Title
Cooperation Incentives for Wireless Networks

Author
Wu, Chuchu

Publication Date
2015

University of California

Los Angeles

Cooperation Incentives for Wireless Networks

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Computer Science

by

Chuchu Wu

2015

ABSTRACT OF THE DISSERTATION

Cooperation Incentives for Wireless Networks

by

Chuchu Wu

Doctor of Philosophy in Computer Science
University of California, Los Angeles, 2015
Professor Mario Gerla, Chair

A large set of protocols for wireless networks require cooperation of the nodes. However, cooperation comes with costs of the contributors without benefit at the same time. Selfish peers may choose to avoid contributing while still expect other peers to serve for them, i.e. choose to be free-riders. Given that selfish behavior seriously degrades system performance, in order to drive selfish peers to cooperate, my work focuses at designing a set of incentive compatible protocols for wireless networks.

In this dissertation, incentive compatible protocols are designed and analyzed for the following scenarios: (1) in mobile ad hoc networks where network coding is applied, to drive selfish intermediate nodes to perform expensive secure network coding and forward packets with redundancy, a social norm based reputation system with fully distributed reputation management is proposed and analyzed; (2) for LTE content distribution in vehicular ad hoc networks, we propose a cluster-based scheme to save LTE bandwidth, improve content download efficiency, and a key-management scheme to incentivize peers to serve as cluster heads. We also investigate related issues on video congestion control, i.e. a priority based queuing scheme to maintain high video quality under congestion.

The dissertation of Chuchu Wu is approved.

Mihaela van der Schaar

Mani Srivastava

Demetri Terzopoulos

Mario Gerla, Committee Chair

University of California, Los Angeles
2015

To my family and friends

Table of Contents

List of Figures

List of Tables

ACKNOWLEDGMENTS

I wish to express my sincere thanks to those who supported and helped me during my Ph.D. study.

First of all, I would like to express my immense gratitude to my advisor Prof. Mario Gerla for his continuous support of my Ph.D study and research. His insight, expertise and valuable advice guides me through the whole process of Ph.D. study. Without his guidance and encouragement this thesis would never have materialized.

I would like to thank Prof. Mihaela van der Schaar, who gave excellent lectures on game theory and network economics, the course that greatly interested me and led me to the research in this field. Her suggestions and advices on my research are also greatly appreciated.

I would also like to thank Prof. Mani Srivastava and Prof. Demetri Terzopoulos for serving on my doctoral committee and providing valuable advice on improving my research work.

My appreciation also goes to Prof. Nadia Boukehatem, Prof. Eduardo Cerqueira and Prof. Nicholoas Mastronarde for their help and suggestions on my research.

I would like to thank the people I worked with at Cisco and Teradata for sharing their knowledge and guiding me on my work. In particular, I would like to thank Xiaoqing Zhu, Toerless Eckert, Wei Tang and Kuorong Chiang.

I would like to thank my colleagues at network research lab of UCLA: You Lu, Yu-Ting Yu, Xiao Li, Jorge Mena, Sungwon Yang, Ruolin Fan, Tuan Le, Vince Rabsatt, Pengyuan Du and many others. It has been a great pleasure of mine to have these nice people around.

I would like to thank my fiance Chenfanfu Jiang. He has been a good example, a great companion and the best friend of mine.

Most of all, I would like to thank my parents for their love and support. I could have never gone this far without their encouragement and help.

VITA

09/2006-07/2010	University of Science and Technology of China B.S. in Electronic Engineering and Information Science
09/2010-06/2012	University of California, Los Angeles, California M.S. in Computer Science
09/2011-06/2015	University of California, Los Angeles, California Teaching Fellow at Computer Science Department
08/2011-09/2011	Research intern at Telecom Paris Institute of Technology, Paris
06/2012-09/2012	Research Intern at Cisco Systems Inc., San Jose, CA
06/2013-09/2013	Research Intern at Cisco Systems Inc., San Jose, CA
06/2014-08/2014	Research Intern at Teradata Corporation, El Segundo, CA

PUBLICATIONS

Wei Hu, Kaiping Xue, Peilin Hong, Chuchu Wu, “ATCS: A Novel Anonymous and Traceable Communication Scheme for Vehicular Ad Hoc Networks”, International Journal of Network Security, Vol.13, No.2, PP.71-78, Sep. 2011

Chuchu Wu, Mario Gerla, Mihaela van der Schaar, “Social Norm Incentives for Secure Network Coding in MANETs”, International Symposium on Network Coding (NetCod), PP.179-184, Jun. 2012

Mario Gerla, Chuchu Wu, Giovanni Pau, Xiaoqing Zhu, “Content Distribution in VANETs”, Elsevier Journal on Vehicular Communications, Vol.1 Iss.1, PP.3-12 Jan. 2014

Chuchu Wu, Mario Gerla, “Game Theoretic Model for Cluster-based Content Distribution in Vehicular Networks”, The Fifth Nordic Workshop on Systems and Networks Optimization for Wireless (SNOW), Are, Sweden, Apr. 2014

Kuorong Chiang, Chuchu Wu, Wei Tang, “Caching Methods and A System for Entropy-based Cardinality Estimation”, U.S. Patent, No. 14/581,797, filed on Dec. 23, 2014

Chuchu Wu, Mario Gerla, Nicholas Mastronarde, “Incentive Driven LTE Content Distribution in VANETs”, The 14th IFIP Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), Jun. 2015

Chuchu Wu, Mario Gerla, Mihaela van der Schaar, “Social Norm Incentives for Secure Network Coding in MANETs”, submitted to IEEE Transactions on Networking, 2015

CHAPTER 1

Introduction

1.1 Background

As the internet access is becoming ubiquitous nowadays, the WiFi access is still limited to mobile users. Given limited LTE network bandwidth resources, when many users download content through the LTE network, congestions will occur, resulting in performance degradation. The situation is getting worse with the increasing demands for high quality video experiences. Various congestion control solutions have been studied, however most traditional congestion control solutions require the senders to reduce the data rate or routers to drop packets. Given massive amount of traffic and limited bandwidth capacity, there was no perfect solution that reduces the congestion without degrading the video quality.

Meanwhile, device-to-device (D2D) communication technologies become available and are focused by more and more researchers. D2D communication refers to the communication between devices directly without the help of infrastructures, as the way that ad hoc networks work. D2D includes not only the communications between cellphones or laptops, but also machine-to-machine (M2M), vehicle-to-vehicle (V2V) communications.

Under this circumstance, one promising solution of mitigating the congestions of LTE network is to offload the traffic from LTE networks to D2D networks, which often refers to wireless ad hoc networks. Specifically, some users will download content from the wireless peers instead of directly from the LTE infrastructure, when they have mutual interests on the content. It thus forms a local peer-to-peer network in the wireless community, illustrated as Fig. 1.1.

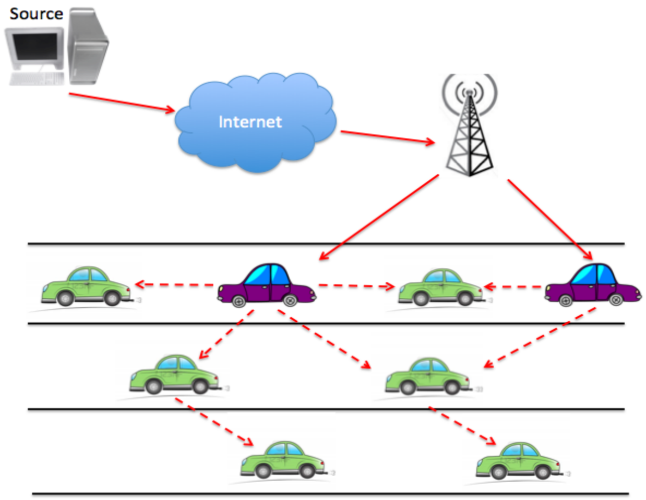

Figure 1.1: The system architecture for offloading traffic from LTE network to wireless peer-to-peer

1.2 Motivation

Offloading the traffic from D2I (device-to-infrastructure) to D2D networks can significantly lighten the burden of LTE networks thus reduce congestions. However, There is clear imbalance of interests in such systems, i.e., some users benefit at the costs of some others. Particularly, there are two forms of imbalance of interests: (1) the downloaders consume LTE connection costs, while other peers receive the data for free; (2) the intermediate nodes relay the packets for others which cost energy but have no benefit for themselves. Such imbalance results in unfairness, and even worse, selfish users are motivated to decline service and free-ride, and finally cause the system to collapse.

There are generally two issues that motivates selfish behavior: (1) LTE connection is expensive; (2) portable devices, such as cellphones and laptops, have energy concerns. As a consequence of the first issue, selfish peers choose to avoid downloading via LTE networks; as a consequence of the second issue, selfish peers choose to decline forwarding. In wireless networks, especially mobile networks, where packet losses due to disruption and interference are serious, network coding [ACL00][KM03][RLM03] can significantly improve packet delivery. However, secure network coding schemes, e.g. homomorphic hash/signature [LGK11][YWR08], require heavy computational cost, thus significant energy cost of the intermediate relay nodes. Hence, avoiding performing network coding becomes another important type of misbehavior of selfish peers, along with avoiding downloading from LTE network and declining forwarding for others.

To sum up, the offloading solution requires the cooperation of wireless peers. The LTE costs and energy concerns promote uncooperative behavior. Hence incentive schemes are needed to enforce the cooperation.

1.3 Contribution

Generally speaking, in this dissertation, we propose a solution architecture that offloads the traffic of LTE network to D2D networks, and design two novel incentive schemes for cooperation enforcement of the above mentioned two misbehavior-prone scenarios. The first one is a social norm incentive that enforces relay nodes to forward packets for others and perform secure network coding when necessary. The second is a key management scheme that enforces the peers to contribute as LTE downloaders (referred as cluster heads in the subsequent content). We model the interactions of the peers using game theory, and systematically evaluate the approaches with game theoretic analysis as well as simulations.

The remainder of this dissertation is summarized in the following.

1.3.1 Chapter 2: Social Norm Incentives for Secure Network Coding in MANETs

Chapter 2 introduces a social norm incentive that drives selfish wireless peers to forward packets and perform secure network coding when needed. The social norm consists of a social strategy and a reputation system with reward and punishment connected with node behavior. Packet coding and forwarding is modeled and formalized as a repeated NC forwarding game. The conditions for the sustainability (or compliance) of the social norm are identified, and a sustainable social norm that maximizes the social utility is designed via selecting the optimal design parameters, including the social strategy, reputation threshold, reputation update frequency and the generation size of network coding. For this game, the impacts of packet loss rate, transmission patterns, fraction of malicious, altruistic and intermittent nodes on performance are evaluated, and their impacts on the decision of selecting the optimal social norm are discussed. Finally, we compare the performance of our protocol with the ideal cooperative scenario and non-cooperative scenario.

1.3.2 Chapter 3: Incentive Driven LTE Content Distribution in VANETs

Chapter 3 introduces a cluster-based content distribution scheme for vehicular networks that addresses the above mentioned issues. Vehicles with common interests form a cluster and take turns to be the cluster head that directly downloads data packets from the Internet and share with others in the cluster. With game theoretic tools, we show that the cluster-based download (or any other cooperative downloading schemes) would not work with fairness and efficiency without a scheduling/management scheme. We design a server-assisting key management scheme that on one hand enforces cooperation of selfish vehicles, and on the other hand ensures fairness, security and efficiency of cooperative downloading. The proposed scheme is analyzed and justified with game theoretic analysis, numerical analysis and simulations.

1.3.3 Chapter 4: Video Congestion Control for VANETs

Chapter 4 considers multiple interests of video flows per cluster, where congestion is more likely to occur than single flow scenarios. In this chapter, we propose a video congestion control solution and perform simulations on multiple flow downloading.

1.3.4 Chapter 5: Conclusion

Chapter 5 concludes the dissertation and includes a discussion about future research directions.

CHAPTER 2

Social Norm Incentives for Secure Network Coding in MANETs

The throughput of mobile ad hoc networks subject to disruption, loss, interference, and jamming can be significantly improved with the use of network coding. However, network coding implies extra work for forwarders. Selfish forwarders may prefer to simply forward packets without coding them because of the processing overhead introduced by network coding. This is especially true in secure network coding where the coded packets are protected from pollution attacks by processor intense homomorphic signatures. To drive selfish nodes to cooperate and encode the packets, this paper introduces social norm based incentives. The social norm consists of a social strategy and a reputation system with reward and punishment connected with node behavior. Packet coding and forwarding is modeled and formalized as a repeated NC forwarding game. The conditions for the sustainability (or compliance) of the social norm are identified, and a sustainable social norm that maximizes the social utility is designed via selecting the optimal design parameters, including the social strategy, reputation threshold, reputation update frequency and the generation size of network coding. For this game, the impacts of packet loss rate, transmission patterns, fraction of malicious, altruistic and intermittent nodes on performance are evaluated, and their impacts on the decision of selecting the optimal social norm are discussed. Finally, we compare the performance of our protocol with the ideal cooperative scenario and non-cooperative scenario.

2.1 Introduction

Mobile devices like smart phones and tablets are becoming increasingly powerful and capable to function not only as clients, but also as peers in a fully fledged ad hoc network. For instance, at a sports arena a spectator may capture a scene on video from a vantage point and peer to peer broadcast the video stream to other spectators with an obstructed view. Similarly, a mobile may propagate to neighbors in ad hoc mode a stream that it is downloading from the Internet via WiFi or 3G. The mobile devices, however, have energy constraints. Since forwarding other devices' packets provides no benefit to a mobile that is not an intended destination, rather, it consumes battery resources, a self-interested relay node chooses not to forward the packets. If every relay node drops others' packets, the video never gets delivered to friends several hops away. This selfish behavior, however, can backfire. When the selfish node transmits its own video file, it will be treated the same way, i.e. its file will be dropped. This behavior is known as "tit for tat" in cooperative P2P distribution protocols (e.g. bit torrent) and is actually a primitive type of incentive to induce rational peers to cooperate and serve each other.

A similar situation occurs when the video stream is network coded. Network coding [ACL00][KM03][RLM03] has been shown to improve streaming performance dramatically in disruptive wireless networks with random loss, jamming or external interference. When the video originator streams the data to one or more receivers in a peer to peer ad hoc network, where the links between intermediate nodes can be stressed with high loss rate, network coding can be used to generate redundant packets and offset the loss. Without coding, the linear forwarding scheme fails. It does not retransmit lost packets, due to the broadcast MAC used for multicast streaming. Hence network coding improves packet deliver rate and stream quality.

Network coding, however, is susceptible to pollution attacks. Upon receiving corrupted packets, the destination cannot decode and must throw away the entire generation. So network coding streams must be protected by special hash functions or signatures that maintain their properties through linear combinations. One such scheme is the homomorphic hash/signature [LGK11][YWR08], which however requires heavy processing overhead, up to 100 times the processing of conventional network coding [RLM03]. Due to the high cost of secure network coding, selfish intermediate nodes are more likely to refuse to perform coding in order to save power for their own future transmissions. Hence, an incentive scheme is needed to encourage intermediate nodes to perform secure network coding. Incentive schemes have been proposed for conventional mobile networks before, e.g. [SSQ08][XS12a][SBH03], etc. In this paper we extend those schemes to the network coding scenario.

To maintain the model analytically tractable, for the first part of this paper we assume a rather simple topology scenario, with unicast from a source to a destination via a single intermediate. The model can be extended to multi-hop and multi-cast scenarios. With the unicast session, the source node injects a network coded stream. The intermediate node may carry out network coding or may opt to simply forward, or even drop the packets, either because it cannot perform coding, or because it wants to avoid the network coding processing O/H and thus energy expenditure, an important consideration in mobile devices. The social norm based incentive scheme we propose prescribes the intermediate nodes to perform coding and inject redundant packets. If the receiver detects non compliance by the intermediate node (i.e. the intermediate node refuses to inject redundant coded packets in presence of loss), it will broadcast the deviating action to a scoped neighborhood and the intermediate node's reputation will be lowered. As a consequence, when a low reputation intermediate node in turn becomes source and sends its own stream, it will be punished; namely, the neighbors will refuse to code its packets according to social strategy. This "punishment" should be sufficient to persuade the nodes that are frequent video originators or receivers to encode the packets, if they can.

The above topology is extremely simple. However, the careful reader will notice that the method and the results of this simple topology can be extended to much more general and complex topologies with multiple intermediate hops, multiple paths and multiple receivers (i.e. broadcast), as discussed later in this paper.

Considering that not all smart phones may have the capability to compute/process homomorphic codes, there are three possible reasons for an intermediate node to refuse coding: it is not homomorphic code enabled, or; it is short of battery, or; it wants to save battery power for its own transmissions (for selfish reasons). In all cases the intermediate node gets punished by a low reputation ranking. However, in the first two cases the low ranking does not matter: the node cannot transmit its own packets in code nor can it decode incoming coded packets. In the third case, the punishment is effective; the node stands to suffer severe degradation when it becomes a transmitter or receiver and discovers that intermediate nodes will refuse to code its packets. It should be mentioned that if an intermediate node does not code, the performance may be degraded but the protection from pollution is still in place. In other words, other homomorphic capable nodes can still detect and discard the polluted packet, with no loss of the entire generation.

In summary, streaming in lossy ad hoc networks greatly benefits from network coding; network coding must be protected from pollution; pollution protection takes a heavy toll on processing overhead and battery consumption, inviting selfish behavior by intermediate nodes; this in turn defies network coding. Even worse, selfish nodes can drop the packets to save transmission costs, and thus cause more severe damage to network performance. It is thus apparent that to break this impasse and jump start network coding among coding capable nodes or at least forwarding among coding incapable nodes, an incentive/punishment mechanism is essential. This main purpose of this paper is to present an efficient incentive scheme for secure network coding.

2.2 Related Works

There are currently some related literatures [BSG06], [PJ08], [ZL08], [ME09a],[ME09b] using game theory as a tool to solve Network Coding problems, however those works are very different from ours in the following aspects:

- The nodes or players in their models are not truly "selfish". Even if the word "selfish" is often used to describe the nodes or players in these literatures, nodes are only selfish in the sense that they take the action to maximize its individual utility function, but the utility function is designable by protocol designers. So the nodes are in fact puppets performing the predefined optimized actions. However in our scheme, the utility functions represent the users' own benefits and costs which cannot be designed by protocol designers. Namely, nodes are truly selfish, and games here are used to model their behaviors, as opposed to an optimization method in these related works.
- The nodes in their model are obliged to perform Network Coding as a common bottom line. However in our scheme, nodes are completely selfish users who are able to decline Network Coding in order to avoid costs.
- Existing game theoretic research on network coding considers only inter-flow network coding but not intra-flow network coding. So the actions in their games are either bandwidth/rate allocation strategies [BSG06][PJ08] [ZL08] or route selections [ME09a],[ME09b]. Differently, we consider intra-flow network coding, so the actions in our game are decisions made by intermediate nodes on whether performing intra-flow NC or not, or even

Literatures	Application/Systems	Protocol type	Incentive type
[DD]	Online trading system	Trading	Reputation
[ZS11][BBL04][FFL04][ZS08]	P2P overlay network	File sharing	Reputation
[LLM01][AG04][PS10]	P2P overlay network	File sharing	Credit/Token
[SSQ08]	Conventional wireless networks	Forwarding	Pairwise strategies
[XS12a][SBH03][ZCY03][XS12b][ZZS11]	Conventional wireless networks	Forwarding	Credit/Token
[MJS06][JS07][CWZ11][AHD05][BBL04]	Conventional wireless networks	Forwarding	Reputation
[CZ10]	Wireless networks with NC	Forwarding	Credit/Token
[CBZ11]	Wireless networks with NC	Forwarding	Reputation

Table 2.1: Related works on incentive design for selfish users

dropping the packets.

Essentially, existing game theoretic research on network coding solve optimization problems among altruistic nodes instead of handling selfish behavior.

There are two ways to deal with selfish behavior: detect and block the selfish users, or incentivize selfish users to cooperate. Schemes like watchdog and pathrater [MGL00], as well as a reputation scheme [BL05], have been proposed to detect and isolate selfish users in wireless networks, however if selfish behavior becomes mainstream or pervasive, isolating selfish users cannot be the ultimate solution.

There are related works designing incentives for selfish users in different types of networks, which can be summarized as Table.2.1. The earliest incentive schemes come from economics and are designed for human networks where users are naturally selfish, e.g. online trading systems like eBay [DD]. Then incentives are introduced to Peer-to-Peer file sharing systems where selfish behavior of peers can seriously degrade the system performance, as shown in the table. Up to here, trading or sharing problems are dominantly modeled as simple binary action games, i.e. players have only two possible actions: $\{Serve, NotServe\}$ or $\{\text{Share}, \text{Not Share}\}$. Such relatively simple games are rigorously formalized and thoroughly analyzed, various types of incentives (bartering, token, reputation) are explored, and solutions are optimized with the consideration of many practical factors.

Similarly, wireless networks can also be seriously affected by selfish behaviors, e.g. dropping others' packets to save its own battery and bandwidth, and it can also be modeled as a simple binary action game for conventional wireless networks without Network Coding, i.e. nodes can only choose to either forward or drop the packets. Yet different from the sharing/trading systems where centralized credit/reputation schemes can be feasible, it is very hard to maintain a centralized control scheme in wireless ad hoc networks. A limited amount of work has been done to design incentives for conventional wireless networks. Among them, only [ZS11] aims at maximizing the overall utility of the network, whereas others each focus on a specific advantage, e.g. [ZZS11] focuses on designing a low cost credit system where virtual tokens are easily distributed and maintained; [XS12b] focuses on designing a scheme where secure signatures are ensured to be checked before forwarding; [BBL04] focuses on the robustness and false detection of the reputation system. Some of the above schemes use centralized authority to maintain the reputations/credits, and some propose fully distributed schemes.

Designing incentives for wireless networks with network coding is very different from the ones for conventional forwarding for three major reasons: first, since NC is used to cover the packet loss, it is not always necessary to perform NC when the network condition is good. Namely the necessity of encouraging NC is not constant but varies with the network condition; second, network coding provides the intermediate nodes one more choice of action and thus significantly increases the complexity of solving the problem of optimal protocol selection, i.e. the problem space is expanded. Since performing network coding requires even more efforts from the selfish forwarders, the need of incentive schemes for wireless networks with network coding is even more badly than conventional wireless networks. However so far very little work has been done to design incentives for networks with NC. The only schemes we know are a credit scheme [CZ10] and a reputation scheme[CBZ11] .

Comparing between the two major types of incentive schemes, reputation schemes are advantageous over credit or token schemes in the following aspects: first, the tokens are very difficult to implement. If tokens are simply cryptographic packets, they can be easily duplicated and cannot be used as currency. So the implementation of tokens needs special hardware imbedded techniques. Second, the transactions of token before each transmission introduce a large overhead and serious delay. Due to the the above reasons, we deploy reputation schemes instead of tokens.

The existing reputation scheme for network coding in wireless network [CBZ11] is far from complete due to the following limitations: first, the existing work does not have rigorous formalism for NC forwarding using games. Second, the incentive compatibility constraints are not considered in the existing work, i.e. the conditions under which the incentive scheme can be complied by selfish users, which is essential and important for any incentive design. Third, the reputation update rule is not optimized. Fourth, the existing work proposed a centralized reputation management scheme with the help of a central authority for wireless mess network, however such scheme is not feasible for pure MANETs in most cases.

Our incentive scheme is also based on reputation, however, is significantly different from the previous one in these aspects:

- for the first time it models the (secure) network coding and forwarding in wireless networks as an infinite and discounted repeated game and provides rigorous game theory formalism.
- it provides a set of social norm based reputation schemes which are specifically designed for mobile networks with selfish users to encourage network coding.

- it formalizes the social norm selection as an optimization problem with incentive compatible (sustainability) constraints, and solves the problem with both theoretic and numerical analysis.
- it analyzes the impacts of network parameters to our scheme including packet loss rate, the costs of forwarding and coding, stability of the network (persistence factor).
- it discusses some practical issues including the implementation details and costs of the distributed reputation scheme, transmission sessions with multiple hops, fraction of malicious and altruistic nodes in the network.
- it compares the performance with two simplified versions of social norm based schemes.

2.3 System Model

2.3.1 Network Model and Network Coding Scenario

We consider a wireless ad hoc network, where mobile devices can act as source, sink or relay nodes at different times or simultaneously in different sessions, as shown in Fig.2.1.

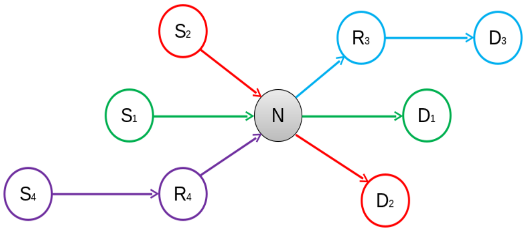

Figure 2.1: The same node N acts as a relay node in session 1 and 2, a source node in session 3 and a sink node in session 4.

Intra-flow network coding is used to increase transmission reliability on lossy wireless channels. The source generates a stream that is segmented generation by generation, where generation size $= K$ packets. The source then injects in the network K packets that are linear network coded, i.e. a linear random combination of the initial generation packets. Redundancy is required to overcome losses on link (R,D) with a packet loss rate denoted as p. When an intermediate node R receives K encoded packets from its predecessor node (in our case, source node S), it generates and forwards to the next hop (i.e. sink node D) n random linear coded packets[RLM03]. The ratio n/K (where $n >= K$) is defined as redundancy rate. If a secure NC scheme such as homormophic signature is used to protect packets from pollutions, the procedure is no different from regular random linear network coding except for the extra processing overhead.

The probability of delivering K packets with n transmissions on a faulty link follows a negative binomial distribution with parameter $1 - p$. So the expected number of transmissions $E(n)$ is $K/(1-p)$, which means the redundancy rate should be set to $1/(1-p)$ to achieve full delivery. If less than K linearly independent packets are received, the entire generation is lost because the receiver cannot decode. "Progressive coding" is used here to overcome this problem, where the intermediate node forwards the full generation without coding but only encodes the redundant packets. The advantage of progressive coding is a lower decoding delay and also the ability to receive some packets at the destination even if the full rank is never received. This scheme is clearly attractive if some of the intermediate nodes in the path are not capable to code. In particular, if the non coding node is in the critical section, the destination receives all (and only) the packets in the generation that survive link loss along the path.

To simplify the model, without loss of generality, we assume that the intermediate node knows a priori loss rate p and introduces redundancy accordingly, i.e. it injects $K/(1-p)$ packets, of which K are the original packets and the rest are coded packets.

The nodes in our model are mobiles with the following characteristics: having limited power resources, following selfish algorithms and being randomly matched as session origin, relay and destination. Here selfishness means taking an action to maximize its own utility. The possible actions of an intermediate node includes performing NC, simple forwarding or dropping the packets totally. Relatively high mobility forces the devices to have different neighbors throughout time. Hence, we use a global distributed reputation system (say, reputation reports are signed and periodically broadcast to a scoped neighborhood) to keep track of the reputation value of each node. This way, each node has a consistent view of the reputations of the other nodes. More detailed discussion on such distributed reputation maintenance system is presented in Section VII.

The model described above is rather simple (unicast session on a single path), yet it captures the worst damage a node R can cause by being not cooperative. In the more general case, a MANET with multiple paths is deployed from S to D. In this case the non cooperation of a node has a less severe effect than in the single path case because of path redundancy (recall, in our case a malicious node can only drop it cannot pollute). The worst case situation is the unicast session we discussed, with a single non cooperating node in the critical section. The single path model reflects the worst case situation over all possible topologies and routing schemes.

2.3.2 Game Setting

We model the unicast session in part A as a one-shot NC forwarding game $G = (\mathcal{N}, \mathcal{A}, u)$. $\mathcal{N}$ is the set of players in this game. There are two players: player 1 is the intermediate node R and player 2 is the source-destination (S-D) pair of the session. So $\mathcal{N} = \{R, S\text{-}D\}$. $\mathcal{A} = (A_R, A_{S\text{-}D})$ is the set of actions for each player. Intermediate node R, has three actions: (1) NC-forward (NC-F) $K/(1-p)$ encoded packets to sink node D; (2) simply forward (F) the original K packets to

D; (3) totally drop (Drop) the packets. So $A_R = \{\text{NC-F}, \text{F}, \text{Drop}\}$. The source-destination pair, has no actions, so $A_{S\text{-}D} = \emptyset$. The utilities of each player when an action is taken are shown in Table.2.2. We assume the S-D pair receives a constant benefit of B for each delievered packet. The intermediate node R has a constant cost of c_1 for encoding each packet, and a constant cost of c_2 for forwarding each packet. Here c_1 and c_2 mainly represent the power consumption of coding and forwarding.

	Intermediate node R		
	NC-F	F	Drop
S-D pair	B, $-(c_1 + c_2)/(1 - p)$	$B(1 - p)$,$-c_2$	0,0

Table 2.2: The utility matrix of one-shot NC forwarding game

As system designers, we encourage nodes to perform NC-F if and only if the overall utility of performing network coding is larger than simple forwarding; and obviously the utility of performing simple forwarding should be larger than dropping (which is 0). So the game itself must satisfy the condition in equ.(2.1).

$$B - (c_1 + c_2)\frac{K}{1 - p} > B(1 - p) - c_2 > 0 \tag{2.1}$$

Nodes in the network are assumed to be selfish strategic players who take the action to maximize their own utilities. If it is a one-shot game, there is a dominant action (or best choice) for intermediate node R, i.e., total drop (Drop). Therefore the network has zero throughput. Besides, one-shot game is not appropriate to model nodes that stay in the network for a relatively long time. So we model it as an infinitely repeated game with persistence factor δ. δ represents the probability of playing the game once again in the future. Namely, it reflects the likelihood that a node will stay in the system for the next time period. While some nodes are leaving the network, some new coming nodes arrive, forming a dynamic balance of the population. δ is decided by the nodes themselves and is not tunable by system designers. Having a low average δ indicates that nodes join and leave the network frequently. $\delta = 1$ indicates that all existing nodes stay in the network whereas no new nodes come. We will first assume δ is constant and given when selecting the optimal social norm. Later we will discuss how different δ affects our design.

As mentioned in the previous section, a relay node in one session may be a source/sink node in another session, and a current relay node will probably become a source/sink node in the future. We assume three nodes in the session are randomly matched and every node has the same chance of being a source, relay or sink node. So expectedly, in every three sessions, a certain node in the network will act as relay, source and sink respectively. Namely, we assume the nodes in the network are uniform, i.e., having the same chance of matching and sharing the same parameters of $\delta, B, c_1, c_2, p, K$, etc. Expectedly, in every three sessions, a certain node in the network will act as relay, source and sink respectively. So the long-term utility of a random node is,

$$v^{\infty} = \sum_{t=1}^{\infty} \delta^{t-1}(v_R^t + v_{SD}^t) \tag{2.2}$$

If every node has fixed neighbors and plays the repeated game with the same opponents over time, simply adopting a tit-for-tat strategy will be enough to enforce the nodes to perform network coding. However, mobiles usually change neighbors throughout the experiment, so they do not have knowledge of their new neighbor's behavior history. Under such circumstance, tagging each node with a reputation value is efficient to record its behavior in the past. By introducing the idea of social norm from economics to wireless networks, we design a novel incentive scheme to encourage nodes to perform network coding despite of the expensive power consumption.

2.3.3 Social Norm

The social norm, denoted by κ, is composed of a social strategy σ and a reputation scheme τ.

In our reputation system, every node in the network is tagged with a reputation θ, an integer from $\Theta = \{0, 1, ..., L\}$ for some L. A high θ means that the node performed well (e.g. coding and/or forwarding) in history, otherwise the node did not perform well in the past. The reputation system has the following structure: (1) every node maintains a table that stores all its neighbors' reputation values; (2) nodes broadcast deviating reports to a scoped neighborhood periodcally (3) each node updates the reputations of its neighbors according to the deviating reports received within the period.

σ is a reputation-based behavioral strategy represented by a mapping $\sigma : \Theta \times \Theta^2 \rightarrow$ A, where the first Θ is the reputation of the intermediate node R, and Θ^2 represents the reputations of the S-D pair. A = {NC-F, F, Drop}. So σ specifies the action $\sigma(\theta_R, \theta_S, \theta_D) \in$ A which node R with reputation θ_R should select when faced with a unicast session consisting of a source node with reputation θ_S and a sink node with reputation θ_D.

Generally, the social strategy rewards high reputation nodes by prescribing intermediate nodes to perform coding and forwarding for them; also, it punishes low reputation nodes by prescribing intermediate nodes to drop their packets. We restrict our attention to a set of threshold-based strategies Γ. Every strategy $\sigma \in \Gamma$ can be characterized by a threshold $k(\sigma) \in \{0, 1, ..., L\}$. Given any threshold k, there are still many choices of σ: (2.3)(2.4) are two examples. Among all possible strategies, σ_2 is the strictest (i.e. node R performs NC-F only if both source and sink nodes have the maximum reputation L). In this paper, we mainly use σ_1 as our proposed social strategy, where R performs NC-F if at least one of the source and sink has reputation of L. We will compare σ_1 with other alternatives and analyze the impacts of choosing different strategies in part A of SectionIV.

$$\sigma_1(\theta_R,\theta_S,\theta_D)=\begin{cases}\text{NC-F} & \text{if } \theta_R=L\cap\max\{\theta_S,\theta_D\}=L\\ & \cap\min\{\theta_S,\theta_D\}\geq k\\ \text{Drop} & \text{if } \min\{\theta_R,\theta_S,\theta_D\}<k\\ \text{F} & \text{Otherwise}\end{cases} \tag{2.3}$$

$$\sigma_2(\theta_R,\theta_S,\theta_D)=\begin{cases}\text{NC-F} & \text{if } \theta_R=L\cap\min\{\theta_S,\theta_D\}=L\\ \text{Drop} & \text{if } \min\{\theta_R,\theta_S,\theta_D\}<k\\ \text{F} & \text{Otherwise}\end{cases} \tag{2.4}$$

Consistent with our threshold-based social strategy σ_1, we propose a "multi-stage punishment reputation scheme", denoted as τ, shown in Fig.2.2, where α'_1, α'_2 are transition probabilities caused by approved behavior, and $\varepsilon'_{11}, \varepsilon'_{12}, \varepsilon'_{21}, \varepsilon'_{22}$ are transition probabilities caused by evil behavior. Specific interpretations can be found in Appendix. Reputations are updated periodically. If an intermediate node always complies with strategy σ_1 during the period, its reputation will increase by 1 until it reaches the maximum L; if it deviates from σ_1 for at least once within the period, its reputation will decrease to either k or 0, depending on its deviating action—if it drops packet totally when it should not, its reputation will decrease to 0; otherwise, only decreases to k. Here μ_1, μ_2, μ_3 respectively denote the proportion of nodes with a high ($\theta = L$), medium ($k \leq \theta < L$) and low ($0 \leq \theta < k$) reputation.

2.4 Game Optimization

2.4.1 Metrics

We design social norm based incentives to encourage nodes in the network to cooperate, aiming at maximizing the network throughput. Throughput here is

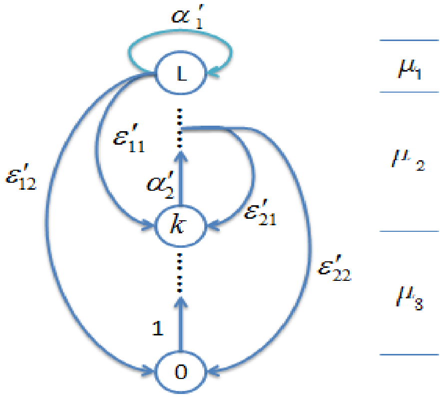

Figure 2.2: Multi-stage punishment reputation scheme τ

represented by all packets delivered in all sessions within each reputation update period. Taking the costs (the power consumptions) into account, we use social utility as a metric of the system. Social utility is defined to be the total utility of all nodes in the network after one period.

Besides social utility, another important metric for an incentive scheme is the obedience of the nodes, i.e. sustainability of the social norm. A social norm is sustained when selfish nodes choose to comply with the social strategy σ instead of deviating from it. Sustainability reflects the probability that a selfish node will choose to cooperate and help others. The more intermediate nodes perform coding, the higher throughput the network will achieve; so the sustainability of the incentive scheme also affects the throughput.

2.4.2 System Convergence

Given the social norm with social strategy σ_1 in (2.3) and reputation scheme τ in Fig.2.2, we can derive the unique corresponding stationary distribution of reputation.

Proposition 1 *Initially every node is assigned with the maximum reputation value L. When all nodes follow the previously defined social norm, the reputa-*

tion distribution of nodes in the network converges to a unique point $\{\eta(\theta)\}$

$$\begin{aligned}
\eta(L)^{-1} &= \frac{(1+\varepsilon'_{21}+\varepsilon'_{22}-k\varepsilon'_{21})(1-\alpha'_1)(1-{\alpha'_2}^{L-k})}{(1-\alpha'_2){\alpha'_2}^{L-k}} \\
&\quad + (k+1)\left(\frac{1-\alpha'_1}{{\alpha'_2}^{L-k}}-\varepsilon'_{11}\right)+1+\varepsilon'_{12} \\
\eta(\theta) &= \frac{1-\alpha'_1}{{\alpha'_2}^{L-\theta}}\eta(L),\ \textit{if}\ k \le \theta \le L-1 \\
\eta(\theta) &= \frac{1-\alpha'_1}{{\alpha'_2}^{L-k}}\eta(L)-\eta(0),\ \textit{if}\ 1 \le \theta \le k-1 \\
\eta(0) &= \varepsilon'_{11}\mu_1+\varepsilon'_{21}\mu_2 \\
\mu_1 &= \eta(L) \\
\mu_2 &= \sum_{\theta=k}^{L-1}\eta(\theta) = \frac{(1-\alpha'_1)(1-{\alpha'_2}^{L-k})}{(1-\alpha'_2){\alpha'_2}^{L-k}}\eta(L)
\end{aligned} \tag{2.5}$$

Where μ_1*,* μ_2 *and* $\mu_3 = 1-\mu_1-\mu_2$ *are respectively the proportions of nodes with a high, medium and low reputation at the steady state, as illustrated in Fig.2.2. Since we assume that the nodes in a unicast session are randomly matched, then each node has a probability of* μ_1 *to have* $\theta = L$*, a probability of* μ_2 *to have* $k \le \theta < L$ *and a probability of* μ_3 *to have* $\theta < k$*. In most cases,* μ_3 *is very close to zero.*

2.4.3 Social Utility at Steady State

In this subsection we determine the social utility at steady state for social norm κ with reputation distribution $\eta(\theta)$ obtained in proposition1. According to the social strategy σ_1, a session containing low reputation nodes (either S, D or R) always has a utility of 0, because low reputation nodes always drop others' packets and their own packets are also dropped by others as punishment. So we only consider the sessions where all nodes have reputations $\theta \ge k$. Let M be the number of sessions where a node acts as a relay node within one reputation update period. We use m_1 and m_2 to denote the number of NC and SF sessions for a high reputation intermediate node (an intermediate node with $\theta_R = L$). We use l to denote the number of SF sessions for a medium reputation intermediate node (an intermediate node with $\theta_R \in [k, L)$). Since nodes are considered uniform, the social utility, or the expected utility of any node in the network per session, is:

$$\begin{aligned} U &= \sum_{\theta} \eta(\theta) v(\theta) \\ &= \frac{1}{3}\mu_1 \times \left[\frac{m_1}{M}(B - \frac{c_1 + c_2}{q}) + \frac{m_2}{M}(Bq - c_2) \right] \\ &+ \frac{1}{3}\mu_2 \times \left[\frac{l}{M}(Bq - c_2) \right] \end{aligned} \tag{2.6}$$

Where,

$$\begin{aligned} m_1 &= M\left(\mu_1^2 + 2\mu_1\mu_2\right) \\ m_2 &= M\mu_2^2 \\ l &= M\left(\mu_1 + \mu_2\right)^2 \end{aligned} \tag{2.7}$$

We also extend the results to a multihop session with H hops, i.e., H intermediate nodes, expectedly there are $\mu_1 H$ high reputation intermediate node which can perform NC-F, and $\mu_2 H$ medium reputation intermediate node which can only do SF. In practice, the fraction of nodes with low reputation, i.e., μ_3, is very close to 0. In other words, we can take an approximation as $\mu_1 + \mu_2 = 1$. So the expected utility of a session with H hops is approximately:

$$\begin{aligned} U^H = &\frac{m_1}{M}\left(Bq^{\mu_1 H} - c_2\mu_2 H - \frac{c_1 + c_2}{q}\mu_1 H\right) \\ &+ \frac{m_2}{M}\left(Bq^H - c_2 H\right) \end{aligned} \tag{2.8}$$

2.4.4 Sustainability Conditions

In this subsection, we determine the sustainability conditions, under which all nodes in the network will rationally choose to comply with the social strategy σ_1 instead of deviating from it. Based on the assumption that all nodes in the network take the action to maximize their own long-term utility, we have the following lemma:

Lemma 1 *When a social norm* $\kappa = (\sigma, \tau)$ *is applied, if the long-term utility, shown in (2.2), of any node in the network by complying with the social strategy is no smaller than the long-term utility by deviating from the prescribed actions in the social strategy, social norm* κ *is sustainable, i.e., for any reputation* $\theta \in \Theta$ *and any selfish strategy* $\sigma' \neq \sigma$*, we must have:*

$$v_{\sigma}^{\infty}(\theta) \geq v_{\sigma'}^{\infty}(\theta) \tag{2.9}$$

where,

$$v^{\infty}(\theta) = E\left[\sum_{t=1}^{\infty} \delta^{t-1} v(\theta_t)\right] = v(\theta_{t=1}) + \delta \sum_{\theta'} p_{\tau}(\theta'|\theta) v^{\infty}(\theta') \tag{2.10}$$

Initially, all nodes have $\theta = L$. So we have $\mu_1 = 1$ and $\mu_2 = \mu_3 = 0$. According to (2.7), we have $m_1 = M$ and $m_2 = l = 0$. Here $l = 0$ is because there is no medium reputation node. In order to make sure all nodes comply with σ_1 initially, all parameters must satisfy the conditions in (2.11) and (2.12). If (2.11) is satisfied, nodes will have a larger (or equal) long-term utility by performing NC-F instead of F; if (2.12) is satisfied, nodes will have a larger (or equal) long-term utility by performing NC-F rather than Drop.

$$\left(B - \frac{c_1 + c_2}{q}\right) \sum_{t=1}^{\infty} \delta^{t-1} \geq B - c_2 + (Bq - c_2) \sum_{t=2}^{\infty} \delta^{t-1} \tag{2.11}$$

$$\left(B - \frac{c_1 + c_2}{q}\right) \sum_{t=1}^{\infty} \delta^{t-1} \geq B - 0 + 0 \cdot \sum_{t=2}^{\infty} \delta^{t-1} = B \tag{2.12}$$

Conclusion 1 *If discount factor* δ *satisfies the condition in (2.13), all nodes will perform NC as prescribed in* σ_1*, and the social norm will be sustained initially.*

$$\delta \geq \frac{c_1 + c_2 - qc_2}{qB(1 - q)} \tag{2.13}$$

At any time after the initial point until the steady state, nodes will always comply with the social strategy iff. the long-term utility will decrease if they deviate. By comparing the long-term utilities of all possible deviations motivated by selfish intention, i.e., NC-F → F, NC-F → Drop and F → Drop, we obtain the sustainability conditions in (2.14)(2.15)(2.16).

$$\frac{\delta}{1-\delta}\left[B - Bq - \left(\frac{\mu_1}{\mu_2}+2\right)\left(\frac{c_1+c_2}{q}-c_2\right)\right]\cdot\mu_1\mu_2 \geq \frac{c_1+c_2(1-q)}{q} \tag{2.14}$$

$$\frac{\delta}{1-\delta}\left[B\left(1+\frac{\mu_1}{\mu_2}\right)+Bq\left(\frac{\mu_2}{\mu_1}\right)-\left(2+\frac{\mu_1}{\mu_2}\right)\frac{c_1+c_2}{q}-c_2\frac{\mu_2}{\mu_1}\right]\cdot\mu_1\mu_2 \geq \frac{c_1+c_2}{q} \tag{2.15}$$

$$\frac{\delta}{1-\delta}\left[\frac{\mu_1}{\mu_2}B+Bq\left(\frac{\mu_2}{\mu_1}+2\right)-\left(\frac{\mu_1}{\mu_2}+2+\frac{\mu_2}{\mu_1}\right)c_2\right]\cdot\mu_1\mu_2 \geq c_2 \tag{2.16}$$

For steady state, all parameters must also satisfy the above three sustainability conditions.

(2.13)(2.14)(2.15)(2.16) can also be written as the form of $\delta \geq \delta_1, \delta \geq \delta_2, \delta \geq \delta_3, \delta \geq \delta_4$ or $\delta \geq \max\{\delta_1,\delta_2,\delta_3,\delta_4\}$. So, if $\max\{\delta_1,\delta_2,\delta_3,\delta_4\} < 1$, the sustainability conditions can always be satisfied if the actual δ of the nodes in the network is close enough to 1, which indicates that the nodes will stay in the system forever. By computing inequalities (2.14)(2.15)(2.16), we obtain a threshold for B in (2.17), above which we have $\max\{\delta_1,\delta_2,\delta_3,\delta_4\} < 1$.

Proposition 2 *If benefit B is large enough or at least satisfies:*

$$B \geq \frac{c_1+c_2-qc_2}{q(1-q)}\left(\frac{\mu_1}{\mu_2}+2\right) \tag{2.17}$$

there always exists a $\underline{\delta} \in [0,1]$, s.t. when $\delta \geq \underline{\delta}$, the incentive scheme can be sustained by all nodes in the network. Here,

$$\underline{\delta} = \max\{\delta_1,\delta_2,\delta_3,\delta_4\} \tag{2.18}$$

Since $\delta \in [0,1]$ is decided by the nodes themselves and not tunable, whether the social norm based incentive scheme can be sustained is greatly decided by the system itself. So we use the interval between $\underline{\delta}$ and 1 as the metric to reflect the sustainability of the scheme.

Definition 1 *The Sustainable Interval for an incentive scheme is defined as:* $SI = 1 - \underline{\delta}$. *In any game theoretic model based on discounted repeated games, SI can be used to represent the sustainability of the incentive scheme. The larger SI, the more compliant and better performing the scheme is.*

Nodes with discount factor δ falling in the interval (i.e., $\delta \geq \underline{\delta}$) will comply to the social strategy (do network coding/forwarding). A larger SI indicates that more nodes will cooperate and do network coding, so the network will achieve a larger throughput.

2.4.5 Optimization Problem Formalization

We formalize the parameter selection as an optimization problem of maximizing both social utility U in (2.6) and $SI = 1 - \underline{\delta}$ for any reputation distribution $\eta(\mu_1, \mu_2)$ from the initial state $\eta_0(1, 0)$ all the way to the steady state $\{\eta(\theta)\}$.

The parameters can be classified into two categories: one category reflects the nature of the network, and is given (not tunable by designers); the other reflects the characteristics of the social norm based scheme we propose, and can be tuned by the protocol designer.

In summary, network nature parameters include: packet loss rate p, benefit-cost ratio $B : c_1 : c_2$, discount factor δ, number of hops H; tunable parameters include: social strategy σ, reputation threshold k, number of sessions M within one reputation update period and NC generation size K.

2.5 Performance Evaluation

In this section, we evaluate the performance of the NC forwarding game using numerical techniques. We compute social utility U and sustainability SI results as a function of system variables and design parameters. In particular we define the optimal selection of reputation thresholds and other design parameters.

2.5.1 Selecting the Social Strategy σ

In part C of section II, we pointed out that there are many design choices for σ, and we presented two examples in (2.3)(2.4). Here we show that there exists a tradeoff between social utility and sustainability when designing the social strategy. Specifically, a stricter social strategy (e.g. a strategy that requires high reputation of both source and destination for coding at the intermediate node) will lead to a higher sustainability (i.e. higher compliance ratio) at the price of a lower social utility. We discover this tradeoff by comparing the social utility U and the sustainability interval SI of σ_1 in (2.3) and σ_2 in (2.4), where σ_2 is stricter than σ_1. We prove that $U_1 \geq U_2$ and $SI_1 \leq SI_2$.

This happens because in σ_2, the intermediate node performs network coding only when all the nodes in the session have maximum reputation of L. Naturally this harms the throughput since it reduces the chance of coding. On the other hand, it enforces the nodes to be more compliant in order to keep a high reputation and avoid severe utility degradation.

2.5.2 Selecting the Design Parameters L, k and M

In practice, there are very few nodes with reputation lower than k, mainly because the probability that all packets are lost is tiny. Also, malicious nodes that intentionally drop packets will be blacklisted and are avoided during routing. Hence the value of k has almost no effect on network performance. So we only consider the interval between L and k. Fig.2.3 and Fig.2.4 respectively show the numerical results of stationary social utilities U and SI under different settings of threshold interval $L-k$ and number of sessions in each period M.

According to Fig.2.3 and Fig.2.4, generally both U and SI are higher when $L-k$ is set small. On the contrary, when $L-k$ is set very large, U approaches the utility value that corresponds to all nodes only performing SF, whereas SI approaches 0. The intuition is that if $L-k$ is very large, medium reputation nodes have to keep performing well for many periods of time in order to reach L. During this long journey from k to L, they are still medium reputation nodes which only perform (and are only served with) SF, even worse they may leave the system before becoming a high reputation node. Therefore the network has a lower throughput and worse sustainability. Also, we prove mathematically that U is a strictly decreasing function of $L-k$.

Fig.2.3 and Fig.2.4 also show that U and SI increase when M gets smaller. This means that if the system updates or refreshes reputations frequently, i.e. there are fewer sessions in an update period, the network will have a higher throughput and better sustainability. Intuitively, if refreshing rate is higher, the nodes' reputations more accurately reflect their behavior history. And they will be rewarded or punished in time. We also prove mathematically that U is a strictly decreasing function of M.

2.5.3 Impact of Untunable System Variables on Performance

NC generation size K belongs to tunable parameters, however according to numerical tests, changing K has little impact on system performance. Intuitively, for both NC-F and SF, we mainly care about the proportion of packets delivered, so the value of K does not affect results.

In part D of section III, we have thoroughly discussed the impacts of discount

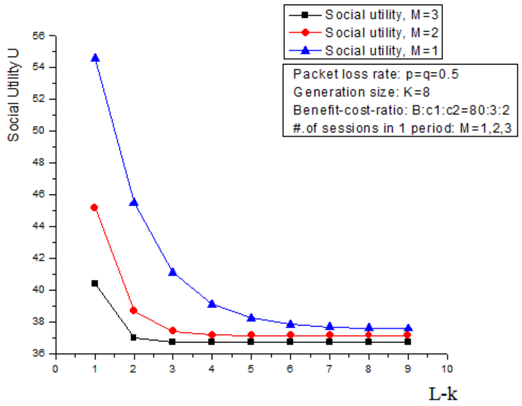

Figure 2.3: Social utilities at steady states (U) when selecting different threshold intervals ($L-k$) and number of sessions M in one reputation update period

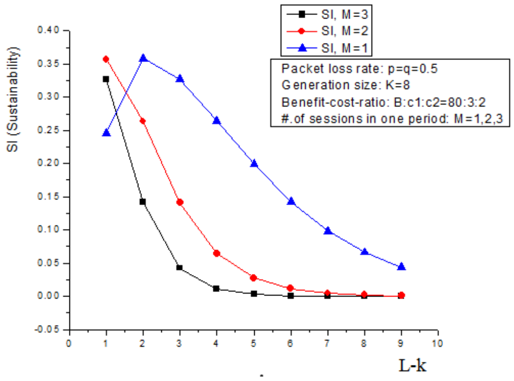

Figure 2.4: Sustainable interval (SI) when selecting different threshold intervals ($L-k$) and number of sessions M in one reputation update period

factor δ (i.e. the probability of a node staying in the system for the next period) on the sustainability of our scheme. The higher δ, the better sustainability. There exists a threshold $\underline{\delta}$, below which nodes will not comply or perform network coding, above which they will. This indicates that a node that stays in the system for a long time is more likely to help others than the node that will leave the system soon.

Also in part D of section III, we show that there exists a $\underline{B}$, below which the scheme can never be sustained. Benefit B is a relative parameter evaluated by each node itself. If an intermediate node highly values the benefit of its own future transmissions, it has more incentives to serve others and pursue a high reputation. If δ is too low, a high B is still insufficient to encourage nodes to do network coding, hence the throughput does not increase after B.

A low packet loss rate p, or a high packet deliver rate q, will lead to a higher throughput and social utility as expected. Fig.2.5 shows how the sustainability of the scheme varies as a function of packet deliver rate. When the packet loss rate is very low, the network condition is good, and nobody does expensive secure NC because the performance of NC is hardly different from SF. When the packet loss rate is very high and the network condition is very bad, nobody does expensive secure NC either, because even if they did, nobody notices and they will be punished anyway due to the severe packet loss. In realistic scenarios, the packet loss rate is generally well below 0.5. With this condition, we may conclude that our incentive scheme is more sustainable (i.e. more nodes are induced to use Network Coding) when packet loss increases, just as NC is beneficial when packet loss is serious.

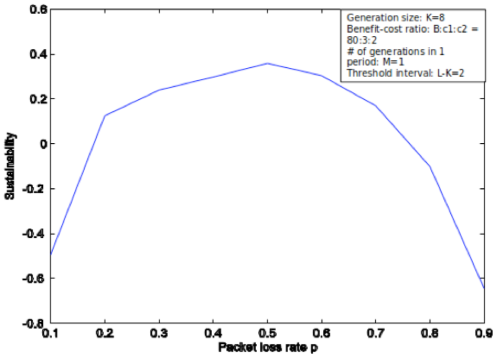

Figure 2.5: Sustainable interval (SI) of the incentive scheme under different packet loss rate p

2.6 Practical Issues

In this section, we consider some practical issues of applying the social norm based protocol to mobile ad hoc networks.

2.6.1 Distributed Reputation System

As mentioned before, centralized or partially centralized reputation management with a reputation administrator or multiple reputation managers is not feasible for wireless ad hoc networks. We adopt a fully distributed reputation management system where nodes disseminate reputation information to a scoped neighbors of limited hops periodically. Research has been done to minimize the cost of reputation data dissemination by encoding the reputation packets[ZS08], as well as to recognize and prevent fake reputation reports[ZZS11]. We try to optimize the reputation system by solving the following three major problems: (1) what's the simplest message format to broadcast and update reputation information; (2) what's the best reputation update frequency if taking the cost of reputation dissemination into account; (3) to what scope of neighbors do nodes disseminate the reputation information.

According to the reputation update rule τ explained in part C of section III, a node's new reputation is decided by its old reputation and its behavior in the last period. Initially nodes have the uniform reputation information: all nodes are tagged with initial reputation θ_0. If all nodes in the network have consistent reputation information at the beginning of period i, as long as they receive consistent information about the behavior of others in period i, and calculate the new reputations according to τ at the end of period i, then they will have consistent updated reputation information at the beginning of period $i+1$. Hence, disseminating reputation values and distinguishing the updated info from out-of-date info is not necessary and should be replaced by disseminating nodes' behavior instead. So problem (1) is reduced to: in what format do nodes broadcast the behavior of their neighbors.

Since reputations are updated only at the end of each period, nodes only need to broadcast a single message which is a summary of its neighbors' behaviors during the period. There are two possible types of deviating actions: perform Drop when it should have performed F or NCF (case 0) and perform F when it should have performed NCF (case 1). The deviating node's reputation should be decreased to 0 for case 0 and decreased to k for case 1. Let x denote deviating actions. If a node has j deviating actions within one period, the worst behavior is considered, i.e., $x = x_1 \cap x_2 \cap ... \cap x_j$. So the deviating action can be represented as a one-bit message: 0 indicating reputation down to 0 and 1 indicating down to k. If Node i has n_i deviating neighbors and their Node IDs are $\{d_{i0}, d_{i1}, ..., d_{i,n_i-1}\}$, its reputation update message should be in the format shown in Table.2.3. For a network graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$ with maximum degree $\bar{k}(\mathcal{G})$, if it has no more than 256 nodes ($|\mathcal{N}| \leq 256$), then the size of node ID is 1 byte, and the largest message has size of $9 * \bar{k}(\mathcal{G})$ bits. Even if $\mathcal{G}$ is one big cluster (i.e., $k(\mathcal{G}) \equiv |\mathcal{N}| - 1$), which is not quite possible for ad hoc networks but is the worst case with the most information redundancy, the message size is approximately $1/4$ of a regular packet size. The reputation information can also be combined as a header of regular video packets when the message size is very small.

NodeID	d_{i0}	d_{i1}	...	d_{i,n_i-1}
Behavior	1	0	...	1

Table 2.3: Reputation Update Message from Node i

If the nodes are static, i.e., every node has fixed neighbors, then it would be sufficient to broadcast reputation update messages within 2 hops. If the network structure and neighbors are changing all the time, broadcasting to farther neighborhood is necessary. Although the scope of reputation broadcasting greatly depends on the mobility of the nodes, however nodes can always broadcast queries and retrieve reputation information from others upon need, and neighbors are relatively fixed within each period due to high reputation update frequency. So, in order to save cost of message dissemination, the reputation message broadcasting can be restricted to 2 hops.

As concluded in part C of section IV, the more frequently reputations are updated, the better system performance can be achieved. However, updating reputation more frequently leads to higher message dissemination costs as well. Let c_3 denotes the cost of broadcasting one reputation update message to two-hop neighbors. If the ratio between the sizes of a reputation update message and a regular packet is f $(0 < f < 1)$, then $c_3 \leq (k(\mathcal{G}) + 1) * f * c_2$. The cost is estimated to be fairly insignificant comparing to regular packets. Even when $M = 1$, i.e., one update for each generation transmission, it would not flood the network. Besides, many routing protocols, such as OLSR, has routing information updates periodically. Hence, the reputation information can be attached to the routing information updates.

2.6.2 Networks with Altruistic, Malicious and Intermittent Nodes

So far we assumes that all the nodes are selfish and rational, i.e., choose the action that maximizes its own utility. Usually such users are called “reciprocative” users. Now we consider the following cases where some users are:

- altruistic, i.e., always perform NC and forward;
- malicious, i.e., drop packets intentionally and upload bogus packets.

Altruistic users themselves do not need to be incentivized, however the existence of altruistic users will encourage selfish behavior of other selfish nodes — some selfish behavior is not punished. Our social strategy prescribes that a relay should take actions corresponding to the reputation of source and destination nodes. In this sense, the altruistic users deviate from the prescribed strategy — they are supposed to drop the packets from low reputation nodes, but they choose to network code / forward the packets. In order to avoid the negative impacts of altruistic nodes, a modification for the reputation update rule should be enough, i.e., include the improper altruistic behavior as a type of deviating action, and punish the altruistic nodes by reducing their reputation.

Malicious nodes who intentionally drop packets will receive low reputation immediately according to our current reputation scheme. So neighbors will refuse to forward the packets coming from malicious nodes, which effectively stops the spread of bogus/spam information. Hence, our incentive scheme has a by-product of protecting the network from malicious attacks.

2.7 Conclusion

In this chapter, we present a social norm based incentive scheme for mobile ad hoc networks, where nodes are selfish, strategic users. For simplicity, we consider a single-path unicast session, where the intermediate node can rationally choose to do secure network coding and forward packets with redundancy (NCF), simply forward the original packets (F) or totally drop the packets (Drop). We model the unicast session as a repeated gift-giving game with persistence factor δ, where nodes are symmetric and randomly matched. Our incentive scheme consists of a social strategy σ and a reputation scheme τ, which reward cooperative nodes by increasing their reputation and promising better transmission quality in the future, and punish uncooperative nodes by decreasing their reputations and refusing coding/forwarding their packets in the future. Most existing incentive schemes only propose a single protocol and prove it can incentivize nodes to perform coding/forwarding. Differently from previous schemes, we propose a set of social norm based incentive schemes with parameters that we adjust to optimize the social utility and sustainability. We first prove that the reputation distribution will converge to a unique steady state, corresponding to a fixed setting of the parameters. Then we present a series of results for parameter selection that are supported by mathematical derivations as well as by numerical analysis: (1) there exists a tradeoff between the social utility/throughput and sustainability — a stricter social strategy will increase the sustainability but decrease the utility; (2) both utility and sustainability will be optimized if the reputation threshold k is set close to the maximum reputation L, and reputations are updated or refreshed frequently; (3) NC generation size does not affect the performance of our incentive scheme. Additionally we show how the network condition or the game itself impacts our incentive scheme. Moreover, practical issues for the proposed reputation system are discussed, including the distributed reputation system, as well as the existence of altruistic and malicious users.

CHAPTER 3

Incentive Driven LTE Content Distribution in VANETs

Content downloading in vehicular networks is increasingly popular and demands for better performance due to existing difficulties and challenges including: limited RSUs available on the road, expensive and limited 3G/LTE resource, etc. By introducing the idea of peer-to-peer content delivery to vehicular networks, cooperative schemes, such as SPAWN [NDP05], significantly improve the efficiency of content downloading in vehicular networks, where the cooperation among vehicles is essential to establish such systems. Hence, uncooperative behavior of peers can severely degrade the system performance, e.g. some vehicles may avoid downloading original data chunks through LTE network in order to save the cost of LTE connection, and only wait for other peers to share with him, i.e. choose to be a free-rider. On the other hand, if too many nodes connect to the 3G/LTE, the 3G/LTE network may encounter serious congestion, and it would be a waste of network resource if many of them are actually downloading the same content. Security is another concern since overhear is an issue whereas peers would prefer to only share content with other contributive peers.

In this paper, we propose a cluster-based content distribution scheme for vehicular networks that addresses the above mentioned issues. Vehicles with common interests form a cluster and take turns to be the cluster head that directly downloads data packets from the Internet and share with others in the cluster. With game theoretic tools, we show that the cluster-based download (or any other cooperative downloading schemes) would not work with fairness and efficiency without a scheduling/management scheme. We design a server-assisting key management scheme that on one hand enforces cooperation of selfish vehicles, and on the other hand ensures fairness, security and efficiency of cooperative downloading. The proposed scheme is analyzed and justified with game theoretic analysis, numerical analysis and simulations.

3.1 Introduction

In the near future, the proliferation of “smart vehicles” is envisaged to become a reality. Smart vehicle on board processors combine the characteristics of both smartphones and laptops: they are as “mobile” as smart phones, i.e., having ubiquitous access to the Internet via 3G/LTE networks, but more powerful in computation and energy supply like laptop computers. Besides safe-driving services, content downloading is expected to be widely popular among drivers and passengers on board just like in traditional networks. Since the number of road side units is very limited, the time when vehicles can get access to Wi-Fi is very short. For most of the time, vehicles must rely on 3G/LTE to download Internet content. However, mobile data customers are increasing, and the amount of data requested are getting larger and larger, due to increasing demands for higher quality of video experience. Consequently, the 3G/LTE resources are getting more and more scarce and thus expensive, resulting in longer delays and higher costs of content downloading.

Often in vehicles at the same location moving towards the same direction, people onboard are very likely to be interested in the same video streaming content. One possible scenario is a certain entertainment mobile application that provides limited options of movies to enjoy on the road. As inter-vehicle communication technology becomes available, cooperative content downloading schemes

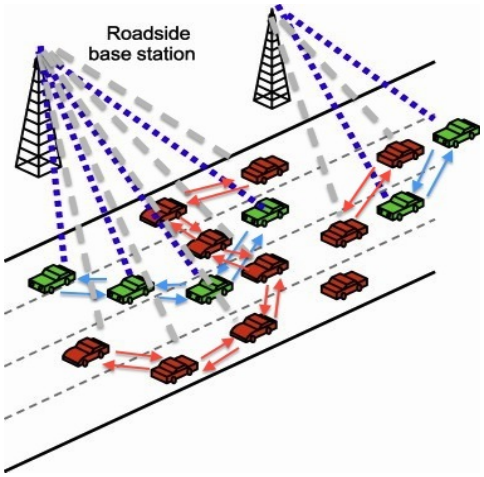

Figure 3.1: Cooperative peer-to-peer content downloading in VANETs: vehicles of common interest (same color in the figure) form a cluster

are proposed to leverage this advantage and bring peer-to-peer overlay network to the vehicular environment. An early example is SPAWN [NDP05], a peer-to-peer content delivery mechanism that utilizes parallel download across a mesh of co-operating peers. Given the limited number of available RSUs, vehicles with common interest download data from the Internet through cellular base stations, instead of RSUs, and share with each other in ad hoc mode, as shown in Fig. 3.1. Such cooperative downloading scheme significantly improves the efficiency of content downloading and can also reduce delays.

Different from traditional peer-to-peer file sharing systems (like Bit-Torrent) where for each peer, downloading is beneficial and uploading is costly, in vehicular networks, downloading content via 3G/LTE is expensive and inter-vehicle sharing is just about free. So instead of avoiding the sharing of files with peers, selfish nodes in vehicular networks will try to avoid downloading original chunks of content, because of the 3G costs. To enforce cooperation and fairness among selfish peers, incentive and scheduling are needed. In our system, every vehicle is connected to LTE network, so it behooves us to utilize a server assisted incentive scheme, which is much easier to implement and maintain than a fully distributed scheme. Hence we propose a server assisted key management scheme, where the server maintains the peer organization centrally, but the data content is distributed through low-cost V2V transmissions.

3.2 Related Works

Most existing approaches that improve the efficiency of content downloading in vehicular networks, such as [FB09, GWP14, JH12, LYL11, LCC13, MCC12b, NDP05, WSH12, ZY11], rely on the cooperation among vehicles. However, misbehavior among the vehicle community motivated by selfish intentions of saving 3G/LTE bandwidth or energy can severely degrade network performance. Hence mechanisms are needed to enforce the cooperation among nodes in vehicular networks. Incentive schemes, such as [CZW11, HHH08, LW09, MPX13, WC07, WGS12, XV13], etc., are proposed to enforce the cooperation of intermediate nodes to forward packets in VANETs or other wireless networks. Generally, incentive schemes can be categorized into two types, i.e., credit based schemes and reputation based schemes. Credit based schemes are adopted by [CZW11, LW09, MPX13, WC07, XV13] where monetary rewards/credits/tokens are allocated/paid to the forwarders. A reputation based scheme is adopted in [WGS12], where users' behavior is recorded by reputation, and reward/punishment is given according to the reputation score. A three-counter scheme was proposed in [HHH08] where nodes count the number of packets requested to be forwarded and the number of packets actually requested as a way of observing and recording the behavior of neighbor nodes. This scheme is fairly similar to the way how reputation schemes work. Table 3.1 summarizes the related work according to the application scenarios, the use of incentives and the type of incentive schemes used.

Power supply in vehicles is not a critical issue, hence nodes in VANETs usually do not have strong motivations to drop (instead of forwarding) packets to save energy. However, 3G/LTE bandwidth is scarce for smart phones as well as smart vehicles. While some smart vehicles download interesting contents from the Internet via 3G/LTE and share with the neighborhood in V2V manner, selfish vehicles try to be free-riders to benefit from contributing nodes without contributing themselves in order to save 3G/LTE cost.

So far, no incentive schemes have been proposed to enforce cooperative, fair LTE downloading. In this paper, we propose an incentive compatible scheme for cooperative downloading in vehicular networks. We first build a game theoretic model of the cluster downloading and sharing. Then we design a server-assisted round robin incentive and scheduling system using key management. Basically, the server encodes the shared contents with a temporary key and periodically updates it. The scheme rewards contributive vehicles by providing them with the keys to decode the content for the next few periods. Non-contributive nodes cannot decode until serving as cluster heads for a certain amount of time. Since the wireless environment of a vehicular network is highly lossy, and network coding can significantly improve the reliability of packet delivery according to existing research [LLL08], we consider that network coding is applied for V2V packet sharing in our scheme.

In summary, the main contributions of our paper are:

- cluster-based content downloading and distribution for VANETs, leveraging both vechicle-to-infrastructure (V2I) and V2V communication in vehicular networks to save LTE bandwidth and cost.
- The cluster-based scheme is modeled as a game and analyzed from a game theoretic perspective with numerical analysis, proving that the cluster member vehicles cannot cooperate with efficiency and fairness without an incentive-compatible management/scheduling scheme.

Literatures	Scenarios	Incentive type
[JH12, LYL11, LCC13, MCC12b, MCC12a, NDP05, WSH12, ZY11]	Cooperative download	None
[CZW11, LW09, WC07]	Packet forwarding	Token/Credit
[HHH08]	Packet forwarding	Counters/Reputation
Our scheme	Cooperative download	Key management

Table 3.1: Related works on cooperative downloading and incentive design in vehicular network

- novel server-assisted key management scheme that enforces the cooperation, fairness and security of cluster based content downloading. Game theoretic analysis is done to verify cooperation enforcement and analyze system utility.
- simulations of a highway scenario with the proposed cluster-based content distribution as well as the key management scheme. Results on selfish behavior impacts, system convergence, number of cluster heads and packet loss rate are presented and analyzed.

3.3 System Model

3.3.1 Network Model

We consider a vehicular network where vehicles the same interest form clusters, as shown in Fig. 3.2. A few cluster heads are selected and download the original data contents from the Internet source via 3G/LTE network. The cluster heads share their data packets with member vehicles in the cluster, and members may share with each other in a peer-to-peer fashion. Since the highly mobile vehicular network usually has a significant packet loss rate, in order to cope with the link loss, the cluster heads may find it beneficial to perform network coding (with or without redundancy) before broadcasting the data. A vehicle then can receive network coded packets from different cluster heads engaged in the same file download, and can still decode and reconstruct the original content.

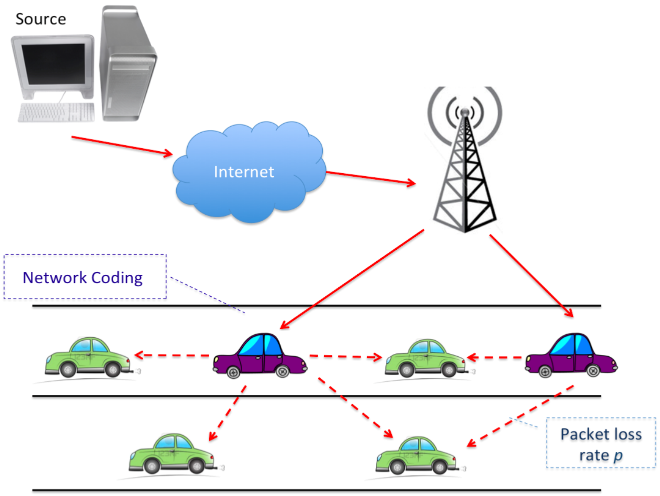

Figure 3.2: Cluster-based content distribution in vehicular networks

3.3.2 Game Setting

We model the cluster-based content downloading and peer-to-peer distribution of each network coding generation as a one-shot cluster game $G = (\mathcal{V}, \mathcal{A}, u)$. $\mathcal{V} = \{v_1, ..., v_M\}$ is the set of players, i.e., the set of vehicles in a cluster, where M denotes the total number of vehicles in a cluster. $\mathcal{A} = \{a_1, a_2, ..., a_M\}$ is the set of actions taken by each vehicle v_i, and $a_i \in \{\text{Head}, \text{Edge}\}$, i.e., each vehicle can choose to be either a cluster head or an edge node for the current round. Let m denote the number of vehicles choosing to be a cluster head, then the value of m represents the outcome of the actions. The utility of a player v_i, denoted as u_i, can be one of the two values, depending on the action set. The player receives an expected utility of $u_H(m)$, which is a constant as shown in (3.1), if he chooses to be a cluster head; on the other hand, it receives an expected utility of $u_E(m)$, as shown in (3.2), if he chooses to be an edge node. We assume each node receives a constant benefit of b for receiving each packet. The cluster head nodes encounter a constant cost of c for downloading each packet using 3G/LTE connection, and a constant cost of c' for performing network coding and broadcasting to other cluster members. Since energy is not a critical issue for vehicles, the cost c' is insignificant and ignored in the subsequent content. Should the model and solution in this paper be extended to MANETs where energy is an important issue, c' should be added to c to indicate the cost of network coding and forwarding.

$$u_H(m) = Kb - Kc \tag{3.1}$$

$$u_E(m) = Kb \cdot P(r \geq K|m) \tag{3.2}$$

where

$$P(r \geq K|m) = \begin{cases} \sum_{r=K}^{mn} \binom{mn}{r} (1-p)^r p^{(mn-r)}, & m \geq 1 \\ 0, & m = 0 \end{cases} \tag{3.3}$$

Here p denotes the link loss rate; K denotes the network coding generation size; n/K denotes the NC redundancy rate; and r denotes the number of packets actually received. $P(r \geq K)$ is the probability that an edge node receives enough encoded packets (at least K) to decode and reconstruct the original data. If a node receives less than K packets for a generation, it cannot decode and receives 0 utility. Here we assume the loss of each packet is independent, so the number of packets received follows a binomial distribution $B(mn, 1-p)$. In future work, we will take burst loss and other loss models into consideration. For multimedia contents, we propose that packets belonging to each I-frame are encoded as one NC generation with redundancy at cluster heads, which serves as a "multi-path compatible network layer FEC". This proposition takes advantage of the following three facts about video coding: (1) I-frames are large, thus have to be cut into multiple packets; (2) I-frames are specifically important to the video quality and need special protection, because the remaining frames in the group of pictures depend on them; and (3) generally, a frame cannot be decoded if one or more packets are lost. The third fact about video coding exactly matches with the characteristic of network coding, hence encoding the frame in the network layer at cluster heads will provide a higher reliability and decrease the chance of losing the whole generation/frame when less than K packets are received.

Table 3.2 gives an example of a 2-player game, i.e., assuming there are two vehicles in the cluster v_1 and v_2. $u_H(m)$ is always a constant u_H; and $u_E(m)$ depends on the number of players choosing to be a cluster head.

Table 3.2: The utility matrix of 2-player cluster game

		v_2	
		Head	Edge
v_1	Head	u_H , u_H	u_H, $u_E(m=1)$
	Edge	$u_E(m=1)$, u_H	0,0

3.3.3 Analysis of the One-shot Cluster Game

In order to analyze the cluster game, we need to find the Nash equilibrium of the m-player game. Below is a definition of Nash equilibrium.

Definition 2 *Nash equilibrium for a strategic game is a profile of actions such that each action is a best response to the other actions.* $a_i \in \mathcal{A}_i$ *is a best response to* $a_{-i} \in \mathcal{A}_{-i}$ *if*

$$u_i(a_i, a_{-i}) \geq u_i(a'_i, a_{-i}), \forall a'_i \in \mathcal{A}_i \tag{3.4}$$

where $a_i \in A_i$ *is the* i*-th user's action and* $a_{-i} \in A_{-i}$ *denotes the action profile of all users except user* i.

To compute Nash equilibrium, we need to know which is larger between $u_H(m)$ and $u_E(m)$. Given a fixed set of NC generation size K, redundancy rate n/K, benefit and cost scalers b and c, according to (3.1), u_H is a constant, i.e., a cluster head always has a fixed utility for downloading packets from LTE. According to (3.2), $u_E(m)$ is an increasing function over the number of cluster heads m, and a decreasing function over the V2V link loss rate p. Fig. 3.3 shows how $u_E(m)$ varies over different link loss rates p comparing with u_H; Fig. 3.4 shows how $u_E(m)$ varies with the number of cluster heads m comparing with u_H. From the figures we can see that for most cases where the link loss rate is acceptable and there is at least one cluster head, the utility of being an edge node is almost twice as much as being a cluster head, i.e., $u_E \simeq 2u_H$, assuming the cost benefit ratio c/b is 0.5. Generally speaking, when the link loss rate is low, u_E approaches $\frac{b}{b-c}u_H$. Only when the V2V link loss rate is so high that almost no packets can get through, the utility of being a cluster head could be higher than being an edge node, i.e., $u_H > u_E$, because edge nodes receive almost nothing. One special case is that nobody is willing to be cluster head, and everyone receives 0 utility, shown as the origin point in Fig. 3.4.

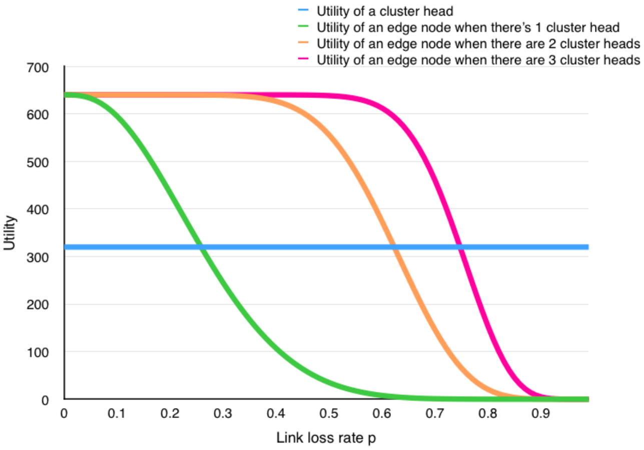

Figure 3.3: How the cluster head and edge node utilities vary with respect to V2V link loss rate. Settings: $K = 8$; $n = 10$; $M = 100$; $b = 80$; $c = 40$

For this game, when $u_H \geq u_E(M-1)$, e.g., when packet loss rate $p = 0.95$ in Fig. 3.4, there is one Nash equilibrium: $a_i = \text{Head}, \forall v_i \in \mathcal{V}$. Practically it refers to the situation where the utility of being a cluster head is larger than being an edge node, even if everyone else chooses to be a cluster head. This situation occurs when V2V communication is too lossy to transmit data effectively. In this case, all nodes prefer to directly connect to the 3G/LTE and download data by themselves. As system designers we do not need to worry about the cooperation enforcement in this extreme case because cooperation is not beneficial.

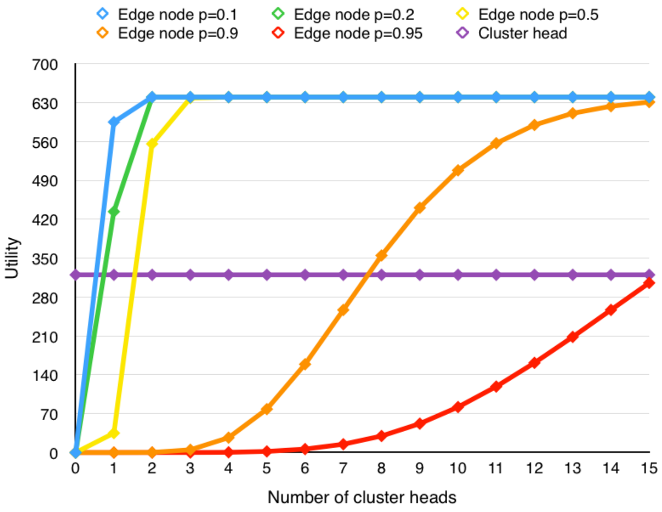

Figure 3.4: How the cluster head and edge node utilities vary with respect to the number of cluster heads. Settings: $K = 8$; $n = 10$; $M = 15$; $b = 80$; $c = 40$

When $u_H < u_E(M-1)$, there is always a $m^* \in (0, M-1]$ such that $u_E(m^*-1) < u_H \leq u_E(m^*)$, because $u_E(m)$ is an increasing function over m, and $u_H > u_E(0) = 0$. In this case, it forms a Nash equilibrium when exactly m^* players choose to be a cluster head, and $M - m^*$ players choose to be an edge node. $m = m^*$ is a Nash equilibrium because it would result in a decrease of utility if any of the $M - m^*$ edge nodes deviates and changes its action to being a cluster head since $u_E(m^*) \geq u_H$; it would also result in a decrease of utility if any of the m^* cluster heads deviates and changes its action to being an edge node, because $u_E(m^*-1) < u_H$.

The game has a Nash equilibrium that already maximizes the overall utility, however, if we think practically, the Nash equilibrium cannot be efficiently achieved without the help of an incentive compatible scheduling mechanism. If we assume nodes take actions all at the same time without negotiation, they would select the actions that maximize their expected utilities according to their prior believes about m. If the nodes believe $m \geq m^*$, all of them would choose to be edge nodes, leading to 0 utility for all nodes; if they believe $m < m^*$, all of them would choose to be cluster heads, resulting in no cooperation and a waste of resource. On the other hand, if we assume nodes choose their actions one by one, and the later ones can observe what the others have chosen, then the Nash equilibrium will be achieved when the first $M - m^*$ nodes choose to be edge nodes and the remaining peers have to become cluster heads, which is unfair and inefficient. Therefore under either assumption, the cluster-based scheme cannot work fairly and efficiently without a scheduling system, and it must be incentive compatible such that selfish nodes would be willing to follow the rule. Hence we propose a server-assisted management system to enforce efficient cooperation and fairness.

3.4 Server-assisted Key Management

Different from pure mobile ad hoc networks where central authority is not available, vehicular network is a mixed network that contains both V2I and V2V communications. Moreover, given the security and privacy concerns, maintaining the vehicle cooperation in a pure distribution way (i.e., by broadcasting the historical behavioral records among peers) is not feasible or efficient. Hence, we choose to let V2I communications to conduct the task of management.

Given the broadcast nature of wireless networks, especially for content distribution applications, nodes can always overhear the data content, even without making any contribution over time. Also, security and privacy has always been a very important concern for vehicular networks. Users may not want to be overheard by unknown vehicles unless those vehicles are interested in the same content and could be potential peers. Hence, to restrict the content distribution and keep it only among verified contributive users, we propose a server or proxy assisted key management scheme.

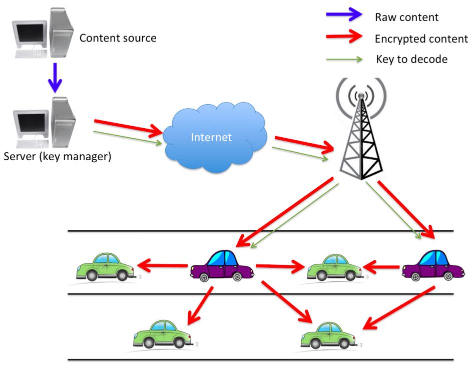

Figure 3.5: Server-assisted key management system architecture

3.4.1 How the Key Management Scheme Works

The server-assisted key management scheme, shown as Fig. 3.5, basically works as follows. The server receives the content from the content source and encrypts the raw content, such as a video stream, with a temporary key. The key is updated periodically. The server rewards the cluster heads by giving them the keys for the next several periods, say P, after they have been serving as cluster head for a period. So for P periods, the previous cluster heads will be able to decode the content. Those who have not recently been a cluster head (and therefore do not have the temporary key) are not able to decode, even if they overhear the V2V broadcast. They must volunteer to serve as a cluster head in order to receive the key from the server.

The functions and operations includes the following:

1. Vehicles register with the server and create an account through an application in order to participate, or log in if they have one.
2. Vehicles submit their interest (say soccer game stream) and location to the server. The server checks if there's an existing download group (or cluster) in the nearby area for the requested content. If yes, the server adds the user to the group; otherwise the server creates a new group for the user, initiates

the content downloading from the original Internet source and encrypts with the key for the current time period $K(t)$.

Note that the range of the area does not have to be specifically small or accurate, because as the vehicles travel, they may meet other vehicles interested in the same content that used to be quite far away 10 minutes ago. For such cases, they don't have to leave the old group and join a different one, instead, they can seamlessly continue the cooperative download. In future work, we will investigate how location information can be used to improve the key management scheme and the cluster head selection algorithm described in Section III.B. Note that all keys are identical across all servers and clusters for the same download group.

3. The server picks m ($m \geq m^*$ according to Section IV.A) vehicles to be cluster heads based on the cluster head selection algorithm described in Section III.B, and starts sending the encrypted content. At the end of the period, it distributes the keys for the current and next M/m periods to the cluster heads as rewards.
4. The server changes the keys and selects the new cluster heads for the next period. Step 3) and 4) are repeated until the content download is completed. Upon completion, the group is dismissed and the records are deleted.

3.4.2 Cluster Head Selection Algorithm

We propose the following cluster head selection algorithm, where the server (or cluster manager) selects nodes that do not have keys to decode with priority in order to promote fairness.

- A list is maintained at the server recoding which nodes have been cluster heads in previous rounds, noted as $\mathcal{H}$, which is a subset of all vehicles in the cluster $\mathcal{V}$.

- At the beginning of each round, m vehicles are selected to be the cluster heads of the current round $\mathcal{H}_c$. These m vehicles are randomly selected among the vehicles that are not in the list i.e., $v \in \mathcal{V} - \mathcal{H}$. Then the m nodes are added to the list, i.e., $\mathcal{H} \leftarrow \mathcal{H} + \mathcal{H}_c$

- If the number of candidates is less than m, i.e., $|\mathcal{V} - \mathcal{H}| < m$, all nodes in $|\mathcal{V} - \mathcal{H}|$ are selected as cluster heads for this round, and the rest $m - |\mathcal{V} - \mathcal{H}|$ cluster heads are chosen from $\mathcal{H}$. Then the list is cleared.

- Newly registered vehicles will be added to the member set $\mathcal{V}$, thus will have equal chance to be selected as the others who have not been selected before. If some vehicles leave the cluster, the server will remove them from $\mathcal{V}$.

It is efficient to prevent white wash (repeatedly leave and join the system to refresh records) by not granting the new-comer nodes the key to decode until they serve as cluster head for the first time. Clearly, new users that just join the group will not be able to benefit until they contribute, therefore they should have strong intention to contribute instead of being free-riders. With this key management scheme, there is a decrease in throughput at the beginning since many nodes do not have keys to decode the content they receive. Such decrease in throughput serves as a tradeoff for the gain of cooperation, privacy and security enforcement. Furthermore, the management process incurs a cost of the server (thus the service provider), however, the cooperative downloading scheme significantly saves LTE bandwidth and reduces LTE network congestions, which is beneficial to the service provider.

3.5 Game Theoretic Analysis

3.5.1 Cooperation Enforcement Condition

In Section II.C, we showed that cluster-based content distribution cannot be implemented fairly and efficiently without a management/scheduling scheme. With the key management scheme and corresponding cluster head selection algorithm presented above, fairness and efficiency of cluster head selection is ensured. In this section, we use game theoretic tools to analyze the cooperation enforcement when the key management scheme is applied.

The game theoretic analysis is conducted based on the user interaction per $\lfloor \frac{M}{m} \rfloor$ to $\lceil \frac{M}{m} \rceil$ periods, within which each cluster member serves as cluster head one time. The game is repeated over time, however here we do not consider adding a discount factor [1] to make it a repeated game, because the rewards (keys) are consumed within the $\frac{M}{m}$ periods and do not affect future decisions. In other words, the user interactions of every $\frac{M}{m}$ periods are independent.

Let u_C denote the expected utility of compliance, i.e., serving as a cluster head for every $\lfloor \frac{M}{m} \rfloor$ to $\lceil \frac{M}{m} \rceil$ periods. Let u_N denote the expected utility of non-compliance. A rational node will choose to comply if and only if $u_C \geq u_N$. u_C and u_N can be expressed with the utility of being a cluster head $u_H(m)$, shown in (3.1) and the utility of being an edge node $u_E(m)$, shown in (3.2).

$$u_C(m) = u_H(m) + u_E(m) \cdot (\frac{M}{m} - 1) \tag{3.5}$$

$$u_N(m) = u_H(m) \cdot \frac{M}{m} \tag{3.6}$$

Since nodes that do not serve as cluster heads will not be able to receive the

---

[1] Discount factor in a repeated game represents the probability that the game will be repeated for an additional round.

keys to decode, they have to download the whole file on their own (if they are actually interested in the content), which results in a utility that is equivalent to being a cluster head all the time. [2]

The necessary and sufficient condition of compliance for any rational and selfish node is: $u_C(m) \geq u_N(m)$. After derivation, it is equivalent to: $u_E(m) \geq u_H(m)$.

According to Fig. 3.3 and the analysis in III.C, $u_E(m) \geq u_H(m)$ is satisfied when the number of cluster head m is no less than a threshold $m^* = \tau(p)$, which is an increasing function of packet loss rate p, i.e., higher packet loss rate requires more cluster heads. When p approaches 1, m^* approaches M, which refers to the extreme case where everyone has to download by themselves (however the size of the cluster M does not affect cooperation enforcement condition); when p approaches 0, m^* approaches 0 but m must be at least 1, which represents the best case where one cluster head is enough to serve a fleet. Hence, generally speaking, cooperation enforcement condition can be interpreted as $m \geq m^* = \tau(p)$. Evidently, the more cluster heads, the higher cooperation enforcement. With intra-flow network coding applied, the more cluster heads also end up with higher throughput and reliability.

3.5.2 Average Utility

Let $\overline{u}$ denote the average utility of the cluster, defined to be the average of all the nodes' utilities in the cluster.

$$\overline{u}(m) = \begin{cases} \frac{m}{M} \cdot u_H(m) + (1 - \frac{m}{M}) \cdot u_E(m), & m \geq 1 \\ u_H(m), & m = 0 \end{cases} \tag{3.7}$$

If $m < m^*$, then $u_E(m) < u_H(m)$ and $\overline{u}$ is increasing in m. This is because when there are too few cluster heads, edge nodes cannot receive sufficient packets.

---

[2] Non-compliant nodes download the content from a different source instead of the cluster server, so they are not actually serving as cluster heads. Only their payoffs/utilities are the same as cluster heads.

On the other hand, if $m \geq m^*$, then $u_E(m) > u_H(m)$ and $\overline{u}$ is decreasing in m. This is because when there are too many cluster heads, the benefit of cooperation decreases (to the non-cooperative case for $m = M$), which is very intuitive. Fig. 3.6 shows how average utility varies over number of cluster heads given different packet loss rates. From the figure we can see that there is always a peak point, which is m^*. According to Section IV.A, $m^* = \tau(p)$. So the peak comes earlier when the packet loss rate p is small, i.e., when V2V communication is reliable, fewer cluster heads are needed; and the peak comes later when p is large, i.e., when V2V communication is very lossy, more cluster heads are required.

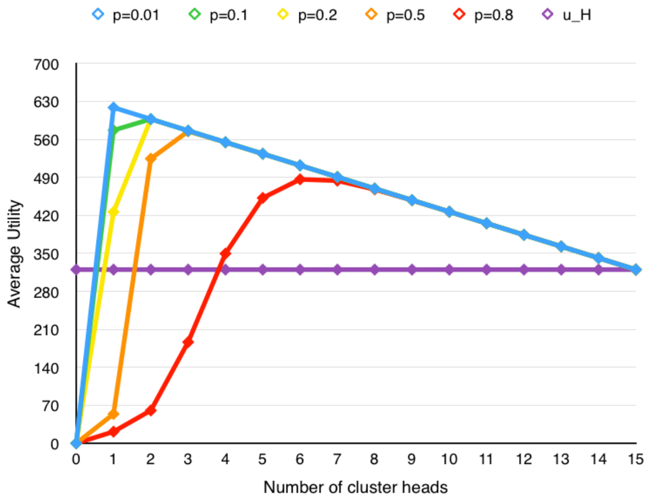

Figure 3.6: How average utility varies with respect to the number of cluster head. Settings: $K = 8$; $n = 10$; $M = 15$; $b = 80$; $c = 40$

3.5.3 Optimization & Performance Evaluation

Comparing the conclusions from Section IV.A and IV.B, we see that there is an interesting relationship between the level of cooperation and the average utility. In order to maximize the average utility, yet enforce cooperation, we formalize the problem as an optimization problem with constraint:

$$\begin{aligned}
&\text{Maximize: } \overline{u}(m) = \frac{m}{M} \cdot u_H(m) + (1 - \frac{m}{M}) \cdot u_E(m) \\
&\text{subject to: } u_E(m) \geq u_H(m)
\end{aligned}$$

The optimization problem is equivalent to finding the minimum number of cluster heads to enforce cooperation, and the solution is clear, i.e., m^*. According to the model provided in Section II, m^* can be obtained by solving the equation $u_H(m) = u_E(m)$ with a set of parameters (p, M, b, c) given, shown as the intersection points in Fig. 3.4 and the peak points in Fig. 3.6. Fig. 3.6 also shows that with a properly selected number of cluster heads, the performance of the cluster based content downloading scheme, taking the average utility as the evaluation metric, is always better than the traditional situation where everybody downloads by themselves, represented by the horizontal line in Fig. 3.6.

3.6 Simualtion

As shown in the previous section, with a properly selected number of cluster heads, the performance of the cluster based content downloading scheme is always better than the traditional situation where everybody downloads by themselves. The proposed cluster-based downloading scheme with key management is generally suitable for a stable group of users in any type of wireless networks. As mentioned in the introduction section, one feasible application is video stream downloading in vehicular ad hoc networks, especially highway scenarios. To bring the analysis one step closer to reality, we performed a simulation on a particular VANET highway scenario to

- study the impacts of selfish behaviors;
- verify the minimum number of cluster heads needed;

- study the impacts of the key management, i.e., the cost of (or the performance degradation due to) incentives.

3.6.1 Simulation Setup

We simulated the VANET highway scenario using SUMO and NS3. The simulation follows the system architecture shown in Fig. 3.5. The highway segment has a total length of 200 meters and contains 3 lanes and 4 exits. 50 vehicles with the same interest move in the same direction at a starting velocity of 60 mph and accelerate afterwards with randomness, as shown in Fig. 3.7. On average 10% of the total traffic (i.e., 5 vehicles) will exit the highway at each exit. The others remain on the highway to the end. We did not set up highway entrances because there are no ramp settings in SUMO that allow entering vehicles to merge into the traffic; instead, both entering and oncoming vehicles must stop and wait as if it were an urban intersection, thus altering flow dynamics.

In the simulation, for simplicity we assume an ON/OFF propagation loss model where receivers at or within the maximum propagation range receive the transmission at the transmit power level, and receivers beyond the maximum range receive at power -1000 dBm (effectively zero). The value of the maximum propagation range for V2V communication impacts the number of edge nodes within the range

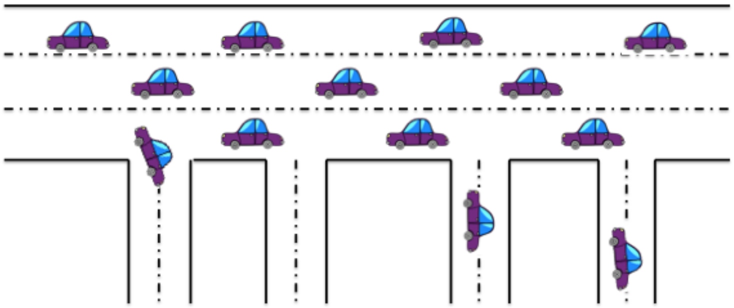

Figure 3.7: Highway layout for the simulation

of each cluster head. Equivalently, it impacts the number of cluster heads within the range of each edge node. When an edge node is not within the range of a cluster head, the packets broadcasted by the cluster head won't be received by the edge node, equivalently lost. Hence, the propagation range impacts packet delivery. Fig. 3.8 shows how packet loss rate varies over propagation range — larger propagation range leads to lower packet loss rate, which is intuitive. Of course, the lost packets from one cluster head may be received from a different cluster head, so the packet loss rate here does not reflect final packet delivery.

For the rest of the simulation, the propagation range is set to 30 meters, which results in an average packet loss rate of $p \simeq 22\%$. According to Fig. 3.6, when the loss rate is close to 20%, the optimal number of cluster head is 2, which will be verified with the simulation result in part B.(1).

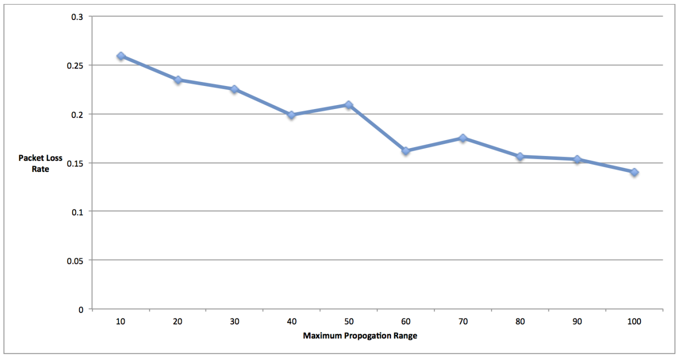

Figure 3.8: Packet loss rate vs. maximum propagation range

Each vehicle has a perfect LTE connection directly to the cluster manager node (acting as the remote server). The node operations of packet downloading, distributions and key management follows the descriptions in section III.

3.6.2 Simulation Results

3.6.2.1 Non-compliant behavior impacts and the number of cluster heads

We consider a probability of being non-compliant for each node in the cluster (equal to the fraction of non-compliant nodes in the system). When the server designates a cluster member to be a cluster head, the node behaves as non-compliant, i.e., it rejects the task, with a probability p_N. Fig. 3.9 reports simulation results for packet receiving rate and decoding rate by the edge nodes, i.e. the ratio of the number of packets received/decoded over the number of packets requested, as a function of non-compliance probability p_N. The number of packets decoded is less than the number of packets received by edge nodes because edge nodes which have not been a cluster head before do not have the keys to decode (as further discussed in part 2 of this section). Here we set the number of cluster heads to 20, which means for each requested packet, each edge node will ideally receive 20 packets if it is within the range of all the cluster heads. An increase in non-compliance rate is equivalent to a decrease of the number of effective cluster heads, i.e., a 95% non-compliance rate of 20 cluster heads is equivalent to having only 1 effective cluster head (the other 19 are non-compliant).

From the result we can see: (1) a higher compliance rate, i.e., a larger number of effective cluster heads, results in higher throughput as well as higher packet loss tolerance; (2) for the scenario we simulated, two cluster heads, i.e. non-compliance rate = 0.9, can on average guarantee that the average deliver rate is above one. It verifies the result in Fig. 3.6 on the minimum number of cluster heads needed.

Without the stimulus of an incentive (and punishment), non-compliance behavior would be pervasive, considering the human nature of being a free-rider. The proportion of "altruistic" users willing to pay for LTE service for the entire download is expected to be very small. For instance, with the optimistic estimate of 5% altruistic users, when the system selects 20 cluster heads, there will be on average only one altruistic cluster head responding, not sufficient (as per Fig. 3.9) to deliver the packets to all the member in the cluster for the highway scenario we simulated. However with the proposed key management scheme, we proved in Section IV.A that as long as the number of selected cluster heads is no less than m^* (in the simulated scenario with $p = 22\%$, $m^* = 2$), the system is incentive-compatible. This means that by selecting two or more cluster heads, selfish users will have no motivation to refuse compliance because compliance yields a higher utility than non-compliance.

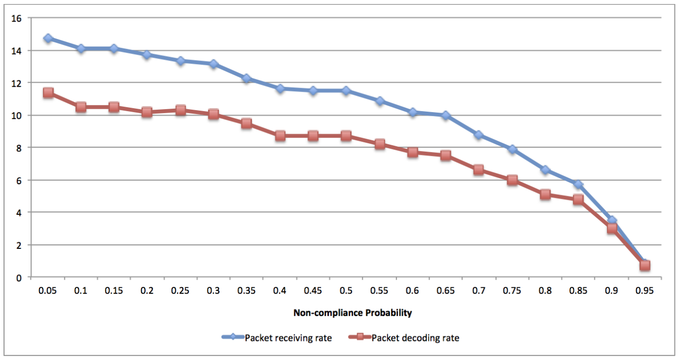

Figure 3.9: Packet receiving rate and decoding rate by edge nodes vs. non-compliance probability

3.6.2.2 System convergence

We will use the following definition as a metric for system convergence.

Definition 3 *Let* n_d *denote the number of packets received by edge nodes who have the key to decode, i.e., the decodable packets. Let* n_t *denote the total number of packets received by all edge nodes, with or without the key to decode. Decoding rate* γ *is defined to be the ratio of* n_d *over* n_t*.*

$$\gamma = \frac{n_d}{n_t} \tag{3.8}$$

As discussed at the end of Section III, since only a few vehicles have the key to decode at the beginning phase, the throughput initially may be low, which is in fact the cost of cooperation enforcement (as well as security/privacy protection). Fig. 3.10 shows the decoding rate (as defined above) of edge nodes over time, both cumulative and per round. We can see that the decoding rate increases very fast in the first a few rounds, and remains very close to 1 starting from round-7. The cumulative average reaches 1 only asymptotically as it must aggregate all the previous rounds.

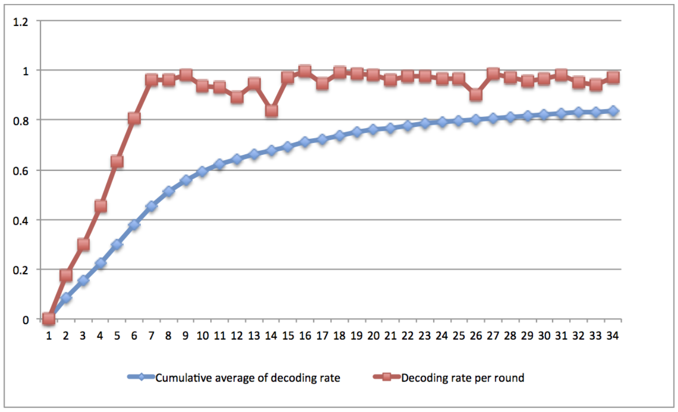

Figure 3.10: Decoding rate (cumulative average and per round) vs. number of rounds

3.7 Conclusion

In this chapter, we propose a cluster-based scheme for content distribution in VANETs leveraging both V2I and V2V communications. By forming a cluster, a few nodes acting as cluster heads download original content from the Internet via 3G/LTE connections, and share with others in a peer-to-peer fashion via V2V communications. With game theoretic tools and numerical analysis, we show that vehicle peers cannot spontaneously cooperate with efficiency and fairness. Hence we propose a server-assisted key management scheme to manage the cluster, enforcing cooperation and promoting security and efficiency. We also propose a cluster head selection algorithm that ensures fairness across the peers. By game theoretic analysis and simulation of a highway scenario, we show that: (1) the system throughput increases with the number of effective cluster heads; (2) non-compliance behavior decreases the number of effective cluster heads drastically, and thus reduces throughput; (3) our scheme converges very rapidly after a "slow-start" of decoding rate at initialization.

In future work, we will investigate how location information can be used to improve the key management scheme and the cluster head selection algorithm. For instance, packet loss rate could be reduced and key distribution could be sped up by aggregating users into clusters based not only on interests, but also on their geographic locations and relative proximities. These clusters could be managed in the same way as discussed in this paper, i.e., they could be split and combined dynamically as proximity changes over time.

CHAPTER 4

Video Congestion Control for VANETs

4.1 Introduction

The previous sections present a cluster based video downloading scheme to save LTE bandwidth and reduce congestions. We assumed that there is one video flow per vehicle cluster. In this chapter, we will consider multiple interests of video flows per cluster. When there are multiple interests of video flows, instead of assigning multiple cluster heads each responsible for one flow, it is more efficient to let the same cluster heads download the multiple requested flows. In this case, it is more likely to cause congestions over the LTE link from the base station to the cluster heads. In this chapter, we propose a video congestion control solution and perform simulations on multiple flow downloading.

4.2 2-staged Scheme for Video Distribution

The VANET may become congested if too many video streams are downloaded from the Internet. This situation is resolved using adaptive video coding. A receiver, upon suffering video packet loss, will shed the video enhancement layers and at the same time it signals the congestion condition to the upstream node. However, the video is generally broadcast among vehicles in a cluster, so it is not scaleable to collect feedback information from each and every receiver. Hence, we propose a novel, two level hierarchical video delivery and congestion control architecture. In our system, a video transmission path from the Internet video

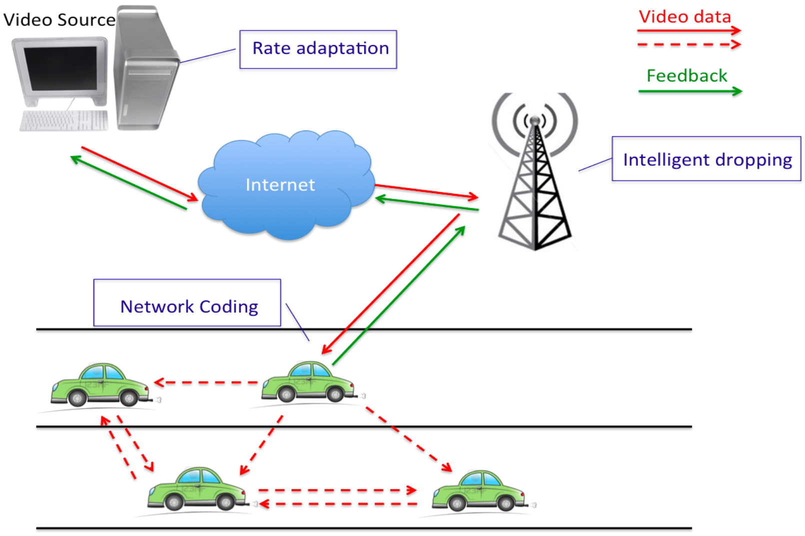

Figure 4.1: System architecture for VANET video congestion control

source to the vehicle receivers, as shown in Fig.4.1, consists of two segments. Segment 1 (the top level) is a unicast RTP session (with feedback flow) from the Internet video source to the cluster head; and segment 2 (at the bottom end of the distribution) is a broadcast UDP session (without feedback flow) from the cluster head to other vehicles in the cluster.

4.3 Stage1: Adaptive Congestion Control

In designing segment 1 congestion control, the vehicle cluster can be viewed as a whole subnet, and the cluster head vehicle as the representative which provides feedback to the source on behalf of the cluster. Traditional point-to-point video congestion control schemes can be directly applied here without specific modification to account for the VANET. Fig.4.2 shows the system architecture for segment 1, where video content is encoded and pushed to the Internet as RTP packet stream; routers in the Internet (as well as LTE towers) perform video-based AQM (Active Queue Management) schemes to mitigate congestion and optimize video performance; the cluster head feedbacks the one-way delay and loss rate, etc. to the source as RTCP packet stream; and video source performs rate adaptation according to the feedback information.

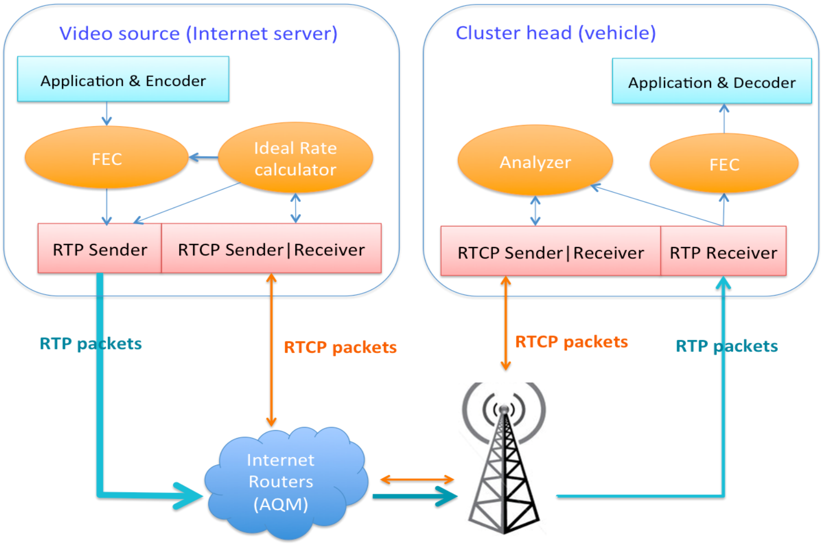

Figure 4.2: System architecture of segment 1: unicast from source to cluster head

4.4 Stage2: Network Coding for Reliability Improvement

In segment 2, i.e. within the vehicle cluster, the cluster head broadcasts the video packets upon request. There is no feedback in this stage to assist in congestion protection. In order to both mitigate congestion and enable the use of Net Coding to cope with the lossy VANET channels, we propose a novel two-stage queue scheme, as shown in Fig.4.3. Upon receiving video packets through direct LTE connection, the cluster head pushes the packets to the first queue, the congestion control queue. In this queue video-based AQM (e.g. differentiated dropping) is performed to mitigate congestion based on WiFi channel congestion while at the same time preserving acceptable video performance. Next, packets are transferred group by group from the front of the first queue to the second buffer. Here packets are network coded generation by generation and are broadcast. No more congestion packet dropping is performed here. Vehicles in the cluster exchange blocks (network coded packets) in a peer-to-peer manner to increase the deliver rate and download speed.

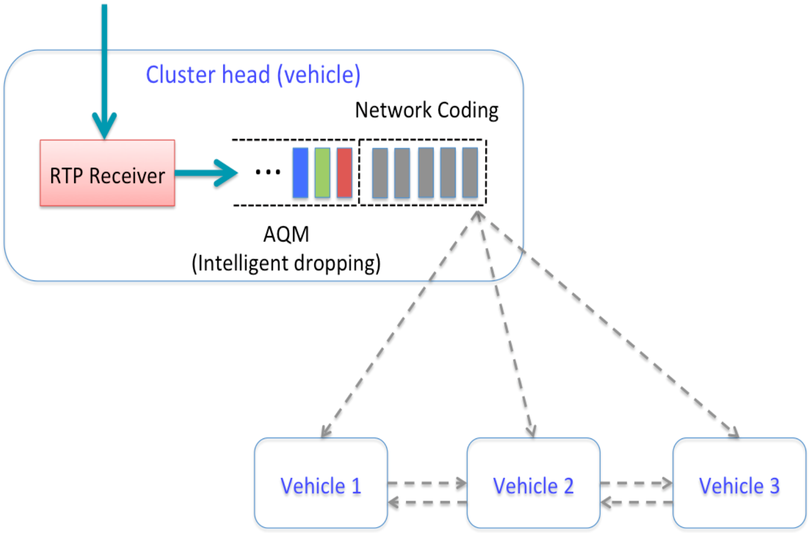

Figure 4.3: Segment 2: broadcast from cluster head to other vehicles

In our system, all video rate adaptation schemes (including conventional equation-based approaches such as TFRC[FF99, SW08], as well as LIVA [ZPP11], a new scheme relying on explicit congestion signaling from the network to achieve proactive and media-aware congestion control, etc.) and AQM schemes (e.g. WRED, CoDel [NJ12], etc.) at the server can be deployed without modification required by the VANET. Recall that different video packet types have different importance in determining video reception quality. In the MPEG-1 codec, the Intra-frame (I-frame) is more important than the Predicted-frame (P-frame); and P-frame is more important than Bidirectional-frame (B-frame). In scalable video coding (SVC), base-layer (BL) packets are more important than enhancement-layer (EL) packets [SMW07]. The importance of different frames for decoding is reflected in the application of dropping policies. Leveraging this characteristic of SVC, a few proposed congestion control schemes [XJN07, KZD04, TLW00] deploy intelligent dropping at the router or relay nodes to improve the video performance. Basically, in these schemes, different dropping thresholds are set for different types of frames according to their importance, i.e. dropping less important packets earlier to save more important packets. In VANET, we propose that the cluster head and intermediate vehicles also perform intelligent dropping to improve video quality when congestion happens. Simulation results in [XJN07, KZD04, TLW00] show that intelligent dropping leads to significant improvements in video QoS/QoE. Note that the dropping must be done in the first queue, before the packets are Network Coded and mixed in the second queue in order not to affect the receiver's ability to reassemble the generation. The careful reader will note that coded packets may still be lost in the VANET during the P2P exchange due to random errors, interference etc. However, these losses are recovered by redundant transmissions from the coded buffer and also by the intrinsic redundancy of the multipath VANET.

4.5 Media Frame Protection

Given different levels of importance of the video frames to video quality, the Media Frame Protection (MFP) method develops an AQM scheme adopting differentiated dropping thresholds for different types of frames. The idea is to drop less important frame packets before the queue is full, so that it leaves space for important frame packets, even when bursts occur. The goal is to smoothen the video conferencing or video streaming experiences.

Fig. 4.4 shows the MFP AQM scheme. The parameter includes two thresholds in terms of queue length. One threshold is for dropping B frames (or discardable P frames), θ_B; the other threshold is for dropping P frame (or super P frame), θ_P, where $\theta_B < \theta_P$.

A set of AQM schemes that are implemented and compared is listed as follow-

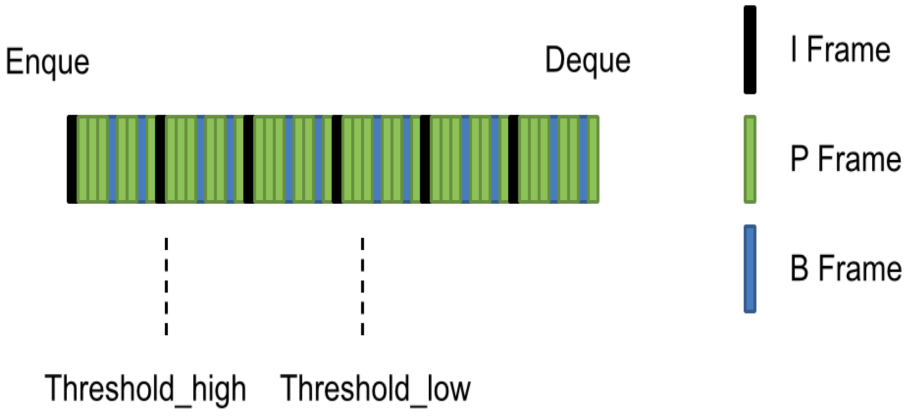

Figure 4.4: Media Frame Protection AQM scheme

ing:

- Droptail: all packets are dropped when the queue is full.
- MFP (with variants):
  - MFP-enque: packets are dropped when they enter the queue.
  - MFP-deque: packets are dropped when they arrive the end of the queue.

4.6 Performance Evaluation

To evaluate the performance of MFP, we conduct simulations with the following settings:

- Two identical video flows go through a bottleneck link with 10Mb bandwidth. Video trances are generated from movie trailer Avenger with Cisco Tanberg equipment. Each flow has a small time shift. The video rates are shown in Fig. 4.5.

- Each packet is marked with its frame type in the IP header. Audio packets are marked as I-frame packets, i.e. both audio packets and I-frame packets have the highest priority; super-P packets have the secondary priority; discardable-P packets have the lowest priority. Fig. 4.6 shows the proportions of different video frame packets in the video source, where discardable-P frame packets take up about 20% of the total number of packets.
- Simulations are run with different AQM algorithms, as well as different dropping thresholds.

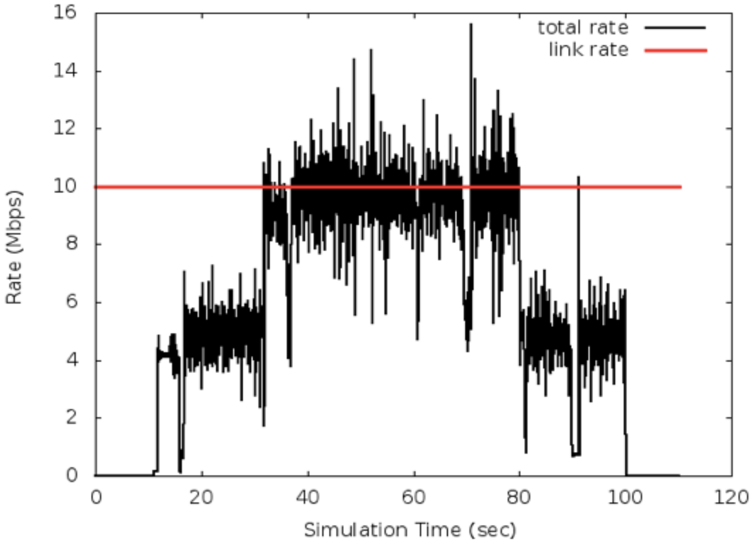

Figure 4.5: Data rate for the two flows. The bottleneck link has a bandwidth of 10Mb

Fig. 4.7 shows the statistics of different frame packet drops. The MFP threshold is in terms of the ratio of current queue length over the total queue length. When the threshold is set to (1.0, 1.0), it is equivalent to the traditional DropTail scheme.

By setting the threshold of super-P packets to 1.0, we tune the threshold for discardable-P frame packets from 0 to 1, and plot the result on packet drops with

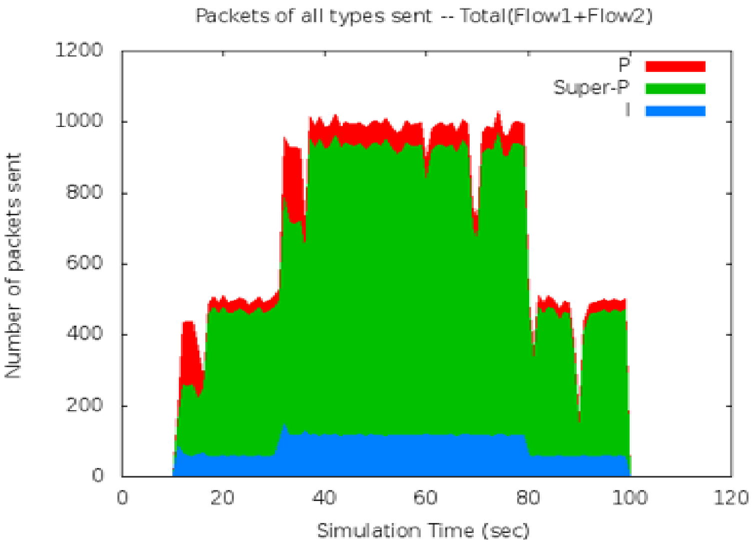

Figure 4.6: The fractions of different video frame packets in the video source

Fig. 4.8. Comparing with Enque versus Deque schemes, i.e. packets are dropped when it enters or leaves the queue, we can see that Enque has a better performance than Deque.

4.7 Conclusion

By performing simulations on different AQM schemes with real video traces, we obtain the following conclusions: (1) MFP saves more I-frame and P-frame (or super-P-frame) packets by dropping B-frame (or discardable-P-frame) ahead of time; (2) according to experiments, selecting the dropping threshed of B frames to be around 0.5% of the queue length leads to good performance, i.e. the least dropping of high priority packets with little increase in total packets drops; (3) dropping packets at the enque side yields a slightly better result over dropping at the deque side.

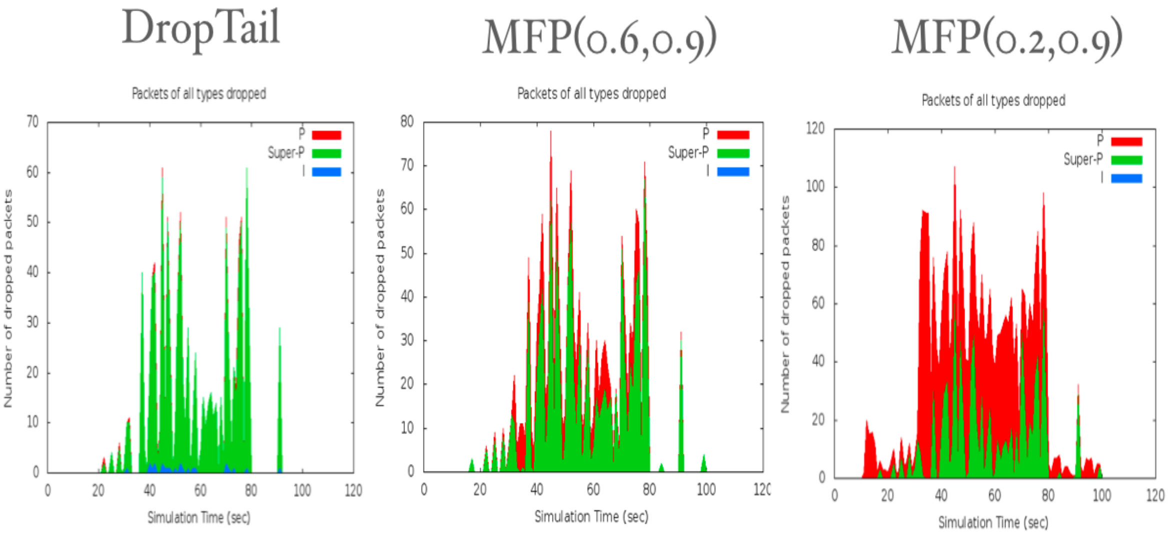

Figure 4.7: The statistics of different video frame packets dropped over different MFP dropping thresholds

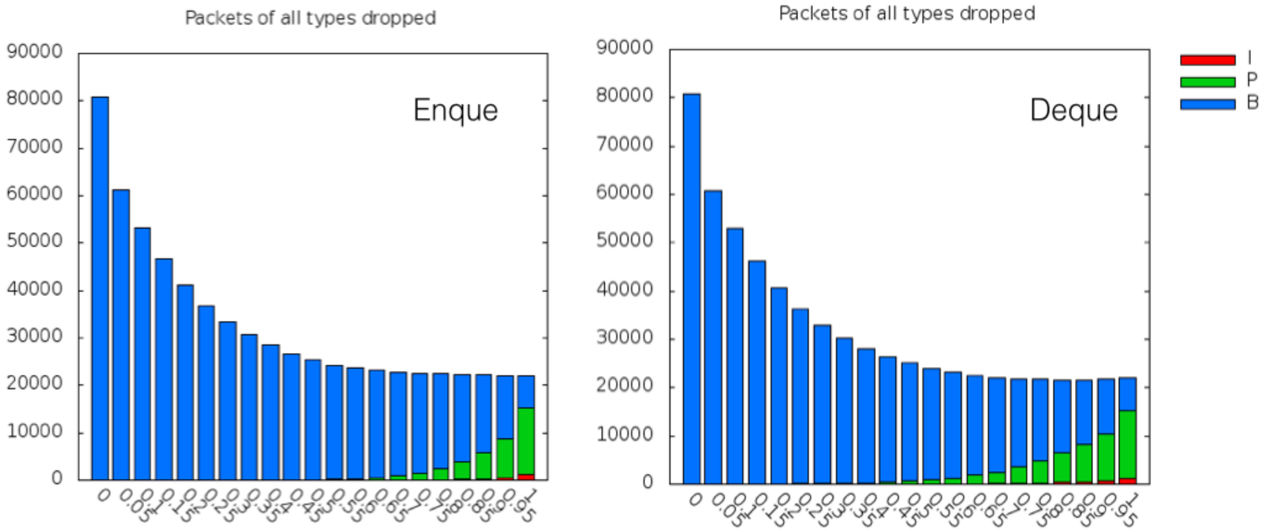

Figure 4.8: The number of packets of different frame types dropped vs. the dropping threshold of discardable-P frame packets.

CHAPTER 5

Conclusion

Cooperative content distribution requires cooperation of mobile users, so as many other cooperative schemes proposed for wireless networks in this field. Selfish mobile users are motivated to decline cooperation due to imbalance of interests. In this dissertation, we presented two incentives schemes for cooperation enforcement in wireless networks.

In chapter 2, we present a social norm based incentive scheme for mobile ad hoc networks, where nodes are selfish, strategic users. For simplicity, we consider a single-path unicast session, where the intermediate node can rationally choose to do secure network coding and forward packets with redundancy (NCF), simply forward the original packets (F) or totally drop the packets (Drop). We model the unicast session as a repeated gift-giving game with persistence factor δ, where nodes are symmetric and randomly matched. Our incentive scheme consists of a social strategy σ and a reputation scheme τ, which reward cooperative nodes by increasing their reputation and promising better transmission quality in the future, and punish uncooperative nodes by decreasing their reputations and refusing coding/forwarding their packets in the future. Most existing incentive schemes only propose a single protocol and prove it can incentivize nodes to perform coding/forwarding. Differently from previous schemes, we propose a set of social norm based incentive schemes with parameters that we adjust to optimize the social utility and sustainability. We first prove that the reputation distribution will converge to a unique steady state, corresponding to a fixed setting of the parameters. Then we present a series of results for parameter selection that are supported by mathematical derivations as well as by numerical analysis: (1) there exists a tradeoff between the social utility/throughput and sustainability — a stricter social strategy will increase the sustainability but decrease the utility; (2) both utility and sustainability will be optimized if the reputation threshold k is set close to the maximum reputation L, and reputations are updated or refreshed frequently; (3) NC generation size does not affect the performance of our incentive scheme. Additionally we show how the network condition or the game itself impacts our incentive scheme. Moreover, practical issues for the proposed reputation system are discussed, including the distributed reputation system, as well as the existence of altruistic and malicious users.

In chapter 3, we propose a cluster-based scheme for content distribution in VANETs leveraging both V2I and V2V communications. By forming a cluster, a few nodes acting as cluster heads download original content from the Internet via 3G/LTE connections, and share with others in a peer-to-peer fashion via V2V communications. With game theoretic tools and numerical analysis, we show that vehicle peers cannot spontaneously cooperate with efficiency and fairness. Hence we propose a server-assisted key management scheme to manage the cluster, enforcing cooperation and promoting security and efficiency. We also propose a cluster head selection algorithm that ensures fairness across the peers. By game theoretic analysis and simulation of a highway scenario, we show that: (1) the system throughput increases with the number of effective cluster heads; (2) non-compliance behavior decreases the number of effective cluster heads drastically, and thus reduces throughput; (3) our scheme converges very rapidly after a "slow-start" of decoding rate at initialization.

Furthermore, we propose a solution architecture for video congestion control, and perform simulations to analyze priority based queuing scheme, the medium frame protection scheme, and shown its significance in protecting high priority video frames.

As future work, the presented schemes can be extended in the following directions.

First, the cluster head selection scheme proposed in chapter 3 is a centralized scheme that took advantage of the V2I communications to conduct the cluster management. With proper design of the initiation phases, the cluster heads can be selected in a distributed manner where each round the right number of nodes volunteer to become cluster heads. In this way, the management overhead is saved, but it is faced with difficulties. For example, when some members leave (or join) the system, the number of volunteers for that round will be affected and thus do not match the optimal number of cluster heads, resulting in a degradation of system performance.

Second, the cluster based content distribution, as well as the key management incentive scheme, is mainly designed for scenarios where steady downloading groups can be formed, e.g. the VANET highway scenario. As for scenarios where nodes meet and depart, e.g. the VANET urban scenario, a different scheme is required. One possible scheme is opportunistic cooperation of wireless peers where each node downloads by themselves and broadcasts a request for missed packets. Peers who have the requested packets can feedback. For such scenario, virtual credits or tokens can be applied to incentivize peers to help each other. Virtual credit based incentives better serve distributed and opportunistic peer interactions, but it is also faced with difficulties such as finding the optimal amount of token supply according to how frequent peers interact as well as many other complicated factors in practice.

References

[ACL00] R. Ahlswede, Ning Cai, S.-Y.R. Li, and R.W. Yeung. "Network information flow." *Information Theory, IEEE Transactions on*, **46**(4):1204–1216, 2000.

[AG04] K.G. Anagnostakis and M.B. Greenwald. "Exchange-based incentive mechanisms for peer-to-peer file sharing." In *Distributed Computing Systems, 2004. Proceedings. 24th International Conference on*, pp. 524–533, 2004.

[AHD05] W.J. Adams, G.C. Hadjichristofi, and IV Davis, N.J. "Calculating a node's reputation in a mobile ad hoc network." In *Performance, Computing, and Communications Conference, 2005. IPCCC 2005. 24th IEEE International*, pp. 303–307, 2005.

[BBL04] Sonja Buchegger, Sonja Buchegger, and Jean-Yves" Le Boudec. "A Robust Reputation System for P2P and Mobile Ad-hoc Networks." 2004.

[BL05] S. Buchegger and J.-Y. Le Boudec. "Self-policing mobile ad hoc networks by reputation systems." *Communications Magazine, IEEE*, **43**(7):101–107, July 2005.

[BSG06] S. Bhadra, S. Shakkottai, and P. Gupta. "Min-Cost Selfish Multicast With Network Coding." *Information Theory, IEEE Transactions on*, **52**(11):5077–5087, 2006.

[CBZ11] Tingting Chen, Ankur Bansal, and Sheng Zhong. "A reputation system for wireless mesh networks using network coding." *J. Netw. Comput. Appl.*, **34**(2):535–541, March 2011.

[CWZ11] Tingting Chen, Fan Wu, and Sheng Zhong. "FITS: A Finite-Time Reputation System for Cooperation in Wireless Ad Hoc Networks." *Computers, IEEE Transactions on*, **60**(7):1045–1056, 2011.

[CZ10] Tingting Chen and Sheng Zhong. "INPAC: An Enforceable Incentive Scheme for Wireless Networks using Network Coding." In *INFOCOM, 2010 Proceedings IEEE*, pp. 1–9, 2010.

[CZW11] Tingting Chen, Liehuang Zhu, Fan Wu, and Sheng Zhong. "Stimulating Cooperation in Vehicular Ad Hoc Networks: A Coalitional Game Theoretic Approach." *Vehicular Technology, IEEE Transactions on*, **60**(2):566–579, 2011.

[DD] Chrysanthos Dellarocas and Chrysanthos" Dellarocas. "How often should reputation mechanisms update a trader's reputation profile." *HANDBOOK ON INFORMATION SYSTEMS AND ECONOMICS*, **2006**.

[FB09] M. Fiore and J.M. Barcelo-Ordinas. "Cooperative download in urban vehicular networks." In *Mobile Adhoc and Sensor Systems, 2009. MASS '09. IEEE 6th International Conference on*, pp. 20–29, Oct 2009.

[FF99] Sally Floyd and Kevin Fall. "Promoting the use of end-to-end congestion control in the Internet." *IEEE/ACM Trans. Netw.*, **7**(4):458–472, August 1999.

[FFL04] Michal Feldman, Michal Feldman, Kevin Lai, Ion Stoica, and John" Chuang. "Robust Incentive Techniques for Peer-to-Peer Networks." 2004.

[GWP14] Mario Gerla, Chuchu Wu, Giovanni Pau, and Xiaoqing Zhu. "Content distribution in VANETs." *Vehicular Communications*, **1**(1):3 – 12, 2014.

[HHH08] Yao H. Ho, Ai H. Ho, Georgiana L. Hamza-Lup, and Kien A. Hua. "Cooperation Enforcement in Vehicular Networks." In *Proceedings of the 2008 International Conference on Communication Theory, Reliability, and Quality of Service*, CTRQ '08, pp. 7–12, Washington, DC, USA, 2008. IEEE Computer Society.

[JH12] Hung-Chin Jang and Tzu-Yao Hsu. "Urban Multi-Layered Chord for P2P over vehicular network." In *Mobile and Wireless Networking (iCOST), 2012 International Conference on Selected Topics in*, pp. 54–59, 2012.

[JS07] Juan José Jaramillo and R. Srikant. "DARWIN: distributed and adaptive reputation mechanism for wireless ad-hoc networks." In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, MobiCom '07, pp. 87–98, New York, NY, USA, 2007. ACM.

[KM03] R. Koetter and M. Medard. "An algebraic approach to network coding." *Networking, IEEE/ACM Transactions on*, **11**(5):782–795, 2003.

[KZD04] Seong-Ryong Kang, Yueping Zhang, Min Dai, and D. Loguinov. "Multi-layer active queue management and congestion control for scalable video streaming." In *Distributed Computing Systems, 2004. Proceedings. 24th International Conference on*, pp. 768–777, 2004.

[LCC13] T. Luan, L. Cai, J. Chen, X. Shen, and F. Bai. "Engineering a Distributed Infrastructure for Large-Scale Cost-Effective Content Dissemination over Urban Vehicular Networks.", 2013.

[LGK11] Seung-Hoon Lee, M. Gerla, H. Krawczyk, Kang-Won Lee, and E.A. Quaglia. "Performance Evaluation of Secure Network Coding Using Homomorphic Signature." In *Network Coding (NetCod), 2011 International Symposium on*, pp. 1–6, 2011.

[LLL08] Seung-Hoon Lee, Uichin Lee, Kang-Won Lee, and M. Gerla. "Content Distribution in VANETs Using Network Coding: The Effect of Disk I/O and Processing O/H." In *Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON '08. 5th Annual IEEE Communications Society Conference on*, pp. 117–125, June 2008.

[LLM01] Kevin Leyton-brown, Kevin Leyton-brown, Ilya Mironov, and Mark" Lillibridge. "Incentives For Sharing in Peer-to-Peer Networks." 2001.

[LW09] Feng Li and Jie Wu. "FRAME: An Innovative Incentive Scheme in Vehicular Networks." In *Communications, 2009. ICC '09. IEEE International Conference on*, pp. 1–6, 2009.

[LYL11] Ming Li, Zhenyu Yang, and Wenjing Lou. "CodeOn: Cooperative Popular Content Distribution for Vehicular Networks using Symbol Level Network Coding." *Selected Areas in Communications, IEEE Journal on*, **29**(1):223–235, 2011.

[MCC12a] F. Malandrino, C. Casetti, C. Chiasserini, and M. Fiore. "Optimal Content Downloading in Vehicular Networks.", 2012.

[MCC12b] F. Malandrino, C. Casetti, C. F. Chiasserini, C. Sommer, and F. Dressler. "Content downloading in vehicular networks: Bringing parked cars into the picture." In *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on*, pp. 1534–1539, 2012.

[ME09a] J.R. Marden and M. Effros. "A game theoretic approach to network coding." In *Networking and Information Theory, 2009. ITW 2009. IEEE Information Theory Workshop on*, pp. 147–151, 2009.

[ME09b] J.R. Marden and M. Effros. "The price of selfishness in network coding." In *Network Coding, Theory, and Applications, 2009. NetCod '09. Workshop on*, pp. 18–23, 2009.

[MGL00] Sergio Marti, T. J. Giuli, Kevin Lai, and Mary Baker. "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks." In *Proceedings of the 6th Annual International Conference on Mobile Computing and*

Networking, MobiCom '00, pp. 255–265, New York, NY, USA, 2000. ACM.

[MJS06] Fabio Milan, Juan José Jaramillo, and R. Srikant. "Achieving cooperation in multihop wireless networks of selfish nodes." In *Proceeding from the 2006 workshop on Game theory for communications and networks*, GameNets '06, New York, NY, USA, 2006. ACM.

[MPX13] N. Mastronarde, V. Patel, Jie Xu, and M. Van Der Schaar. "Learning relaying strategies in cellular D2D networks with token-based incentives." In *Globecom Workshops (GC Wkshps), 2013 IEEE*, pp. 163–169, Dec 2013.

[NDP05] A. Nandan, Shirshanka Das, G. Pau, M. Gerla, and M.Y. Sanadidi. "Co-operative downloading in vehicular ad-hoc wireless networks." In *Wireless On-demand Network Systems and Services, 2005. WONS 2005. Second Annual Conference on*, pp. 32–41, Jan 2005.

[NJ12] Kathleen Nichols and Van Jacobson. "Controlling queue delay." *Commun. ACM*, **55**(7):42–50, July 2012.

[PJ08] J. Price and T. Javidi. "Network Coding Games with Unicast Flows." *Selected Areas in Communications, IEEE Journal on*, **26**(7):1302–1316, 2008.

[PS10] Jaeok Park and M. van der Schaar. "Pricing and Incentives in Peer-to-Peer Networks." In *INFOCOM, 2010 Proceedings IEEE*, pp. 1–9, 2010.

[RLM03] Shuo yen Robert Li, Shuo-yen Robert Li, Senior Member, Raymond W. Yeung, and Ning" Cai. "Linear network coding." *IEEE TRANSACTIONS ON INFORMATION THEORY*, **49**:371–381, 2003.

[SBH03] Naouel Ben Salem, Levente Buttyán, Jean-Pierre Hubaux, and Markus Jakobsson. "A charging and rewarding scheme for packet forwarding in multi-hop cellular networks." In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, MobiHoc '03, pp. 13–24, New York, NY, USA, 2003. ACM.

[SMW07] H. Schwarz, D. Marpe, and T. Wiegand. "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard." *Circuits and Systems for Video Technology, IEEE Transactions on*, **17**(9):1103–1120, 2007.

[SSQ08] U. Shevade, Han Hee Song, Lili Qiu, and Yin Zhang. "Incentive-aware routing in DTNs." In *Network Protocols, 2008. ICNP 2008. IEEE International Conference on*, pp. 238–247, Oct 2008.

[SW08] J. Padhye S. Floyd, M. Handley and J. Widmer. "TCP Friendly Rate Control (TFRC): Protocol Specification.", 9 2008.

[TLW00] Tao Tian, A.H. Li, Jiangtao Wen, and J.D. Villasenor. "Priority dropping in network transmission of scalable video." In *Image Processing, 2000. Proceedings. 2000 International Conference on*, volume 3, pp. 400–403 vol.3, 2000.

[WC07] Zhou Wang and Chunxiao Chigan. "Countermeasure Uncooperative Behaviors with Dynamic Trust-Token in VANETs." In *Communications, 2007. ICC '07. IEEE International Conference on*, pp. 3959–3964, 2007.

[WGS12] Chuchu Wu, M. Gerla, and M. van der Schaar. "Social norm incentives for secure network coding in MANETs." In *Network Coding (NetCod), 2012 International Symposium on*, pp. 179–184, 2012.

[WSH12] Tianyu Wang, Lingyang Song, and Zhu Han. "Collaborative data dissemination in cognitive VANETs with sensing-throughput tradeoff." In *Communications in China (ICCC), 2012 1st IEEE International Conference on*, pp. 41–45, 2012.

[XJN07] Yang Xiaogang, Liu JiQiang, and Li Ning. "Congestion Control Based on Priority Drop for H.264/SVC." In *Multimedia and Ubiquitous Engineering, 2007. MUE '07. International Conference on*, pp. 585–589, 2007.

[XS12a] Jie Xu and M. van der Schaar. "Designing incentives for wireless relay networks using tokens." In *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2012 10th International Symposium on*, pp. 169–176, 2012.

[XS12b] Yichao Xu and Kouichi Sakurai. "Cooperatively Securing Network Coding Against Pollution Attacks with Incentive Mechanism." In *Proceedings of the 6th International Conference on Ubiquitous Information Management and Communication*, ICUIMC '12, pp. 52:1–52:10, New York, NY, USA, 2012. ACM.

[XV13] Jie Xu and M. Van Der Schaar. "Token System Design for Autonomic Wireless Relay Networks." *Communications, IEEE Transactions on*, **61**(7):2924–2935, July 2013.

[YWR08] Zhen Yu, Yawen Wei, B. Ramkumar, and Yong Guan. "An Efficient Signature-Based Scheme for Securing Network Coding Against Pollution Attacks." In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pp. 1409–1417, 2008.

[ZCY03] S. Zhong, J. Chen, and Y.R. Yang. “Sprite: a simple, cheat-proof, credit-based system for mobile ad-hoc networks.” In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pp. 1987–1997 vol.3, 2003.

[ZL08] Xinyu Zhang and Baochun Li. “Dice: a game theoretic framework for wireless multipath network coding.” In *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc ’08, pp. 293–302, New York, NY, USA, 2008. ACM.

[ZPP11] Xiaoqing Zhu, Rong Pan, M.S. Prabhu, N. Dukkipati, V. Subramanian, and F. Bonomi. “Layered Internet Video Adaptation (LIVA): Network-Assisted Bandwidth Sharing and Transient Loss Protection for Video Streaming.” *Multimedia, IEEE Transactions on*, **13**(4):720–732, 2011.

[ZS08] Yingwu Zhu and Haiying Shen. “TrustCode: P2P reputation-based trust management using network coding.” In *Proceedings of the 15th international conference on High performance computing*, HiPC’08, pp. 378–389, Berlin, Heidelberg, 2008. Springer-Verlag.

[ZS11] Yu Zhang and Mihaela van der Schaar. “Peer-to-Peer Multimedia Sharing based on Social Norms.” *Archived journal http://arxiv.org/abs/1102.1503*, 02 2011.

[ZY11] Da Zhang and Chai-Kiat Yeo. “A Cooperative Content Distribution System for Vehicles.” In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pp. 1–6, 2011.

[ZZS11] Chi Zhang, Xiaoyan Zhu, Yang Song, and Yuguang Fang. “C4: A new paradigm for providing incentives in multi-hop wireless networks.” In *INFOCOM, 2011 Proceedings IEEE*, pp. 918–926, 2011.