
\frac{d\psi_2}{d\tau} &= \psi_2(-r_2^{(2)} + p_2^{(2)}\chi - q_{12}^{(2)}\psi_1 - q_{22}^{(2)}\psi_2 - s^{(2)}\omega),
\end{aligned} \tag{4.19}$$

and
$$\frac{d\omega}{d\tau} = \omega(-r^{(3)} + p^{(3)}(\psi_1 + \psi_2) - q^{(3)}\omega),$$

where $s_j^{(1)}$, $j \in \{1,2\}$ is the SP$_j$-to-EU service availability rate, $s^{(2)}$ is the InP-to-SP service availability rate, $p_j^{(2)} = \frac{s_j^{(1)}}{m_j}$ is the EU-to-SP$_j$ payoff rate, $p^{(3)} = \frac{s^{(2)}}{n}$ is the SP-to-InP payoff rate, $q_{jj}^{(2)}$ is the SP$_j$ congestion rate, $q_{Jj}^{(2)}$, $j \neq J$ is the SP$_J$-to-SP$_j$ competition rate, and $q^{(3)}$ is the InP congestion rate.

For analysis convenience, the dimensionless form of (4.19) for a two-SP, one-InP NV system is given by

$$\frac{dx}{dt} = x(1 - x - w_1 y_1 - w_2 y_2) = xf(x, y_1, y_2, z),$$

$$\frac{dy_1}{dt} = y_1(-w_3 + w_4 x - w_5 y_1 - w_6 y_2 - w_7 z) = y_1 g_1(x, y_1, y_2, z),$$

$$\frac{dy_2}{dt} = y_2(-w_8 + w_9 x - w_{10} y_1 - w_{11} y_2 - w_7 z) = y_2 g_2(x, y_1, y_2, z), \tag{4.20}$$

and
$$\frac{dz}{dt} = z(-w_{12} + w_{13} y_1 + w_{14} y_2 - w_{15} z) = zh(x, y, z),$$

where

- x is the EU capacity demand;
- y_j, $j \in \{1,2\}$ is the VN$_j$ capacity supply/demand;
- z is the PN capacity supply; and
- t is the dimensionless time.

In this work, it is assumed that the SPs and InPs accept and allow only the connection requests they receive, and operate and manage only the nodes that are active. Thus, their VNs and PNs capacities, y_j and z, respectively, are the same as the total virtual/physical node capacities. That is, a network capacity is equal the sum of its node capacities.

The parameters of (4.20) are listed in Table 4.3. Since it has been established in Section 4.1 that $p_j^{(2)} = \frac{s_j^{(1)}}{m_j}$ for SP_j and that $p^{(3)} = \frac{s^{(2)}}{n}$ for the sole InP, then

$$w_4 = \frac{w_1}{\eta_1^{(2)}}, \quad w_9 = \frac{w_2}{\eta_2^{(2)}}, \quad w_{13} = \frac{w_7}{\eta_1^{(3)}}, \quad \text{and} \quad w_{14} = \frac{w_7}{\eta_2^{(3)}}, \tag{4.21}$$

where the quantities

$$\eta_j^{(2)} = \frac{\kappa_j^{(2)}}{\kappa^{(1)}/n_j} \quad \text{and} \quad \eta_j^{(3)} = \frac{\kappa^{(3)}}{\kappa_j^{(2)}/m}, \tag{4.22}$$

are the SP and InP network coverage ratios, respectively.

The relationship between the demand x and its supplies y_1 and y_2, as well as the demands y_1 and y_2 and their supply z, are implicitly defined by the equations

$$\frac{dy_1}{dx} = \frac{y_1(-w_3 + w_4 x - w_5 y_1 - w_6 y_2 - w_7 z)}{x(1 - x - w_1 y_1 - w_2 y_2)}, \tag{4.23}$$

$$\frac{dy_2}{dx} = \frac{y_2(-w_8 + w_9 x - w_{10} y_1 - w_{11} y_2 - w_7 z)}{x(1 - x - w_1 y_1 - w_2 y_2)}, \tag{4.24}$$

$$\frac{dz}{dy_1} = \frac{z(-w_{12} + w_{13} y_1 + w_{14} y_2 - w_{15} z)}{y_1(-w_3 + w_4 x - w_5 y_1 - w_6 y_2 - w_7 z)}, \tag{4.25}$$

and

$$\frac{dz}{dy_2} = \frac{z(-w_{12} + w_{13} y_1 + w_{14} y_2 - w_{15} z)}{y_2(-w_8 + w_9 x - w_{10} y_1 - w_{11} y_2 - w_7 z)}. \tag{4.26}$$

4.2.3 N-SP, M-InP Network Virtualization System

In this subsection, the system model is generalized to an N-SP, M-InP NV system. Consider an NV system in a metropolitan area with multiple EUs, N SPs, and M InPs. Let χ_j be the number of EUs in the area that submits URs to SP_j, $j \in \{1, 2, ..., N\}$. Assuming an EU submits only one UR at a time, the total number of URs in the area is $\chi = \sum_{j=1}^{N} \chi_j$. Meanwhile, suppose ψ_{jk} is the number of VRRs in SP_j to any InP_k, $k \in \{1, 2, ..., M\}$. Thus, the total number of VRRs needed by SP_j to realize its VN is $\psi_j = \sum_{k=1}^{M} \psi_{jk}$. Lastly, let ω_k be the number of active PRs in InP_k. As in the previous subsection, the flow of payment from the EUs to the SPs, and from the SPs to the InP is followed.

Suppose $X = \sum_{j=1}^{N} c_j^{(1)} \chi_j$ is the potential "revenue" at the area from all the URs to any SP_j, $Y_j = \sum_{k=1}^{M} c_k^{(2)} \psi_{jk}$ is the potential revenue of SP_j from all the VRRs to any InP_k, and $Z_k = c_k^{(3)} \omega_k$ is the potential revenue of the InP_k for leasing and operating its PN. The revenue at the area is reduced by the service cost charged by every SP_j to the EUs, $S_{11}^{(1)} Y_1 + S_{12}^{(1)} Y_2 + ... + S_{1N}^{(1)} Y_N$, where the first subscript "1" refers to the metropolitan area (i.e., $i = 1$) where the NV system is. Consequently, the net "income" rate $\frac{1}{X}\frac{dX}{d\tau}$ in (4.2) can now be stated as

$$\frac{1}{X}\frac{dX}{d\tau} = r^{(1)} \left(1 - \frac{X}{K^{(1)}}\right) - \sum_{j=1}^{N} S_{1j}^{(1)} Y_j. \tag{4.27}$$

Meanwhile, the net income rate $\frac{1}{Y_j}\frac{dY_j}{d\tau}$ of SP_j is now the sum of the failed VRR rate $r_j^{(2)}$, the payoff $P_j^{(2)} X$ from the EUs at the service area, the cost of *congestion* $Q_{jj} Y_j$ within SP_j, the cost of *competition* $Q_{j1}^{(2)} Y_1 + Q_{j2}^{(2)} Y_2 + ... + Q_{jN}^{(2)} Y_N$ from every other SP_J $J \in \{1, 2, ..., N\} \setminus j$, and the cost service availability $S_{j1}^{(2)} Z_1 + S_{j2}^{(2)} Z_2 + ... + S_{jM}^{(2)} Z_M$ from every InP_k. Therefore, the net income rate equation (4.4) for SP_j becomes

$$\frac{1}{Y_j}\frac{dY_j}{d\tau} = -r_j^{(2)} + P_j^{(2)} X - \sum_{J=1}^{N} Q_{jJ}^{(2)} Y_J - \sum_{k=1}^{M} S_{jk}^{(2)} Z_k. \tag{4.28}$$

Lastly, the net income rate $\frac{1}{Z_k}\frac{dZ_k}{d\tau}$ of InP_k becomes the sum of the unavailable PR rate $r_k^{(3)}$, the payoff $P_{k1}^{(3)} Y_1 + P_{k2}^{(3)} Y_2 + ... + P_{kN}^{(3)} Y_N$ from every SP_j, the cost of *congestion* $Q_{kk} Z_k$ within InP_k, and the cost of *competition* $Q_{k1}^{(3)} Z_1 + Q_{k2}^{(3)} Z_2 + ... + Q_{jJ}^{(3)} Z_N$ from every other InP_K, $K \in \{1, 2, ..., M\} \setminus k$. Hence, (4.5) for InP_k can be written as

$$\frac{1}{Z_k}\frac{dZ_k}{d\tau} = -r_k^{(3)} + \sum_{j=1}^{N} P_{kj}^{(3)} Y_j - \sum_{K=1}^{M} Q_{kK}^{(3)} Z_K. \tag{4.29}$$

Using the average VR bandwidth cost, $c^{(2)} = \left(\sum_k^M c_k^{(2)}\right)/M$, $Y_j \approx c^{(2)} \psi_j$. Substituting $K^{(1)} = c^{(1)} \kappa^{(1)}$, $P_{\{j,kj\}}^{(l)} = p_{\{j,kj\}}^{(l)} / c^{(l-1)}$, $Q_{\{jJ,kK\}}^{(l)} = q_{\{jJ,kK\}}^{(l)} / c^{(l)}$, and $S_{\{1j,jk\}}^{(l)} =$

$s^{(l)}_{\{1j,jk\}}/c^{(l+1)}$ to (4.27), (4.28), and (4.29) yields the system of equations

$$\frac{d\chi}{d\tau} = \chi \left(r^{(1)} \left(1 - \frac{\chi}{\kappa^{(1)}}\right) - \sum_{j=1}^{N} s^{(1)}_{1j} \psi_j \right) ,$$

$$\frac{d\psi_j}{d\tau} = \psi_j \left(-r^{(2)}_j + p^{(2)}_j \chi - \sum_{J=1}^{N} q^{(2)}_{jJ} \psi_J - \sum_{k=1}^{M} s^{(2)}_{jk} \omega_k \right) , \tag{4.30}$$

and
$$\frac{d\omega_k}{d\tau} = \omega_k \left(-r^{(3)}_k + \sum_{j=1}^{N} p^{(3)}_{kj} \psi_j - \sum_{K=1}^{M} q^{(3)}_{kK} \omega_K \right) .$$

The parameters in an N-SP, M-InP NV system in (4.30) are listed in Table 4.4. Meanwhile, the dimensionless form of (4.30) is

$$\frac{dx}{dt} = x \left(1 - x - \sum_{j=1}^{N} w_{s_{1j}} y_j \right) ,$$

$$\frac{dy_j}{dt} = y_j \left(-w_{r_j} + w_{p_j} x - \sum_{J=1}^{N} w_{q_{jJ}} y_J - \sum_{k=1}^{M} w_{s_{jk}} z_k \right) , \tag{4.31}$$

and
$$\frac{dz_k}{dt} = z_k \left(-w_{r_k} + \sum_{j=1}^{N} w_{p_{kj}} y_j - \sum_{K=1}^{M} w_{q_{kK}} z_K \right) ,$$

where the corresponding parameters are defined in Table 4.5.

4.2.4 Network Economics

The relationship between a capacity demand and a capacity supply in an NV system can be found by combining their time rate equations through chain rule and solving the resulting differential equation numerically. Specifically, if the interest is in the relationship between the capacity demand x at the EU level and the capacity supply y_j of SP_j, then find $\frac{dy_j}{dx}$ and solve it numerically. Similarly, if the interest is in how the capacity supply z_k of InP_k behaves with respect to the capacity demand y_j of SP_j, then determine $\frac{dz_k}{dy_j}$ and solve it using a numerical method. In this work, the numerical method used to solve the resulting differential equation is the 4th-order Runge-Kutta method [86].

4.3 Properties and Analyses

The following section provides the properties of NV systems modeled as a prey-predator food chain system. In addition, this section presents the derivation of the solution point of an NV system. In a prey-predator food chain system, this solution point, also known as the equilibrium point, refers to the state at which the number of agents at each level of the chain does not change [78]. In NV, this equilibrium point refers to the resource strategy (i.e., amount of resources) at each level of the NV system based on the system dynamics such as service, payoff, failure, and congestion/competition rates. In other words, if the dynamics of the NV system can be quantified, the resource strategy of each entity in an NV system can be computed based on these dynamics using the prey-predator food chain model.

This section also introduces the conditions at which the equilibrium point is stable. A stable equilibrium point means that the state of the system goes back to its equilibrium after a small perturbation in the number of agents. In NV, this means that the resource strategy remains the same even after the introduction of a small change in the number of bandwidth requests from the EUs the SPs or allocation from the InPs. For more comprehensive analysis on the prey-predator food chain model, the reader may refer to [79], [80], and [81].

This section is organized as follows. First, the properties and analysis of a one-SP, one-InP NV system is presented. Then, these properties are analyzed and extended to a two-SP, one-InP NV system. Next, an analysis for an N-SP, M-InP NV system is presented. Finally, a summary of this section is provided.

4.3.1 One-SP, One-InP Network Virtualization System

This subsection is devoted to the properties and analysis of the one-SP, one-InP NV system in (4.14). These properties support the use of the prey-predator food chain model to represent an NV system. The network parameters w_1 through w_8 described and derived in the previous section dictate the behavior and properties of a virtualized network. Moreover, an analysis to compute the solution point to the system in (4.14) is also discussed. Theoretically, this solution (i.e., the positive equilibrium point) signifies the values of x, y, and z for which the system is stable indefinitely with respect to time given that the system parameters remain the same. In NV, this solution refers to the resource strategy at each level given the economic dynamics of the NV system.

The prey-predator food chain model exhibits properties that an NV system also possess. The following lemma states the effect of SP service availability to the entire system.

Lemma 1. *If* $w_1 = 0$*, and consequently,* $w_3 = 0$*, then* $\lim_{t\to\infty} x(t) = 1$ *and* $\lim_{t\to\infty} y(t) = \lim_{t\to\infty} z(t) = 0$.

Proof. If $w_1 = w_3 = 0$ is substituted to (4.14), its first equation becomes $\frac{dx}{dt} = x(1-x)$. The solution of this differential equation is $x(t) = \frac{x_0}{x_0+(1-x_0)e^{-t}}$, where $x_0 = x(0)$. Evaluating its limit yields $\lim_{t\to\infty} x(t) = 1$. Repeating this procedure to the second and third equations of (4.14) results to $\lim_{t\to\infty} y(t) = \lim_{t\to\infty} z(t) = 0$. ■

That is, if the SP is unavailable, the EU demand still reaches maximum while the VN supply/demand, and consequently, the PN supply eventually becomes zero.

Likewise, the statement regarding the availability of the InP is stated as follows.

Proposition 1. *If* $w_5 = 0$*, and hence,* $w_7 = 0$*, then it follows that* $\lim_{t\to\infty} x(t) = \frac{w_1 w_2 + w_4}{w_1 w_3 + w_4} = \bar{x} > 0$, $\lim_{t\to\infty} y(t) = \frac{-w_2 + w_3 \bar{x}}{w_4} = \bar{y} > 0$*, and* $\lim_{t\to\infty} z(t) = 0$.

The following lemma states that if the payoff of the SP is high enough, it can continue to demand from the InP, and hence, can continue to support the EUs.

Lemma 2. *If* $w_3 > \frac{w_2 + w_5 z}{x}$*, then* $y > 0$.

Proof. The solution to the second equation of (4.14) is $y(t) = \frac{b_1 y_0}{b_2 y_0 + (b_1 - b_2 y_0)e^{-b_1 t}}$, where $y_0 = y(t=0)$, $b_1 = -w_2 + w_3 x - w_5 z$ and $b_2 = w_4$. Consequently, $\lim_{t\to\infty} y(t) = \frac{b_1}{b_2} > 0$, where it follows that $w_3 > \frac{w_2 + w_5 z}{x}$. ■

A similar statement can be expressed for the InP as well.

Proposition 2. *If* $w_7 > \frac{w_6}{y}$*, then* $z > 0$*.*

The analysis of the system in (4.14) starts with finding its equilibrium point. The following statement expresses the definition of an equilibrium point.

Definition 6. *The equilibrium points of a system* $U(t) = \begin{bmatrix} x(t) & y(t) & z(t) \end{bmatrix}^\top$ *are the solutions to* $\frac{dU}{dt} = 0$*.*

Consequently, the system defined by (4.14) has four equilibrium points:

- $E_0(0, 0, 0)$ indicates the absence of all the system levels;
- $E_1(1, 0, 0)$ shows that the EU will still reach its maximum capacity even in the absence of VN and PN networks;
- $E_2(\bar{x}, \bar{y}, 0)$ denotes the absence of the PN; and
- $E_3(x^*, y^*, z^*)$ specifies the strategies of the SP and the InP given the capacity demand at the EUs.

The first three equilibrium points are irrelevant to our study of a fully functional NV environment. Therefore, the focus is on the positive equilibrium point $E_3(x^*, y^*, z^*)$ or simply $E(x^*, y^*, z^*)$. The existence of this point is defined as follows.

Definition 7. *The positive equilibrium point* $E(x^*, y^*, z^*)$ *exists if there is a positive solution* $\{(x^*, y^*, z^*) | x^* > 0, y^* > 0, z^* > 0\}$ *to*

$$\frac{dx}{dt} = \frac{dy}{dt} = \frac{dz}{dt} = 0. \tag{4.32}$$

Solving (4.32) simultaneously gives the respective resource strategy of the SP and the InP

$$y^* = \frac{1 - x^*}{w_1}, \tag{4.33}$$

and $$z^* = \frac{-w_6 + w_7 y^*}{w_8}. \tag{4.34}$$

Since x^*, y^*, and $z^* > 0$, it follows that

$$1 - w_1 y^* > 0, \quad w_3 w_8 + w_5 w_6 > w_2 w_8, \quad \text{and} \quad w_7 y^* > w_6. \tag{4.35}$$

In order for the system in (4.14) to be stable near or at its equilibrium point $E(x^*, y^*, z^*)$, it must satisfy certain conditions. According to [82], a locally asymptotically stable system is defined as follows:

Definition 8. *A system* $U(t) = \begin{bmatrix} x(t) & y(t) & z(t) \end{bmatrix}^\top$ *is locally asymptotically stable near or at an equilibrium point* U_E *if there is a constant* $K > 0$ *such that* $\|U(0) - U_E\| \leqslant K$*. Thus,* $U(t) \to U_E$ *as* $t \to \infty$*.*

To determine the behavior of the system near or at its equilibrium points, its Jacobian matrix must be computed first. For a system of the form

$$\frac{dx}{dt} = xf(x,y,z), \quad \frac{dy}{dt} = yg(x,y,z), \quad \text{and} \quad \frac{dz}{dt} = zh(x,y,z), \tag{4.36}$$

the Jacobian matrix is defined by

$$J(x^*, y^*, z^*) = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \\ J_{31} & J_{32} & J_{33} \end{bmatrix} = \begin{bmatrix} x\frac{\partial f}{\partial x} + f & x\frac{\partial f}{\partial y} & x\frac{\partial f}{\partial z} \\ y\frac{\partial g}{\partial x} & y\frac{\partial g}{\partial y} + g & y\frac{\partial g}{\partial z} \\ z\frac{\partial h}{\partial x} & z\frac{\partial h}{\partial y} & z\frac{\partial h}{\partial z} + h \end{bmatrix}. \tag{4.37}$$

The characteristic polynomial of $J(x^*, y^*, z^*)$ is

$$p_J(\lambda) = \lambda^3 + A_2\lambda^2 + A_1\lambda + A_0, \tag{4.38}$$

where $A_0 = -J_{11}J_{22}J_{33} + J_{11}J_{23}J_{32} + J_{12}J_{21}J_{33}$, $A_1 = J_{11}J_{22} + J_{11}J_{33} + J_{22}J_{33} - J_{23}J_{32} - J_{12}J_{21}$, $A_2 = -(J_{11} + J_{22} + J_{33})$.

The Routh-Hurwitz stability criterion provides the necessary and sufficient condition for the stability of linear time-invariant (LTI) systems [83], [84]. According to this criterion, a third-order system characterized by its third-order characteristic polynomial (4.38) is locally asymptotically stable near or its equilibrium point if $A_2 > 0$, $A_0 > 0$, and $A_2 A_1 > A_0$.

Lemma 3. *If $J_{11} < 0$, $J_{22} < 0$, and $J_{33} < 0$, then the system in (4.14) is locally asymptotically stable near or at the equilibrium point $E(x^*, y^*, z^*)$. That is, $J_{11} = 1 - 2x^* - w_1 y^* < 0$, $J_{22} = -w_2 + w_3 x^* - 2w_4 y^* - w_5 z^* < 0$, and $J_{33} = -w_6 + w_7 y^* - 2w_8 z^* < 0$.*

Proof. All the terms of A_0 are greater than zero if and only if J_{11}, J_{22}, and J_{33} are all less than zero. Similarly, $A_2 > 0$ if and only if all of its terms are greater than zero. Consequently, the condition $A_2 A_1 > A_0$ is also satisfied. ∎

4.3.2 Two-SP, One-InP Network Virtualization System

The properties of the system in (4.14) can be extended for the system in (4.20). From Lemma 1 about the SP availability, if $w_1 = w_2 = 0$, and consequently, $w_4 = w_9 = 0$, then $\lim_{t\to\infty} x(t) = 1$ and $\lim_{t\to\infty} y_1(t) = \lim_{t\to\infty} y_2(t) = \lim_{t\to\infty} z(t) = 0$. Regarding the InP availability, if $w_7 = 0$, and consequently, $w_{13} = w_{14} = 0$, then, $\lim_{t\to\infty} x(t) = \bar{x} > 0$, $\lim_{t\to\infty} y_1(t) = \bar{y}_1 > 0$, $\lim_{t\to\infty} y_2(t) = \bar{y}_2 > 0$, and $\lim_{t\to\infty} z(t) = 0$. Extending Lemma 2 to (4.20), the respective payoff of each SP must be high enough to continue supporting the URs of the EUs. That is,

$$w_4 > \frac{w_3 + w_6 y_2 + w_7 z}{x} \quad \text{and} \quad w_9 > \frac{w_8 + w_{10} y_1 + w_7 z}{x}. \tag{4.39}$$

The positive equilibrium point $E(x^*, y_1^*, y_2^*, z^*)$ of (4.20) is found by letting

$$\frac{dx}{dt} = \frac{dy_1}{dt} = \frac{dy_2}{dt} = \frac{dz}{dt} = 0. \tag{4.40}$$

Hence, $E(x^*, y_1^*, y_2^*, z^*)$ is the solution to the system of equations

$$1 - x - w_1 y_1 - w_2 y_2 = 0,$$

$$-w_3 + w_4 x - w_5 y_1 - w_6 y_2 - w_7 z = 0,$$

$$-w_8 + w_9 x - w_{10} y_1 - w_{11} y_2 - w_7 z = 0, \tag{4.41}$$

and

$$-w_{12} + w_{13} y_1 + w_{14} y_2 - w_{15} z = 0.$$

Solving (4.41) for x^*, y_1^*, y_2^*, and z^* and setting x^*, y_1^*, y_2^*, and $z^* > 0$ provide the following conditions to be satisfied for the existence of the positive equilibrium point $E(x^*, y_1^*, y_2^*, z^*)$

$$w_1 y_1^* + w_2 y_2^* < 1,$$

$$\left(w_2 w_9 + w_{11} + \frac{w_7 w_{14}}{w_{15}}\right)\left(-w_3 + w_4 + \frac{w_7 w_{12}}{w_{15}}\right) >$$
$$\left(w_2 w_4 + w_6 + \frac{w_7 w_{14}}{w_{15}}\right)\left(-w_8 + w_9 + \frac{w_7 w_{12}}{w_{15}}\right),$$

$$\left(w_1 w_4 + w_5 + \frac{w_7 w_{13}}{w_{15}}\right)\left(-w_8 + w_9 + \frac{w_7 w_{12}}{w_{15}}\right) > \tag{4.42}$$
$$\left(w_1 w_9 + w_{10} + \frac{w_7 w_{13}}{w_{15}}\right)\left(-w_3 + w_4 + \frac{w_7 w_{12}}{w_{15}}\right),$$

and

$$w_{13} y_1^* + w_{14} y_2^* > w_{12}.$$

The behavior of the system in (4.20) near or at $E(x^*, y_1^*, y_2^*, z^*)$ can be analyzed by computing its Jacobian matrix J at $E(x^*, y_1^*, y_2^*, z^*)$. The characteristic polynomial of $J(x^*, y_1^*, y_2^*, z^*)$ is

$$p_J(\lambda) = \lambda^4 + A_3\lambda^3 + A_2\lambda^2 + A_1\lambda + A_0, \tag{4.43}$$

where the leading coefficients are found by solving for the determinant

$$p_J(\lambda) = \det(J - \lambda I). \tag{4.44}$$

For a fourth-degree polynomial $p_J(\lambda)$, the conditions for stability are $A_3 > 0$, $A_2 > 0$, $A_3 A_2 > A_1 > 0$, and $(A_3 A_2 A_1 - A_1^2)/A_3^2 > A_0 > 0$ [85].

Lemma 4. *If* $J_{11} < 0$, $J_{22} < 0$, $J_{33} < 0$, *and* $J_{44} < 0$, *then the system in* (4.20) *is locally asymptotically stable near or at the equilibrium point* $E(x^*, y_1^*, y_2^*, z^*)$. *That is,* $J_{11} = 1 - 2x^* - w_1 y_1^* - w_2 y_2^* < 0$, $J_{22} = -w_3 + w_4 x^* - 2w_5 y_1^* - w_6 y_2^* - w_7 z^* < 0$, $J_{33} = -w_8 + w_9 x^* - w_{10} y_1^* - 2w_{11} y_2^* - w_7 z^*$, *and* $J_{44} = -w_{12} + w_{13} y_1^* + w_{14} y_2^* - 2w_{15} z^* < 0$.

Proof. Based on (4.41), $J_{11}, J_{22}, J_{33}, \text{and } J_{44} < 0$ when $x^*, y_1^*, y_2^*, \text{and } z^* > 0$. ■

4.3.3 N-SP, M-InP Network Virtualization System

Referring to the previous subsection, the properties and analysis of (4.14) and (4.20) can be generalized to a more general case of a N-SP, M-InP NV system. For instance, for an NV system over one metropolitan area, the positive equilibrium point $E^*(x^*, y_1^*, ..., y_N^*, z_1^*, ..., z_M^*)$ of (4.31) is the solution to the system of equations

$$\frac{dx}{dt} = \frac{dy_1}{dt} = ... = \frac{dy_N}{dt} = \frac{dz_1}{dt} = ... = \frac{dz_M}{dt} = 0. \tag{4.45}$$

Combining (4.31) and (4.45), the system has only one positive equilibrium point where $x, y_j, z_k > 0$ since the resulting $N + M + 1$ simultaneous equations

$$1 - x - \sum_{j=1}^{N} w_{s_{1j}} y_j = 0,$$

$$-w_{r_j} + w_{p_j} x - \sum_{J=1}^{N} w_{q_{jJ}} y_J - \sum_{k=1}^{M} w_{s_{jk}} z_k = 0, \tag{4.46}$$

and

$$-w_{r_k} + \sum_{j=1}^{N} w_{p_{kj}} y_j - \sum_{K=1}^{M} w_{q_{kK}} z_K = 0,$$

are all linear in variables x, y_j, and z_k.

The stability analysis of the NV system involves computing for its $(N + M + 1) \times (N + M + 1)$ Jacobian matrix at $E^*(x^*, y_1^*, ..., y_N^*, z_1^*, ..., z_M^*)$, as well as conforming with the Routh-Hurwitz stability criteria for a $(N + M + 1)$-degree characteristic polynomial.

4.3.4 Network Resource Strategies

This section has focused on solving for the equilibrium point or resource strategies in an NV system. From the first equation of (4.46), the strategy of SP_j is

$$y_j^* = \frac{1}{w_{s_{1j}}}\Bigg(1 - x^* - \sum_{\substack{J=1 \\ J\neq j}}^{N} w_{s_{1J}} y_J^*\Bigg). \tag{4.47}$$

Meanwhile, the strategy of InP_k can be derived from the third equation of (4.46),

$$z_k^* = \frac{1}{w_{q_{kk}}}\Bigg(- w_{r_k} + \sum_{j=1}^{N} w_{p_{kj}} y_j^* - \sum_{\substack{K=1 \\ K\neq k}}^{M} w_{q_{kK}} z_K^*\Bigg). \tag{4.48}$$

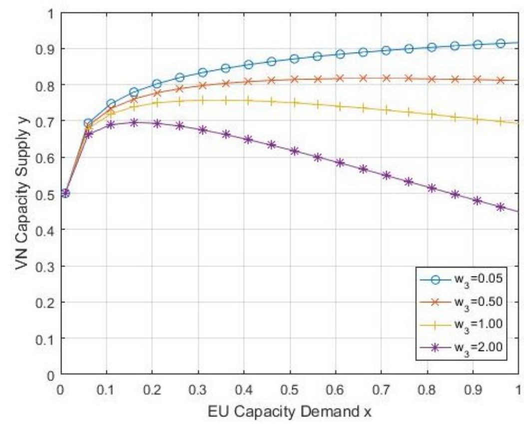
Equations (4.47) and (4.48) signify the optimal strategies of the SPs and the InPs given the dynamics of the NV system. These are helpful in analyzing how the network parameters affect the network resource strategies of the SPs and the InPs.

4.4 Simulation Results

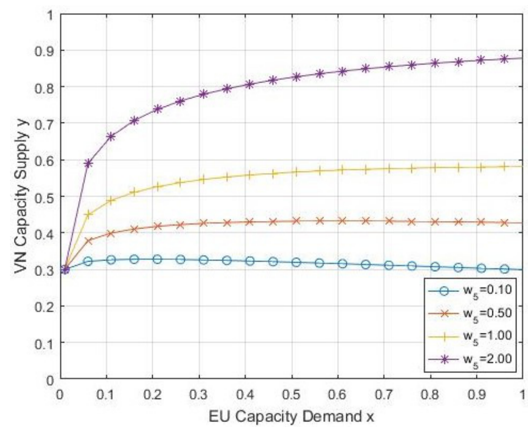
In this section, numerical simulations that demonstrate the prey-predator food chain model as a mathematical framework for NV are presented. For simplicity, it is assumed that the VN/PN network capacities are the same as the total virtual/physical node capacities. That is, a network capacity is equal the sum of its node capacities.

4.4.1 One-SP, One-InP Network Virtualization System

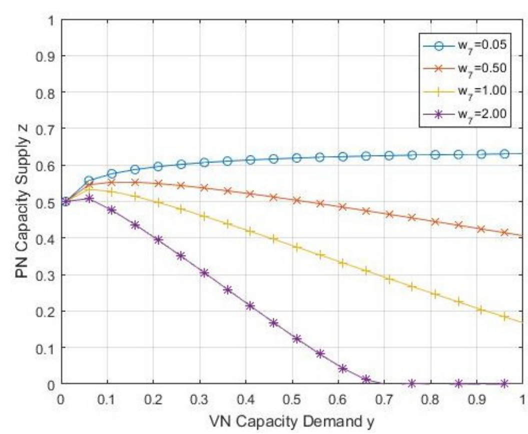
In this subsection, the economics of a simple one-SP, one-InP NV system is investigated. Assume the following setup. Suppose that the number of URs (or EUs) in the serviced area is limited to 1000. The sole SP can expand its VN up to a maximum of 50 virtual nodes and allow 25 EUs per VR. In addition, the sole InP can scale its PN up to 50 physical nodes



(a) The effect of VN payoff to the VN capacity supply.



(b) The effect of PN service availability to the VN capacity supply.



(c) The effect of PN payoff to the PN capacity supply.

Figure 4.2: The network economics of a one-SP, one-InP NV system.

and allow 4 VRs to share a single PR. Let each virtual and physical node have a capacity of 25 and 100 Mbps, respectively. In each simulation, the values for w_1 through $w_8 > 0$ that satisfy Lemma 3 and (4.35) are provided. Table 4.6 lists the values used in this subsection, unless otherwise stated.

Network Resource Strategies

The following simulations shows the effect of NV parameters to the resource strategy $E(x^*, y^*, z^*)$ by solving the system defined by (4.14) using (4.32).

First, the influence of the VN service availability w_1 to the SP resource strategy y^* is studied. Hence, w_1 is set to four different values, 0.10, 0.50, 1.00, and 2.00. From (4.15) and (4.16), increasing w_1, while keeping other parameters the same, indicates that the SP has either increased n or expanded $k^{(2)}$. The computed value for w_7 is 0.125. The results of the simulations are listed in Table 4.7. When w_1 is increased from 0.10 to 2.00, while keeping other parameters constant, especially the VN payoff $w_3 = \frac{w_1}{\eta_2}$, the SP adapts by decreasing its VN capacity demand strategy y^* from 0.70, or equivalently, $0.70 \cdot 50\,\mathrm{VR} \cdot 25\,\frac{\mathrm{Mbps}}{\mathrm{VR}} = 875$ Mbps to 0.31 or 388 Mbps. The reason is that the SP has no incentive to increase its demand strategy from the InP when its payoff from the EU is not improving. Likewise, the InP reduces its supply strategy z^* in response to the lower VN demand.

Next, the effect of the PN payoff factor w_7 to the InP resource strategy z^* is investigated. Thus, w_7 is allowed to have the values 0.10, 0.15, 0.20, and 0.25. From (4.15) and (4.16), the InP has to either reduce m or scale down $k^{(3)}$ to increase w_7 while keeping the other parameters constant. The computed value w_3 is 0.40. The results are listed in Table 4.7. Assume that the SP can only lease one portion of a PR. It follows that, there are $20\,\frac{\mathrm{Mbps}}{\mathrm{PR}}$. When $w_7 = 0.10$, z approaches $z^* = 0.45 \cdot 50\,\mathrm{PR} \cdot 25\,\frac{\mathrm{Mbps}}{\mathrm{PR}} = 563$ Mbps; however, when $w_7 = 0.25$, $z^* = 0.61$ or 763 Mbps. That is, when the payoff w_7 increases, especially with PN service availability $w_5 = \frac{w_5}{\eta_3}$ constant, the InP can afford to supply a higher additional capacity z^* to the SP. On the other hand, the resource strategy y^* of the SP is to reduce its demand from the InP because as w_7 becomes higher, n or k_3 is reduced. Consequently, the high payoff that the InP acquires will come from the larger share of resources, and hence cost, the SP has to pay in return.

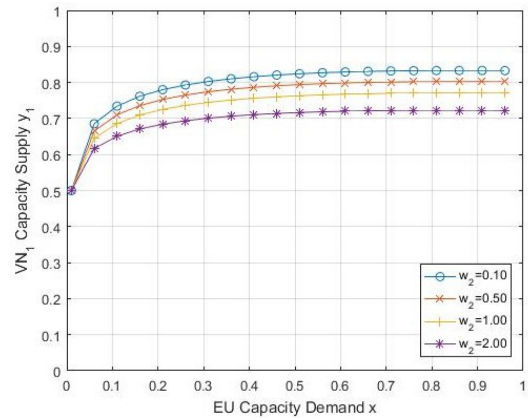
Network Economics

The relationship between the EU capacity demand and the VN capacity supply is explored through solving (4.17) by implementing a numerical method called the 4th-order Runge-Kutta method [86].

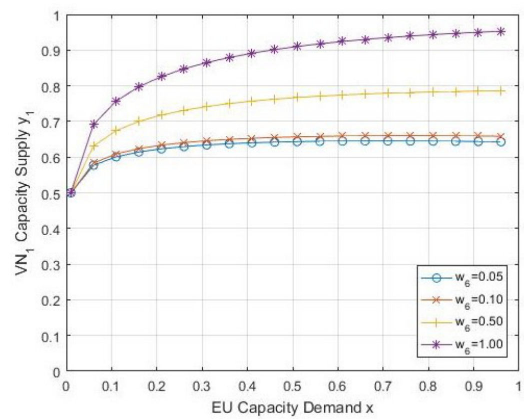
First, the effect of the VN payoff factor w_3 to the VN capacity supply is discovered. Consider four different values for w_3: 0.05, 0.50, 1.00, and 2.00. Also, set $z = 0.50$ and $w_1 = 5.00$. The results of this case is illustrated in Fig. 4.2a. As w_3 increases from 0.05 to 2.00 at 50% EU activity (i.e., $x = 0.50$), the available VN supply decreases from $y = 0.87$ to 0.62 . This happens since an increase in w_3 means that $\eta^{(2)}$ has been decreased, which is caused by a small n or $k^{(2)}$. Hence, when a VR can accommodate lesser EUs or when the SP can only manage a smaller VN, the SP adapts by decreasing its VN demand from the InP, and thus, its VN supply to the EUs.

Now, the influence of the PN service availability w_5 to the VN capacity supply is studied. Assume $w_1 = 10.00$, which is quite high just to accentuate the difference in the four curves to be plotted, $y_0 = 0.30$, and $z = 0.50$. Allow w_5 to have these values: 0.10, 0.50, 1.00, and 2.0. Fig. 4.2b shows the results of the simulation. Hence, a higher w_5 allows the SP to increase its VN demand from the InP, and hence, its VN supply to the EU, and accommodate more EUs. For example, at $x = 0.50$, the VN capacity supply goes from $y = 0.32$ to 0.82 when w_5 varies from 0.10 to 2.00.

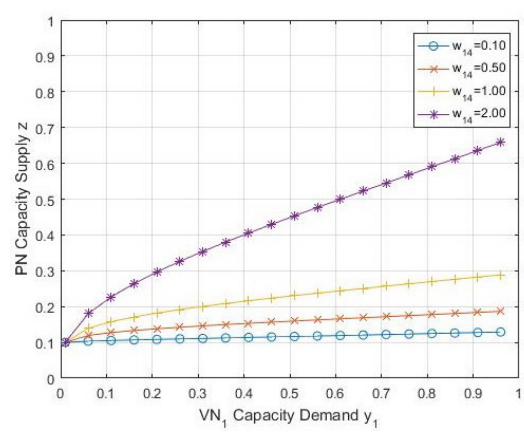
Lastly, consider the connection between the VN capacity demand and the PN capacity supply through solving (4.18) using the Runge-Kutta method [86]. To investigate the consequence of the PN payoff factor to its own capacity, vary w_7 to these values: 0.05, 0.50,



(a) The effect of SP_2-to-EU service availability to the VN_1 capacity supply.



(b) The effect of SP_2-to-SP_1 competition to the VN_1 capacity supply.



(c) The effect of SP_2-to-InP payoff to the PN capacity supply.

Figure 4.3: The network economics of a two-SP, one-InP NV system.

1.00, and 2.00. Then, set $x = 0.50$, $w_3 = 0.05$, and $w_5 = 2.00$. The result is plotted in the Fig. 4.2c. At $y = 0.50$, the PN supply decreases from 0.62 to 0.13 as w_7 is increased from 0.05 to 2.00. It can be deduced from the definition of w_7 that it increases when m or $k^{(3)}$ is reduced. Thus, when w_7 is high, the PN decreases its supply because n is low.

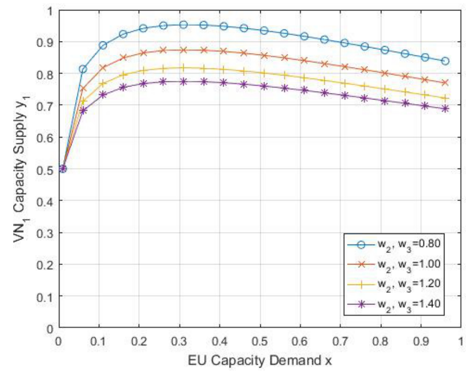
4.4.2 Two-SP, One-InP Network Virtualization System

In this subsection, the focus is on the effect of the existence of an additional SP in the economics of an NV system. In the following simulations, the values in Table 4.8 are used, unless otherwise stated. Entries labeled with * are computed values using (4.21), while those with $^+$ are varied. These values are chosen to satisfy Lemma 4 and (4.42).

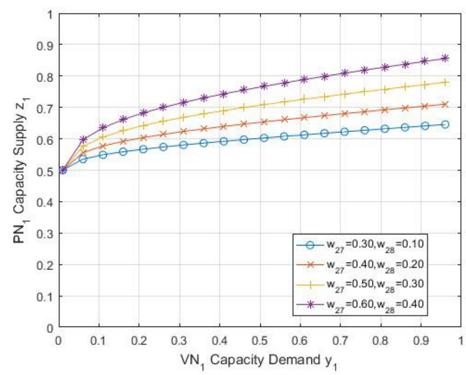
Network Resource Strategies

The network strategy for a two-SP, one InP NV system of (4.20) is found by solving (4.41). Table 4.9 shows the effect of the SP_2 service availability w_2 to the network strategies. The values for w_2 are 0.20, 0.25, 0.30, and 0.40, while the corresponding computed values for w_9 are 0.25, 0.31, 0.38, and 0.50. As w_2 is increased from 0.20 to 0.40, the strategy of SP_2 increases from 0.15 to 0.91 (i.e., from 150 to 910 Mbps) due to higher demand from EUs. However, the strategy of SP_1 has diminished from 0.96 to 0.46 (i.e., from 960 to 460 Mbps) since EUs have an alternative SP.

Next, w_9 assumes values of 0.25, 0.30, 0.40, and 0.50, and the computed values for w_2 are 0.20, 0.24, 0.32, and 0.40. When SP_2 has an increasing payoff w_9 from 0.25 to 0.50, its strategy also increases from 0.15 to 0.91 (i.e., from 150 to 910 Mbps). However, SP_1, whose payoff w_4 remains constant at 0.40, decreases its strategy from 0.96 to 0.46 (i.e., from 960 to 460 Mbps). This conflicting behavior between the two SPs allows the EUs to still get service even if one of the SPs is decreasing its strategy.



(a) The effect of service availabilities to the VN_1 capacity supply y_1.



(b) The effect of InP_1 payoffs to the PN_1 capacity supply.

Figure 4.4: The network economics of a three-SP, two-InP NV system.

Network Economics

Fig. 4.3a illustrates the effect of SP_2 service availability w_2 to the VN capacity of SP_1 as a result of solving (4.23) using the Runge-Kutta method. At $x = 0.5$ and as w_2 is increased, the VN capacity of SP_1 y_1 decreases from 0.83 to 0.72. That is, a more reachable competing SP can accommodate more users, causing the demand for other SPs to decrease.

Likewise, from numerically solving (4.23), the influence of the SP_2 competition w_6 to the VN capacity of SP_1 is plotted in Fig. 4.3b. It shows that SP_1 increases its VN capacity when the effect of competition from SP_2 w_6 is greater. For instance, at $x = 0.5$, y_1 goes from 0.65 to 0.92. In other words, the SP_1 will try to increase its capacity to satisfies its users.

Lastly, the PN capacity as a function of the VN_1 capacity is found by solving (4.25). Fig. 4.3c shows the changes to the PN capacity with respect to the SP_2-to-InP payoff w_{14}, with all initial capacities lowered to 0.10. At $y_1 = 0.5$, as the payoff w_{14} of SP_2 is increased, the PN capacity supply z increases from 0.12 to 0.42 to keep up with the increasing demand from SP_2.

4.4.3 Three-SP, Two-InP Network Virtualization System

This section is concluded by showing the economics of a three-SP, two-InP NV system. From (4.31), this configuration includes 36 network parameters. Consequently, for brevity, only the values of parameters that vary in each setup are shown.

Network Resource Strategies

To show the effect of the service availabilities w_2 and w_3 of SP_2 and SP_3 to the equilibrium point $E^*(x, y_1^*, y_2^*, y_3^*, z_1^* z_2^*)$, set w_2 and w_3 to 0.50, 0.55, 0.60, and 0.65 while keeping all the other parameters constant. The results are tabulated in 4.10. Thus, when the service availabilities of SP_2 and SP_3 increase, the resource strategy y_1^* of SP_1 decreases. Moreover, the resource strategies y_2^* of SP_2 and y_3^* of SP_3 also decrease.

Network Economics

The effect of service availabilities w_2 and w_3 of SP_2 and SP_3, respectively, to the VN capacity supply of SP_1 is shown in Fig. 4.4a. It is evident that as the other two SPs become more available to the EUs, with w_2 and w_3 taking values 0.80, 1.00, 1.20, and 1.40, the capacity supply of SP_1 decreases. This may be attributed to the behavior of the EUs having more choices for the SPs. However, the trend for VN_1 capacity supply y_1 stays the same where it decreases with increasing EUs demand.

In Fig. 4.4b, the effect of SP_2-to-InP_1 payoff w_{27} and SP_3-to-InP_1 payoff w_{28} to the PN capacity supply of InP_1 z_1 is shown. As w_{27} and w_{28} increases, z_1 increases with the demand from SP_1 as well. That is, InP_1 can afford to increase its supply as long as its payoff for doing so increases, no matter from which SP the increase in payoff comes from. Finally, for networks with higher number of SP and InP, they can be analyzed and simulated in a similar way.

4.5 Related Works

To the best of our knowledge, the prey-predator food chain model has not been used to model the resource allocation and economics in an NV system. However, several related research papers have inspired us to develop a new mathematical framework for NV. Some of these works utilized game theory, contract theory, and queuing theory. These frameworks have not only been applied to traditional NV but also to wireless NV and to network function virtualization (NFV).

The authors of [87] proposed a non-cooperative game theoretic approach to bandwidth allocation. They developed an algorithm that iteratively solved for the Nash equilibrium of the game, and in turn, found the solution to the bandwidth allocation problem. Meanwhile, the authors of [88] proposed a two-stage non-cooperative game for bandwidth allocation. In the first stage, the bandwidth negotiation game, the SPs request bandwidth from multiple InPs. In the second stage, the bandwidth provisioning game, the SPs sharing the same physical link of an InP compete for bandwidth. Game theory was also used to improve the utilization of resources. The authors of [89] proposed a two-stage Stackelberg game-based mechanism that allows the InP to determine a reselling rate that maximizes its revenue and enables the SP to calculate the amount of bandwidth that minimizes its expenses.

Virtualization in wireless networks, or wireless NV, allows multiple wireless networks operated by different SPs to dynamically share a common physical infrastructure operated by mobile network operators (MNO) [90]. Game theory was also applied to wireless NV, where the MNOs are responsible for dynamic wireless resource management, while the SPs aim at providing proper bandwidth allocation to their users. In [91], the interactions between the SPs and the NO were modeled as a stochastic game, since the demand of the users of the SPs varies with time. However, since the resulting stochastic game has strong dependency among SPs and on future actions, the authors utilized the Vickrey-Clarke-Groves (VCG) auction mechanism and conjectural pricing to transform the game into a series of independent games. Other papers that used game theory to formulate and solve an NV resource allocation problem include [92], [93], [94], and [95].

In NFV, software implementations of network functions are decoupled from the hardware [96]. Hence, multiple virtual network functions (VNFs) can be multiplexed in common physical servers. The authors of [97] used mixed queuing model to allocate the resources of the servers to VNFs. Each user of the network requires different combinations of VNFs, or service chains. Thus, each VNF was modeled as a queue, and the model was designed to minimize the waiting time of service chains. Since VNFs are normally "chained" to each other, and thus may be utilized in succession by its users, effective placement and scheduling of these VNF chains in physical servers have been proposed in [98] to achieve high network resource utilization and low network response latency. The authors proposed a priority-driven weighted algorithm to maximize the average resource utilization, and a novel heuristic algorithm to minimize the average response latency. When the central offices (COs), where physical servers are housed, are geo-distributed, assigning VNFs to servers can be challenging. Consequently, to provide load balancing and cost efficiency, the authors of [99] proposed a framework that selects a set of COs that minimizes the communication cost within the set and then uses a shadow-routing approach that jointly optimizes VNF-CO and VNF-server assignment.

Many other approaches have been used to model the allocation of resources in NV. The authors of [100] proposed a parallel and distributed algorithm based on alternating direction method of multipliers (ADMM) for bandwidth allocation and routing problem that minimizes the traffic that the SP encounters. Information asymmetry in wireless NV, a situation where the NO has no access to private information about the SPs, was investigated by the authors of [101]. They proposed a solution to the resource allocation problem using contract theory, where the NO offers each SP a contract that allows each to maximize its own payoff. In [102], when the SPs could not report their traffic, a traffic predictor was integrated into the bandwidth allocation algorithm so that the InPs can supply accurate amount of resources to the SPs. Other approaches to resource allocation in NV include matching theory [103] [104], machine learning [105] [106], and fuzzy logic [107].

4.6 Conclusion

In this work, a novel approach to NV resource allocation and economics based on prey-predator food chain model has been developed. Models for one-SP, one-InP and two-SP, one-InP NV systems have been formulated, as well as the generalized N-SP, M-InP NV system. The properties of these configurations have also been discussed. Furthermore, the derivation of the equilibrium point of these models, as well as the conditions for its existence and stability, have also been presented. Economic models that associate the EU demand to the SP supply and the SP demand to the InP supply have been proposed. Numerical simulations have been performed to demonstrate the characteristics of these models. The results have indicated that the SP and the InP resource strategies depend on NV parameters such as payoff, failure, service, and congestion factors. Therefore, the SP and InP can control their resource strategies by changing the relevant network parameters. Moreover, the capacity supplies of the SP and the InP as a function of their respective demand also change with these network parameters.

Table 4.1: Network Virtualization Parameters.

Symbol	Definition
χ	number of user requests (URs)
ψ	number of virtual resource requests (VRRs)
ω	number of active physical resources (PRs)
τ	time
$r^{(1)}$	arrival rate of user requests (per unit time)
$r^{(2)}$	failure rate of virtual resource requests (per unit time)
$r^{(3)}$	unavailability rate of physical resources (per unit time)
$\kappa^{(1)}$	maximum number of user requests (URs)
$\kappa^{(2)}$	maximum number of virtual resource requests (VRRs)
$\kappa^{(3)}$	maximum number of physical resources (PRs)
s	service rate (per unit time per VRR (or PR))
p	payoff rate (per unit time per UR (or VRR))
q	congestion/competition rate (per unit time per VRR (or PR))
n	max. no. of EUs per VRRs (URs per VRR)
m	max. no. of VRs per PR (VRRs per PR)

Table 4.2: One-SP, One-InP NV Dimensionless Parameters.

Parameter	Definition	Name
x	$\frac{\chi}{\kappa^{(1)}}$	EU capacity demand
y	$\frac{\psi}{\kappa^{(2)}}$	VN capacity supply/demand
z	$\frac{\omega}{\kappa^{(3)}}$	PN capacity supply
t	$r^{(1)}\tau$	dimensionless time
w_1	$\frac{s^{(1)}}{r^{(1)}/\kappa^{(2)}}$	SP service availability ratio
w_2	$\frac{r^{(2)}}{r^{(1)}}$	SP failure ratio
w_3	$\frac{p^{(2)}}{r^{(1)}/\kappa^{(1)}}$	SP payoff factor
w_4	$\frac{q^{(2)}}{r^{(1)}/\kappa^{(2)}}$	SP congestion factor
w_5	$\frac{s^{(2)}}{r^{(1)}/\kappa^{(3)}}$	InP service availability ratio
w_6	$\frac{r^{(3)}}{r^{(1)}}$	InP unavailability ratio
w_7	$\frac{p^{(3)}}{r^{(1)}/\kappa^{(2)}}$	InP payoff factor
w_8	$\frac{q^{(3)}}{r^{(1)}/\kappa^{(3)}}$	InP congestion factor

Table 4.3: Two-SP, One-InP NV Dimensionless Parameters.

Parameter	Definition	Name
x	$\frac{\chi}{\kappa^{(1)}}$	EU capacity demand
y_j	$\frac{\psi_j}{\kappa_j^{(2)}}$	VN$_j$ capacity supply/demand
z	$\frac{\omega}{\kappa^{(3)}}$	PN capacity supply
t	$r^{(1)}\tau$	dimensionless time
w_1	$\frac{s_1^{(1)}}{r^{(1)}/\kappa_1^{(2)}}$	SP$_1$-to-EU service availability factor
w_2	$\frac{s_2^{(1)}}{r^{(1)}/\kappa_2^{(2)}}$	SP$_2$-to-EU service availability factor
w_3	$\frac{r_1^{(2)}}{r^{(1)}}$	SP$_1$ failure ratio
w_4	$\frac{p_1^{(2)}}{r^{(1)}/\kappa^{(1)}}$	EU-to-SP$_1$ payoff factor
w_5	$\frac{q_1^{(2)}}{r^{(1)}/\kappa_1^{(2)}}$	SP$_1$ congestion factor
w_6	$\frac{q_2^{(2)}}{r^{(1)}/\kappa_2^{(2)}}$	SP$_2$-to-SP$_1$ competition factor
w_7	$\frac{s^{(2)}}{r^{(1)}/\kappa^{(3)}}$	InP-to-SP service availability factor
w_8	$\frac{r_2^{(2)}}{r^{(1)}}$	SP$_2$ failure ratio
w_9	$\frac{p_2^{(2)}}{r^{(1)}/\kappa^{(1)}}$	EU-to-SP$_2$ payoff factor
w_{10}	$\frac{q_1^{(2)}}{r^{(1)}/\kappa_1^{(2)}}$	SP$_1$-to-SP$_2$ competition factor
w_{11}	$\frac{q_2^{(2)}}{r^{(1)}/\kappa_2^{(2)}}$	SP$_2$ congestion factor
w_{12}	$\frac{r^{(3)}}{r^{(1)}}$	InP unavailability factor
w_{13}	$\frac{p^{(3)}}{r^{(1)}/\kappa_1^{(2)}}$	SP$_1$-to-InP payoff factor
w_{14}	$\frac{p^{(3)}}{r^{(1)}/\kappa_2^{(2)}}$	SP$_2$-to-InP payoff factor
w_{15}	$\frac{q^{(3)}}{r^{(1)}/\kappa^{(3)}}$	InP congestion factor

Table 4.4: N-SP, M-InP NV System Parameters.

Symbol	Definition
χ	number of user requests (URs) at service area
ψ_j	number of virtual resource requests (VRRs) at SP_j
ω_k	number of active physical resources at InP_k
τ	time
$r^{(1)}$	arrival rate of URs at service area
$r_j^{(2)}$	failure rate of VRRs at SP_j
$r_k^{(3)}$	unavailability rate of PRs at InP_k
$s_{1j}^{(1)}$	SP_j-to-EU service rate
$p_j^{(2)}$	EU-to-SP_j payoff rate
$q_{jJ}^{(2)}$	SP_J-to-SP_j congestion/competition rate
$s_{jk}^{(2)}$	InP_k-to-SP_j service rate
$p_{kj}^{(3)}$	SP_j-to-InP_k payoff rate
$q_{kK}^{(3)}$	InP_K-to-InP_k congestion/competition rate

Table 4.5: N-SP, M-InP NV System Dimensionless Parameters.

Symbol	Definition	Name
x	$\frac{\chi_i}{\kappa_i^{(1)}}$	EU demand
y_j	$\frac{\psi_j}{\kappa_j^{(2)}}$	VN$_j$ network demand
z_k	$\frac{\omega_k}{\kappa_k^{(3)}}$	PN$_k$ network supply
t	$r^{(1)}\tau$	time
w_{r_j}	$\frac{r_j^{(2)}}{r^{(1)}}$	failure rate of VRRs at SP$_j$
w_{r_k}	$\frac{r_k^{(3)}}{r^{(1)}}$	unavailability rate of PRs at InP$_k$
$w_{s_{1j}}$	$\frac{s_{1j}^{(1)}}{r^{(1)}/\kappa_j^{(2)}}$	SP$_j$-to-EU service rate
w_{p_j}	$\frac{p_j^{(2)}}{r^{(1)}/\kappa^{(1)}}$	EU-to-SP$_j$ payoff rate
$w_{q_{jJ}}$	$\frac{q_{jJ}^{(2)}}{r^{(1)}/\kappa_j^{(2)}}$	SP$_J$-to-SP$_j$ congestion/competition rate
$w_{s_{jk}}$	$\frac{s_{jk}^{(2)}}{r^{(1)}/\kappa_k^{(3)}}$	InP$_k$-to-SP$_j$ service rate
$w_{p_{kj}}$	$\frac{p_{kj}^{(3)}}{r^{(1)}/\kappa_j^{(2)}}$	SP$_j$-to-InP$_k$ payoff rate
$w_{q_{kK}}$	$\frac{q_{kK}^{(3)}}{r^{(1)}/\kappa_k^{(3)}}$	InP$_K$-to-InP$_k$ congestion/competition rate

Table 4.6: Parameter values for One-SP, One-InP NV System.

EU		SP		InP	
Parameter	Value	Parameter	Value	Parameter	Value
$\kappa^{(1)}$	1000	$\kappa^{(2)}$	50	$\kappa^{(3)}$	50
		n	25	m	4
		w_1	0.50	w_5	0.50
		w_2	0.10	w_6	0.10
		w_3	0.50	w_7	0.50
		w_4	0.10	w_8	0.10
x_0	0.50	y_0	0.50	z_0	0.50

Table 4.7: The effect of VN service availability and PN payoff factor on the equilibrium.

E	w_1				w_7			
	0.10	0.50	1.00	2.00	0.10	0.15	0.20	0.25
x^*	0.93	0.72	0.56	0.37	0.73	0.79	0.83	0.86
y^*	0.70	0.55	0.44	0.31	0.55	0.42	0.34	0.28
z^*	0.77	0.59	0.45	0.29	0.45	0.53	0.58	0.61

Table 4.8: Parameter values for Two-SP, One-InP NV System.

EU		SP_1		SP_2		InP	
Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
$\kappa^{(1)}$	1000	$\kappa_1^{(2)}$	50	$\kappa_2^{(2)}$	40	$\kappa^{(3)}$	50
		n_1	25	n_2	20	m	4
		w_1	0.50	w_2	+	w_7	0.20
		w_3	0.01	w_8	0.01	w_{12}	0.01
		w_4	0.40*	w_9	+	w_{13}	0.05*
		w_5	0.10	w_{11}	0.10	w_{14}	0.04*
		w_6	0.01	w_{10}	0.01	w_{15}	0.10
x_0	0.50	$y_{1(0)}$	0.50	$y_{2(0)}$	0.50	z_0	0.50

Table 4.9: The effect of service availability and payoff factor on the equilibrium.

E	w_2				w_9			
	0.20	0.25	0.30	0.40	0.25	0.30	0.40	0.50
x^*	0.49	0.48	0.47	0.41	0.49	0.49	0.46	0.41
y_1^*	0.96	0.85	0.72	0.46	0.96	0.87	0.66	0.46
y_2^*	0.15	0.38	0.59	0.91	0.15	0.33	0.66	0.91
z^*	0.44	0.47	0.49	0.49	0.44	0.47	0.50	0.49

Table 4.10: The effect of service availabilities on the equilibrium.

E	w_2, w_3			
	0.50	0.55	0.60	0.65
x^*	0.7773	0.7621	0.7475	0.7332
y_1^*	0.0589	0.0577	0.0565	0.0553
y_2^*	0.1485	0.1459	0.1434	0.1411
y_3^*	0.2380	0.2341	0.2304	0.2286
z_1^*	1.0331	1.0125	0.9924	0.9731
z_2^*	1.8463	1.8099	1.7747	1.7406

Chapter 5

Conclusions and Future Works

5.1 Conclusions

In this dissertation, the application of mean field games (MFGs) to future generation of communication networks has been proposed. Based on the results of this work, MFG is an excellent framework to analyze, model, and simulate these networks in order to address their issues and features including massive connectivity, energy efficiency, low latency, and improved network performance.

The first work deals with the analysis and modeling of user behavior in social networks. A multiple-population MFG approach has been proposed to gain knowledge about the belief and opinion evolution of social network users. Furthermore, the presence of stubborn users, or users that are reluctant to change their initial opinion, has been integrated in the MFG-based behavior modeling. The results have indicated that the proposed MFG-based framework provides insights on how users and populations behave on a multiple-population social network. Using a social evolution dataset as a benchmark, the proposed MFG-based method has also allowed for an effective approach to estimate and predict the distribution of opinions of social network users on a social topic.

The second work focuses on energy-efficient and low-latency computation offloading in multi-access edge computing networks (MECN). The computation offloading problem has been formulated as a mean-field-type game (MFTG), which is a relaxed version of MFG. Then, non-cooperative and cooperative computation offloading algorithms have been proposed. These algorithms compute the optimal offloading control or the optimal number of computation tasks to be offloaded by each computing node from the MECN. The non-cooperative algorithm is a decentralized approach where each computing node determines its own offloading control, whereas the cooperative algorithm is a centralized approach where the MECN determines the offloading control of each computing node. The results have proven that the MFTG-based algorithms are efficient and effective in implementing computation offloading in MECNs.

Finally, the last work introduces a dynamic hierarchical game approach to resource allocation and economics in network virtualization (NV). An NV system is a dynamic hierarchical system that has three levels, the end users (EUs), the service providers (SPs), and the infrastructure providers (InPs). The prey-predator food chain model has been utilized to model the dynamic interactions that exist between these entities. Furthermore, the proposed models have been used to determine the network resource strategies or the amount of network resources demanded and/or supplied by each entity in an NV system. The results have indicated that the network resource strategies of the SPs and the InPs depend on NV parameters such as payoff, failure, service, and congestion factors. Therefore, the SP and InP can control their resource strategies by changing the relevant network parameters. This work is a preparation for a future work that integrates MFG into the resource allocation and economics framework of an NV system.

5.2 Future Works

5.2.1 Unmanned Aerial Vehicle Networks

Unmanned aerial vehicles (UAVs) have played major roles in recent advances in communications, civil, and military applications [108]. UAVs in communication networks can be utilized either as aerial base stations to extend the coverage, increase the capacity, and improve the reliability and energy efficiency of wireless communication networks, or as mobile access points to enable applications such as real-time video streaming and content delivery [109]. Meanwhile, UAVs can be integrated in civil services such as search and rescue operations, remote sensing, and security and surveillance due to their ease of deployment, low maintenance cost, high-mobility, and ability to hover [110]. UAVs can speed-up search and rescue operations in case of natural or man-made disasters, assist in gathering and delivering data from ground sensors to base stations, and improve safety, coverage, range, robustness, and efficiency of surveillance [111].

A special application of UAVs is pursuit and evasion games between pursuer UAVs and target UAVs. This application is prevalent in security and surveillance operations administered by the police, military, and private institutions. In a pursuit and evasion game, the goal of a pursuer UAV is to follow and capture a target UAV while minimizing the associated cost; the goal of a target UAV is to move away and prevent capture by a pursuer UAV while minimizing the associated cost. Since UAVs have limited resources such as energy supply and computation capability, they must be able to move according to an optimizing strategy and must be able to do so through a computationally efficient algorithm.

In a future work, the authors propose a special form of MFG to formulate and model pursuit and evasion games involving a large number of UAVs. For games with Stackelberg structure where leading players enforce strategies to a large number of following players with each player performing based on their optimal control that minimizes their cost, a mean-field Stackelberg game [112] can be used to find the optimal control of each player. Consequently, for pursuit and evasion scenarios with one target UAV and a large number of pursuer UAVs, the authors propose a mean-field Stackelberg game to determine the optimal control of each UAV that minimizes their cost function.

5.2.2 Network Virtualization

In Chapter 4, a unique approach to analyzing and modeling of resource allocation and economics in network virtualization (NV) has been introduced. NV is a dynamic system where the quantities of network resource trading between the consumers and the producers are changing with time. Due to these inter-level and intra-level interactions in NV, a conventional game theoretical approach becomes more complicated. Thus, a dynamic hierarchical game model has been proposed to capture these interactions in an NV system.

There are three levels in the dynamic hierarchical game model: the lowest level consists of the end users (EUs); the middle level consists of the service providers (SPs); and the top level consists of the infrastructure provider (InPs). An interaction between two levels in the model is represented by service payment (e.g., money) in NV, while an interaction within the same level in the model is represented by competition (e.g., loss) in NV. Thus, the proposed dynamic hierarchical model represents the dynamic allocation of resources between each level as well as the economics (i.e., transfer of wealth) in the NV system.

As part of a future work, the authors propose to integrate MFG in order to model the resource allocation and economics of a virtualized network with large number of virtual networks and users. Since the hierarchy denotes level of precedence among the entities (i.e., InPs have more influence than SPs and SPs have more influence than EUs), the authors propose MFG with major and minor players [113]. In this MFG configuration, the major players have non-negligible influence to the behavior of the system while minor players have negligible individual effect to the system. In NV, the absence or presence of a few EUs can have negligible effect on the SPs and the absence or presence of a few SPs can have negligible effect on InPs. However, the InPs have non-negligible effect on the SPs and the SPs have non-negligible effect on the EUs.

5.2.3 Internet of Things

The Internet of things (IoT) allows physical objects to communicate with each other, make decisions and deliver smart services together [114]. It should be capable of interconnecting billions or trillions of heterogeneous objects through the internet. The emerging

IoT applications include smart home, intelligent transportation system, smart city, industrial, and smart healthcare [115]. In order to successfully implement these applications, the key requirements include low deployment cost, long battery life, low device cost, extended coverage area, support for massive number of connected devices (scalability), and security and privacy.

Since IoT consists of a large number of heterogeneous devices that make network modeling and optimization more complicated, the authors propose MFG as a method to model and optimize such networks. MFG can help devices decide their strategies in order for an IoT network to perform a specific functionality satisfactorily. Specifically, as in the first part of this dissertation, MFG can be applied to model the behavior of a variety of IoT devices, and hence help the network optimize its performance based on the estimated or predicted behavior of the devices. Furthermore, MFG can also be used to provide solution to challenges that IoT is facing. For instance, in IoT security and privacy, an IoT device can base its strategy (e.g., whether to continue communication with the network) on the perceived level of network security. In this scenario, MFG can be used to estimate the perceived level of security in the network given the behavior of other devices in the network.

References

[1] Cisco, "Cisco Annual Internet Report (2018–2023) White Paper," Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html, Mar. 2020.

[2] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, "5G: A Tutorial Overview of Standards, Trials, Challenges, Deployment, and Practice," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1201–1220, Jun. 2017.

[3] W. Saad, M. Bennis, and M. Chen, "A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems," *IEEE Network*, Early Access, pp. 1–9, Oct. 2019.

[4] A. Gupta and R. Jha, "A Survey of 5G Network: Architecture and Emerging Technologies," *IEEE Access*, vol. 3, pp. 1206–1232, Jul. 2015.

[5] M. Newman, "The Structure and Function of Complex Networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, May 2003.

[6] N. Vastardis and K. Yang, "Mobile Social Networks: Architectures, Social Properties, and Key Research Challenges," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1355–1370, 3rd Qtr. 2013.

[7] N. Kayastha, D. Niyato, P. Wang, and E. Hossain, "Applications, Architectures, and Protocol Design Issues for Mobile Social Networks: A Survey," *Proceedings of the IEEE*, vol. 99, no. 12, pp. 2130–2158, Dec. 2011.

[8] S. Wasserman and F. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press, UK, Nov. 1994.

[9] L. Jin, Y. Chen, T. Wang, P. Hui, and A. Vasilakos, "Understanding User Behavior in Online Social Networks: A Survey," *IEEE Communications Magazine*, vol. 51, no. 9, pp. 144–150, Sep. 2013.

[10] C. Chamley, A. Scaglione, and L. Li, "Models for the Diffusion of Beliefs in Social Networks," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 16–29, May 2018.

[11] Z. Han, D. Niyato, W. Saad, T. Basar, and A. Hjørungnes, *Game Theory in Wireless and Communication Networks: Theory, Models and Applications*, Cambridge University Press, UK, Jan. 2012.

[12] Z. Han and K. J. Ray Liu, *Resource Allocation for Wireless Networks: Basics, Techniques, and Applications*, Cambridge University Press, UK, Apr. 2008.

[13] J. M. Lasry, and P. L. Lions, "Mean Field Games," *Japanese Journal of Mathematics*, vol. 2, no. 1, pp. 229–260, Mar. 2007.

[14] A. Bensoussan, T. Huang, and M. Lauriere, "Mean Field Control and Mean Field Game Models with Several Populations," *arXiv Optimization and Control*, Available: https://arxiv.org/abs/1810.00783.

[15] R. A. Banez, H. Gao, L. Li, C. Yang, Z. Han, and H. V. Poor, "Belief and Opinion Evolution in Social Networks Based on a Multi-Population Mean Field Game Approach," *Proc. IEEE International Conference on Communications (ICC)*, Dublin, Ireland, Jun. 2020.

[16] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A Survey on Mobile Edge Networks: Convergence of Computing, Caching and Communications," *IEEE Access*, vol. 5, pp. 6757–6779, Mar. 2017.

[17] R. A. Banez, L. Li, C. Yang, L. Song, and Z. Han, "A Mean-Field-Type Game Approach to Computation Offloading in Mobile Edge Computing Networks," *Proc. IEEE International Conference on Communications (ICC)*, Shanghai, China, May 2019.

[18] N. M. Chowdhury and R. Boutaba, "Network Virtualization: State of the Art and Research Challenges," *IEEE Communications Magazine,* vol. 47, no. 7, pp. 20–26, Jun. 2009.

[19] A. Belbekkouche, M. Hasan, and A. Karmouch, "Resource Discovery and Allocation in Network Virtualization," *IEEE Communications Surveys & Tutorials,* vol. 14, no. 4, pp. 1114–1128, 4th Qtr, 2012.

[20] D. A. Gomes, L. Nurbekyan, and E. A. Pimentel, "Economic Models and Mean-field Game Theory," *30º Colóquio Brasileiro de Matemática*, Rio de Janeiro, Brazil, Jul. 2015.

[21] C. Yang, J. Li, P. Semasinghe, E. Hossain, S. Perlaza, and Z. Han, "Distributed Interference and Energy-Aware Power Control for Ultra-Dense D2D Networks: A Mean Field Game," *IEEE Transactions on Wireless Communications*, vol. 16, no. 2, pp. 1205–1217, Feb. 2017.

[22] Y. Xu, L. Li, Z. Zhang, K. Xue, and Z. Han, "A Discrete-Time Mean Field Game in Multi-UAV Wireless Communication Systems," *Proc. IEEE/CIC International Conference on Communications in China*, pp. 714–718, Beijing, China, Aug. 2018.

[23] J. Li, R. Bhattacharyya, S. Paul, S. Shakkottai, and V. Subramanian, "Incentivizing Sharing in Realtime D2D Streaming Networks: A Mean Field Game Perspective," *IEEE/ACM Transactions on Networking*, vol. 25, no. 1, pp. 3–17, Feb. 2017.

[24] S. Maghsudi and E. Hossain, "Distributed User Association in Energy Harvesting Dense Small Cell Networks: A Mean-Field Multi-Armed Bandit Approach," *IEEE Access*, vol. 5, pp. 3513–3523, Mar. 2017.

[25] L. Li, Y. Xu, Z. Zhang, J. Yin, W. Chen, and Z. Han, "A Prediction-Based Charging Policy and Interference Mitigation Approach in the Wireless Powered Internet of Things," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 2, pp. 439–451, Feb. 2019.

[26] M. E. Mkiramweni, C. Yang, and Z. Han, “Mean Field Games for 5G Ultra-Dense Networks: A Resource Management Perspective,” *Ultra-Dense Networks for 5G and Beyond: Modelling, Analysis and Design*, Wiley, Hoboken, NJ, editors: T. Q. Duong, X. Chu, and H. A Suraweera, 2019.

[27] R. Couillet, S. Perlaza, H. Tembine, and M. Debbah, “Electric Vehicles in the Smart Grid: A Mean Field Game Analysis,” *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 6, pp. 1086–1096, Jul. 2012.

[28] Y. Wang, F. Yu, H. Tang, and M. Huang, “A Mean Field Game Theoretic Approach for Security Enhancements in Mobile Ad Hoc Networks,” *IEEE Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1616–1627, Mar. 2014.

[29] X. Ge, H. Jia, Y. Zhong, Y. Xiao, Y. Li, and B. Vucetic, “Energy Efficient Optimization of Wireless-Powered 5G Full Duplex Cellular Networks: A Mean Field Game Approach,” *IEEE Transactions on Green Communications and Networking*, vol. 3, no. 2, pp. 455–467, Jun 2019.

[30] A. Madan, M. Cebrian, S. Moturu, K. Farrahi, and A. Pentland, “Sensing the ‘Health State’ of a Community,” *Pervasive Computing*, vol. 11, no. 4, pp. 36–45, Oct. 2012.

[31] R. Hegselmann and U. Krause, “Opinion Dynamics and Bounded Confidence Models, Analysis and Simulations,” *Journal of Artificial Societies and Social Simulation*, vol. 5, no. 3, pp. 1–33, Jun. 2002.

[32] G. Albi, L. Pareschi, and M. Zanella, “On the Optimal Control of Opinion Dynamics on Evolving Networks,” *System Modeling and Optimization*, Springer, vol. 494, pp. 58–67, Apr. 2017.

[33] N. Friedkin and E. Johnsen, “Social Influence Networks and Opinion Change,” *Advances in Group Processes*, vol. 16, no. 1, pp. 1–29, Jan. 1999.

[34] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Springer, NY, Jan. 2008.

[35] J. M. Schulte, *Adjoint Methods for Hamilton-Jacobi-Bellman Equations*, Diploma Thesis, University of Munster, Germany, Nov. 2010.

[36] J. S. Rosenthal, *A First Look at Rigorous Probability Theory, 2nd Edition*, World Scientific Publishing Co. Pte. Ltd., London, UK, 2006.

[37] M. H. DeGroot, "Reaching a Consensus," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, Oct. 1974.

[38] C. Castellano, S. Fortunato, and V. Loreto, "Statistical Physics of Social Dynamics," *Reviews of Modern Physics*, vol. 81, no. 2, pp. 591–646, May 2009.

[39] A. Proskurnikova and R. Tempo, "A Tutorial on Modeling and Analysis of Dynamic Social Networks, Part I," *Annual Reviews in Control*, vol. 43, no. 1, pp. 65–79, Mar. 2017.

[40] H. Noorazar, K. Vixie, A. Talebanpour, and Y. Hu, "From Classical to Modern Opinion Dynamics," *Preprints*, Feb. 2020.

[41] A. Nordio, A. Tarable, C. F. Chiasserini, and E. Leonardi, "Belief Dynamics in Social Networks: A Fluid-Based Analysis," *IEEE Transactions on Network Science and Engineering*, vol. 5, no. 4, pp. 276–287, Oct.–Dec. 2018.

[42] D. Bauso, H. Tembine, and T. Basar, "Opinion Dynamics in Social Networks Through Mean Field Games," *SIAM Journal of Control and Optimization*, vol. 54, no. 6, pp. 3225–3257, Dec. 2016.

[43] H. Tembine, D. Bauso, and T. Basar, "Robust Linear Quadratic Mean-Field Games in Crowd-Seeking Social Networks," *52nd IEEE Conference on Decision and Control*, pp. 3134-3139, Florence, Italy, Dec. 2013.

[44] L. X. Wang and J. M. Mendel, "Fuzzy Opinion Networks: A Mathematical Framework for the Evolution of Opinions and Their Uncertainties Across Social Networks," *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 4, pp. 880–905, Aug. 2016.

[45] S. E. Parsegov, A. V. Proskurnikov, R. Tempo, and N. E. Friedkin, "Novel Multidimensional Models of Opinion Dynamics in Social Networks," *IEEE Transactions on Automatic Control*, vol. 62, no. 5, pp. 2270–2285, May 2017.

[46] Z. Bu, H. J. Li, C. Zhang, J. Cao, A. Li, and Y. Shi, "Graph K-Means Based on Leader Identification, Dynamic Game and Opinion Dynamics," *IEEE Transactions on Knowledge and Data Engineering*, Early Access, 2019.

[47] L. Gyarmati and T. A. Trinh, "Measuring User Behavior in Online Social Networks," *IEEE Network*, vol. 24, no. 5, pp. 26–31, Sep.–Oct. 2010.

[48] X. Chen, "Decentralized Computation Offloading Game for Mobile Cloud Computing," *IEEE Transactions on Distributed and Parallel Computing*, vol. 26, no. 4, pp. 974–983, Apr. 2015.

[49] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman, "Cloud-Vision: Real-Time Face Recognition Using a Mobile-Cloudlet-Cloud Acceleration Architecture", *Proc. IEEE Symposium on Computers and Communications*, pp. 59–66, Jul. 2012.

[50] M. Greer, *Electricity Cost Modeling*, Elsevier Inc., 2010.

[51] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. Wu, "Energy-Optimal Mobile Cloud Computing Under Stochastic Wireless Channel," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.

[52] S. Samarakoon, M. Bennis, W. Saad, M. Debbah, and M. Latva-aho, "Ultra Dense Small Cell Networks: Turning Density Into Energy Efficiency," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1267–1280, May 2016.

[53] B. Jovanovic, "Selection and the Evolution of Industry," *Econometrica*, vol. 50, no. 3, pp. 649–670, May 1982.

[54] B. Jovanovic and R. W. Rosenthal, "Anonymous Sequential Games," *Journal of Mathematical Economics*, vol. 17, no. 1, pp. 77-87, 1988.

[55] J. Bergin and D. Bernhardt, "Anonymous Sequential Games with Aggregate Uncertainty," *Journal of Mathematical Economics*, vol. 21, no. 6, pp. 543–562, 1992.

[56] B. Djehiche, A. Tcheukam, and H. Tembine, "Mean-Field-Type Games in Engineering," *AIMS Electronics and Electrical Engineering*, vol. 1, no. 1, pp 18–73, 2017.

[57] A. Tcheukam and H. Tembine, "Mean-Field-Type Games for Distributed Power Networks in Presence of Prosumers," *Proc. Chinese Control and Decision Conference (CCDC)*, pp. 446–451, Yinchuan, China, May 2016.

[58] A. Siwe and H. Tembine, "Network Security as Public Good: A Mean-Field-Type Game Theory Approach," *Proc. 13th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 601–606, Leipzig, Germany, Mar. 2016.

[59] B. Djehiche, A. Tcheukam, and H. Tembine, "A Mean-Field Game of Evacuation in Multilevel Building," *IEEE Transactions on Automatic Control*, vol. 62, no. 10, pp. 5154–5169, Oct. 2017.

[60] T. E. Duncan and H. Tembine, "Linear-Quadratic Mean-Field-Type Games: A Direct Approach," *Games*, vol. 9, no. 1, pp. 7–24, Feb. 2018.

[61] Y. Mao, J. Zhang, and K. Letaief, "Dynamic Computation Offloading for Mobile-Edge Computing with Energy Harvesting Devices," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[62] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.

[63] H. Guo and J. Liu, "Collaborative Computation Offloading for Multiaccess Edge Computing Over Fiber-Wireless Networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4514–4526, May 2018.

[64] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint Optimization of Radio and Computational Resources for Multicell Mobile-Edge Computing," *IEEE Transactions on Signal and Information Processing Over Networks*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[65] S. Guo, B. Xiao, Y. Yang, and Y. Yang, "Energy-Efficient Dynamic Offloading and Resource Scheduling in Mobile Cloud Computing," *Proc. IEEE INFOCOM*, San Francisco, CA, Apr. 2016.

[66] C. Wang, F. Yu, C. Liang, Q. Chen, and L.Tang, "Joint Computation Offloading and Interference Management in Wireless Cellular Networks with Mobile Edge Computing," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, 7432–7445, Aug. 2017.

[67] S. Yu, R. Langar, X. Fu, L. Wang, and Z. Han, "Computation Offloading With Data Caching Enhancement for Mobile Edge Computing," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11098–11112. Nov. 2018.

[68] C. You, K. Huang, and H. Chae, "Energy Efficient Mobile Cloud Computing Powered by Wireless Energy Transfer," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1757–1771, May 2016.

[69] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint Offloading and Computing Optimization in Wireless Powered Mobile-Edge Computing Systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[70] K. Zhang, Y. Mao, S. Leng, Q. Zhao, L. Li, X. Peng, L. Pan, S. Maharjan, and Y. Zhang, "Energy-Efficient Offloading for Mobile Edge Computing in 5G Heterogeneous Networks," *IEEE Access*, vol. 4, pp. 5896–5907, Aug. 2016.

[71] M. Chen and Y. Hao, "Task Offloading for Mobile Edge Computing in Software Defined Ultra-Dense Network," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 587–597, Mar. 2018.

[72] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-Edge Computing: Partial Computation Offloading Using Dynamic Voltage Scaling," *IEEE Transactions on Communications*, vol. 64. no. 10, pp. 4268–4282, Oct. 2016.

[73] Z. Ning, P. Dong, X. Kong, and F. Xia, "A Cooperative Partial Computation Offloading Scheme for Mobile Edge Computing Enabled Internet of Things," *IEEE Internet of Things Journal*, Early Access, Sep. 2018.

[74] J. Barreiro-Gomez, T. E. Duncan, and H. Tembine, "Linear-Quadratic Mean-Field-Type Games-Based Stochastic Model Predictive Control: A Microgrid Energy Storage Application," *Proc. 2019 American Control Conference (ACC)*, Philadelphia, PA, Jul. 2019.

[75] J. Gao and H. Tembine, "Distributed Mean-Field-Type Filter for Vehicle Tracking," *Proc. 2017 American Control Conference (ACC)*, Seattle, WA, May 2017.

[76] J. Barreiro-Gomez and H. Tembine, "Blockchain Token Economics: A Mean-Field-Type Game Perspective," *IEEE Access*, vol. 7, pp. 64603–64613, Jul. 2019.

[77] E. Weisstein, *Logistic Equation*, Available: http://mathworld.wolfram.com/Logistic Equation.html, Apr. 2017.

[78] A. Hastings, *Population Biology: Concepts and Models*, Springer, Dec. 1996.

[79] H. Freedman and P. Waltman, "Mathematical Analysis of Some Three-Species Food-Chain Models," *Mathematical Biosciences*, vol. 33, no. 3–4, pp. 257–276, 1977.

[80] R. Naji and A. Balasim, "Dynamical Behavior of a Three Species Food Chain Model with Beddington-DeAngelis Functional Response," *Science Direct Chaos, Solitons, and Fractals*, vol. 32, no. 1, pp. 1853–1866, Dec. 2005.

[81] P. Pang and M. Wang, "Strategy and Stationary Pattern in a Three-Species Predator-Prey Model," *Journal of Differential Equations*, vol. 200, no. 2, pp. 245–273, Jun. 2004.

[82] S. Boyd, *Basic Lyapunov Theory,*Available: https://stanford.edu/class/ee363/lectures/lyap.pdf, Apr. 2017.

[83] E. Routh, "A Treatise on the Stability of a Given State Motion: Particularly Steady Motion," *Macmillan*, 1877.

[84] H. Bergmann, "On the Conditions Under which an Equation has Roots with Negative Real Parts," *Selected Papers on Mathematical Trends in Control Theory*, pp. 70–82, New York, USA, 1964.

[85] E. Weisstein, *Stable Polynomial*,
Available: http://mathworld.wolfram.com/StablePolynomial.html, Apr. 2017.

[86] W. Press, B. Flannery, S. Teukolsky, and W. Vetterline, *Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd Edition*, Cambridge University Press, pp. 704–716, Sep. 1992.

[87] Y. Zhou, Y. Li, G. Sun, D. Jin, L. Su, and L. Zeng, "Game Theory Based Bandwidth Allocation Scheme for Network Virtualization," *Proc. IEEE Global Communications Conference*, Miami, FL, Dec. 2010.

[88] M. Seddiki and M. Frikha, "A Non-Cooperative Game Theory Model for Bandwidth Allocation in Network Virtualization," *Proc. 15th International Telecommunications Network Strategy and Planning Symposium*, Rome, Italy, Oct. 2012.

[89] W. Xie, J. Zhu, C. Huang, M. Luo, and W. Chou, "Network Virtualization with Dynamic Resource Pooling and Trading Mechanism," *Proc. IEEE Global Communications Conference*, Austin, TX, Dec. 2014.

[90] C. Liang and F. Yu, "Wireless Network Virtualization: A Survey, Some Research Issues and Challenges," *IEEE Communication Surveys & Tutorials*, vol. 17, no. 1, 1st Qtr. 2015.

[91] F. Fu and U. Kozat, "Stochastic Game for Wireless Network Virtualization," *IEEE/ACM Transactions on Networking*, vol. 21, no. 1, pp. 84–97, Feb. 2013.

[92] B. Liu and H. Tian, "A Bankcruptcy-Game Based Resource Allocation Approach Among Virtual Mobile Operators," *IEEE Communications Letter*, vol. 17, no. 7, pp. 1420–1423, Jul. 2013.

[93] J. Guo, F. Liu, J. Lui, and H. Jin, "Falloc: Fair Network Bandwidth Allocation in IaaS Datacenters via a Bargaining Game Approach," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 873–886, Apr. 2016.

[94] F. Fu and U. Kozat, "Wireless Network Virtualization as a Sequential Auction Game," *Proc. IEEE INFOCOM*, San Diego, CA, Mar. 2010.

[95] X. Lv, A. Xiong, S. Zhang, and X. Qiu, "VCG-Based Bandwidth Allocation Scheme for Network Virtualization," *Proc. IEEE Symposium on Computers and Communications*, Cappadocia, Turkey, Jul. 2012.

[96] B. Han, V. Gopalakrishnan, L. Ki, and S. Lee, "Network Function Virtualization: Challenges and Opportunities for Innovations," *IEEE Communications Magazine*, vol. 53, no. 2, pp. 90–97, Feb. 2015.

[97] M. Yoon and A. Kamal, "NFV Resource Allocation Using Mixed Queuing Network Model," *Proc. IEEE Global Communications Conference*, Washington D.C., Dec. 2016.

[98] Q. Zhang, Y. Xiao, F. Liu, J. Lui, J. Guo, and T. Wang, "Joint Optimization of Chain Placement and Request Scheduling for Network Function Virtualization," *Proc. IEEE International Conference on Distributed Computing Systems*, Atlanta, GA, Jun. 2017.

[99] X. Fei, F. Liu, H. Lu, and H. Jin, "Towards Load-Balanced VNF Assignment in Geo-distributed NFV Infrastructure," *Proc. IEEE/ACM International Symposium on Quality of Service*, Vilanova i la Geltru, Spain, Jun. 2017.

[100] H. Nguyen, Y. Zhang, Z. Chang, and Z. Han, "Parallel and Distributed Resource Allocation With Minimum Traffic Disruption for Network Virtualization," *IEEE Transactions on Communications*, vol. 65, no. 3, pp. 1162–1175, Mar. 2017.

[101] D. Nguyen, Y. Zhang, and Z. Han, "A Contract-Theoretic Approach to Spectrum Resource Allocation in Wireless Virtualization," *Proc. IEEE Global Communications Conference*, Washington D.C., Dec. 2016.

[102] Y. Wei, J. Wang, C. Wang, and X. Hu, "Bandwidth Allocation in Virtual Network Based on Traffic Prediction," *Proc. 6th International Conference on Wireless Communications Networking and Mobile Computing*, Qinhuangdao, China, Sep. 2010.

[103] E. Datsika, A. Antonopoulos, N. Zorba, and C. Verikoukis, "Matching Game Based Virtualization in Shared LTE-A Networks," *Proc. IEEE Global Communications Conference*, Washington D.C., Dec. 2016.

[104] Y. Gu, W. Saad, M. Bennis, M. Debbah, and Z. Han, "Matching Theory for Future Wireless Networks: Fundamentals and Applications," *IEEE Communications Magazine*, vol. 53, no. 5, pp. 52–59, May 2015.

[105] R. Mijumbi, J. Gorricho, J. Serrat, M. Claeys, F. Turck, and S. Latre, "Design and Evaluation of Learning Algorithms for Dynamic Resource Management in Virtual Networks," *Proc. IEEE Network Operations and Management Symposium*, Krakow, Poland, May 2014.

[106] R. Shi, J. Zhang, W. Chu, Q. Bao, X. Jin, C. Gong, Q. Zhu, C. Yu, and S. Rosenberg, "MDP and Machine Learning-Based Cost-Optimization of Dynamic Resource Allocation

for Network Function Virtualization," *Proc. IEEE Conference on Services Computing*, pp. 65–73, New York City, Jul. 2015.

[107] J. Asmuss and G. Lauks, "A Fuzzy Logic Based Approach to Bandwidth Allocation in Network Virtualization," *Proc. Science and Information Conference*, pp. 507–513, London, UK, Oct. 2013.

[108] L. Gupta, R. Jain, and G. Vaszkun, "Survey of Important Issues in UAV Communication Networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, Nov. 2015.

[109] M. Mozaffari, W. Saad, M. Bennis, Y. H. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Communications Surveys & Tutorials*, Early Access, Mar. 2019.

[110] S. Hayat, E. Yanmaz, and R. Muzaffar, "Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2624–2661, 4th Qtr. 2016.

[111] H. Shakhatreh, A. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges," *IEEE Access*, vol. 7, no. 1, pp. 48572–48634, Apr. 2019.

[112] J. Moon and T. Baser, "Linear-Quadratic Mean Field Stackelberg Differential Games," *Automatica*, vol. 97, no. 1, pp. 200–213, Nov. 2018.

[113] R. Carmona and X. Zhu, "A Probabilistic Approach to Mean Field Games with Major and Minor Players," *The Annals of Applied Probability*, vol. 26, no. 3, pp. 1535–1580, Jun. 2016.

[114] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, Jun. 2015.

[115] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, "A Survey on 5G Networks for the Internet of Things: Communication Technologies and Challenges," *IEEE Access*, vol. 6, pp. 3619–3647, Dec. 2017.