Université de Nice - Sophia Antipolis – UFR Sciences
École Doctorale STIC

THÈSE

Présentée pour obtenir le titre de :
Docteur en Sciences de l'Université de Nice - Sophia Antipolis

Spécialité : INFORMATIQUE

par

Dinesh KUMAR

Thèse dirigée par Dr. Eitan ALTMAN
et préparée à l'équipe MAESTRO - INRIA Sophia Antipolis

Optimisation et Contrôle dans les Réseaux Sans-Fil et Informatiques

Soutenue publiquement à l'INRIA le 26 Novembre 2008 devant le jury composé de :

Président :	Mérouane	DEBBAH	Ecole Supérieure d'Électricité
Directeur :	Eitan	ALTMAN	INRIA Sophia Antipolis
Rapporteurs :	Bruno	TUFFIN	INRIA Rennes
	Tijani	CHAHED	Institut TELECOM SudParis
Examinateurs :	Jean-Marc	KELIF	France Telecom R&D
	Corinne	TOUATI	INRIA Grenoble - Rhône-Alpes

Optimisation et Contrôle dans les Réseaux Sans-Fil et Informatiques

Dinesh Kumar

Titre de la thèse en anglais :

Optimization and Control in Wireless and Computer Networks

To my mother, grand parents and all teachers, professors and colleagues who have imparted knowledge in me.

Thought, knowledge, education, science, technology and moral values
are superior than all religions, cultures and geographies.

They enlighten the God in Man and kill the evil in Him.

Acknowledgments

I would like to acknowledge the co-existence of curiousity and ignorance in the minds of myself and all those with whom I interacted during my Ph.D studies. With the former persistently trying to efface the latter, it transformed my mind from an amateur to a careful thinker.

I do not have enough words to gratefully acknowledge my advisor Dr. Eitan Altman for giving me a priveleged chance and fostering me as his Ph.D student. I am very thankful and indebted to him for providing all the opportunities that kept up a stimulating and learning research environment. His guidance and advisory support greatly helped me to steer through my Ph.D. I admire his intellectual thinking and hardworking capabilities and have been trying to instill the same in myself through out our interaction. I would also like to thank the director of Maestro group Dr. Philippe Nain for supporting all my endeavors such as conference and workshop travels, course work, etc.

I am thankful to Dr. Tijani Chahed and Dr. Bruno Tuffin for carefully reviewing my thesis and providing me with important feedback on various aspects. Many thanks also go to Dr. Mérouane Debbah, Dr. Jean Marc Kelif and Dr. Corinne Touati for being part of the jury.

I wish to express my gratitude towards France Telecom R&D and Dr. Jean Marc Kelif for giving me an opportunity to work in collaboration with them on the user-network association and 3G UMTS related problems. This collaboration partly financed my thesis work.

My deep gratitude goes to Dr. Anurag Kumar and Dr. Arzad Alam Kherani for hosting me as a visiting scholar at IISc and IIT-Delhi and for giving me a priveleged chance to have enriching and fruitful discussions with them on MANET and VANET related work.

It was a great privelege to pursue a short summer internship with Prof. Tamer Basar at UIUC. Though our work on Wiener filters and H^∞ control remains incomplete, it was surely an enriching experience to work with him. I do plan to finalize our work.

My sincere thanks go to Dr. Li Zhang and Dr. Zhen Liu at IBM Research for offering me an internship position to work on the real-time performance modeling and Inferencing related work. I am also thankful to Dr. Asser Tantawi for all our collaborative work. Research and prototype development experience at IBM boosted my motivation to pursue my career in an industry research lab.

At INRIA, my research work was carried out in a very lively and friendly atmosphere. My appreciations go to Giovanni Neglia, Nicolas Bonneau, Natalia Osipova, Balakrishna Prabhu, Ahmad Al Hanbali, Mouhamad Ibrahim, Danil Nemirovsky, Sara Alouf, Konstantin Avrachenkov, Avik Bhattacharya, Robin Groenevelt and Alonso Silva for creating a cheerful and supportive atmosphere. Special thanks and gratitude goes to Ephie Deriche, our administrative assistant, for helping in all administrative and travel related issues and for all her cheerful discussions and support. I also thank my friends outside INRIA, Simona Jakobson, Francois Sabatier, Guillaume Bonvillain, Raghav Bhaskar, Parijat Dube, Pauline Duprez, Amar Prakash Azad and Laura Cottatellucci for their moral and cheerful support.

Lastly, I would like to thank all my family members for their consistent moral support throughout my Ph.D. Specifically, I dedicate this thesis to my mother: Om Kumari, my grand parents: Niranjan Daas Sachdeva and Prakaash Vati Sachdeva, and all my teachers and professors at school and university who made me capable of pursuing this Ph.D.

Dinesh Kumar

September 2008
INRIA Sophia Antipolis

Table of Contents

Chapter 1

Introduction

Wireless communication and computer networks have experienced an exponential growth over the last decade. Cellular phones and wireless connectivity devices have become a critical business tool and part of everyday life in many countries. While third generation (3G) cell phone technologies such as Universal Mobile Telecommunications System (UMTS) have recently been deployed in some countries, wireless local area networks (WLANs) such as those based on IEEE 802.11 standard have been supplementing or replacing wired networks in many homes, businesses and campuses since a couple of years. Many industrial and government organizations have also been involved in developing wireless vehicular ad hoc networks (VANETs) to enable communications among nearby vehicles and between vehicles and nearby roadside network infrastructure.

The explosive growth of wireless systems coupled with the proliferation of wireless devices such as 3G phones, WiFi laptops and vehicular wireless devices, suggests a bright future for wireless networks, both as stand-alone ad hoc systems and as part of the larger networking infrastructure. However, many technical challenges still remain in designing robust wireless networks that can deliver the necessary performance for supporting emerging applications. These technical challenges have given rise to numerous research problems that have attracted significant attention in the research community. This thesis deals with some of these research problems and provides several new results that can be used to improve the performance of existing wireless systems and technology. In addition to wireless systems, we also study and present an innovative approach for performance modeling of computer network systems in *real-time*.

Wireless networks can be broadly classified into two categories, namely, wireless *access* networks and wireless *ad hoc* networks. Wireless access networks usually provide connectivity to the wired infrastructure network through the wireless medium. Examples of wireless access networks include cellular systems such as 2G GSM, 3G UMTS, etc. and WiFi sys-

tems such as 802.11 WLANs. Wireless ad hoc networks on the other hand are decentralized wireless networks that may be setup and dismantled dynamically. Wireless ad hoc networks may be further classified as mobile ad hoc networks (MANETs), wireless mesh networks and wireless sensor networks. VANETs are a special kind of MANETs that exhibit drastically different behavior from the usual MANETs. In this thesis, we shall limit our focus to the study of 3G UMTS and 802.11 WLAN access technologies and mobile and vehicular ad hoc networks.

With declining hardware costs, some devices are now available in the market that provide both WiFi and cellular connectivity at affordable prices. These devices can detect the presence of either of the two networks and switch between them when available. With more and more users subscribing to wireless broadband services, these devices allow them to have access to both WLAN hot-spot and UMTS cellular networks. Thus network operators are seeking to offer seamless and ubiquitous connectivity to mobile users through integrated 802.11 WLAN and 3G UMTS *hybrid* networks. Such hybrid networks are being increasingly studied by network operators as a converged medium of providing connectivity for both voice and data applications. In this thesis, we have two chapters dedicated to the study of optimal user-network association in a WLAN and UMTS hybrid cell.

Power control and PHY data rate control are two central mechanisms that are critical in achieving an efficient functioning of a WLAN. Power control and rate control are also very often used for providing quality of service (QoS) and both are useful for obtaining a radio channel with a low bit error rate (BER). The IEEE 802.11 specification for mobile devices allows customization of certain critical operation parameters like PHY data rate and MAC layer frame size and any manufacturer or even user can adaptively select the PHY data transmission rates for rate control. For this purpose, several auto-rate selection algorithms [KL97, HVB01, GCNC01, AKKD01, QCJS03] have been proposed in the literature and most of them allow WLAN nodes to adapt their PHY rates non-cooperatively, by definition. In this thesis, we have one chapter that studies the inefficiency of IEEE 802.11 MAC protocol under non-cooperative power and rate control.

Fountain Codes are rateless erasure codes and offer a very promising future for ameliorating existing data packet transmission techniques. Specific type of fountain codes such as Luby Transform (LT) Codes, Raptor codes, etc., can be used by a sender to generate encoded packets from source data packets *on the fly* and the number of such encoded packets can be potentially limitless. If the original file at sender side comprises N_p packets, then by decoding *any* set of $N_p(1+\epsilon)$ (slightly more than N_p) fountain coded packets received, the receiver can recover the whole file with probability $1-\delta$. The probability of failure to decode the file, δ, is bounded above by $\delta \leq 2^{-\epsilon N_p}$ and depends on the *degree distribution* used to code the packets at sender side. A Fountain Codes based Transport (FCT) protocol relies on an alternate paradigm to that of the ubiquitous TCP. It abolishes the need for a reverse feedback mechanism usually essential to provide reliability in packet data transmission. Absence of a reverse feedback mechanism can substantially improve the performance of networks with half-duplex wireless channels (such as 802.11 WLANs), where collisions between forward and reverse MAC frame transmissions contribute significantly towards performance degradation. In this thesis, we have one chapter dedicated to performance modeling and

analysis of a simple FCT protocol in a single cell IEEE 802.11 WLAN.

There are two types of layer-2 downlink transport channels that have been provided in UMTS: *dedicated* channels and *common* channels. A common channel is a resource shared between all or a group of users in a cell, where as a dedicated channel is a resource identified by a certain code on a certain frequency and is reserved for a single user only. The only dedicated channel is termed as DCH and one of the six common transport channels that is mainly used for packet data on the downlink is the FACH channel [HT01]. According to the WCDMA (Wideband-CDMA) specifications detailed by the 3GPP group, for a particular user, long flows with large number of packets can be transmitted on the user dedicated DCH channel and short flows of few packets can be transmitted on the common FACH channel which is shared by all users. However, the 3GPP specifications do not provide any standardization of such a channel selection/switching policy. A network operator is free to choose its own proprietary channel switching policy. In this thesis, we have one chapter in which we propose a new *cross-layer* channel switching policy for TCP transmission in 3G UMTS.

Mobile ad hoc networks (MANETs) consist of wireless mobile nodes that can freely and dynamically self-organize. In this way they form arbitrary and temporary ad hoc network topologies, allowing devices to seamlessly interconnect in areas with no pre-existing infrastructure. MANETs have been a popular subject for research since the widespread use of 802.11 WiFi devices. In this thesis, we study optimal next hop distance that maximizes the system end-to-end flow throughput in a MANET subject to network average power constraints.

Vehicles equipped with communication devices can form Vehicular ad hoc networks (VANETs) for tasks such as inter-vehicle collision avoidance, road-accident notification, traffic situation update, coordinated driving systems or simply inter-vehicle voice communication. Like MANETs, VANETs do not rely on any fixed infrastructure and instead depend on intermediate relay nodes for route establishment protocols and data transmission. However, VANETs tend to exhibit a drastically different behavior from the usual MANETs. High speeds of vehicles, mobility constraints on a straight road and driver behavior are some factors due to which VANETs possess very different characteristics from the typical MANETs. VANETs are being researched upon and developed by many industrial and government organizations. In this thesis, we have one chapter dedicated to the study of optimal next hop selection in a route between two vehicles for a simple scenario of VANETs on a fast track highway. There is another related chapter that discusses in detail a special structure inherent to the solution of Dynamic Programming (DP) problem arising in path optmization problems such as the one considered for VANETs.

Performance modeling of computer network systems is a crucial step in capacity planning of computer networks. Performance models of complex software and hardware network architectures can be very helpful in accurately predicting their performance for varying data traffic patterns and workloads. Performance modeling becomes more challenging if the workload (e.g., number of jobs or transactions handled per unit of time) for an IT system used in production environment is non-stationary in nature and has fast changing characteristics. Moreover, state of the art transaction-based software applications are quite complex

in design. Due to this, end-to-end transactions may incur workload dependent and variable service times. In this thesis, we have one chapter where we study and present an innovative approach of using Kalman filtering with queueing theory based performance models. This approach allows us to pursue *real-time* performance modeling of online computer network systems that process fast changing non-stationary workload and have workload dependent service times.

1.1 Thesis Organization and Contribution

This thesis may be considered to be divided into two parts. The first major part comprises Chapters 2-9 that deal with various research problems related to optimization and control in wireless networks. While Chapters 2-6 deal with wireless access networks, the focus of Chapters 7-9 is on wireless ad hoc networks. The second part comprises Chapter 10 that presents estimation of workload dependent model parameters and real-time performance modeling of computer network systems.

In Chapter-2, we study globally optimal user-network association in an integrated 802.11 WLAN and 3G UMTS *hybrid cell*. The association problem is formulated as a generic SMDP (semi-Markov decision process) connection routing decision problem. We then solve this SMDP problem using a particular network model for WLAN and UMTS networks and with rewards comprising financial and aggregate throughput components. The network model assumes saturated resource allocation in WLAN and UMTS networks and a single QoS class of mobiles arriving at an *average* location in the hybrid cell. The corresponding Dynamic Programming equation is solved using Value Iteration and a stationary optimal policy with neither convex nor concave type switching curve structure is obtained. Threshold type and symmetric switching curves are observed for the analogous homogenous network cases.

In Chapter-3, we study individually optimal user-network association in an integrated WLAN and UMTS *hybrid cell*. The association problem is formulated within a non-cooperative game framework. In the formulation, mobile arrivals are assumed to follow the Poisson process and each mobile considers its average service time in each network as the decision criteria to connect to either of the WLAN or UMTS networks. We seek to compute the optimal association or decision policy that achieves the Nash equilibrium. For this we develop a generic system of linear equations for estimating the average service time of a mobile. This system is then solved assuming a particular model for the WLAN and UMTS networks and we explicitly compute the optimal association policy that is observed to possess a descending staircase curve structure.

The IEEE 802.11 MAC protocol, DCF (Distributed Coordination Function), allows nodes in a WLAN to choose an appropriate PHY data transmission rate for each frame transmission. For this purpose, several auto-rate selection algorithms have been proposed in the literature and most of them allow nodes to adapt their rates non-cooperatively, by definition. Under a non-cooperative game setting, each node would choose its rate so as to optimize its own, appropriately defined payoff. In Chapter-4, we formulate a payoff function comprising throughput and costs related to power consumption and derive explicit

expressions for the optimal rates under non-cooperative and cooperative rate selection. We consider optimization problems for both finite number of nodes n and for the limit $n \to \infty$ and single node throughputs corresponding to the optimal PHY rate under non-cooperative game are compared with those obtained under a cooperative selection of PHY rates. The comparisons reveal that network performance in the non-cooperative game scenario is inefficient as compared to the cooperative scenario.

In Chapter-5, we propose a Markovian stochastic framework to model the performance of a simple Fountain Codes based Transport (FCT) protocol in a single cell IEEE 802.11 WLAN. Our model allows the WLAN Access Point to employ a generic rate control algorithm for MAC frame transmissions on the downlink. Using renewal theory we provide an explicit expression for the average downlink throughput. *ns2* simulations are used to validate our model and the analytically obtained throughput metric. A detailed performance analysis study is then carried out to provide insights into the choice of various system parameters that can lead to optimal network performance. Finally we present a brief comparison between the performance of FCT and TCP through simulations.

In 3G UMTS, two main transport channels have been provided at the layer-2 (MAC) for downlink data transmission: a common FACH channel and a dedicated DCH channel. The performance of TCP in UMTS depends much on the channel switching policy used. In Chapter-6, we first propose and analyze three new basic threshold-based channel switching policies for UMTS that we name as QS (Queue Size), FS (Flow Size) and QSFS (QS & FS combined) policy. These policies significantly improve over a 'modified threshold policy' in [PAAD03] by about 17% in response time metrics. We further propose and evaluate a new improved switching policy that we call FS-DCH (*at-least* flow-size threshold on DCH) policy. This policy is biased towards short TCP flows of few packets. It is thus a *cross-layer* policy that improves the performance of TCP by giving priority to the initial few packets of a flow on the fast DCH channel. Extensive simulation results show that FS-DCH policy improves over others by about 30% to 36% in response time metrics for a particular case.

In Chapter-7, in a dense multi-hop network of mobile nodes capable of applying adaptive power control, we consider the problem of finding the optimal hop distance that maximizes a certain throughput measure in *bit-metres/sec*, subject to average network power constraints. The mobility of nodes is restricted to a circular periphery area centered at the nominal location of nodes. We incorporate only randomly varying path-loss characteristics of channel gain due to the random motion of nodes, excluding any multi-path fading or shadowing effects. Computation of the throughput metric in such a scenario leads us to compute the probability density function of random distance between points in two circles. Using numerical analysis we discover that choosing the nearest node as next hop is *not always* optimal. Optimal throughput performance is also attained at non-trivial hop distances depending on the available average network power.

The main goal of Chapter-8 is to better understand the routing dynamics in VANETs that are a special class of MANETs but exhibit very different behavior from them. We consider the problem of optimal next hop selection in a route between two vehicles for a simple scenario of VANETs on a fast track highway. For a given choice of number of hops between the source and destination, we seek the optimal choice of next hop based on its

speed and inter-node distance so as to maximize the expected *route lifetime*. Our analytical model accounts for the randomly changing speeds of nodes (vehicles) over time and hence the optimal choice depends on the dynamics of the stochastic process corresponding to the speed of nodes. Under a Markovian assumption on the process of speed of nodes, we show that the optimal choice of speeds is such that the lifetimes of adjacent links are as close as possible. Explicit expressions for optimal choice of next hop node's speed and inter-node distance are obtained for certain fast track highway scenarios of interest. A monotone variation property of the speed of relay nodes under the optimal policy is proved. These properties have been confirmed with simulations. The optimal policies and their structures can assist in enhancing the performance of existing VANET routing protocols.

The inherent nature of the physical setup and transmission mechanism in wireless ad hoc networks with random channel access, results in correlation between the *link metrics* of adjacent links, when considering path optimization problems. In Chapter-9, we identify a special structure inherent to the solution of Dynamic Programming (DP) problem arising in such an optimization over paths. According to this structure, the optimal policy tries to *equalize* the link metrics of adjacent links in a multi-hop route. We validate this structural property with a VANET simulator.

In Chapter-10, we are concerned with performance modeling of transaction-based distributed software applications deployed over an arbitrary computer network architecture. AMBIENCE, which is a research prototype tool that has been developed at IBM Research, makes use of the powerful *Inferencing* technique that allows one to generate a workload-independent parameters based performance model. In this work, we extend the Inferencing technique for generating arriving workload dependent parameter based performance models. We call this extended form of Inferencing as *Enhanced Inferencing*. Implementation of this Enhanced Inferencing in AMBIENCE shows significant improvement in performance model fitting and approximation. We further present an innovative approach of using Kalman filtering with Inferencing like performance models in order to be able to pursue *real time* performance modeling of production environment computer application systems that face fast changing *non-stationary* arriving workloads. This real time performance model not only works reliabily for *non-stationary* workloads, but also incorporates *workload dependent* model parameters.

Chapter 2

Globally Optimal User-Network Association in a WLAN & UMTS Hybrid Cell

With more and more users subscribing to wireless broadband services, it is desirable for them to have access to both WLAN hot-spot and UMTS cellular networks. In this chapter, we study globally optimal user-network association in an integrated 802.11 WLAN and 3G UMTS *hybrid cell*. The association problem is formulated as a generic SMDP (semi-Markov decision process) connection routing decision problem. We then solve this SMDP problem using a particular network model for WLAN and UMTS networks and with rewards comprising financial and aggregate throughput components. The network model assumes saturated resource allocation in WLAN and UMTS networks and a single QoS class of mobiles arriving at an *average* location in the hybrid cell. The corresponding Dynamic Programming equation is solved using Value Iteration and a stationary optimal policy with neither convex nor concave type switching curve structure is obtained. Threshold type and symmetric switching curves are observed for the analogous homogenous network cases.

Note: The material in this chapter has appeared in [KAK07a]. This work was sponsored by France Telecom R&D.

2.1 Introduction

As 802.11 WLANs and 3G UMTS cellular coverage networks are being widely deployed, network operators are seeking to offer seamless and ubiquitous connectivity for high-speed wireless broadband services, through integrated WLAN and UMTS hybrid networks. For efficient performance of such an hybrid network, one of the core decision problems that a network operator is faced with is that of optimal user-network association or load balancing, i.e., optimally routing an arriving mobile user's connection to one of the two constituent networks. We study this decision problem assuming a specific network model for the WLAN and UMTS networks. To be more precise, consider a hybrid network comprising two independent 802.11 WLAN and 3G UMTS networks, that offers connectivity to mobile users arriving in the combined coverage area of these two networks. By independent we mean that transmission activity in one network does not create interference in the other. Our goal in this chapter is to study the dynamics of optimal user-network association in such a WLAN-UMTS hybrid network. We concentrate only on streaming and interactive data transfers. Moreover, we consider only a single QoS class of mobiles arriving at an *average* location in the hybrid cell and these mobiles have to be admitted to one of the WLAN and UMTS networks.

Note that we do not propose a full fledged cell-load or interference based connection admission control (CAC) policy in this work. We instead assume that a CAC precedes the association decision control. A connection admission decision is taken by the CAC controller before any mobile is considered as a candidate to be connected to either of the WLAN or UMTS networks. Thereafter, an association decision only ensures an optimal performance of the hybrid cell and it is not proposed as an alternative to the CAC decision. However, the association decision controller can still reject mobiles for optimal performance of the network.

In the network model for WLAN and UMTS networks, we introduce certain simplifying assumptions to make the SMDP formulation analytically tractable. Without these assumptions it may be very hard to study the dynamics of user-network association in a WLAN-UMTS hybrid network.

2.1.1 Related Work and Contributions

Study of WLAN-UMTS hybrid networks is an emerging area of research and not much related work is available. Authors in some related papers [MYLR04, SLC03, LZ05, Jas03, VGN05, KRPR05, FC05] have studied issues such as vertical handover and coupling schemes, integrated architecture layout, radio resource management (RRM) and mobility management. However, questions related to load balancing or optimal user-network association have not been explored much. Premkumar et al. in [PK06] propose a *near optimal* solution for a hybrid network within a combinatorial optimization framework, which is different from our approach. To the best of our knowledge, ours is the first attempt to present a generic formulation of the user-network association problem under an SMDP decision control framework. Moreover, this work is the first we know of that obtains an explicit optimal association policy for the specific WLAN-UMTS hybrid network model that we consider.

2.2 Framework for the Decision Control Problem

A hybrid network may be composed of several 802.11 WLAN Access Points (APs) and 3G UMTS Base Stations (NodeBs) that are operated by a single network operator. However, our focus is only on a single pair of an AP and a NodeB that are located sufficiently close to each other so that mobile users arriving in the combined coverage area of this AP-NodeB pair have a choice to connect to either of the two networks. We call the combined coverage area of a single AP cell and a single NodeB micro-cell [HT01] as a *hybrid cell*. The cell coverage radius of a UMTS micro-cell is usually around $400m$ to $1000m$ whereas that of a WLAN cell varies from a few tens to a few hundreds of meters. Therefore, some mobiles arriving in the hybrid cell may only be able to connect to the NodeB, either because they fall outside the transmission range of the AP or they are equipped with only 3G technology electronics. While other mobiles that are equipped with only 802.11 technology can connect exclusively to the WLAN AP. Apart from these two categories, mobiles equipped with both 802.11 WLAN and 3G UMTS technologies can connect to any one of the two networks.

The decision to connect to either of the two networks may involve a utility criteria that could comprise the total throughput of the hybrid network. Moreover, the connection or association decision involves two different decision makers, the mobile user and the network operator. We assume that the mobile user is selfish and takes its decision only to maximize its own utility while ignoring overall network performance. With this assumption, leaving the decision choice with the mobile user may result in less efficient use of the network resources. We thus focus only on the globally optimal control problem in which the network operator dictates the decision of mobile users to connect to one of the two networks, so as to optimize a certain global cell utility. In Section 2.3, we model this global optimality problem under an SMDP (semi-Markov decision process) control framework. Our SMDP control formulation is a *generic* formulation of the user-network association problem in a WLAN-UMTS hybrid network and is independent of the network model assumed for WLAN and UMTS networks. Thereafter in Section 2.5, we solve the SMDP problem assuming a particular network model (described in Section 2.4) which is based on some reasonable simplifying assumptions.

2.2.1 Mobile Arrivals

We model the hybrid cell of an 802.11 WLAN AP and a 3G UMTS NodeB as a two-server processing system (Figure 2.1) with each server having a separate finite pole capacity of M_{AP} and M_{3G} mobiles, respectively. We will give further clarifications on the pole capacity of each server later in Sections 2.4.2 and 2.4.3. For simplification we assume that mobile users are stationary, having no mobility. As discussed previously, mobiles are considered as candidates to connect to the hybrid cell only after being admitted by a CAC such as the one described in [YK05]. Some of the admitted mobiles can connect only to the WLAN AP and some others only to the UMTS NodeB. These two set of arriving mobiles are each assumed to constitute two separate dedicated arrival streams with Poisson rates λ_{AP} and λ_{3G}, respectively. The remaining set of mobiles which can connect to both networks form a common arrival stream with Poisson rate λ_{AP3G}. The mobiles of the two dedicated streams can either directly join

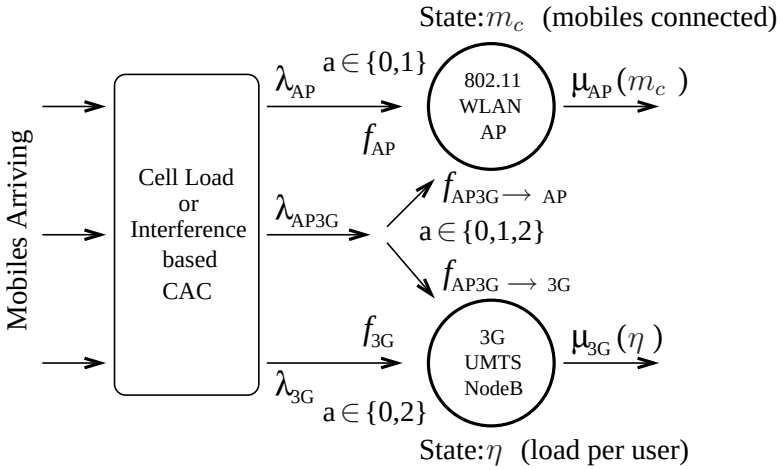


Figure 2.1: Hybrid cell scenario

their respective AP or NodeB network without any connection decision choice involved, or they can be rejected. For mobiles of common stream, either a rejection or a connection routing decision has to be taken as to which of the two networks will the arriving mobiles join while optimizing a certain utility.

2.2.2 Service Requirements and Departure Rates

It is assumed that all arriving mobiles have a downlink data service requirement which is exponentially distributed with parameter ζ. In other words, every arriving mobile seeks to download a data file of average size $1/\zeta$ bits on the downlink. Let $\theta_{AP}(m_c)$ denote the downlink throughput of each mobile in the AP network when m_c mobiles are connected to it at any given instant. If η_{DL} denotes the downlink *cell load* of the NodeB cell, then assuming N active mobiles to be connected to the NodeB, $\eta \stackrel{\Delta}{=} \frac{\eta_{DL}}{N}$ denotes the average *load per user* in the cell. Let $\theta_{3G}(\eta)$ denote the downlink throughput of each mobile in the NodeB network when its average load per user is η. With the above notations, the effective departure rates of mobiles in each network or server can be denoted by,

$$\mu_{AP}(m_c) = \zeta \times \theta_{AP}(m_c) \tag{2.1}$$

and

$$\mu_{3G}(\eta) = \zeta \times \theta_{3G}(\eta). \tag{2.2}$$

2.3 SMDP Control Formulation

As mentioned previously, for a globally optimal decision control it is the network operator that takes the decision for each mobile as to which of the AP or NodeB networks the mobile will connect to, after it has been admitted into the hybrid cell by the CAC controller (Figure 2.1). Since decisions have to be made at each arrival epoch, this gives an SMDP structure

(see Chapter 11 in [Put94]) to the decision problem and we state the equivalent SMDP problem as follows:

- *States:* The state of a hybrid cell system is denoted by the tuple (m_c, η) where m_c ($m_c \in \mathbb{Z}, 0 \leq m_c \leq M_{AP}$) denotes the number of mobiles connected to the AP and η ($\eta \in \mathbb{R}, 0.05 \leq \eta \leq 0.9$) is the load per user of the NodeB cell (see Sections 2.4.2 & 2.4.3 for details on bounds).

- *Events:* We consider two distinguishable events: (i) arrival of a new mobile after it has been admitted by CAC and (ii) departure of a mobile after service completion.

- *Decisions:* For mobiles arriving in the common stream a decision action $a \in \{0, 1, 2\}$ has to be taken. $a = 0$ represents rejecting the mobile, $a = 1$ represents routing the mobile connection to AP network and $a = 2$ represents routing the mobile connection to NodeB network. For the dedicated arrival streams to AP and NodeB, a decision action $a \in \{0, 1\}$ and $a \in \{0, 2\}$, respectively, has to be taken.

- *Rewards:* Whenever a new incoming mobile is either rejected or routed to one of the two networks, it generates a certain *state-dependent* reward. $R_{AP}(m_c, \eta; a)$ and $R_{3G}(m_c, \eta; a)$ denote the rewards generated at dedicated arrival streams for AP and NodeB, respectively, when action 'a' is taken and the state of the system is (m_c, η). Similarly, $R_{AP3G}(m_c, \eta; a)$ denotes the reward generated at the common stream.

- *Criterion:* The optimality criterion is to maximize the total expected discounted reward over an infinite horizon and obtain a *deterministic* and *stationary* optimal policy.

Note that in the SMDP problem statement above, state transition probabilities have not been mentioned because depending on the action taken, the system moves into a unique new state deterministically, i.e., w.p. 1. For instance when action $a = 1$ is taken, the state evolves from (m_c, η) to the unique new state $(m_c + 1, \eta)$.

Applying the well-known *uniformization* technique from [Lip75], we can say that events (i.e., arrival or departure of mobiles) occur at the jump times of the combined Poisson process of all types of events with rate $\Lambda := \lambda_{AP} + \lambda_{3G} + \lambda_{AP3G} + \check{\mu}_{AP} + \check{\mu}_{3G}$, where $\check{\mu}_{AP} := \max_{m_c} \mu_{AP}(m_c)$ and $\check{\mu}_{3G} := \max_{\eta} \mu_{3G}(\eta)$. The departure of a mobile is now considered as either a real departure, or an *artificial* departure [Lip75] when from a single mobile's point of view the corresponding server slows down due to large number of mobiles in the network. Then, any event occurring corresponds to an arrival on the dedicated streams with probability λ_{AP}/Λ and λ_{3G}/Λ, an arrival on the common stream with probability λ_{AP3G}/Λ, a real departure with probability $\mu_{AP}(m_c)/\Lambda$ or $\mu_{3G}(\eta)/\Lambda$ and an artificial departure with probability $1 - (\lambda_{AP} + \lambda_{3G} + \lambda_{AP3G} + \mu_{AP}(m_c) + \mu_{3G}(\eta))/\Lambda$. As a result, the time *periods* between consecutive events (including artificial departures) are i.i.d. and we can consider an $n-$stage SMDP decision problem [Put94]. Let $V_n(m_c, \eta)$ denote the maximum expected $n-$stage discounted reward for the hybrid cell when the system is in state (m_c, η). The stationary optimal policy that achieves the maximum total expected discounted reward over

an infinite horizon can then be obtained as a solution of the $n-$stage problem as $n \to \infty$ [Put94].

The discount factor is denoted by γ ($\gamma \in \mathbb{R}, 0 < \gamma < 1$) and determines the relative worth of present reward v/s future rewards. State (m_c, η) of the system is observed right after the occurrence of an event, for example, right after a newly arrived mobile in the common stream has been routed to one of the two networks, or right after the departure of a mobile. Given that an arrival event has occurred and that action 'a' will be taken for this newly arrived mobile, let $U_n(m_c, \eta; a)$ denote the maximum expected $n-$stage discounted reward for the hybrid cell when the system is in state (m_c, η). We can then write down the following recursive Dynamic Programming (DP) [Put94] equation to solve our SMDP decision problem, $\forall n \geq 0$ and $0 \leq m_c \leq M_{AP}$, $0.05 \leq \eta \leq 0.9$,

$$\begin{aligned} V_{n+1}(m_c, \eta) = & \frac{\lambda_{AP}}{\Lambda} \max_{a \in \{0,1\}} \{R_{AP}(m_c, \eta; a) + \gamma U_n(m_c, \eta; a)\} \\ & + \frac{\lambda_{3G}}{\Lambda} \max_{a \in \{0,2\}} \{R_{3G}(m_c, \eta; a) + \gamma U_n(m_c, \eta; a)\} \\ & + \frac{\lambda_{AP3G}}{\Lambda} \max_{a \in \{0,1,2\}} \{R_{AP3G}(m_c, \eta; a) + \gamma U_n(m_c, \eta; a)\} \\ & + \frac{\mu_{AP}(m_c)}{\Lambda} \gamma V_n(m_c', \eta) \\ & + \frac{\mu_{3G}(\eta)}{\Lambda} \gamma V_n(m_c, \eta') \\ & + \frac{\Lambda - (\lambda_{AP} + \lambda_{3G} + \lambda_{AP3G} + \mu_{AP}(m_c) + \mu_{3G}(\eta))}{\Lambda} \gamma V_n(m_c, \eta), \end{aligned} \tag{2.3}$$

where, states (m_c', η) and (m_c, η') are the new states that the system evolves into when a departure occurs at AP and NodeB, respectively. The fact that dedicated stream mobiles can only join one network or the other has been incorporated in the first two terms in R.H.S. Equation 2.3 is a very generic formulation of our user-network association decision problem and it can be solved using any particular definition for the rewards and the new states (m_c', η) and (m_c, η'). In Section 2.5, we will solve the DP formulation of Equation 2.3 assuming a specific definition for the rewards based on throughput expressions obtained from a specific network model for the WLAN and UMTS networks. We first present this network model in the following section along with some simplifying assumptions.

2.4 WLAN and UMTS Network Models

Before discussing the network models adopted from previous work, we first state below some simplifying assumptions along with their justification. Since the bulk of data transfer for a mobile engaged in streaming or interactive data transmission is carried over the downlink

(AP to mobile or NodeB to mobile) and since TCP is the most common transport protocol, we are interested here in network models for computing TCP throughput on only downlink.

2.4.1 Simplifying Assumptions

2.4.1.1 Assumption on QoS and TCP:

We assume a single QoS class of arriving mobiles so that each mobile has an identical minimum downlink throughput requirement of θ_{min}, i.e., each arriving mobile must achieve a downlink throughput of at least θ_{min} bps in either of the two networks. It is further assumed that each mobile's or receiver's advertised window W^* is set to 1 in the TCP protocol. This is in fact known to provide the best performance of TCP (see [ea03, LP05] and references therein).

2.4.1.2 Resource allocation in AP:

We further assume *saturated resource allocation* in the downlink of AP and NodeB networks. Specifically, this assumption for the AP network means the following. Assume that the AP is *saturated* and has infinitely many packets backlogged in its transmission buffer. In other words, there is always a packet in the AP's transmission buffer waiting to be transmitted to each of the connected mobiles. Now, in a WLAN cell resource allocation to an AP on the downlink is carried out through the contention based DCF (Distributed Coordination Function) protocol. If the AP is saturated for a particular mobile's connection and W^* is set to 1, then this particular mobile can benefit from higher number of transmission opportunities ($TxOPs$) won by the AP for downlink transmission to this mobile (hence higher downlink throughput), than if the AP was not saturated or W^* was not set to 1. Thus with the above assumptions, mobiles can be allocated downlink throughputs greater than their QoS requirements of θ_{min} and cell resources in terms of $TxOPs$ on the downlink will be maximally utilized.

2.4.1.3 Resource allocation in NodeB:

For the NodeB network the saturated resource allocation assumption has the following elaboration. It is assumed that at any given instant, the NodeB cell resources on downlink are fully utilized resulting in a constant maximum cell load of η_{DL}^{max}. This is analogous to the maximal utilization of $TxOPs$ in the AP network discussed in the previous paragraph. With this maximum cell load assumption even if a mobile has a minimum throughput requirement of only θ_{min} bps, it can actually be allocated a higher throughput if additional unutilized cell resources are available, so that the cell load is always at its maximum of η_{DL}^{max}. If say a new mobile j arrives and if it is possible to accommodate its connection while maintaining the QoS requirements of the presently connected mobiles (this will be decided by the CAC), then the NodeB will initiate a *renegotiation* of QoS attributes (or bearer attributes) procedure with all the presently connected mobiles. All presently connected

mobiles will then be allocated a lower throughput than the one prior to the set-up of mobile j's connection. However, this new lower throughput will still be higher than each mobile's QoS requirement. This kind of a renegotiation of QoS attributes is indeed possible in UMTS and it is one of its special features (see Chapter 7 in [HT01]). Also note a very key point here that the average load per user η as defined previously in Section 2.2.2, decreases with increasing number of mobiles connected to the NodeB. Though the total cell load is always at its maximum of η_{DL}^{max}, contribution to this total load from a single mobile (i.e., load per user, η) decreases as more mobiles connect to the NodeB cell. We define $\Delta(\eta)$ as the average change in η caused by a new mobile that connects to the NodeB cell. Therefore, when a new mobile connects the load per user drops from η to $\eta - \Delta(\eta)$ and when a mobile disconnects the load per user increases from η to $\eta + \Delta(\eta)$.

2.4.1.4 Power control & location of mobiles in NodeB:

In downlink, the inter-cell to intra-cell interference ratio denoted by i_j and the orthogonality factor denoted by α_j are different for each mobile j depending on its location in the NodeB cell. Moreover, the throughput achieved by each mobile is interference limited and depends on the signal to interference plus noise ratio (SINR) received at that mobile. Thus, in the absence of any power control the throughput also depends on the location of mobile in the NodeB cell. We however assume a uniform SINR scenario where closed-loop fast power control is applied in the NodeB cell so that each mobile receives approximately the same SINR (see Section 3.5 in [HT01]). We therefore assume that all mobiles in the NodeB cell are allocated equal throughputs. This kind of a power control will allocate more power to users far away from the NodeB that are subject to higher path-loss, fading and neighboring cell interference. Users closer to the NodeB will be allocated relatively less power since they are susceptible to weaker signal attenuation. In fact, such a fair throughput allocation can also be achieved by adopting a fair and power-efficient channel dependent scheduling scheme as described in [ZHZ03]. Now since all mobiles are allocated equal throughputs, it can be said that mobiles arrive at an *average* location in the NodeB cell (see Section 8.2.2.2 in [HT01]). Therefore all mobiles are assumed to have an identical average inter-cell to intra-cell interference ratio $\bar{i}$ (see Section 8.2.2.2 in [HT01]) and an identical average orthogonality factor $\bar{\alpha}$ (see Section 8.2.2.2 in [HT01]).

2.4.1.5 Justification:

The assumption on saturated resource allocation is a standard assumption, usually adopted to simplify modeling of complex network frameworks like those of WLAN and UMTS (see for e.g., [HT01, KAMG05a]). Mobiles in NodeB cell are assumed to be allocated equal throughputs in order to have a comparable scenario to that of an AP cell, in which mobiles are also known to achieve fair and equal throughput allocation (see Section 2.4.2). Moreover, such fair throughput allocation is known to result in a better delay performance for typical file transfers in UMTS [Bon04]. The assumption of mobiles arriving at an average location in the NodeB cell is essential in order to simplify our SMDP formulation. For instance,

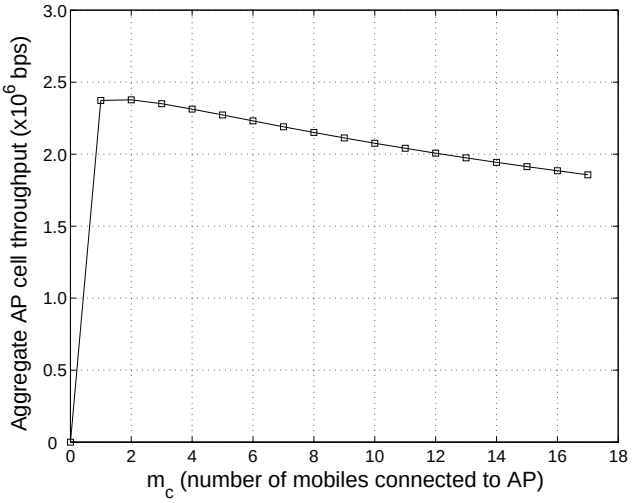


Figure 2.2: Throughput of all users in AP cell

without this assumption the hybrid network system state will have to include the location of each mobile. This will result in an SMDP problem with higher dimensional state space which is known to be analytically intractable and not have an exact solution [Put94]. We therefore assume mobiles arriving at an average location and seek to compute the optimal association policy more from a network planning and dimensioning point of view.

2.4.2 Downlink Throughput in 802.11 WLAN AP

We reuse the downlink TCP throughput formula for a mobile in a WLAN from [MKA06]. For completeness, here we briefly mention the network model that has been extensively studied in [MKA06] and then simply restate the throughput expression without going into much details. Each mobile connected to the AP uses the Distributed Coordination Function (DCF) protocol with an RTS/CTS frame exchange before any data-ack frame exchange and each mobile (including the AP) has an equal probability of the channel being allocated to it. The AP does not employ any rate control algorithm and transmits at a fixed PHY data rate of R_{data} bps to all mobiles. With the assumption of W^* being set to 1 (Section 2.4.1), any mobile will always have a TCP ack waiting to be sent back to the AP with probability 1/2, which is also the probability that it contends for the channel. This is however true only for those versions of TCP that do not use delayed acks. If the AP is always saturated or backlogged, the average number of backlogged mobiles contending for the channel is given by $m_b = 1 + \frac{m_c}{2}$. Based on this assumption and since for any connection an ack is sent by the mobile for every TCP packet received, the downlink TCP throughput of a single mobile is given by Section 3.2 in [MKA06] as,

$$\theta_{AP}(m_c) = \frac{L_{TCP}}{m_c(T_{TCPdata} + T_{TCPack} + 2T_{tbo} + 2T_w)}, \tag{2.4}$$

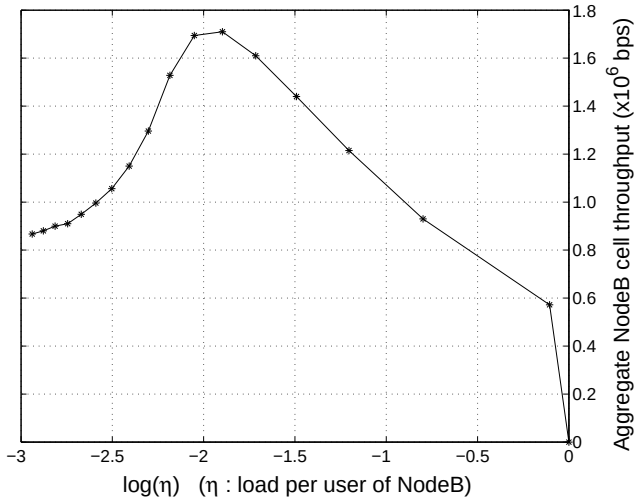


Figure 2.3: Throughput of all users in NodeB cell

where L_{TCP} is the size of TCP packets and $T_{TCPdata}$ and T_{TCPack} are the raw transmission times of a TCP data and a TCP ack packet, respectively. T_{tbo} and T_w denote the mean total time spent in *back-off* and the average total time wasted in collisions for any successful packet transmission and are computed assuming m_b backlogged mobiles. The explicit expressions for $T_{TCPdata}$, T_{TCPack}, T_{tbo} and T_w can be referred to in [MKA06]. However, we mention here that they depend on certain quantities whose numerical values have been provided in Section 2.5.2. Note that all mobiles connected to the AP achieve equal downlink TCP throughputs (given by Equation 2.4) in a fair manner [MKA06]. Figure 2.2 shows a plot of *total* cell throughput in an AP cell for an example scenario. Since the total throughput monotonically decreases with increasing number of mobiles, the pole capacity of an AP cell, M_{AP}, is limited by the QoS requirement θ_{min} bps of each mobile.

2.4.3 Downlink Throughput in 3G UMTS NodeB

We consider a standard model for data transmission on downlink in a 3G UMTS NodeB cell. Let W be the WCDMA modulation bandwidth and if $SINR$ denotes the signal to interference plus noise ratio received at a mobile then its energy per bit to noise density ratio is given by,

$$\frac{E_b}{N_o} = \frac{W}{\theta_{3G}} \times SINR. \tag{2.5}$$

Now, under the assumptions of identical throughput allocation to each mobile arriving at an average location and application of power control so that each mobile receives the same SINR (Section 2.4.1), we deduce from Equation 2.5 that each mobile requires the same E_b/N_o ratio in order to be able to successfully decode NodeB's transmission. From Chapter 8 in [HT01] we can thus say that the downlink TCP throughput θ_{3G} of any mobile, in a

η	$log(\eta)$	$N(\eta)$	$SINR$ (dB)	θ_{3G} $(kbps)$	$\frac{E_b}{N_o}$ (dB)
0.9	−0.10536	1	0.8423	572	9.0612
0.45	−0.79851	2	−2.1804	465	6.9503
0.3	−1.204	3	−3.7341	405	5.7894
0.225	−1.4917	4	−5.1034	360	5.0515
0.18	−1.7148	5	−6.0327	322	4.5669
0.15	−1.8971	6	−6.5093	285	4.3052
0.1286	−2.0513	7	−7.2075	242	4.3460
0.1125	−2.1848	8	−8.8312	191	4.7939
0.1	−2.3026	9	−8.9641	144	5.5091
0.09	−2.4079	10	−9.1832	115	6.0281
0.0818	−2.5033	11	−9.9324	96	6.3985
0.0750	−2.5903	12	−10.1847	83	6.6525
0.0692	−2.6703	13	−10.7294	73	6.8625
0.0643	−2.7444	14	−10.9023	65	7.0447
0.06	−2.8134	15	−10.9983	60	7.0927
0.0563	−2.8779	16	−11.1832	55	7.1903
0.0529	−2.9386	17	−11.3802	51	7.2549
0.05	−2.9957	18	−11.9231	47	7.3614

Table 2.1

NodeB cell with saturated resource allocation, as a function of load per user η is given by,

$$\theta_{3G}(\eta) = \frac{\eta W}{(E_b/N_o)(1-\bar{\alpha}+\bar{i})}, \tag{2.6}$$

where $\bar{\alpha}$ and $\bar{i}$ have been defined earlier in Section 2.4.1.

For an example scenario, Figure 2.3 shows a plot of *total* cell throughput of all mobiles (against $log(\eta)$) in a UMTS NodeB cell. The load per user η has been stretched to a logarithmic scale for better presentation. Also note that throughput values have been plotted in the second quadrant. As we go away from origin on the horizontal axis, $log(\eta)$ (and η) decreases or equivalently number of connected mobiles increase. The equivalence between η and $log(\eta)$ scales and number of mobiles $N(\eta)$ can be referred to in Table 2.1.

It is to be noted here that the required E_b/N_o ratio by each mobile is a function of its throughput. Also, if the NodeB cell is fully loaded with $\eta_{DL} = \eta_{DL}^{max}$ and if each mobile operates at its minimum throughput requirement of θ_{min} then we can easily compute the pole capacity, M_{3G}, of the cell as,

$$M_{3G} = \frac{\eta_{DL}^{max} W}{\theta_{min}(E_b/N_o)(1-\bar{\alpha}+\bar{i})}. \tag{2.7}$$

For $\eta_{DL}^{max} = 0.9$ and a typical NodeB cell scenario that employs the closed-loop fast power

control mechanism mentioned previously in Section 2.4.1, Table 2.1 shows the SINR (fourth column) received at each mobile as a function of the avg. load per user (first column). Note that we consider a maximum cell load of 0.9 and not 1 in order to avoid instability conditions in the cell. These values of SINR have been obtained at France Telecom R&D from radio layer simulations of a NodeB cell. The fifth column shows the downlink throughput with a block error rate (BLER) of 10^{-2} that can be achieved by each mobile as a function of the SINR observed at that mobile. And the sixth column lists the corresponding values of E_b/N_o ratio (obtained from Equation 2.5) that are required at each mobile to successfully decode NodeB's transmission.

2.5 Solving the SMDP Control Problem

With the network model defined in previous section, we now solve our SMDP formulation of Section 2.3. As mentioned earlier, the SMDP formulation can be solved for any given definition of rewards. Here we will motivate the choice of a particular definition based on aggregate throughput of WLAN and UMTS networks.

2.5.1 Defining the Rewards and State Evolution

If we consider the global performance of hybrid cell in terms of throughput and financial revenue earned by the network operator, it is natural from the network operator's point of view to maximize both aggregate network throughput and financial revenue. Except for a certain band of values of η (or $log(\eta)$), generally the aggregate throughput of an AP or NodeB cell drops when an additional new mobile connects to it (see Figures 2.2 & 2.3). However, the network operator gains some financial revenue from the mobile user at the same time. There is thus a trade-off between revenue gain and the aggregate network throughput which motivates us to formulate a *state-dependent* linear (non-linear can also be considered) reward as follows. The reward consists of the sum of a fixed financial revenue price component and β times an aggregate network throughput component which is state-dependent. Here β is an appropriate proportionality constant. When a mobile of the dedicated arrival streams is routed to the corresponding AP or NodeB, it generates a financial revenue of f_{AP} and f_{3G}, respectively. A mobile of the common stream generates a financial revenue of $f_{AP3G \rightarrow AP}$ on being routed to the AP and $f_{AP3G \rightarrow 3G}$ on being routed to the NodeB. Any mobile that is rejected does not generate any financial revenue. The throughput component of the reward is represented by the aggregate network throughput of the corresponding AP or NodeB network to which a newly arrived mobile connects, taking into account the *change* in the state of the system caused by this new mobile's connection. Whereas, if a newly arrived mobile in a dedicated stream is rejected then the throughput component represents the aggregate network throughput of the corresponding AP or NodeB network, taking into account the *unchanged* state of the system. For a rejected mobile belonging to the common stream, it is the maximum of the aggregate throughputs of the two networks that is considered.

With the foregoing discussion in mind, we may define the reward functions R_{AP}, R_{3G} and R_{AP3G} introduced earlier in Section 2.3 as,

$$R_{AP}(m_c,\eta;a)=\begin{cases}\beta\ m_c\ \theta_{AP}(m_c) & :a=0\\ f_{AP}+\beta\ (m_c+1)\ \theta_{AP}(m_c+1) & :a=1,m_c<M_{AP}\\ \beta\ m_c\ \theta_{AP}(m_c) & :a=1,m_c=M_{AP}\end{cases} \tag{2.8}$$

$$R_{3G}(m_c,\eta;a)=\begin{cases}\beta\ N(\eta)\ \theta_{3G}(\eta) & :a=0\\ f_{3G}+\beta\ N(\eta-\Delta(\eta))\ \theta_{3G}(\eta-\Delta(\eta)) & :a=2,N(\eta)<M_{3G}\\ \beta\ N(\eta)\ \theta_{3G}(\eta) & :a=2,N(\eta)=M_{3G}\end{cases} \tag{2.9}$$

$$R_{AP3G}(m_c,\eta;a)=\begin{cases}\max\{\beta\ m_c\ \theta_{AP}(m_c),\beta\ N(\eta)\ \theta_{3G}(\eta)\} & :a=0\\ f_{AP3G\rightarrow AP}+\beta\ (m_c+1)\ \theta_{AP}(m_c+1) & :a=1,m_c<M_{AP}\\ \beta\ m_c\ \theta_{AP}(m_c) & :a=1,m_c=M_{AP}\\ f_{AP3G\rightarrow 3G}+\beta\ N(\eta-\Delta(\eta))\ \theta_{3G}(\eta-\Delta(\eta)) & :a=2,N(\eta)<M_{3G}\\ \beta\ N(\eta)\ \theta_{3G}(\eta) & :a=2,N(\eta)=M_{3G}\end{cases} \tag{2.10}$$

where, $\theta_{AP}(m_c)$ and $\theta_{3G}(\eta)$ have been defined earlier in Equations 2.4 and 2.6 and $N(\eta)$ can be obtained from Table 2.1. Also, based on the discussion in Section 2.4.1 we may define the new states at departure events as,

$$(m_c',\eta)=((m_c-1)\vee 0,\eta)\quad\text{and}\quad(m_c,\eta')=(m_c,(\eta+\Delta(\eta))\wedge 0.9), \tag{2.11}$$

for departures at AP and NodeB, respectively. Additionally, the following entities that were introduced in Section 2.3 may be defined as, $U_n(m_c,\eta;0):=V_n(m_c,\eta)$, $U_n(m_c,\eta;1):=V_n((m_c+1)\wedge M_{AP},\eta)$ and $U_n(m_c,\eta;2):=V_n(m_c,(\eta-\Delta(\eta))\vee 0.05)$ for $\theta_{min}=46$ kbps (refer Table 2.1).

2.5.2 Numerical Analysis

The focus of our numerical analysis is to study the optimal association policy under an ordinary network scenario. We do not investigate in detail the effects of specific TCP parameters and it is outside the scope of this work. Plugging Equations 2.8, 2.9, 2.10 & 2.11 in the Dynamic Programming Equation 2.3, we solve it for an ordinary scenario using the Value Iteration method [Put94]. The scenario that we consider is as follows: $L_{TCP}=8000$ bits (size of TCP packets), $L_{MAC}=272$ bits, $L_{IPH}=320$ bits (size of MAC and TCP/IP headers), $L_{ACK}=112$ bits (size of MAC layer ACK), $L_{RTS}=180$ bits, $L_{CTS}=112$ bits (size of RTS and CTS frames), $R_{data}=11$ Mbits/s, $R_{control}=2$ Mbits/s (802.11 PHY data transmission and control rates), $CW_{min}=32$ (minimum 802.11 contention window), $T_P=144\mu s$, $T_{PHY}=48\mu s$ (times to transmit the PLCP preamble and PHY layer header), $T_{DIFS}=50\mu s$, $T_{SIFS}=10\mu s$ (distributed inter-frame spacing time and short inter-frame spacing time), $T_{slot}=20\mu s$ (slot size time), $K=7$ (*retry limit* in 802.11

standard), $b_0 = 16$ (initial mean back-off), $p = 2$ (exponential back-off multiplier), $\gamma = 0.8$, $\lambda_{AP} = 0.03$, $\lambda_{3G} = 0.03$, $\lambda_{AP3G} = 0.01$, $1/\zeta = 10^6$ bits, $\beta = 10^{-6}$, $M_{AP} = 18$ and $M_{3G} = 18$ for $\theta_{min} = 46$ kbps, $\bar{\alpha} = 0.9$ for ITU Pedestrian A channel, $\bar{i} = 0.7$, $W = 3.84$ Mcps and other values as illustrated in Table 2.1.

The DP equation has been solved for three different kinds of network setups. We first study the simple *homogenous* network case where both networks are AP and hence an incoming mobile belonging to the common stream is offered a connection choice between two identical AP networks. Next, we study an analogous case where both networks are NodeB terminals. We study these two cases in order to gain some insight into connection routing dynamics in simple homogenous network setups before studying the third more complex, hybrid AP-NodeB scenario. Figures 2.4-2.8 show the optimal connection routing policy for the three network setups. Note that the plot in Figure 2.5 is in 3^{rd} quadrant and the plots in Figures 2.6-2.8 are in 2^{nd} quadrant. In all these figures a square box symbol ($\Box$) denotes routing a mobile's connection to the *first* network, a star symbol ($*$) denotes routing to the *second* network and a cross symbol ($\times$) denotes rejecting a mobile all together.

AP-AP homogenous case: In Figure 2.4, optimal policy for the common stream in an AP-AP homogenous network setup is shown with $f_{AP1AP2\rightarrow AP1} = f_{AP1AP2\rightarrow AP2} = 5$ (with some abuse of notation). The optimal policy routes mobiles of common stream to the network which has lesser number of mobiles than the other one. We refer to this behavior as *mobile-balancing* network phenomenon. This happens because the total throughput of an AP network decreases with increasing number of mobiles (Figure 2.2). Therefore, an AP network with higher number of mobiles offers lesser reward in terms of network throughput and a mobile generates greater incentive by joining the network with fewer mobiles. Also note that the optimal routing policy in this case is *symmetric* and of *threshold type* with the threshold switching curve being the coordinate line $y = x$.

NodeB-NodeB homogenous case: Figure 2.5 shows optimal routing policy for the common stream in a NodeB-NodeB homogenous network setup. With equal financial incentives for the mobiles, i.e., $f_{3G13G2\rightarrow 3G1} = f_{3G13G2\rightarrow 3G2} = 5$ (with some abuse of notation), we observe a very interesting switching curve structure. The state space in Figure 2.5 is divided into an *L-shaped* region (at bottom-left) and a *quadrilateral shaped* region (at top-right) under the optimal policy. Each region separately, is *symmetric* around the coordinate diagonal line $y = x$. Consider the point $(log(\eta_1), log(\eta_2)) = (-0.79851, -1.4917)$ on logarithmic scale in the upper triangle of the quadrilateral region. From Table 2.1 this corresponds to the network state when load per user in the first NodeB network is 0.45 which is more than the load per user of 0.225 in the second NodeB network. Equivalently, there are less mobiles connected to the first network as compared to the second network. Ideally, one would expect new mobiles to be routed to the first network rather than the second network. However, according to Figure 2.5 in this state the optimal policy is to route to the second network even though the number of mobiles connected to it is more than those in the first. We refer to this behavior as *mobile-greedy* network phenomenon and explain the intuition behind it in the following paragraph. The routing policies on boundary coordinate lines are clearly comprehensible. On $y = -2.9957$ line when the first network is full (i.e., with least possible

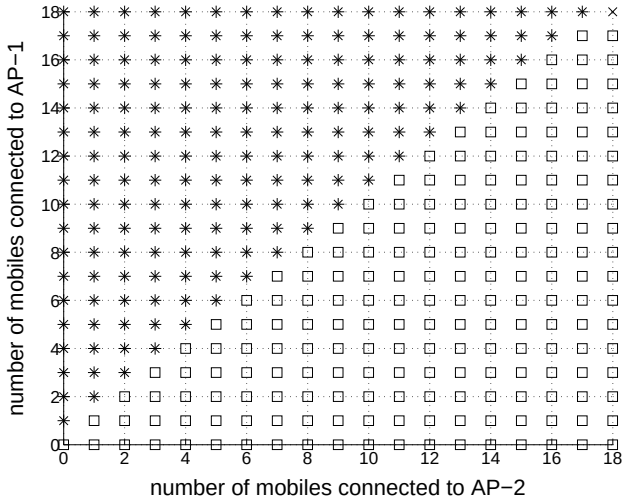


Figure 2.4: Optimal policy for common flow in AP-AP setup. *First* network: AP1, *Second* network: AP2.

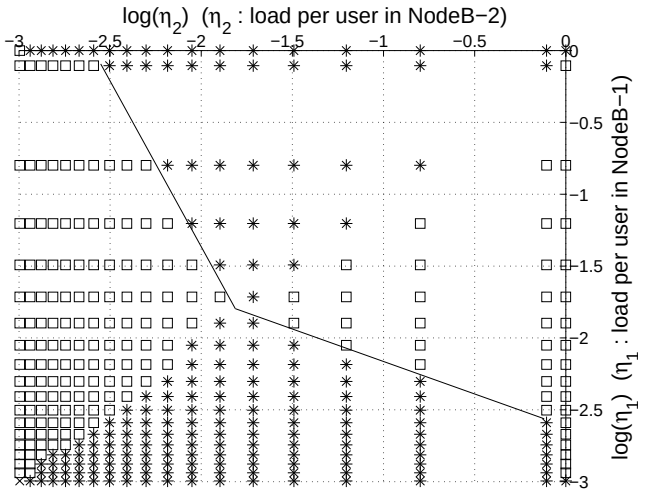


Figure 2.5: Optimal policy for common flow in NodeB-NodeB setup. *First* network: NodeB1, *Second* network: NodeB2.

load per user), incoming mobiles are routed to second network (if possible) and vice-versa for the line $x = -2.9957$. When both networks are full, incoming mobiles are rejected which is indicated by the cross at coordinate point $(x, y) = (-2.9957, -2.9957)$.

The reason behind the mobile-greedy phenomenon in Figure 2.5 can be attributed to the fact that in a NodeB network, the total throughput increases with decreasing avg. load per user up to a particular threshold (say η_{thres}) and then decreases thereafter (see Figure 2.3). Therefore, routing new mobiles to a network with lesser (but greater than η_{thres}) load per user (greater number of mobiles) results in a higher reward in terms of total network throughput, than routing new mobiles to the other network with greater load per user (lesser number of mobiles). However, the mobile-greedy phenomenon is only limited

to the quadrilateral shaped region. In the L-shaped region, the throughput of a NodeB network decreases with decreasing load per user, contrary to the quadrilateral region where the throughput increases with decreasing load per user. Hence, in the L-shaped region higher reward is obtained by routing to the network having higher load per user (lesser number of mobiles) than by routing to the network with lesser load per user (greater number of mobiles). In this sense the L-shaped region shows similar characteristics to mobile-balancing phenomenon observed in AP-AP network setup (Figure 2.4).

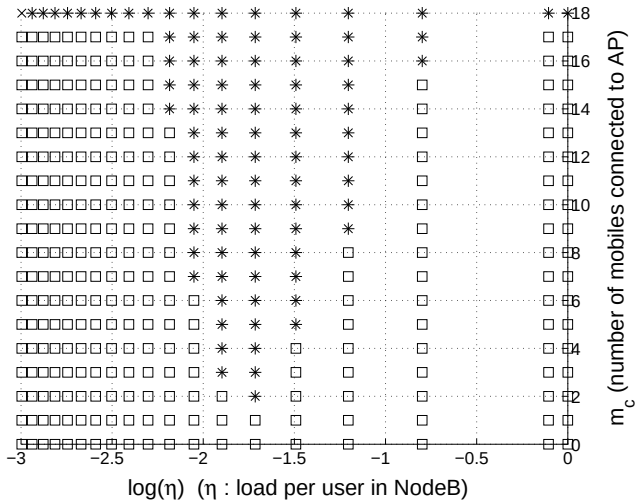


Figure 2.6: Optimal policy for common flow in AP-NodeB hybrid cell. *First* network: AP, *Second* network: NodeB.

AP-NodeB hybrid cell: We finally discuss now the hybrid AP-NodeB network setup. Here we consider financial revenue gains of $f_{AP3G \to AP} = 5$ and $f_{AP3G \to 3G} = 6$ motivated by the fact that a network operator can charge more for a UMTS connection since it offers a larger coverage area. Moreover, UMTS equipment is more expensive to install and maintain than WLAN equipment. In Figure 2.6, we observe that the state space is divided into two regions by the optimal policy switching curve which is *neither convex nor concave*. Besides, in some regions of state space the mobile-balancing network phenomenon is observed, where as in some other regions the mobile-greedy network phenomenon is observed. In some sense, this can be attributed to the symmetric threshold type switching curve and the symmetric L-shaped and quadrilateral shaped regions in the corresponding AP-AP and NodeB-NodeB homogenous network setups, respectively.

Figures 2.7 and 2.8 show the optimal policies for dedicated streams in an AP-NodeB hybrid cell with $f_{AP} = f_{3G} = 0$. The optimal policy accepts new mobiles in the AP network only when there are none already connected. This happens because initially the network throughput of an AP is zero when there are no mobiles connected and a non-zero reward is obtained by accepting a mobile. Thereafter, since $f_{AP} = 0$ the policy rejects all incoming mobiles due to decrease in network throughput with increasing number of mobiles and hence decrease in corresponding reward. Similarly, for the dedicated mobiles to the NodeB network, the optimal policy accepts new mobiles until the network throughput increases

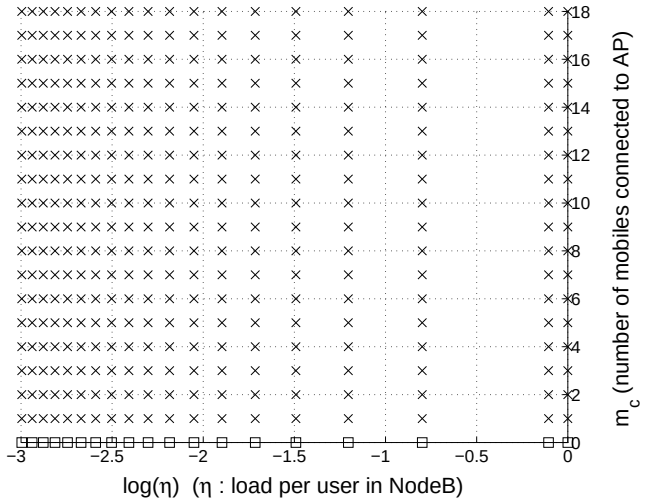


Figure 2.7: Optimal policy for AP dedicated flow in AP-NodeB hybrid cell

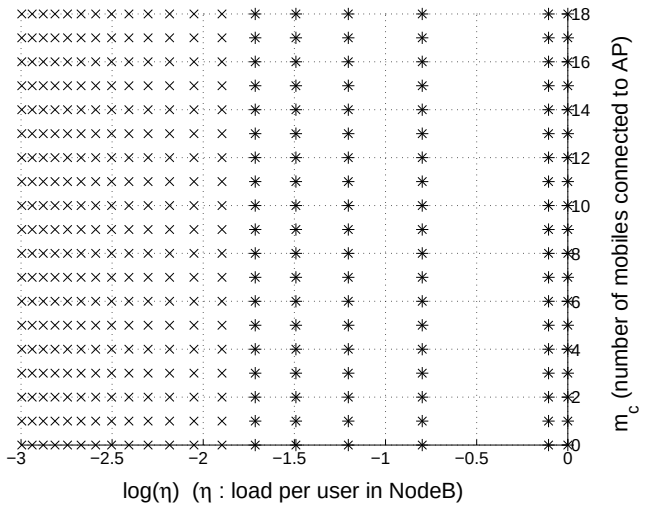


Figure 2.8: Optimal policy for NodeB dedicated flow in AP-NodeB hybrid cell

(Figure 2.3) and rejects them thereafter due to absence of any financial reward component and decrease in the network throughput. Note that we have considered zero financial gains here ($f_{AP} = f_{3G} = 0$) to be able to exhibit existence of these *threshold type* policies for the dedicated streams.

2.6 Conclusion

In this chapter, we have considered globally optimal user-network association (load balancing) in an AP-NodeB hybrid cell. To the best of our knowledge this study is the first of its kind. Since it is infeasible to solve an SMDP formulation for an exhaustive set of network scenarios, we have considered an ordinary network scenario and computed the optimal association policy. Even though the characteristics of the solution to our particular scenario

are not depictive of the complete solution space, they can certainly be helpful in acquiring an intuition about the underlying dynamics of user-network association in a hybrid cell.

2.6.1 Alternate Approach with Individual Optimality

In the following chapter we shall study the same user-network association problem from a different perspective. There the association problem is formulated within a non-cooperative game framework and individual optimality is studied. Though both chapters focus on user-network association, they are not directly related since the performance metric considered in both chapters is different: 'total cell throughput' in this chapter and 'average service time' in the following chapter. Besides, it is only a matter of 'operating policy' adopted by the network operator, whether it prefers global performance optimization at the expense of user's choice or whether it wants to provide more options to the user while being unsure if the individual user's choice would result in globally optimal use of resources. Thus, we do not explicitly compare the results of Chapter 2 and 3.

Chapter 3

Individually Optimal User-Network Association in a WLAN & UMTS Hybrid Cell

In this chapter, we study individually optimal user-network association in an integrated WLAN and UMTS *hybrid cell*. The association problem is formulated within a non-cooperative game framework. In the formulation, mobile arrivals are assumed to follow the Poisson process and each mobile considers its average service time in each network as the decision criteria to connect to either of the WLAN or UMTS networks. We seek to compute the optimal association or decision policy that achieves the Nash equilibrium. For this we develop a generic system of linear equations for estimating the average service time of a mobile. This system is then solved assuming a particular model for the WLAN and UMTS networks and we explicitly compute the optimal association policy that is observed to possess a descending staircase curve structure.

Note: The material in this chapter has appeared in [KAK07b]. This work was sponsored by France Telecom R&D.

3.1 Introduction

As 802.11 WLANs and 3G UMTS cellular coverage networks are being widely deployed, network operators are seeking to offer seamless and ubiquitous connectivity for wireless broadband services through integrated WLAN and UMTS hybrid networks. One of the core decision problems faced in such a hybrid network is that of user-network association, i.e., decision of an arriving mobile user to connect to one of the two constituent networks. We study this decision problem in the framework of individual optimality where arriving mobile users selfishly connect to one of the two networks (WLAN or UMTS) based on an individual decision cost criteria. This gives a non-cooperative game structure to the decision problem and we compute the Nash equilibrium achieving optimal policy assuming a specific network model for the WLAN and UMTS networks. To be more precise consider a hybrid network, comprising two independent 802.11 WLAN and 3G UMTS networks, that offers connectivity to mobile users arriving in the combined coverage area of these two networks. By independent we mean that transmission activity in one network does not create interference in the other. Our goal in this chapter is to study the dynamics of individually optimal user-network association in such a WLAN-UMTS hybrid network. An alternate approach based on globally optimal user-network association is envisaged to be studied as part of our future work. Note that we do not propose a full fledged cell-load or interference based connection admission control (CAC) policy in this chapter. We instead assume that a CAC precedes the association decision control. A connection admission decision is taken by the CAC controller before any mobile is allowed to connect to either of the WLAN or UMTS networks. Thereafter, the mobile's association decision only optimizes its individual performance and it is not proposed as an alternative to the CAC decision.

3.1.1 Related Work and Contributions

Study of WLAN-UMTS hybrid networks is an emerging area of research and not much related work is available. Authors in some related papers [MYLR04, SLC03, LZ05, Jas03, VGN05, KRPR05, FC05] have studied issues such as vertical handover and coupling schemes, integrated architecture layout, radio resource management (RRM) and mobility management. However, questions related to user-network association have not been explored much. Premkumar et al. in [PK06] propose a *near optimal* solution for a hybrid network within a combinatorial optimization framework, which is different from our approach. To the best of our knowledge, ours is the first attempt to present a generic formulation of the user-network association problem under a non-cooperative game framework. Moreover, this work is the first we know of that obtains an explicit threshold based policy for the WLAN-UMTS hybrid network model that we consider.

3.2 Framework for the Decision Control Problem

A hybrid network may be composed of several 802.11 WLAN Access Points (APs) and 3G UMTS Base Stations (NodeBs) that are operated by a single network operator. However,

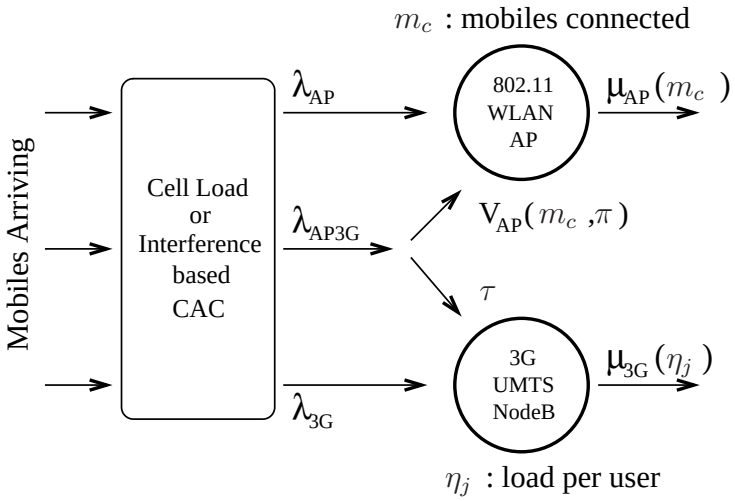


Figure 3.1: Hybrid cell scenario

our focus is only on a single pair of an AP and a NodeB that are located sufficiently close to each other so that mobile users arriving in the combined coverage area of this AP-NodeB pair have a choice to connect to either of the two networks. We call the combined coverage area of a single AP cell and a single NodeB micro-cell [HT01] as a *hybrid cell*. The cell coverage radius of a UMTS micro-cell is usually around $400m$ to $1000m$ whereas that of a WLAN cell varies from a few tens to a few hundreds of meters. Therefore, some mobiles arriving in the hybrid cell may only be able to connect to the NodeB, either because they fall outside the transmission range of the AP or they are equipped with only 3G technology electronics. While other mobiles that are equipped with only 802.11 technology can connect exclusively to the WLAN AP. Apart from these two categories, mobiles equipped with both 802.11 WLAN and 3G UMTS technologies can connect to any one of the two networks.

The decision to connect to either of the two networks may involve a cost criteria based on the average service time of a mobile in the hybrid network. Moreover, the connection or association decision involves two different possible decision makers, the mobile user and the network operator. We focus only on the individually optimal control problem in which the mobile users take a selfish decision to connect to one of the two networks while optimizing only their own individual costs. In Section 3.3, we motivate a non-cooperative game formulation for this problem. Our game formulation is a *generic* formulation of the user-network association problem in a WLAN-UMTS hybrid network and is independent of the network model assumed for WLAN and UMTS networks. Thereafter in Section 3.5, we solve the problem assuming a particular network model described in Section 3.4.

3.2.1 Mobile Arrivals

We model the hybrid cell of an 802.11 WLAN AP and a 3G UMTS NodeB as a two-server processing system (Figure 3.1) with each server having a separate finite capacity of M_{AP} and M_{3G} mobiles, respectively. For simplification we assume that mobile users are stationary, having no mobility. As discussed previously, mobiles are considered as candidates to connect

to the hybrid cell only after being admitted by a CAC such as the one described in [YK05]. Assuming mobile arrivals to be Poisson it can be easily shown using the PASTA property (Poisson arrivals see time averages) that mobile arrivals after the CAC are also Poisson with a reduced effective arrival rate. The reduced effective arrival rate can be obtained by incorporating the fraction of mobiles that are dropped by the CAC. Some of the mobiles (after they have been admitted by the CAC) can connect only to the WLAN AP and some others only to the UMTS NodeB. These two set of mobiles (or sessions) are each assumed to constitute two separate dedicated arrival streams with effective Poisson rates λ_{AP} and λ_{3G}, respectively. The remaining set of mobiles which can connect to both networks form a common arrival stream with effective Poisson rate λ_{AP3G}. The mobiles of the two dedicated streams directly join their respective AP or NodeB network without any connection decision choice involved. Mobiles of the common stream decide to connect to one of the two networks while optimizing their own cost.

3.2.2 Service Requirements and Departure Rates

It is assumed that all arriving mobiles have a downlink data service requirement which is exponentially distributed with parameter ζ. In other words, every arriving mobile seeks to download a data file of average size $1/\zeta$ bits on the downlink. Let $\theta_{AP}(m_c)$ denote the downlink packet (or file) throughput of each mobile in the AP network when m_c mobiles are connected to it at any given instant. If η_j denotes the *load factor* of a mobile j in the NodeB cell (see Chapter 8 in [HT01]) then $\theta_{3G}(\eta_j)$ denotes the downlink packet (or file) throughput of this mobile in the NodeB network. With these notations, the effective departure rates of mobiles (or sessions) in each server (or network) can be denoted by,

$$\mu_{AP}(m_c) = \zeta \times \theta_{AP}(m_c) \tag{3.1}$$

and

$$\mu_{3G}(\eta_j) = \zeta \times \theta_{3G}(\eta_j), \tag{3.2}$$

where, $\mu_{AP}(m_c)$ is identical for each mobile in the AP network and $\mu_{3G}(\eta_j)$ is different for each mobile j in the NodeB network and is a function of its load factor, η_j. The load factor, η_j, in turn depends on the location of mobile j in the NodeB cell.

3.3 Non-cooperative Game Formulation

As discussed earlier, a mobile arriving in the common stream selfishly decides to join one of the two networks so that its own cost is optimized. We consider the *average service time* of a mobile as the decision cost criteria and an incoming mobile connects to either the AP or NodeB network depending on which of them offers the minimum average service time. Service time here represents the time required for a mobile to accomplish its file download. Therefore, higher is the packet throughput achieved by a mobile, lesser will be its service time. We develop this model as an extension to the framework of [AS98] where an incoming

user can either join a shared server with a PS service mechanism or any of several dedicated servers. Based on the estimate of its expected service time on each of the two servers, a user takes a decision to join the server on which its expected service time is least. This framework can be readily applied to our hybrid cell scenario so that the AP is modeled by the shared server and the dedicated DCH channels [HT01] of the NodeB are modeled by the dedicated servers. For simplicity, we refer to the several dedicated servers in [AS98] as one single dedicated server that consists of a pool of dedicated servers. Then the NodeB comprising dedicated DCH channels is modeled by this single dedicated server and this type of framework then fits well with our original setting in Section 3.2. We thus preserve the two-server processing system as in Figure 3.1.

As mentioned earlier, the mobiles of dedicated streams directly join their respective AP or NodeB network. Mobiles arriving in the common stream decide to join one of the two networks based on their *estimate* of the expected service time in each one of them. However, an estimate of the expected service time of an arriving mobile j must be made taking into account the effect of subsequently arriving mobiles. But these subsequently arriving mobiles are themselves faced with a similar decision problem and hence their decision will affect the performance of mobile j (which is presently attempting to connect) or other mobiles already in service. This dependance thus induces a non-cooperative game structure to the decision problem and we seek here to study the Nash equilibrium solution of the game. The existence, uniqueness and structure of the equilibrium point have been proved in [AS98] already. Here we seek to analytically determine the service time estimate (Section 3.3.2) and explicitly compute the equilibrium achieving threshold policy (Section 3.5). But before proceeding further in this direction, we briefly present below a background on the results of [AS98] adapted to our hybrid cell framework.

3.3.1 Background

A decision rule (or policy) for a new mobile is represented by a function $u : \{0, 1, \ldots, M_{AP} - 1\} \rightarrow [0, 1]$, where M_{AP} is the capacity of the AP network. Thus for each possible state of the AP network denoted by number of mobiles already connected, m_c, a new mobile takes a randomized decision $u(m_c) \in [0, 1]$, that specifies the probability of connecting to the AP. $1 - u(m_c)$ then represents either the probability of connecting to the NodeB or abandoning to seek a connection altogether if both networks are full to their capacity. A policy profile $\pi = (u_1, u_2, u_3, \ldots)$ is defined as a collection of decision rules followed by all arriving mobiles indexed $1, 2, 3, \ldots$.

Define $V_{AP}(m_c, \pi)$ as the expected service time of a mobile in the AP network, given that it joins that network when m_c mobiles are already connected and all subsequently arriving mobiles follow the policy profile π. Mobiles in a NodeB network are allocated the *dedicated* DCH channels on which they are guaranteed throughputs greater than a worst case lower bound [HT01]. Equivalently, they are guaranteed service times lesser than a worst case upper bound. For simplification we assume a worst case estimate for the expected service time of a mobile in the NodeB network. Denote $\hat{\mu}_{3G} \stackrel{\Delta}{=} \min_{\eta_j} \mu_{3G}(\eta_j)$ and let $\tau \stackrel{\Delta}{=} 1/\hat{\mu}_{3G}$

be the maximum (or worst case) service time of a mobile j in the NodeB cell, which is *independent* of its load factor η_j.

Now, for some q ($q \in [0,1]$, $q \in \mathbb{R}$), define the *decision* policy $u(m_c)$ to be the best response of a new mobile (against the policy profile $\pi = (u_1, u_2, u_3, \ldots)$ followed by all subsequently arriving mobiles), if,

$$u(m_c) = \begin{cases} 1 & : V_{AP}(m_c, \pi) < \tau \\ q & : V_{AP}(m_c, \pi) = \tau \\ 0 & : V_{AP}(m_c, \pi) > \tau \end{cases}$$

Further, define a special kind of policy namely the *threshold* type policy as follows. Given q and L such that $q \in [0,1]$, $q \in \mathbb{R}$ and $L \in \mathbb{Z}^+ \cup \{0\}$, an L, q threshold policy $u_{L,q}$ is defined as,

$$u_{L,q}(m_c) = \begin{cases} 1 & : m_c < L \\ q & : m_c = L \\ 0 & : m_c > L \end{cases} \tag{3.3}$$

This L, q threshold policy will be denoted by $[L, q]$ or more compactly by $[g]$ where $g = L+q$. Note that the threshold policies $[L, 1]$ and $[L+1, 0]$ are identical. We also use the notation $[g]^\infty \equiv [L, q]^\infty$ to denote the policy profile $\pi = ([g], [g], \ldots)$.

Now, it has been proved in Theorem 1 in [AS98] that a *Nash equilibrium achieving* decision policy (or simply an *equilibrium* policy), $u^*(m_c)$, exists, and it is actually the threshold policy, $[L^*, q^*]$, which can be computed as follows. If

$$V_{AP}(M_{AP} - 1, [M_{AP}]^\infty) < \tau,$$

then $[L^*, q^*] = [M_{AP}, 0]$. Otherwise, let

$$L^{min} \triangleq \min\{L \in \mathbb{Z}^+ \cup \{0\} : V_{AP}(L, [L, 1]^\infty) > \tau\}.$$

Now, if

$$V_{AP}(L^{min}, [L^{min}, 0]^\infty) \geq \tau,$$

then the threshold policy is given by $[L^*, q^*] = [L^{min}, 0]$. Else if

$$V_{AP}(L^{min}, [L^{min}, 0]^\infty) < \tau,$$

then it is given by $[L^*, q^*] = [L^{min}, q^*]$, where q^* is the unique solution of the equation,

$$V_{AP}(L^{min}, [L^{min}, q^*]^\infty) = \tau. \tag{3.4}$$

For explicitly computing the equilibrium policy $[L^*, q^*]$ we thus need to compute $V_{AP}(L, [L, 1]^\infty)$ for all possible values of L.

3.3.2 Determining Expected Service Time in AP

The entity equivalent to $V_{AP}(m_c, \pi)$ in [AS98] has been derived using *constant* departure rates. This scenario differs from our hybrid cell framework, since we consider a *state dependent* departure rate, $\mu_{AP}(m_c)$, for the shared AP server and moreover in our framework we have dedicated arrivals in addition to the common arrivals. Due to this difference we can not adopt the derivation of the entity equivalent to $V_{AP}(m_c, \pi)$ in [AS98]. Therefore, with the state dependent departure rate, $\mu_{AP}(m_c)$, we now compute $V_{AP}(m_c, \pi)$ analytically here.

For notational convenience if we define,

$$V(m_c) \stackrel{\Delta}{=} V_{AP}(m_c, [L, q]^{\infty}), \quad 0 \leq m_c \leq M_{AP} - 1,$$

then the set of indeterminates $\{V(m_c) : 0 \leq m_c \leq M_{AP}-1\}$ can be obtained as a solution of the following system of M_{AP} linear equations, where $\alpha \stackrel{\Delta}{=} \lambda_{AP}+\lambda_{AP3G}+\mu_{AP}(m_c)$ (dependence of α on m_c has been suppressed in the notation):

Case 1: $4 \leq L \leq M_{AP} - 2$,

$$\begin{aligned}
V(0) &= \frac{1}{\alpha} + \frac{\lambda_{AP} + \lambda_{AP3G}}{\alpha} V(1) \\
V(m_c) &= \frac{1}{\alpha} + \frac{\mu_{AP}(m_c)}{\alpha} \frac{m_c}{m_c + 1} V(m_c - 1) \\
&\quad + \frac{\lambda_{AP} + \lambda_{AP3G}}{\alpha} V(m_c + 1), \quad 1 \leq m_c \leq L - 2 \\
V(L-1) &= \frac{1}{\alpha} + \frac{\mu_{AP}(L-1)}{\alpha} \frac{L-1}{L} V(L-2) \\
&\quad + \frac{\lambda_{AP} + q\, \lambda_{AP3G}}{\alpha} V(L) + \frac{\lambda_{AP3G}}{\alpha}(1-q) V(L-1) \\
V(L) &= \frac{1}{\lambda_{AP} + \mu_{AP}(L)} + \frac{\mu_{AP}(L)}{\lambda_{AP} + \mu_{AP}(L)} \frac{L}{L+1} V(L-1) \\
&\quad + \frac{\lambda_{AP}}{\lambda_{AP} + \mu_{AP}(L)} V(L+1) \\
V(m_c) &= \frac{1}{\lambda_{AP} + \mu_{AP}(m_c)} + \frac{\mu_{AP}(m_c)}{\lambda_{AP} + \mu_{AP}(m_c)} \frac{m_c}{m_c + 1} \\
&\quad \times V(m_c - 1) + \frac{\lambda_{AP}}{\lambda_{AP} + \mu_{AP}(m_c)} V(m_c + 1), \\
&\qquad\qquad L + 1 \leq m_c \leq M_{AP} - 2 \\
V(M_{AP} - 1) &= \frac{1}{\mu_{AP}(M_{AP} - 1)} + \frac{M_{AP} - 1}{M_{AP}} V(M_{AP} - 2)
\end{aligned} \tag{3.5}$$

Case 2: $L = M_{AP} - 1$,

$$\begin{aligned}
V(0) &= \frac{1}{\alpha} + \frac{\lambda_{AP} + \lambda_{AP3G}}{\alpha} V(1) \\
V(m_c) &= \frac{1}{\alpha} + \frac{\mu_{AP}(m_c)}{\alpha} \frac{m_c}{m_c + 1} V(m_c - 1) \\
&\quad + \frac{\lambda_{AP} + \lambda_{AP3G}}{\alpha} V(m_c + 1), \quad 1 \leq m_c \leq L - 2 \\
V(L-1) &= \frac{1}{\alpha} + \frac{\mu_{AP}(L-1)}{\alpha} \frac{L-1}{L} V(L-2) \\
&\quad + \frac{\lambda_{AP} + q\,\lambda_{AP3G}}{\alpha} V(L) + \frac{\lambda_{AP3G}}{\alpha}(1-q) V(L-1) \\
V(L) &= \frac{1}{\mu_{AP}(L)} + \frac{L}{L+1} V(L-1).
\end{aligned} \tag{3.6}$$

Each of the above equations says that the expected service time of a new mobile in AP cell ($V(m_c)$), when m_c other mobiles are connected, is given by the expected time till the next event (either arrival or departure) plus the expected service time from this event onwards. Note that $m_c = M_{AP}$ need not be considered since in that case the AP cell will be full to its capacity and can not accept a new mobile. The above system of M_{AP} linear equations in the framework of the non-cooperative game formulation is a very *generic* formulation of our user-network association decision problem and it can be solved using any particular WLAN and UMTS network models. In Section 3.5, we will solve this problem assuming a specific definition for the packet throughputs (or departure rates) obtained from specific models for the WLAN and UMTS networks. We first present these network models in the following section along with some assumptions.

3.4 WLAN and UMTS Network Models

Before discussing the network models adopted from previous work, we first state below some assumptions along with their justification. Since the bulk of data transfer for a mobile engaged in streaming or interactive (HTTP like) data transmission is carried over the downlink (AP to mobile or NodeB to mobile) and since TCP is the most commonly used transport protocol (streaming protocols based on TCP also exist, e.g., Real Time Streaming Protocol), we are interested in network models for computing TCP throughput on only downlink.

3.4.1 Assumptions

3.4.1.1 Assumption on QoS and TCP

We assume a single QoS class of arriving mobiles so that each mobile has an identical minimum downlink throughput requirement of θ_{min}, i.e., each arriving mobile must achieve

a downlink packet throughput of at least θ_{min} bps in either of the two networks.

Several versions of TCP have been proposed in literature for wireless environments. For our purposes we assume that the wireless TCP algorithm operates in *split mode* [TXA05]. In brief, the split mode divides a TCP connection into wireless and wired portions, and acks are generated for both portions separately. Therefore, in our hybrid cell scenario TCP acks are generated separately for the single hop between mobiles and AP or NodeB. We also assume that TCP acks are not delayed and every received data packet is acknowledged with an ack. It is further assumed that each mobile's or receiver's advertised window W^* is set to 1 in the wireless portion of TCP protocol.

3.4.1.2 Resource allocation in AP and NodeB

We assume *saturated resource allocation* in the downlink of AP and NodeB networks. Specifically, this assumption for the AP network means that the AP is *saturated* and has infinitely many packets backlogged in its transmission buffer. In other words, there is always a packet in the AP's transmission buffer waiting to be transmitted to each of the connected mobiles. With this assumption, mobiles can be allocated downlink packet throughputs greater than their QoS requirements of θ_{min} and cell resources in terms of transmission opportunities ($TxOPs$) on the downlink will be maximally utilized.

In the NodeB network the saturated resource allocation assumption implies that at any given instant the NodeB cell resources on downlink are fully utilized. This is analogous to the maximal utilization of $TxOPs$ in the AP network discussed in previous paragraph. With this maximum resource allocation assumption even if a mobile has a minimum packet throughput requirement of only θ_{min} bps, it can actually be allocated a higher throughput if additional unutilized cell resources are available.

3.4.1.3 Justification

The assumption of W^* being set to 1 is required for the WLAN model that we adopt and in fact it is known to provide the best performance of TCP in a single hop case (see [ea03, LP05] and references therein).

Saturated resource allocation is a standard assumption, usually adopted to simplify modeling of complex network frameworks like those of WLAN and UMTS (see for e.g., [HT01, KAMG05a]).

3.4.2 Downlink Throughput in 802.11 WLAN AP

We reuse the downlink TCP throughput formula for a mobile in an 802.11 WLAN from [MKA06]. For completeness, here we briefly mention the network model that has been extensively studied in [MKA06] and then simply restate the throughput expression without going into much details.

Each mobile connected to the AP uses the Distributed Coordination Function (DCF) protocol with an RTS/CTS frame exchange before any data-ack frame exchange and each mobile (including the AP) has an equal probability of the channel being allocated to it. The

AP does not employ any rate control algorithm and transmits at a fixed PHY data rate of R_{data} bps to all mobiles. With the assumption of W^* being set to 1 (Section 3.4.1), any mobile will always have a TCP ack waiting to be sent back to the AP with probability 1/2, which is also the probability that it contends for the channel. This is however true only for those versions of TCP that do not use delayed acks. If the AP is always saturated or backlogged, the average number of backlogged mobiles contending for the channel is given by $m_b = 1 + \frac{m_c}{2}$. Based on this assumption and since for any connection an ack is sent by the mobile for every TCP packet received, the downlink TCP throughput of a single mobile is given by Section 3.2 in [MKA06] as,

$$\theta_{AP}(m_c) = \frac{L_{TCP}}{m_c(T_{TCPdata} + T_{TCPack} + 2T_{tbo} + 2T_w)}, \tag{3.7}$$

where L_{TCP} is the size of TCP packets and $T_{TCPdata}$ and T_{TCPack} are the raw transmission times of a TCP data and a TCP ack packet, respectively. T_{tbo} and T_w denote the mean total time spent in *back-off* and the average total time wasted in collisions for any successful packet transmission and are computed assuming m_b backlogged mobiles. The explicit expressions for $T_{TCPdata}$, T_{TCPack}, T_{tbo} and T_w can be referred to in [MKA06]. However, we mention here that they depend on certain quantities whose numerical values have been provided in Section 3.5. Note that independent of their location in the AP cell, all mobiles achieve equal downlink TCP throughputs (given by Equation 3.7) in a fair manner [MKA06].

3.4.3 Downlink Throughput in 3G UMTS NodeB

We consider a standard model for data transmission on downlink in a 3G UMTS NodeB cell. Let W be the WCDMA modulation bandwidth and if $SINR_j$ denotes the signal to interference plus noise ratio received at a mobile j then its energy per bit to noise density ratio is given by,

$$\left(\frac{E_b}{N_o}\right)_j = \frac{W}{\theta_{3G}} \times SINR_j. \tag{3.8}$$

From Chapter 8 in [HT01] we can then say that in a NodeB cell with saturated resource allocation, the downlink TCP throughput, θ_{3G}, of any mobile j as a function of its load factor η_j is given by,

$$\theta_{3G}(\eta_j) = \frac{\eta_j W}{(E_b/N_o)_j\,(1 - \alpha_j + i_j)}, \tag{3.9}$$

where i_j and α_j are mobile j's inter-cell to intra-cell interference ratio and orthogonality factor, respectively (see Section 8.2.2.2 in [HT01]).

It is to be noted here that the required $(E_b/N_o)_j$ ratio by each mobile j is a function of its throughput. Also, if each mobile operates at its minimum throughput requirement of θ_{min} then we can easily compute the capacity, M_{3G}, of the cell as,

$$M_{3G} = \sum_j \frac{\eta_j W}{\theta_{min}\,(E_b/N_o)_j\,(1 - \alpha_j + i_j)}, \tag{3.10}$$

where, the summation is over all mobiles in the NodeB cell. Note that the load factor, η_j, of a mobile j decreases with increasing number of total mobiles in a NodeB cell [HT01].

For some values of η_j in the interval $[0.09, 0.9]$ and $\theta_{min} = 115$ kbps, Table 2.1 in previous chapter shows the SINR (fourth column) received at any mobile j as a function of its load factor (first column). Note that we consider a maximum load factor of 0.9 and not 1 in order to avoid instability conditions in the cell. These values of SINR have been obtained from radio layer simulations of a NodeB cell. The fifth column shows the downlink packet throughput with a block error rate (BLER) of 10^{-2} that can be achieved by a mobile as a function of the SINR observed at that mobile. And the sixth column lists the corresponding values of $(E_b/N_o)_j$ ratio (obtained from Equation 3.8) that are required at mobile j to successfully decode NodeB's transmission.

3.5 Computing the Equilibrium Policy

With the network models defined in previous section, we now solve the system of M_{AP} linear equations in order to obtain the set of indeterminates $\{V(m_c) : 0 \leq m_c \leq M_{AP} - 1\}$ and finally the equilibrium threshold policy $[L^*, q^*]$, as described in Section 3.3. The focus of our numerical analysis here is to study the equilibrium policy under an ordinary network scenario. We do not investigate in detail the effects of specific TCP parameters and it is outside the scope of this work. The network scenario that we consider is as follows: $L_{TCP} = 8000$ bits (size of TCP packets), $L_{MAC} = 272$ bits, $L_{IPH} = 320$ bits (size of MAC and TCP/IP headers), $L_{ACK} = 112$ bits (size of MAC layer ACK), $L_{RTS} = 180$ bits, $L_{CTS} = 112$ bits (size of RTS and CTS frames), $R_{data} = 11$ Mbits/s, $R_{control} = 2$ Mbits/s (802.11 PHY data transmission and control rates), $CW_{min} = 32$ (minimum 802.11 contention window), $T_P = 144\mu s$, $T_{PHY} = 48\mu s$ (times to transmit the PLCP preamble and PHY layer header), $T_{DIFS} = 50\mu s$, $T_{SIFS} = 10\mu s$ (distributed inter-frame spacing time and short inter-frame spacing time), $T_{slot} = 20\mu s$ (slot size time), $K = 7$ (*retry limit* in 802.11 standard), $b_0 = 16$ (initial mean back-off), $p = 2$ (exponential back-off multiplier), $\lambda_{AP} = 3$, $\lambda_{3G} = 3$, $\lambda_{AP3G} = 10$, $1/\zeta = 10^5$ bits, $\alpha_j \in [0.6, 0.9]$ uniformly randomly [HT01], $i_j \in [0.4, 0.7]$ uniformly randomly [HT01] and $W = 3.84$ Mcps.

Plugging Equation 3.7 in Equation 3.1 and then Equation 3.1 in the system of M_{AP} linear equations, it can be solved with $m_c = L$ and $q = 1$ to obtain $V_{AP}(L, [L, 1]^\infty)$ for different values of L. Figure 3.2 shows an example plot of $V_{AP}(L, [L, 1]^\infty)$ for $1/\zeta = 10^5$ bits, $\lambda_{AP} = 3$, $M_{AP} = 10$ and other numerical values for various entities in WLAN and UMTS networks being those listed in the previous paragraph.

Assuming a certain NodeB cell capacity, M_{3G}, τ can be computed from its definition, Equation 3.10 and Table 2.1. Knowing τ, one can compute L^{min} from Figure 3.2. It is simply the smallest integer value of L for which $V_{AP}(L, [L, 1]^\infty) > \tau$. Using this value of L^{min} one can finally compute q^* from Equation 3.4 which will give us the equilibrium threshold policy.

As an example, a capacity of $M_{3G} = 10$ (i.e., $\theta_{min} = 115$ kbps from Equation 3.10 and

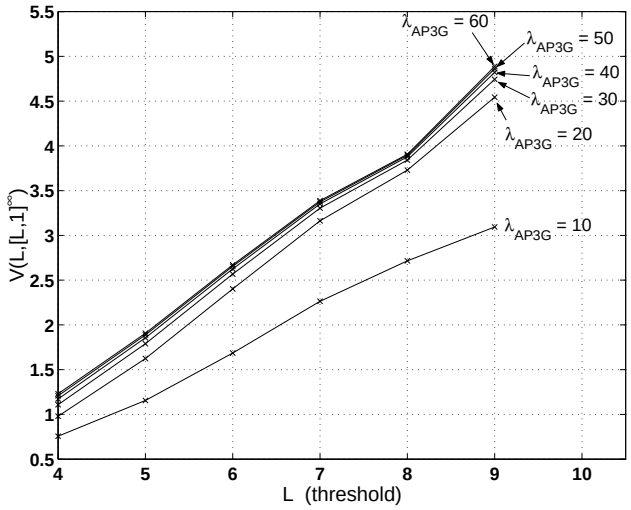


Figure 3.2: $V_{AP}(L,[L,1]^{\infty})$ v/s L for $\lambda_{AP}=3$ and $M_{AP}=10$

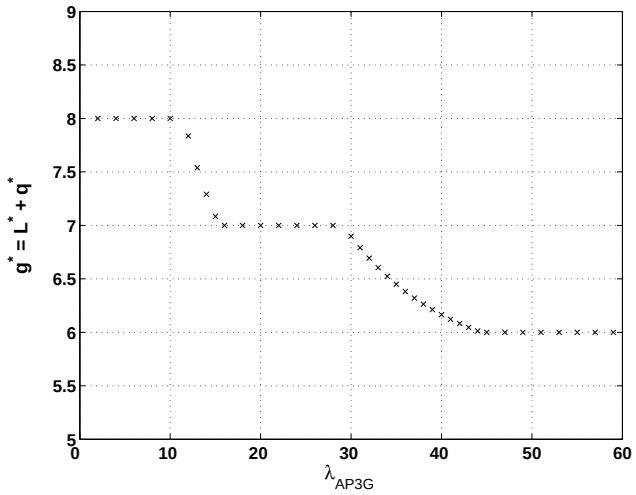


Figure 3.3: g^* v/s λ_{AP3G} for $\lambda_{AP}=3$, $M_{AP}=10$ and $\tau=2.5$

Table 2.1) and $\lambda_{AP}=3$ gives rise to a value of $\tau=2.5$. For $\tau=2.5$, we obtain different values of L^{min} for different values of λ_{AP3G} from Figure 3.2. Finally, these values of L^{min} are then used to compute q^* from Equation 3.4 and Figure 3.3 shows a plot of the equilibrium threshold, $g^*=L^*+q^*=L^{min}+q^*$, against various values of λ_{AP3G}. In this figure a value of $g^*=7.34$ implies that $L^*=7$ and $q^*=0.34$. We clearly observe here that the equilibrium threshold has a special structure of *descending staircase* with increasing arrival rate (λ_{AP3G}) of mobiles in common stream. This special structure is due to the way the threshold type policy has been defined in Equation 3.3. As the value of λ_{AP3G} increases the equilibrium threshold decreases. This implies that for high values of λ_{AP3G} (>44), it is (individually) optimal for mobiles of common stream to join the AP network (w.p. 1) only if there are less than 6 mobiles already connected to AP. Otherwise, it is (individually) optimal for them to connect to the NodeB network (see Equation 3.3). For low values of λ_{AP3G} (<28), it

is (individually) optimal for common stream mobiles to join the AP even if there are 7 or 8 (or less) mobiles already connected to AP. Recall that the AP cell capacity considered in this example is $M_{AP} = 10$. For certain values of λ_{AP3G} we get a non-integer threshold. This implies that (individually) optimal performance (in terms of average service time of each mobile) is achieved when the common stream mobiles connect to AP with a certain probability (q^*, or q in Equation 3.3) and to NodeB otherwise.

3.6 Conclusion

In this chapter, we have considered individually optimal user-network association in an AP-NodeB hybrid cell. To the best of our knowledge this study is the first of its kind. Since it is infeasible to obtain the equilibrium policy for an exhaustive set of network scenarios, we have considered here an ordinary network scenario and explicitly computed the equilibrium policy. Even though the characteristics of the solution to our particular scenario are not depictive of the complete solution space, they can certainly be helpful in acquiring an intuition about the underlying dynamics of individually optimal user-network association in a hybrid cell.

Chapter 4

On the Inefficiency of 802.11 MAC Protocol under Non-Cooperative Control

The IEEE 802.11 MAC protocol, DCF (Distributed Coordination Function), allows nodes in a WLAN to choose an appropriate PHY data transmission rate for each frame transmission. For this purpose, several auto-rate selection algorithms have been proposed in the literature and most of them allow nodes to adapt their rates non-cooperatively, by definition. Under a non-cooperative game setting, each node would choose its rate so as to optimize its own, appropriately defined payoff. In this chapter, we formulate a payoff function comprising of throughput and costs related to power consumption and derive explicit expressions for the optimal rates under non-cooperative and cooperative rate selection. We consider optimization problems for both finite number of nodes n and for the limit $n \to \infty$ and single node throughputs corresponding to the optimal PHY rate under non-cooperative game are compared with those obtained under a cooperative selection of optimal PHY rates. The comparisons reveal that network performance in the non-cooperative game scenario is inefficient as compared to the cooperative scenario.

Note: The material in this chapter has appeared in [AKKV05b].

4.1 Introduction

Mobile devices and Wireless local area networks (WLANs) based on the IEEE 802.11 technologies are becoming more and more popular. The technological challenge for these devices is to achieve maximum throughput levels constrained by the limited available power sources and operating physical data rates. While power is limited due to the limited amount of utilizable battery energy, operating physical data rates depend on the different modulation and channel coding schemes used. For example, the 802.11b PHY (physical layer) provides 4 PHY data rates ranging from 1 to 11 Mbps at the 2.4 GHz band and the high-speed 802.11a PHY, which has been developed to extend the 802.11 operation in the 5 GHz unlicensed band, provides 8 PHY data rates starting from 6 up to 54 Mbps. Power control and PHY data rate control are two central mechanisms that are critical in achieving an efficient functioning of a WLAN. Power control and rate control are also very often used for providing quality of service (QoS) and both are useful for obtaining a radio channel with a low bit error rate (BER). In cellular networks, fixed rate services are often proposed, where power control is used to guarantee a required signal to interference ratio (SIR), see e.g. [Yat95, WB01, JZS04]. However, in IEEE 802.11a/b/g, power control is not a *de facto* standard. In fact, the operating distance of 802.11a/b/g devices decreases with increasing PHY rate and thus additional power control is required. Only 802.11h which is being developed as an extension of 802.11a includes Transmission Power Control (TPC).

The IEEE 802.11 specification for mobile devices allows customization of certain critical operation parameters like PHY data rate and MAC layer frame size and any manufacturer or even user can adaptively select the PHY data transmission rates according to their own criteria, like for example the channel SNR. For this purpose, several auto-rate selection algorithms [KL97, HVB01, GCNC01, AKKD01, QCJS03] have been proposed in the literature and most of them allow WLAN nodes to adapt their PHY rates non-cooperatively, by definition. These algorithms or control schemes either consider only rate control [KL97, HVB01] or only power control [GCNC01, AKKD01] to optimize some kind of a utility or cost function, for example the application throughput or node's power consumption. Some other schemes like MiSer [QCJS03] use both rate and power control to maximize the energy efficiency which is defined as the ratio of the expected delivered data payload to the expected total energy consumption. However, all these schemes operate only in a non-cooperative game setting. That is, they attempt to optimize an individual node's performance in terms of throughput or power consumption. But optimizing an individual node's performance may cause the overall network performance to suffer. Interestingly, Tan et al. in [TG05] have confirmed this possible behavior. They have shown using both simulations and a game-theoretic approach that in a non-cooperative game scenario under DCF, a "rational" node may achieve a higher throughput by using a lower transmission rate than by using a higher transmission rate, but at the expense of a reduced overall network throughput.

In this chapter, we analytically demonstrate the inefficiency of DCF protocol under non-cooperative rate control. The scope of this work is not to propose a distributed cooperative scheme as an alternative to the existing non-cooperative auto-rate selection algorithms.

We only focus on illustrating the inefficiency of non-cooperative rate selection. We formulate a generic payoff function W_n for n nodes which comprises a utility part representing the throughput and a cost part related to power consumption. The utility or throughput is modeled by an explicit expression of the throughput of a node in a typical WLAN and power consumption costs are modeled in two different ways, using a *linear* and an *exponential* relation between power consumption and PHY rates. This kind of a payoff function could be a generalization of the optimization objective of any of the auto-rate selection algorithms mentioned in the previous paragraph. We consider optimizing either the achieved aggregate network (WLAN cell) throughput (cooperative approach) or an individual node's achieved throughput (in a non-cooperative setup) and obtain explicit expressions for the optimal PHY data rates in each case. In the cooperative case the global payoff comprising the total network throughput and total power consumption costs of all mobile nodes is maximized. In the non-cooperative game case, each player seeks to maximize its own payoff and the corresponding solution concept is then the Nash equilibrium. The optimal PHY rates that we obtain analytically in each of the two cases would be the rates that the previously mentioned auto-rate selection algorithms would ideally apply in order to achieve an optimal performance of the network in terms of the payoff function and its composites. However, by comparing single node throughputs of each node at optimal operation in the two different cases of cooperative and non-cooperative settings, it is observed here that the non-cooperative setting is inefficient. At optimal operation in each of the two settings, single node throughputs under the non-cooperative game setting are worse than in the cooperative setting. Indeed, the DCF protocol as it is defined today does not give the best performance in a non-cooperative setting and can be improved to achieve higher overall network efficiency.

In the cooperative control analysis, we seek to maximize the payoff with two different approaches: (i) *global multirate* approach, where we allow each node to use a different PHY rate and seek to obtain the optimal rate for each node. In this approach, the optimal PHY rate used by each node will depend on its channel conditions; (ii) *max-min* approach, where we obtain an optimal fair assignment of PHY rates to all nodes irrespective of their channel conditions (of course, this means that a channel with bad conditions will have to use larger power). In the non-cooperative game analysis, only *multirate* assignment of PHY rates is considered which is similar to the global multirate case in cooperative control. Even though we use simple differential calculus to solve the optimization problems, our main contribution in this work is in obtaining elegant and explicit expressions for the optimal PHY rates. To the best of our knowledge, such explicit expressions for optimal PHY rates using an explicit expression for single node WLAN throughput, have not been obtained before. Our contributions with respect to [TG05] are (i) we demonstrate inefficiency in a non-cooperative setting by deriving explicit expressions for optimal PHY rate whereas [TG05] does not have any such explicit expressions, (ii) we use explicit formulas for the equilibrium throughputs, where as in [TG05] the throughputs are obtained via numerical assumptions, (iii) the formula used in [TG05] for the throughput (as a function of the parameters choice) depends on the attempt rate for which there is no analytical expression

in [TG05], whereas we use an explicit expression for the attempt rate.

There are two different medium access control (MAC) protocols that have been specified for the IEEE 802.11 compliant devices. One is the contention-based Distributed Coordination Function (DCF) protocol, which is mandatory for all IEEE 802.11 devices and the other is the polling-based Point Coordination Function (PCF) protocol, which is optional. In DCF, which has been derived from the CSMA/CA protocol, contending nodes attempt to share channel resources by going into randomly chosen *back-off* durations (in units of *time-slots*) before carrying out a transmission. DCF does not require the presence of a central channel-resource allocating authority point and hence can be used in both *ad-hoc* and *infrastructure* networks. In PCF, access to the channel resources is centrally controlled by the Access Point (AP) and hence PCF can be used only in *infrastructure* networks. In this work, our analysis is based on the DCF protocol that uses the RTS/CTS frames before any data transmission attempts. Ergo, our discussion takes into account both *ad-hoc* and *infrastructure* networks.

4.2 Network model and background

We reuse the network model and explicit expression for the throughput of a node in a typical WLAN setting obtained in [KAMG05a], for constructing our payoff function. Since we only reuse this network model for our analysis and the background work is not the main focus of our work, we mention here a few assumptions and notations about the model that are directly related to our problem and are sufficient for understanding the analysis in this chapter. Further details that are not directly related to the scope of this chapter can be referred to in [KAMG05a].

Let there be n active nodes in a *single cell* IEEE 802.11 WLAN contending to transmit data. Each node uses the Distributed Coordination Function (DCF) protocol with an RTS/CTS frame exchange before any data-ack frame exchange and each node has an equal probability of the channel being allocated to it. It is assumed that every node has infinitely many packets backlogged in its transmission buffer. In other words, the transmission buffer of each node is *saturated* in the sense that there are always packets to transmit when a node gets a chance to do so. It is also assumed that all the nodes use the same *back-off* parameters. Let β denote the long run average attempt rate per node per *slot* ($0 < \beta < 1$) in *back-off time*[1]. By the fact that all nodes use the same back-off parameters and by symmetry, it is assumed that all nodes achieve the same value of β. Conditions for the existence of a unique such β are given in [KAMG05a] and it is to be noted that β is a function of n, i.e., $\beta := \beta(n)$. Let the quantity K be defined such that, at the $(K+1)$th attempt of grabbing the channel by a node for a packet transmission, either the packet transmission succeeds or the packet is discarded. The quantity b_k is defined as the mean back-off duration (in slots)

[1] If we plot transmission attempts as a function of "real" time, and then *cut out* from the plot the channel activity periods (during which all mobiles freeze their back-off), then the new horizontal axis is called the "back-off time", see Section II.A of [KAMG05a].

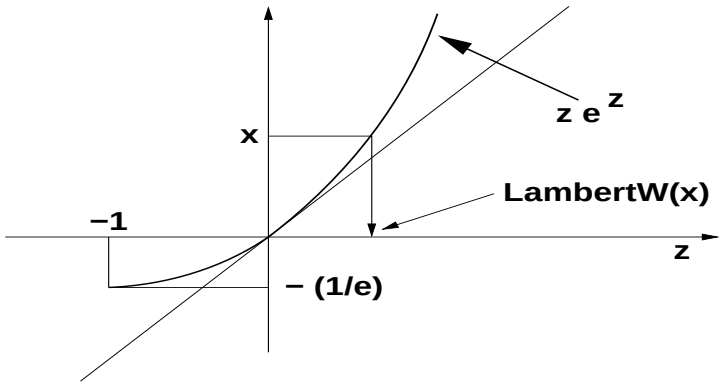


Figure 4.1: The LambertW function is the inverse function of ze^z; notice that for $x \geq -\frac{1}{e}$, $LambertW(x) \leq x$, with equality only for $x = 0$.

at the kth attempt for a packet with $0 \leq k \leq K$ and p is the exponential back-off multiplier, i.e., $b_k = p^k b_0$, $1 \leq k \leq K$. According to the IEEE 802.11 specifications, $p = 2$. We assume here that $p > 1$. We also assume the *decoupling approximation* made by Bianchi in [Bia00a] which says that from the point of view of a given node, the number of attempts by the other nodes in successive slots are i.i.d. binomial random variables with parameters $(n-1)$ and β. Let the MAC frame size of node i be L_i bits and let the PHY rate used by this node be denoted by C_i bits per slot. We assume that $\mathbf{C} \stackrel{\Delta}{=} (C_1, \ldots, C_n)$ denotes the set of all PHY rates used by the n nodes. Let T_o be defined as the transmission overhead in slots related to a frame transmission, which comprises of the SIFS/DIFS, etc and let T_c be defined as the fixed overhead for an RTS collision in slots. Then it follows from [KAMG05a] that the throughput of node i is given by

$$\theta_i(\mathbf{C}, n) = \frac{\beta(1-\beta)^{n-1} L_i}{1 + n\beta(1-\beta)^{n-1}\left(T_o - T_c + \frac{1}{n}\sum_{i=1}^{n}\frac{L_i}{C_i}\right) + \left(1 - (1-\beta)^n\right)T_c}, \tag{4.1}$$

where $\beta := \beta(n)$ for the case when $K \to \infty$ is given by [KAMG05a]

$$\beta = \frac{\eta p - LambertW(\eta(p-1)e^{\eta p})}{\eta b_0},$$

with $\eta = \frac{n-1}{b_0}$. Please see Figure 4.1 for the definition of $LambertW$ function. Note that β does not depend on L_i or C_i, and all nodes achieve the *same* single node throughput even if they use different rates. As is the case in IEEE 802.11, for all nodes that use an RTS/CTS frame exchange before the data-ack frame transmission, we assume throughout our discussion that

$$T_o \geq T_c. \tag{4.2}$$

In our analysis in the following sections, we will consider optimization problems for both finite n and for the limit $n \to \infty$. For handling the latter case, we identify here the

asymptotic aggregate throughput as $n \to \infty$ (this derivation can be found in [KAMG05a] for the special symmetric case where all L_i's and C_i's are equal). An appealing feature of the asymptotic case is the *explicit* expression for β.

Asymptotic throughput: In our discussion we will use asymptotic expressions for the throughput in the following two contexts:

(i) In the case where we consider global multirate PHY rate assignment to all nodes, i.e., each node uses one of c distinct available values of the parameters (C_i, L_i) with $(C_i, L_i) \in \{(C_1, L_1), ..., (C_c, L_c)\}$, we derive here the corresponding asymptotic throughput. Assume that there are m_i nodes using parameters (C_i, L_i). Denote by $\alpha_i(n) = m_i/n$ the fraction of the nodes using (C_i, L_i) among all nodes in the cell. Then the throughput of all nodes using (C_i, L_i) is given by

$$\theta(\alpha_i(n)) = \frac{m_i \beta e^{-n\beta} L_i}{1 + n\beta e^{-n\beta}\left(T_o - T_c + \sum_{i=1}^{c} \frac{\alpha_i(n) L_i}{C_i}\right) + \left(1 - e^{-n\beta}\right) T_c}, \tag{4.3}$$

where we use the Binomial to Poisson approximated version of the throughput expression for the asymptotic case mentioned in Section VII.C of [KAMG05a]. It is assumed that $\alpha_i(n)$ converges to a limit α_i which is a probability measure. Note again that the attempt rate $\beta := \beta(n)$ and the collision probability γ as defined in [KAMG05a] are not functions of L_i nor C_i. Now, first taking $K \to \infty$ [KAMG05a] and then taking the limit $n \to \infty$, it can be observed that

$$\lim_{n\to\infty} n\beta(n) \uparrow \ln\left(\frac{p}{p-1}\right) \tag{4.4}$$

(see Theorem VII.2 in [KAMG05a]). Combining this result with Equation 4.3 we get as $n \to \infty$ the following expression for the aggregate throughput of all nodes using (C_i, L_i):

$$\tau(\alpha_i) = \frac{\alpha_i L_i \left(1 - \frac{1}{p}\right)}{\frac{1}{\ln\left(\frac{p}{p-1}\right)} + \left(1 - \frac{1}{p}\right)\left(T_o - T_c + \sum_{i=1}^{c}\left(\frac{\alpha_i L_i}{C_i}\right)\right) + \frac{T_c}{p \ln\left(\frac{p}{p-1}\right)}}. \tag{4.5}$$

Denote $E_\alpha[L/C] = \sum_{i=1}^{c} \frac{\alpha_i L_i}{C_i}$ and $E_\alpha[L] = \sum_{i=1}^{c} \alpha_i L_i$. Then it follows from Equation 4.5 that the asymptotic global throughput is given by

$$\tau(\alpha) = \frac{E_\alpha[L]\left(1 - \frac{1}{p}\right)}{\frac{1}{\ln\left(\frac{p}{p-1}\right)} + \left(1 - \frac{1}{p}\right)\left(T_o - T_c + E_\alpha[L/C]\right) + \frac{T_c}{p \ln\left(\frac{p}{p-1}\right)}}. \tag{4.6}$$

(ii) In the max-min fair (MMF) case where we assign the same PHY rate to all mobile nodes, we consider all nodes to be symmetric, i.e., they all use the same PHY rate C (they still may have different channel conditions). In this case, if first $K \to \infty$ [KAMG05a] and

then $n \to \infty$, the global throughput is given by Sec. VII.C in [KAMG05a] as:

$$\tau(C) = \frac{L\left(1-\frac{1}{p}\right)}{\frac{1}{\ln\left(\frac{p}{p-1}\right)} + \left(1-\frac{1}{p}\right)\left(T_o - T_c + \left(\frac{L}{C}\right)\right) + \frac{T_c}{p\ln\left(\frac{p}{p-1}\right)}}, \tag{4.7}$$

where p is the exponential back-off multiplier as defined earlier.

4.3 Defining the payoff function

In an efficiently operating WLAN, the goal of mobile nodes is to achieve maximum throughput levels with minimized power consumption costs. In a cooperative scenario, the nodes should cooperate to achieve maximum overall network throughput at minimum combined power consumption. If each node uses the highest available PHY rate, which is say common for all nodes, it may not be the best strategy to achieve the most efficient overall network performance. The reason being that under the given channel conditions, a node may be unnecessarily consuming more power by transmitting at the highest available rate if transmitting at a lower PHY rate does not degrade the combined network throughput. Based on this thought and the fact that under DCF, each node has an equal probability of gaining access to the channel, we define a long-term individual payoff function $\Omega_i(\mathbf{C}, n)$ for each active node i in a WLAN as

$$\Omega_i(\mathbf{C}, n) := \theta_i(\mathbf{C}, n) - \zeta_i Q_i(C_i), \qquad 1 \leq i \leq n, \tag{4.8}$$

where $\theta_i(\mathbf{C}, n)$ is the throughput of node i as defined in Equation 4.1 and $\mathbf{C} = (C_1, \ldots, C_n)$. $Q_i(C_i)$ is a cost related to the power consumption of node i and is a function of the PHY rate C_i and ζ_i is a weight that gives relative importance to the cost versus the throughput, for each node i. These individual payoff functions can be used to construct a global network payoff function defined simply as,

$$W_n(\mathbf{C}) := \sum_{i=1}^{n} \Omega_i(\mathbf{C}, n) = \sum_{i=1}^{n} (\theta_i(\mathbf{C}, n) - \zeta_i Q_i(C_i)). \tag{4.9}$$

Note that, maximizing the payoff functions leads to maximizing the throughput and minimizing the costs related to power consumption.

Now, we consider here two different formulations for the power consumption cost function $Q_i(C_i)$ for each node i. First is a simple *linear* cost function of PHY rates C_i and the second one is an *exponential* cost function of the same. These two formulations are independent of each other and are based on separate experimental results obtained in previous work. Experiments conducted by Gruteser et al. in [GJD+01] with IEEE 802.11 equipment reveal that under given channel conditions and a low transmission power range, the power consumed by a mobile node can be approximated as being linearly proportional to

the PHY rate used. Consequently, $Q_i(C_i)$ can be considered as a linear power consumption cost function of the form,

$$Q_i^{lin}(C_i) = a_i C_i, \tag{4.10}$$

where a_i is a random variable that depends on the path attenuation under given channel conditions, for each node i. Alternatively, an exponential cost function can be derived through simple observation as follows. Motivated by the Shannon's theorem and assuming an AWGN channel that uses complex symbols, the transmission rate of a node is of the form,

$$C(\pi) = W \log_2 \left(1 + \frac{\pi}{z}\right), \tag{4.11}$$

where W is the pass-band spectrum in $Hertz$. π is the transmission power of the node and $z = WN_o/h$, where N_o is the one-sided *power spectral density* of the channel noise and h is a random variable that characterizes the signal attenuation. z is therefore a random variable that depends on channel fading and shadowing. The previous equation can be rewritten as

$$\pi(C) = z(e^{\psi C} - 1), \tag{4.12}$$

where $\psi = \frac{ln2}{W}$. Now, it has been seen in the results of experiments in [GJD+01] that the power consumed by mobile nodes is piecewise linearly proportional to the transmission power. Therefore, an exponential power consumption cost function $Q_i^{exp}(C_i)$ for each node i, can be assumed, which is of the form,

$$Q_i^{exp}(C_i) = z_i(e^{\psi C_i} - 1). \tag{4.13}$$

From the definitions of a_i and z_i in the foregoing discussion it is evident that their average means may vary from one mobile node to another. We denote the expected values of a_i and z_i by $E[a_i]$ and $E[z_i]$.

4.4 Cooperative approach

In the cooperative approach to PHY rate and power control, we shall consider two different scenarios. In the first *global multirate* scenario, we allow each node to use a different PHY rate C_i depending on its channel conditions and we seek to obtain the globally optimal PHY rates for all nodes. In the second *max-min fair* scenario, we assign each node the same PHY rate C and MAC frame size L, at all channel states and we seek to obtain the optimal PHY rate that will maximize the overall payoff of the network. This will of course require an appropriate power control so that in bad channel conditions the transmitted power is larger. In fact, in both the scenarios, it is assumed that all nodes use the same MAC frame size L. We pursue analysis for finite n number of nodes and also consider the asymptotic situation when $n \to \infty$ for both scenarios. Note that equal rate assignment to all nodes in the max-min fair scenario is actually a particular case of the global multirate scenario obtained by simply applying an additional constraint that all rates be equal. However, the asymptotic analysis of the global multirate scenario results in a non-linear optimization problem, whose

solution study is outside the scope of this work. We therefore study the particular case of equal rate assignment with max-min fairness, which gives an explicit expression for the globally optimal PHY rate for the asymptotic analysis. As discussed before, an optimal PHY rate may not be the highest available PHY rate. We shall consider the set of possible values of C or C_i, as the case may be, to lie in an interval of the form $\underline{\mathcal{C}} := [C_l, C_u]$. In 802.11a, where two of the possible PHY rates are 6 Mbps and 54 Mbps, this interval could be $[6, 54]$.

4.4.1 Global multirate (channel dependent) optimization

In this section, we consider global optimization in which we allow each node to use a different PHY rate C_i and we seek to obtain the best choice of C_i, $i = 1..n$. We assume that all values of $E[a_i]$, $E[z_i]$ and ζ_i are known.

4.4.1.1 Finite number of nodes, channel-dependent case

Consider the following problem:

$$\text{Find } \mathbf{C}^* = (C_1^*, ..., C_n^*) \text{ that maximizes } W_n(\mathbf{C}) := \sum_{i=1}^{n} (\theta_i(\mathbf{C}, n) - \zeta_i Q_i(C_i)), \tag{4.14}$$

where $\theta_i(\mathbf{C}, n)$ is defined in Equation 4.1. Then we have,

$$W_n^{lin}(\mathbf{C}) = \frac{q_1}{q_2 + \frac{q_1}{n}\sum_{i=1}^{n}(\frac{1}{C_i})} - \sum_{i=1}^{n} \zeta_i E[a_i] C_i \tag{4.15}$$

and

$$W_n^{exp}(\mathbf{C}) = \frac{q_1}{q_2 + \frac{q_1}{n}\sum_{i=1}^{n}(\frac{1}{C_i})} - \sum_{i=1}^{n} \zeta_i E[z_i](e^{\psi C_i} - 1), \tag{4.16}$$

where q_1 and q_2 are defined as follows,

$$q_1 := n\beta(1-\beta)^{n-1}L, \quad q_2 := 1 + n\beta(1-\beta)^{n-1}(T_o - T_c) + (1 - (1-\beta)^n)\, T_c. \tag{4.17}$$

It can be easily shown that the Hessian of $W_n^{lin}(\mathbf{C})$ and $W_n^{exp}(\mathbf{C})$ are negative definite by the assumption made in Equation 4.2 and therefore $W_n^{lin}(\mathbf{C})$ and $W_n^{exp}(\mathbf{C})$ are strictly concave w.r.t. C_i for every i. Let

$$q_2^i := q_2 + \frac{q_1}{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{1}{C_j}.$$

Denote by $\hat{C}$ the harmonic average of C_i's, i.e.,

$$\hat{C} = \left(\sum_{i=1}^{n} \frac{1}{C_i} \right)^{-1}. \tag{4.18}$$

With these definitions and by partially differentiating $W_n^{lin}(\mathbf{C})$ in Equation 4.15 we get

$$\frac{\partial W_n^{lin}}{\partial C_i} = \frac{nq_1^2}{(nq_2^i C_i + q_1)^2} - \zeta_i E[a_i] = \frac{q_1^2}{\left(q_2 + \frac{q_1}{n\hat{C}}\right)^2 nC_i^2} - \zeta_i E[a_i] \tag{4.19}$$

and similarly by partially differentiating $W_n^{exp}(\mathbf{C})$ in Equation 4.16 we get

$$\frac{\partial W_n^{exp}}{\partial C_i} = \frac{nq_1^2}{(nq_2^i C_i + q_1)^2} - \psi\zeta_i E[z_i]e^{\psi C_i} = \frac{q_1^2}{\left(q_2 + \frac{q_1}{n\hat{C}}\right)^2 nC_i^2} - \psi\zeta_i E[z_i]e^{\psi C_i}. \tag{4.20}$$

Denote

$$H(\hat{C}) = \frac{q_1^2}{\left(q_2 + \frac{q_1}{n\hat{C}}\right)^2 n}. \tag{4.21}$$

Now by equating the derivatives in Equations 4.19 and 4.20 to zero, we obtain:

(i) In the linear case,

$$C_i = \sqrt{\frac{H(\hat{C})}{\zeta_i E[a_i]}}. \tag{4.22}$$

Let

$$Y := \sum_{i=1}^{n} \sqrt{\zeta_i E[a_i]}. \tag{4.23}$$

Then

$$\hat{C} = \left(\sum_{i=1}^{n} \sqrt{\frac{\zeta_i E[a_i]}{H(\hat{C})}}\right)^{-1} = \frac{\sqrt{H(\hat{C})}}{Y},$$

which implies that the solution $\hat{C}^*$ is given by

$$\hat{C}^* = \frac{1}{n}\frac{q_1}{q_2}\left(\frac{\sqrt{n}}{Y} - 1\right).$$

Substituting this solution in Equation 4.22 gives the C_i^*'s.

(ii) In the exponential case, we get from Equation 4.20

$$C_i = \frac{2}{\psi} LambertW\left(\frac{1}{2}\sqrt{\frac{\psi H(\hat{C})}{\zeta_i E[z_i]}}\right). \tag{4.24}$$

Therefore, using the definition of $\hat{C}$ from Equation 4.18, it implies that $\hat{C}^*$ is the solution of

$$\hat{C} = \frac{2}{\psi \sum_{i=1}^{n} \left[LambertW\left(\frac{1}{2}\sqrt{\frac{\psi H(\hat{C})}{\zeta_i E[z_i]}} \right) \right]^{-1}},$$

which yields the C_i^*'s through Equation 4.24.

The above solutions are globally optimal provided they are within the range $\underline{\mathcal{C}}$. We defer the discussion on numerical computations of C_i^*'s to Section 4.6.

4.4.1.2 Large number of nodes

To model the case of a large number of users we shall use a fluid approximation in which there are (non-countably) infinite number of users. We introduce R population classes of mobiles. The parameter z in the exponential cost function will be the same for all mobiles of the same type $r, r = 1, 2, ..., R$ so that mobiles belonging to a given class r have the same channel conditions. We shall thus use the notation $z^{(r)}$ to indicate this dependence. We shall use similarly the notation $a^{(r)}$ for the coefficient appearing in the linear cost. In short, mobiles with the same value of $(a^{(r)}, \zeta^{(r)})$ (in the linear case) or $(z^{(r)}, \zeta^{(r)})$ (in the exponential case) are said to belong to the same class of mobiles having identical channel conditions. We define for each r the vector $\mathbf{x}^{(r)} = (x_1^{(r)}, ..., x_c^{(r)})$ to be the amount of r-type mobiles that use each of the rates $C_1, ..., C_c$. Define $\mathbf{x} = (\mathbf{x}^{(1)}, ..., \mathbf{x}^{(R)})$ to be a multi-strategy for all mobiles. With some abuse of notation, let $x_i := \sum_{r=1}^{R} x_i^{(r)}$ denote the global amount of mobiles that use the rate C_i under $\mathbf{x}$. Denote $\overline{\nu}$ to be the total amount of users. Then $\overline{\nu} = \sum_{i=1}^{c} x_i$. Define $\alpha_i(\mathbf{x}) = x_i/\overline{\nu}$. It follows from Equation 4.6 that

$$\tau(\alpha(\mathbf{x})) = \frac{E_{\alpha(\mathbf{x})}[1] q_1}{q_2 + q_1 E_{\alpha(\mathbf{x})}[1/C]} = \frac{\overline{\nu} q_1}{\overline{\nu} q_2 + q_1 \sum_{i=1}^{c} x_i C_i^{-1}},$$

where q_1 and q_2 are defined as follows,

$$q_1 := L\left(1 - \frac{1}{p}\right), \qquad q_2 := \frac{1 + T_c/p}{\ln\left(\frac{p}{p-1}\right)} + \left(1 - \frac{1}{p}\right)(T_o - T_c). \tag{4.25}$$

To simplify, we shall denote $\tau(\mathbf{x}) = \tau(\alpha(\mathbf{x}))$. Define $Q_i^{(r)}(x_i^{(r)}) = a^{(r)} C_i$ for the linear cost and $Q_i^{(r)}(x_i^{(r)}) = z^{(r)}\left(e^{\psi C_i} - 1\right)$ for the exponential cost. Then our problem of maximizing

the payoff function turns out to be a non-linear optimization problem defined by:

$$\begin{aligned}\max_{\mathbf{x}} W(\mathbf{x}) \quad \text{where } W(\mathbf{x}) &:= \tau(\mathbf{x}) - \sum_{r=1}^{R} \zeta^{(r)} \sum_{i=1}^{c} x_i^{(r)} Q_i^{(r)}(x_i^{(r)}) \\ &= \frac{\overline{\nu} q_1}{\overline{\nu} q_2 + q_1 \sum_{i=1}^{c} \left(\sum_{r=1}^{R} x_i^{(r)} \right) C_i^{-1}} - \sum_{r=1}^{R} \zeta^{(r)} \sum_{i=1}^{c} x_i^{(r)} Q_i^{(r)}(x_i^{(r)}) \\ \text{subject to} \qquad & \sum_{i=1}^{c} x_i^{(r)} = g_r, \forall r, \quad x_i^{(r)} \geq 0, \forall i, r,\end{aligned}$$

where g_r is a predefined constraint on the number of mobiles in class r. Studying the solution for this problem is outside the scope of this work and we therefore study the max-min fair assignment scenario in the following section, which leads to a simple explicit expression for the optimal PHY rate for asymptotically large number of nodes.

4.4.2 Obtaining the max-min fair PHY rates

A max-min assignment of resources to users is a fairness concept characterized by the property that no user i can be assigned more resources unless we decrease the resource assignment to another user j who already has an equal or lesser amount of resources than user i. This is an efficient assignment in the Pareto sense. In our case, it is the PHY rates that are assigned according to the max-min approach. Since all users are assigned an identical max-min objective comprising the common resource entity 'PHY rate' and since the total rate of all users must sum to a finite quantity, the max-min fair assignment leads to an identical rate assignment to all users.

4.4.2.1 Finite number of nodes

We seek to maximize the payoff function defined in Section 4.3 while assigning the same PHY rate C to each node irrespective of the channel conditions. Consider the following problem:

$$\text{Find } C^* \text{ that maximizes } W_n(C) := \sum_{i=1}^{n} (\theta_i(C, n) - \zeta_i Q_i(C)). \tag{4.26}$$

W_n is strictly concave with respect to C (see Appendix for proof) and thus has a unique maximizer C^*. In particular, we have the linear and the exponential costs as,

$$Q_i^{lin}(C) = E[a_i]C \qquad \text{and} \qquad Q_i^{exp}(C) = E[z_i]\Big(e^{\psi C} - 1\Big).$$

Denote

$$u^{lin} = \sum_{i=1}^{n} \zeta_i a_i, \qquad \text{and} \qquad u^{exp} = \sum_{i=1}^{n} \zeta_i z_i,$$

and therefore,

$$E[u^{lin}] = \sum_{i=1}^{n} \zeta_i E[a_i], \qquad \text{and} \qquad E[u^{exp}] = \sum_{i=1}^{n} \zeta_i E[z_i].$$

Then we have

$$W_n^{lin}(C) = \frac{q_1}{q_2 + q_1/C} - E[u^{lin}]C, \qquad W_n^{exp}(C) = \frac{q_1}{q_2 + q_1/C} - E[u^{exp}]\Big(e^{\psi C} - 1\Big). \quad (4.27)$$

where q_1 and q_2 are as defined in Equation 4.17. Differentiating w.r.t. C,

$$\frac{dW_n^{lin}(C)}{dC} = \frac{q_1^2}{(q_2C + q_1)^2} - E[u^{lin}], \qquad \frac{dW_n^{exp}(C)}{dC} = \frac{q_1^2}{(q_2C + q_1)^2} - \psi E[u^{exp}]e^{\psi C} \quad (4.28)$$

and equating the derivative to zero, we obtain the following results:

(i) In the linear case, the unique positive solution of $\frac{dW_n^{lin}(C)}{dC} = 0$ is given by

$$C^* = \frac{q_1}{q_2}\left(\frac{1}{\sqrt{E[u^{lin}]}} - 1\right) \quad (4.29)$$

provided that $0 < E[u^{lin}] < 1$. If $E[u^{lin}] \geq 1$ then there is no positive solution.

(ii) In the exponential case the unique positive solution of $\frac{dW_n^{exp}(C)}{dC} = 0$ is given by

$$C^* = \frac{2}{\psi} LambertW\left(\frac{1}{2}\frac{q_1}{q_2}\sqrt{\frac{\psi}{E[u^{exp}]}}\exp\left(\frac{1}{2}\frac{q_1}{q_2}\psi\right)\right) - \frac{q_1}{q_2} \quad (4.30)$$

In either of linear or exponential case, if C^* lies within $\underline{\mathcal{C}}$ then it is the unique globally optimal rate assignment solution for problem in Equation 4.26. If not, then the optimal solution is obtained on one of the two boundary points of $\underline{\mathcal{C}}$. We defer the discussion on numerical computations of C^* to Section 4.6.

4.4.2.2 The asymptotic case

We present here the asymptotic behavior for large number of nodes. The optimization is based on the expression for the asymptotic throughput given by Equation 4.7. As a simplifying assumption, we assume here that a_i and z_i have the same distribution for all mobiles and all ζ_is are identical. Consider the following problem:

$$\text{Find } C^* \text{ that maximizes } W(C) := \tau(C) - \zeta Q(C), \quad (4.31)$$

where $Q(C) = E[a]C$ for the linear cost and $Q(C) = E[z]\Big(e^{\psi C} - 1\Big)$ for the exponential one. $W(C)$ turns out to be strictly concave in C (see [AKKV05a] for proof) and therefore it has a unique maximizer. Writing $W(C)$ for the linear and exponential cases as

$$W^{lin}(C) = \frac{q_1}{q_2 + q_1/C} - \zeta E[a]C \quad \text{and} \quad W^{exp}(C) = \frac{q_1}{q_2 + q_1/C} - \zeta E[z]\Big(e^{\psi C} - 1\Big),$$

where q_1 and q_2 are defined in Equation 4.25. The optimal C is obtained by differentiating $W^{lin}(C)$ and $W^{exp}(C)$ and equating them to zero, which gives the following unique positive solution for the linear and exponential cases, respectively:

$$C^*_{lin} = \frac{q_1}{q_2}\left(\frac{1}{\sqrt{\zeta E[a]}} - 1\right), \quad C^*_{exp} = \frac{2}{\psi} LambertW\left(\frac{1}{2}\frac{q_1}{q_2}\sqrt{\frac{\psi}{\zeta E[z]}}\exp\left(\frac{1}{2}\frac{q_1}{q_2}\psi\right)\right) - \frac{q_1}{q_2}.$$

If C^* lies within $\underline{\mathcal{C}}$ then it is the unique globally optimal rate assignment solution for problem in Equation 4.31. If not then the optimal solution is obtained on one of the two boundary points of $\underline{\mathcal{C}}$. Also note that C^* here has the same form as in the finite n case but with different q_1 and q_2.

4.5 Non-cooperative game

In this section we analyze the non-cooperative behavior of mobile nodes. We shall model this situation using non-cooperative game theory and obtain the equilibrium.

4.5.1 Finite number of nodes

In a non-cooperative rate control setting with finite number of nodes, each node uses the same MAC frame size L and is allowed to use a different PHY rate as in the global multirate allocation in the cooperative approach. But here the objective of each node is to maximize its individual payoff function $\Omega_i(\mathbf{C}, n)$ as defined before in Equation 4.8,

$$\Omega_i(\mathbf{C}, n) = \theta_i(\mathbf{C}, n) - \zeta_i Q_i(C_i). \tag{4.32}$$

For every i, Ω_i is strictly concave w.r.t. C_i and continuous w.r.t. $C_j, j \neq i$. It then follows from Rosen [Ros65] that a Nash equilibrium exists. In particular, we shall be interested in the linear $Q_i^{lin}(C_i) = E[a_i]C_i$ and the exponential $Q_i^{exp}(C_i) = E[z_i]\left(e^{\psi C_i} - 1\right)$ cases. We have

$$\frac{\partial \Omega_i^{lin}}{\partial C_i} = \frac{H(\hat{C})}{nC_i^2} - \zeta_i E[a_i], \qquad \frac{\partial \Omega_i^{exp}}{\partial C_i} = \frac{H(\hat{C})}{nC_i^2} - \psi\zeta_i E[z_i]e^{\psi C_i}, \tag{4.33}$$

where $H(\hat{C})$ is the same as defined before in Equation 4.21. These equations are similar to the ones we obtained in Section 4.4.1 with an extra factor of n in the denominator. Equating them to zero:

(i) For the linear case we obtain,

$$\hat{C}^* = \frac{1}{n}\frac{q_1}{q_2}\left(\frac{1}{Y} - 1\right), \tag{4.34}$$

where Y is the same as defined in Equation 4.23 and C_i^*'s are obtained from

$$C_i = \sqrt{\frac{H(\hat{C})}{n\zeta_i E[a_i]}}. \tag{4.35}$$

(ii) Similarly for the exponential case, we obtain $\hat{C}^*$ as a solution of

$$\hat{C} = \frac{2/\psi}{\sum_{i=1}^{n}\left[LambertW\left(\frac{1}{2}\sqrt{\frac{\psi H(\hat{C})}{n\zeta_i E[z_i]}}\right)\right]^{-1}},$$

which yields the C_i^*s through

$$C_i^* = \frac{2}{\psi} LambertW\left(\frac{1}{2}\sqrt{\frac{\psi H(\hat{C}^*)}{n\zeta_i E[z_i]}}\right). \tag{4.36}$$

We defer the discussion on numerical computations of C_i^*'s to Section 4.6.

4.5.2 Non-cooperative game with infinitely many users

In the case of a very large number of players, a reasonable model is of a game with a continuum set of players. We are motivated by the Wardrop equilibrium concept in road traffic in which routes of cars are chosen so as to minimize the cars' delays, [War52, Pat94], and where the set of cars is assumed to be infinite. If we denote the throughput of all nodes in class r by $\tau_r(\mathbf{x})$ and the payoff of a node in class r by $W_r(\mathbf{x}, C')$ then for the exponential power consumption costs, the payoff is given by

$$W_r(\mathbf{x}, C') := \frac{1}{g_r}\tau_r(\mathbf{x}) - \zeta^{(r)} z^{(r)}\Big(e^{\psi C'} - 1\Big), \tag{4.37}$$

where g_r is the same as defined in Section 4.4.1. Now, $\mathbf{x}^* = (\mathbf{x}^{(1)*}, ..., \mathbf{x}^{(R)*})$ is called a Wardrop type equilibrium if for each r we have

$$(x_i^{(r)})^* \geq 0, \forall i, r \qquad \sum_{i=1}^{c}(x_i^{(r)})^* = g_r, \forall r, \qquad W_r(\mathbf{x}^*, C') \geq W_r(\mathbf{x}^*, C),$$

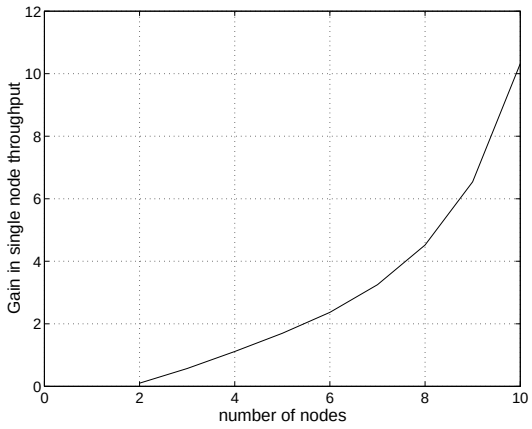


Figure 4.2: Gain in Cooperative multirate over Non-cooperative multirate with linear power cost.

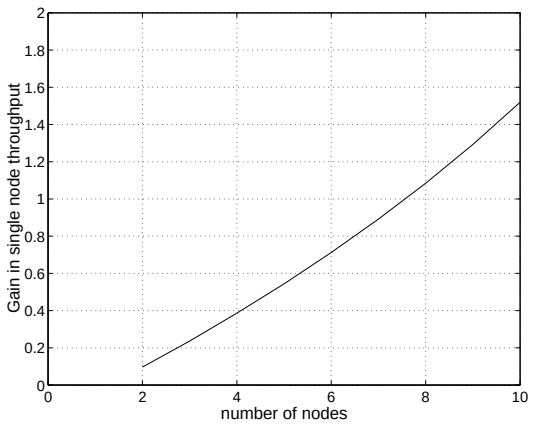


Figure 4.3: Gain in Cooperative multirate over Non-cooperative multirate with exponential power cost.

where the last inequality holds for all $C \in \underline{\mathcal{C}}$, and for all C' in the support of $\mathbf{x}^{(r)*}$ (i.e., for all i for which $(x_i^{(r)})^* > 0$. It is seen from Equation 4.37 that for any value $\mathbf{x}$, the maximization of $W_r(\mathbf{x}, C')$ over C' is obtained by choosing the smallest available C'. This implies that as the number of nodes becomes large, the only Wardrop equilibrium that is highly inefficient is when all nodes end up using the smallest available physical rate. The reason behind it is that in the case of a lot of users, the choice of C' by a single node has a negligible impact on its own throughput, and hence the node only considers minimizing its own cost related to power consumption.

4.6 Numerical Studies

In this section, we numerically examine the closed form expressions for the optimal PHY transmission rates, single node throughputs and overall payoffs. We compute these quantities as a function of number of nodes. The following set of parameters have been used to study the optimal transmission rates and the corresponding single node throughputs and overall payoffs. In the linear cost, $E[a_i]$ is set to vary uniformly from $a_i^{min} = 0.5 * 10^{-3}$ to $a_i^{max} = 1 * 10^{-3}$ $watts$ per $bits/slot$ for each mobile i. In the exponential cost, $E[z_i]$ is set to vary uniformly from $z_i^{min} = WN_o/h_i^{min}$ to $z_i^{max} = WN_o/h_i^{max}$ where value of W (pass-band spectrum) is taken as 20 MHz for an 802.11a system, N_o (one-sided power spectral density) is taken as $5.52 * 10^{-21}$ $watts/Hz$ and for the Rayleigh fading case $h_i^{min} = 10^{-11}$ and $h_i^{max} = 10^{-8}$. The back-off multiplier $p = 2$ and $b_0 = 16$ slots in $b_k = p^k b_0$. The data frame transmission overhead $T_o = 52$ slots, the RTS collision overhead $T_c = 17$ slots and the MAC frame size $L = 12000bits(1500bytes)$. The slot size is taken as $20\mu s$ and $K = 10$ [KAMG05a]. For simplification the parameter ζ_i is taken to be the same for all nodes $i = 1..n$ and its values are displayed below each plot. The plots obtained from the numerical computations are presented at the end of the chapter.

Comparison of cooperative and non-cooperative solutions: In the PHY rate plots (Figures 4.4, 4.6, 4.8, 4.10, 4.12 and 4.14), we observe that the optimal PHY rate of each node decreases with increasing number of nodes. It can be seen that in cooperative global multirate and non-cooperative multirate allocations, for a given number of nodes, each node is assigned a different rate depending on the parameters $E[a_i]$ and $E[z_i]$ for channel conditions. When we have a *linear* cost associated with the power consumption, then for $n = 2$, the single node throughput in the cooperative global multirate case (Figure 4.7) is around 0.1 times higher than the non-cooperative multirate case (Figure 4.9). In fact with increasing n, the single node throughput *gain* in the cooperative global multirate scenario over the non-cooperative multirate scenario varies from 0.1 times for $n = 2$ to up to more than 10 times for $n = 10$ (Figure 4.2). This is a significant amount of increase in single node throughputs. When the cost associated with power consumption is *exponential*, the single node throughput gain in cooperative allocation (Figures 4.3,4.13) over non-cooperative allocation (Figure 4.15) varies from around 0.1 times for $n = 2$ to up to 1.5 times for $n = 10$. In both cases, this illustrates that the gain in single node throughput becomes more significant with increasing number of users. We also observe that the cooperative max-min fair scheme (Figures 4.5,4.11) performs almost equally well as the cooperative global multirate scheme (Figures 4.7,4.13). All these observations clearly illustrate that cooperative rate allocation strategy results in higher single node throughputs and hence higher total network throughput as against non-cooperative strategy. Our analysis thus confirms that the DCF protocol under non-cooperative rate control setting is not efficient.

4.7 Conclusion

In this chapter, we have analyzed cooperative and non-cooperative rate control in an IEEE 802.11 WLAN, by optimizing a payoff function that comprises throughput and costs related to power consumption. It is observed through numerical studies that cooperative control is more efficient than non-cooperative control. With a linear cost approximation, the single node throughput in the cooperative approach is observed to be 0.1 to 10 times more than in the non-cooperative game approach. The gain varies from 0.1 to 1.5 times in the exponential cost approximation case. Thus a first glimpse of cooperative and non-cooperative control in an 802.11 WLAN by our analysis shows that the currently used DCF protocol in 802.11 does not perform with highest efficiency in a non-cooperative rate control setting.

Appendix: Consider the linear and exponential payoff functions in Section 4.4.2. The second derivatives of the linear and exponential payoffs are given by

$$-\frac{2q_1^2 q_2}{(q_2 C + q_1)^3} \qquad \text{and} \qquad -\frac{2q_1^2 q_2}{(q_2 C + q_1)^3} - \psi^2 \zeta E[z] e^{\psi C},$$

respectively, where $q1$ and $q2$ for finite n and $n \to \infty$ are given by Equations 4.17 and 4.25, respectively. The second derivatives are strictly negative by the assumption made in Equation 4.2 and hence the payoff functions considered in Section 4.4.2 are indeed (strictly) concave.

Variation with n and linear cost

Cooperative Approach (Max-min fair)

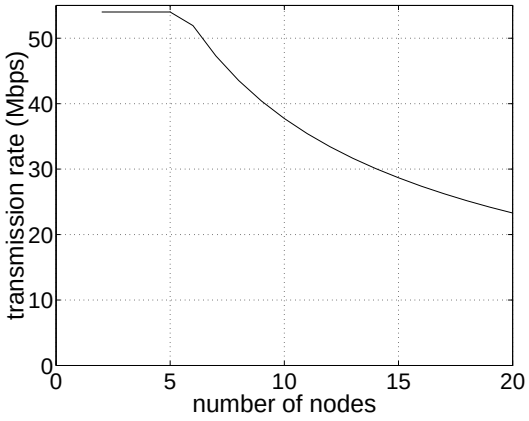


Figure 4.4: Using Eqn. 4.29, $\zeta_i = 6$.

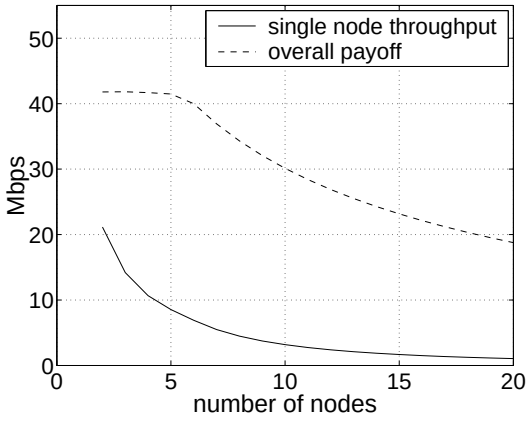


Figure 4.5: Using Eqn. 4.27, $\zeta_i = 6$.

Cooperative Approach (Global multirate)

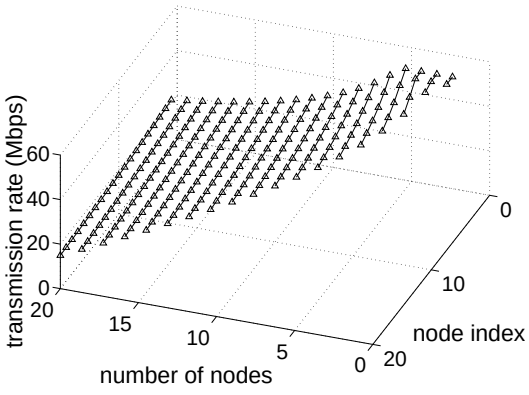


Figure 4.6: Using Eqn. 4.22, $\zeta_i = 9$.

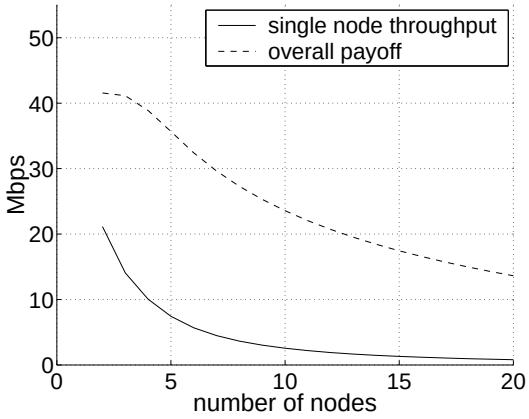


Figure 4.7: Using Eqn. 4.15, $\zeta_i = 9$.

Non-cooperative game (Multirate)

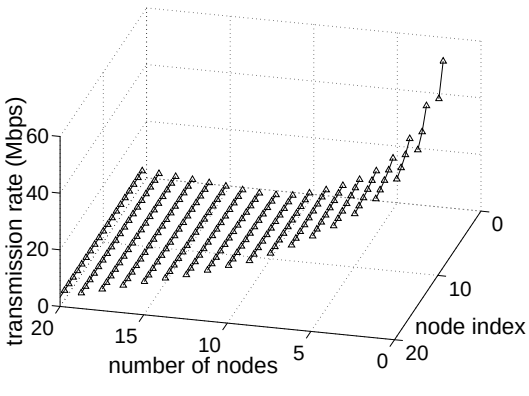


Figure 4.8: Using Eqn. 4.35, $\zeta_i = 9$.

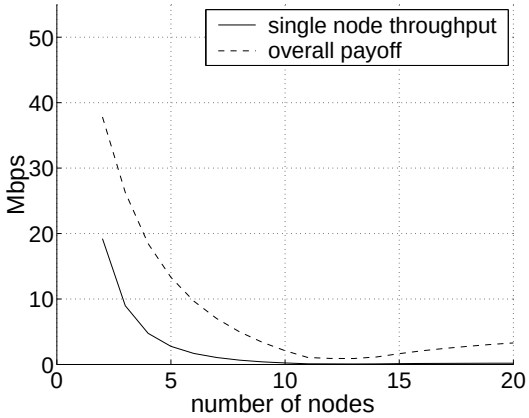


Figure 4.9: Using Eqn. 4.32, $\zeta_i = 9$.

Variation with n and exponential cost

Cooperative Approach (Max-min fair)

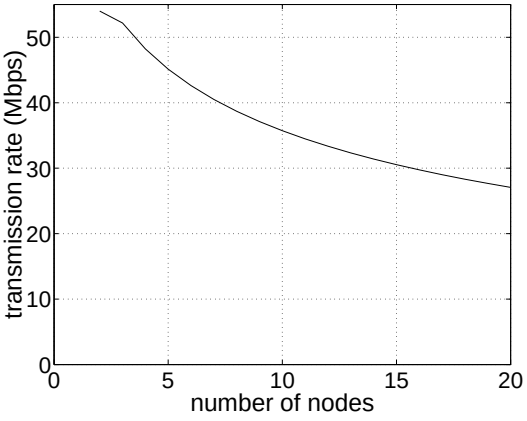


Figure 4.10: Using Eqn. 4.30, $\zeta_i = e^5$.

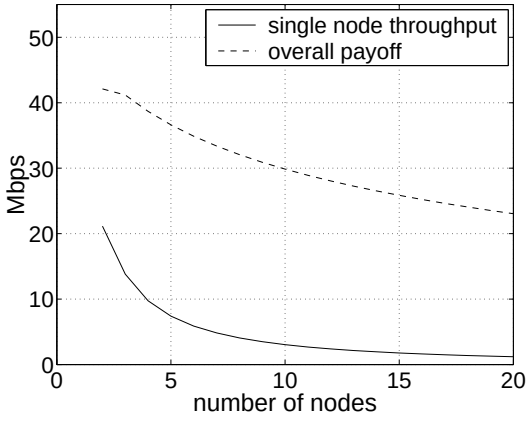


Figure 4.11: Using Eqn. 4.27, $\zeta_i = e^5$.

Cooperative Approach (Global multirate)

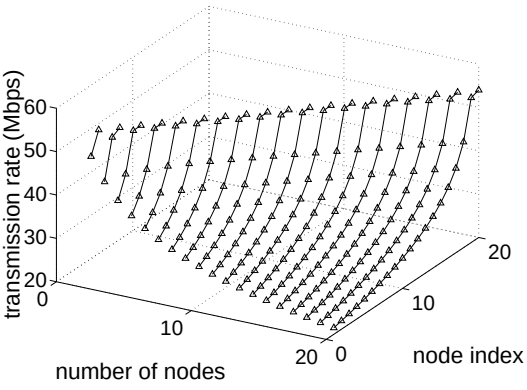


Figure 4.12: Using Eqn. 4.24, $\zeta_i = e^5$.

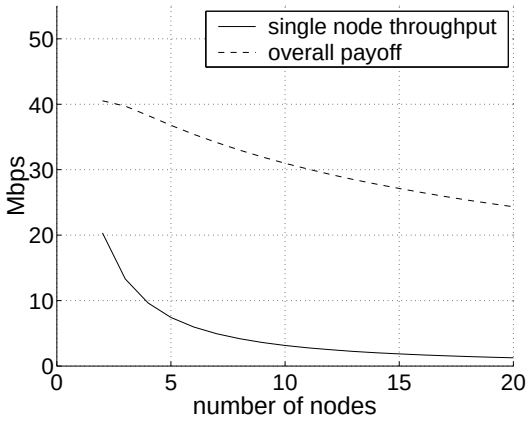


Figure 4.13: Using Eqn. 4.16, $\zeta_i = e^5$.

Non-cooperative game (Multirate)

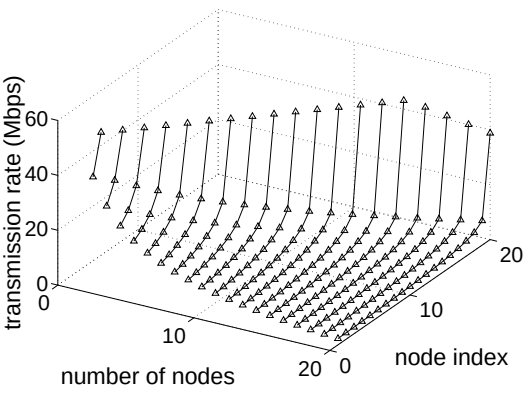


Figure 4.14: Using Eqn. 4.36, $\zeta_i = e^5$.

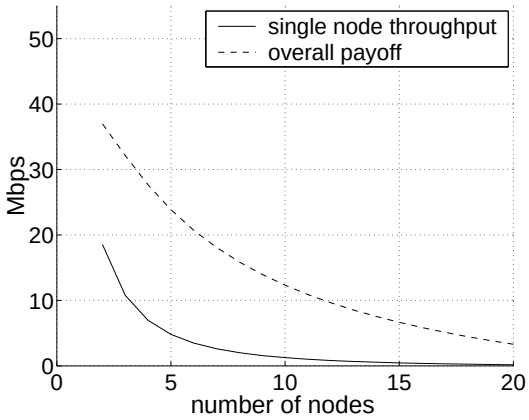


Figure 4.15: Using Eqn. 4.32, $\zeta_i = e^5$.

Chapter 5

Performance Analysis of a Fountain Codes based Transport in a WLAN Cell

A Fountain Codes based Transport (FCT) protocol relies on an alternate paradigm to that of the ubiquitous TCP. It abolishes the need for a reverse feedback mechanism usually essential to provide reliability in packet data transmission. Absence of a reverse feedback mechanism can substantially improve the performance of networks with half-duplex wireless channels (such as 802.11 WLANs), where collisions between forward and reverse MAC frame transmissions contribute significantly towards performance degradation. We propose a Markovian stochastic framework to model the performance of a simple FCT protocol in a single cell IEEE 802.11 WLAN. Our model allows the WLAN Access Point to employ a generic rate control algorithm for MAC frame transmissions on the downlink. Using renewal theory we provide an explicit expression for the average downlink throughput. *ns2* simulations are used to validate our model and the analytically obtained throughput metric. A detailed performance analysis study is then carried out to provide insights into the choice of various system parameters that can lead to optimal network performance. Finally we present a brief comparison between the performance of FCT and TCP through simulations.

Note: Note: The material in this chapter has appeared in [KA]. This work was sponsored by EuroNGI.

5.1 Introduction

Fountain Codes [Mac05, BLM02] are rateless erasure codes and offer a very promising future for ameliorating existing data packet transmission techniques. Specific type of fountain codes such as Luby Transform (LT) Codes, Raptor codes, etc. [Lub02, Sho03], can be used by a sender to generate encoded packets from source data packets *on the fly* and the number of such encoded packets can be potentially limitless. Unlike traditional end-to-end transmission mechanisms like the popular TCP protocol, Fountain Codes based Transport (FCT) protocols do not require re-transmission of lost packets. If the original file at sender side comprises N_p packets, then by decoding *any* set of $N_p(1+\epsilon)$ (slightly more than N_p) fountain coded packets received, the receiver can recover the whole file with probability $1-\delta$. The probability of failure to decode the file, δ, is bounded above by $\delta \leq 2^{-\epsilon N_p}$ and depends on the *degree distribution* used to code the packets at sender side [Mac05]. The quantity $1+\epsilon$ denotes the *decoding inefficiency* and its value may vary from 1.05 to 1.4 depending on the coding scheme used [Mac05, KPR05]. End-to-end transport protocols based on Fountain Coded Packets ($FCPs$) can thus offer reliable transmission without the need for re-transmitting lost information.

It is well known that the standard TCP for wired networks performs very poorly for assymetric wireless channels [ea97, ea00b, ea05b]. Assymetry in wireless channels arises mainly due to difference in bandwidth and latency on the forward and reverse paths. Asymmetric bandwidth can easily break the TCP ack clocking mechanism, i.e., acks get spaced farther apart due to queuing at the bottleneck link with lower bandwidth (see [ea05b] and references there in). On the other hand, asymmetric latency may cause high transmission delays due to half-duplex radios, overhead per packet due to MAC protocol, etc. [ea05b]. Moreover, networks with half-duplex channels such as 802.11 WLANs may suffer from severe performance degradation due to large number of MAC frame collisions when TCP ack traffic is present on the reverse link [ea99]. One solution to all these problems could be to adopt an FCT protocol instead of TCP. A few FCT protocols have been proposed by some authors in [LFC05, KPR05]. These protocols aim to replace TCP and hence increase a sender's transmission efficiency by abandoning altogether the need for TCP ack packets. In this chapter, we propose a simple stochastic model for the performance analysis of an FCT protocol in a single cell 802.11 WLAN. Our model allows for the Access Point (AP) to employ a *generic* adaptive rate control algorithm such as Adaptive Auto Rate Fallback (AARF) or Adaptive Multi Rate Retry Algorithm (AMRR) [LMT04]. Such algorithms are used to adaptively select PHY (physical layer) rates for packet transmission, depending on the varying wireless medium characteristics such as fading, attenuation, etc.

5.1.1 Motivation and Related Work

FCT protocols are expected to exhibit improved efficiency for various different kinds of networks and applications (see [Mit04] and references there in). For instance in multicast networks they can assist in avoiding feedback implosion and provide simple means of han-

dling heterogeneous users and disparate start times [Mit04]. Other than that they can also be effective in overlay networks [Mit04]. However, network architectures in the future are more likely to be *heterogeneous* in nature with the communication path from one end to another consisting of both (multi-hop) wired and (last-hop) wireless links, such as in the case of 802.11 WLANs [ea02]. As compared to wired links, wireless links in a heterogeneous network possess certain undesired characteristics such as high link error rates, added delays due to coding, limited capacity, etc. Thus, performance of an FCT protocol is more critical over the last-hop wireless link. Moreover, since 802.11 WLANs are becoming ubiquitous, we are motivated in this work to carry out a performance analysis study of an FCT protocol for the specific case of last-hop wireless link in a single cell WLAN.

Due to patent protection of fountain codes (LT codes to be specific) [Lub03], research efforts in areas related to fountain codes have been limited. Consequently, very little research literature on FCT protocols is available. Nonetheless, some authors in [KPR05, LFC05, Yan05, UD05] have proposed and studied a few FCT protocols. Lopez et al. in [LFC05] present a game theoretic analysis of protocols based on fountain codes and their results suggest that in a generic setting, hosts using TCP have an incentive to switch to an FCT protocol. However, as per our knowledge none of the previous works have mathematically modeled and analyzed the performance of an FCT protocol in an 802.11 WLAN setting.

5.1.2 Main Contributions

The contributions of this work are *threefold*. Firstly, this work is the first we know of that provides a simple stochastic model for an FCT protocol in an 802.11 WLAN. In addition, our model allows for the WLAN AP to employ a *generic* rate control algorithm. Secondly, we provide a detailed performance analysis study of the FCT protocol in a WLAN setting in which we observe certain trade-offs and performance variations w.r.t. various system parameters. These observations lead to hints for choosing system parameter values that can achieve optimal network performance. Thirdly, we provide some performance comparisons between FCT and TCP through a simulation study.

5.2 A Fountain Codes based Transport (FCT)

For our analysis and simulation purposes, we consider a very basic FCT protocol similar to the one proposed by authors in [LFC05]. The protocol is described as follows. Given a data file of size N_p packets, the application or transport layer at the sender side encodes these packets *on the fly* using a suitable fountain code, say for e.g., the LT codes. These fountain coded packets (FCPs) are then passed on to the lower layers which in turn transmit these FCPs to the receiver. The number of FCPs that can be generated from the original N_p packets is potentially limitless due to the design of LT codes. Thus, the sender keeps on transmitting a new set of FCPs pertaining to the same original file of N_p packets, until it receives some kind of acknowledgement from the receiver indicating the receipt of a sufficient number of FCPs that may now be decoded to recover the original file. Every FCP is useful

and can be exploited to decode the complete file once the receiver has received at least $N_p(1+\epsilon)$ FCPs (slightly more than N_p). After collecting at least $N_p(1+\epsilon)$ FCPs, the receiver sends an *un-coded* 'STOP' message packet to the sender indicating the receipt of sufficient number of FCPs for being able to decode the complete file of N_p packets. The sender then immediately ceases to transmit any more FCPs. Note that this FCT protocol is different from the usual UDP protocol mainly because of the 'STOP' message acknowledgement. We will see later in Section-5.3.4 that this additional 'STOP' message can not be neglected in the stochastic model.

As for the 802.11 WLAN, we assume that the transport layer operates in a *split mode* similar to that in the case of Wireless TCP (see [ea05b] and references there in). In split mode the WLAN AP acts as a terminal node for both the wired network and the last-hop wireless link. Both sender and receiver communicate independently with the WLAN AP router without any knowledge of each other. In our case, the split mode is assumed to operate such that TCP is employed for the wired network and FCT protocol is employed for the last-hop wireless link.

Remark: In our protocol description the receiver sends 'STOP' message after it has received at least $N_p(1+\epsilon)$ FCPs. However, our protocol could be modified such that the receiver sends 'STOP' message *not* after receiving $N_p(1+\epsilon)$ FCPs but only after it has successfully decoded and recovered the original file (since FCPs in excess of $N_p(1+\epsilon)$ will reduce the bound $2^{-\epsilon N_p}$ on the failure probability). It is to be noted that this modification or any such similar modification does not affect our modeling and analysis that follow because our model is valid for *any* number of FCPs received by the receiver.

5.3 Model

Assume there are N 802.11 nodes present in a single cell WLAN including the AP. Therefore, there are $N-1$ wireless nodes connected to an AP, engaged in either streaming (e.g., video streaming) or interactive (e.g., HTTP like traffic) data transmissions. We limit our focus to *downlink* transmissions in the WLAN cell since the bulk of data transfers are usually carried out from an AP to the wireless nodes. Assume a simple homogeneous scenario where each of the $N-1$ nodes downloads a file of size N_p un-coded packets from the AP. Each node must therefore receive at least $D \triangleq N_p(1+\epsilon)$ FCPs to decode the original file with probability $1-\delta$. Time is divided into a discrete set of 802.11 *slots*. For instance, in IEEE 802.11b standard each slot is $20\mu s$ long. The AP is assumed to implement the optional RTS/CTS mechanism in order to reserve the channel medium for transmission of each MAC frame. Figure-5.1 illustrates a MAC frame transmission *sequence* between the AP and a single wireless node, comprising RTS, CTS, MAC data and ACK frames. The MAC frame here consists of the FCP passed on to the MAC layer by upper layers. The commencement of every frame transmission coincides with the beginning of an 802.11 time slot. Various time intervals of Distributed Inter-Frame Spacing (DIFS), Short Inter-Frame Spacing (SIFS) and a random back-off, separate the commencement of frame exchanges as shown in Figure-5.1.

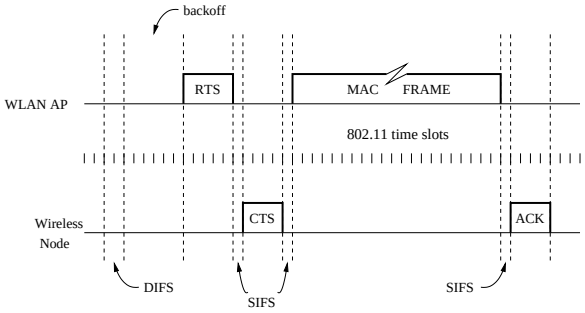


Figure 5.1: MAC frame transmission *sequence*

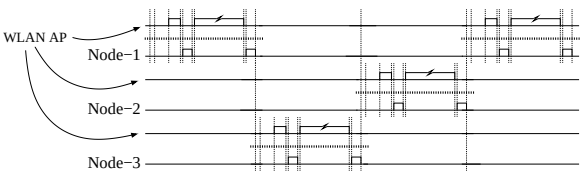


Figure 5.2: Balanced resource allocation by AP

5.3.1 Assumptions on the Back-off Process

As part of the IEEE 802.11 standard, the length of back-off interval (in terms of number of slots) is uniformly randomly chosen from a set $\{0, 1, \ldots, CW - 1\}$, where CW denotes the current size of the contention window corresponding to the CSMA/CA based Distributed Coordination Function (DCF). The value of CW may vary from CW_{min} to CW_{max} in exponential powers of two. If any of the RTS, CTS, MAC or ACK frame transmissions is unsuccessful due to either collision or wireless loss, the value of CW is doubled before a re-transmission attempt is made for the lost frame. For our model, we instead assume that the value of CW remains constant and is not increased exponentially as described in the foregoing discussion. Moreover, we assume that the length of back-off interval, T_B, is actually sampled from a geometric distribution with mean $\frac{1}{p} = E[T_B] = \frac{CW-1}{2}$ and not uniformly randomly as mentioned earlier. These two simplifying assumptions have been used previously in other works as well (see [ea00a, KS04b, KS04a] and Chapter-8 in [KMK04]). An 802.11 node operating under similar assumptions is in fact a p-persistent type 802.11 (see [ea06a] and references there in). Our simulation results in Section-5.5 however exhibit that in terms of the throughput metric that we are interested in, these assumptions cause insignificant deviation from the original behavior of an 802.11 AP or node.

5.3.2 Balanced Resource Allocation by AP

Considering only downlink traffic from the AP to the $N-1$ wireless nodes, it has been shown in a previous work [MKA06] that for HTTP like traffic that employs TCP, the downlink throughput of AP to all wireless nodes is equally and fairly distributed among the $N-1$ nodes. In other words, the average number of transmission opportunities ($TxOPs$) acquired by the AP for MAC frame transmission to each wireless node are nearly equal over a long period of transmission. We can safely assume a similar balanced resource allocation in terms of $TxOPs$ for our simple FCT due to the way it has been defined in Section-5.2. Later, simulation results in Section-5.5 actually confirm this assumption.

5.3.3 Generic Rate Control and Mobility

Most of the IEEE 802.11 standards (e.g., 802.11 a/b/g) support multiple PHY data transmission rates. In our framework, we allow the AP to dynamically select a PHY rate adapted to the continuously varying wireless medium characteristics such as fading, attenuation, etc. We further assume that the corresponding rate control algorithm ensures a balanced resource allocation (in terms of *TxOPs*) among the $N-1$ wireless nodes, as discussed in Section-5.3.2. Many rate control algorithms have been proposed in the literature. Some of these include the Auto Rate Fallback (ARF) algorithm, Receiver Based Auto Rate (RBAR) algorithm and the Adaptive Multi Rate Retry (AMRR) algorithm (see [LMT04] and references there in). Whichever algorithm is employed, we assume that the AP may dynamically select up to K distinct transmission rates for the downlink. Let $\{r_1, \ldots, r_K\}$ denote these K different rates in units of *bits/slot* and let $\mathcal{K} = \{1, \ldots, K\}$ denote their index set.

In brief, the aforementioned rate control algorithms function in the following manner. Either the receiver's measure of the Signal to Noise Ratio (SNR) of received signal or the sender's estimation of the channel loss probability are exploited to adaptively select a higher or lower PHY rate among the K available rates. The PHY rate is increased or decreased either on a MAC frame by frame basis (see [LMT04] and references there in), or after a suitably calculated number of MAC frame transmission attempts elapse or at large ($\gg$ one 802.11 slot) periodic intervals with varying periodicity (see [LMT04] and references there in). In order to *generalize* these different set of rate selection criteria, for our model we introduce a generic PHY rate transition matrix $H = [h_{k,j}]_{K \times K}$, $k, j \in \mathcal{K}$, which is common to all $N-1$ AP to wireless node connections. The elements of this matrix denote the *mean* probability, $h_{k,j}$, with which the chosen rate control algorithm will alter the PHY rate to r_j for the next MAC frame transmission attempt, given that the PHY rate for the ongoing frame transmission is r_k. Such a matrix can be appropriately defined as a function of the channel conditions for any given rate control algorithm. As a crucial basis for the discussion in Section-5.3.4, we assume that the PHY rate transition process for each node is stationary. Equivalently, H is assumed to be a time homogeneous matrix.

Wireless nodes in the AP cell are allowed to be mobile in our framework. However, we do not explicitly model the mobility of such nodes. Instead, we assume that the PHY rate transition matrix H can be tuned in real time in order to incorporate an additional rate control strategy that may be required to take care of the effects on channel conditions due to mobility of nodes.

5.3.4 Formulation as Markov Renewal Reward Process

Due to its design, the DCF protocol defined in 802.11 standard allows only one successful MAC frame exchange to be ongoing at any time. Then if we focus only on downlink, the AP will be engaged in a MAC frame transmission sequence with only one of the $N-1$ nodes at any instant. Moreover, by the balanced resource allocation assumption, such frame transmissions will be nearly equally distributed over time for all $N-1$ wireless nodes. Figure-5.2 shows a snapshot of such a scenario for $N = 4$. Therefore, from now onwards

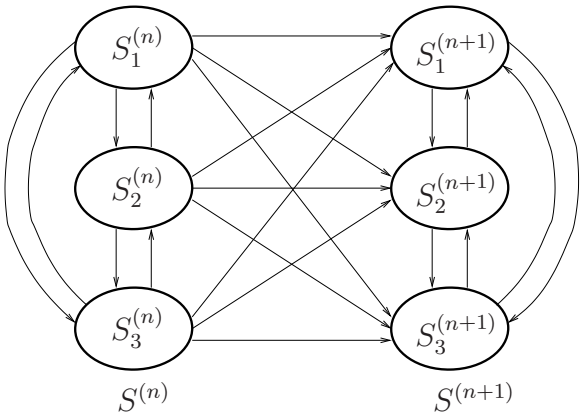


Figure 5.3: Single step of Markov chain for $K = 3$

we will consider FCP transmissions from AP to only one of the $N-1$ nodes and we argue that the same analysis holds for any of the other nodes in the AP cell. Note that we are *not* excluding the effects of other $N-2$ nodes on the transmission between AP and the chosen node, in terms of the model formulation.

Now, note that the PHY rate transition from one MAC frame sequence to another will be independent of time since H is time homogeneous. Also, by the *decoupling approximation* in [Bia00b, KAMG05b], our assumptions on the back-off process and time homogeneity of H, every successful or unsuccessful MAC frame transmission sequence occurs independently of all the preceding MAC frame transmission sequences. We can thus assume the FCP transmission process, between the AP and any given node, being represented by a Markov renewal reward process whose renewal instants J_{t_i} are embedded at the time instants t_i of completion of an i^{th} successful or unsuccessful MAC frame transmission sequence. The reward W_i for each i^{th} renewal cycle is denoted by whether the MAC frame was successfully transmitted or not. In order to formalize this notion, we construct the corresponding discrete time Markov chain as follows. First, we assume that all FCPs are of identical size. Though the 802.11 standard permits packet fragmentation into MAC frames, we assume that packet size is equal to a single MAC frame size and that the whole FCP is transmitted as a single MAC frame. Recall that a node must receive at least D FCPs in order to recover the original file of N_p packets. Now, at the beginning of each successful or unsuccessful MAC frame transmission sequence between the AP and a given node, let r_k, $k \in \mathcal{K}$, be the PHY rate that will be employed during this transmission sequence and let n denote the total number of FCPs or MAC frames that have already been received by the node. Thus $n = 0$ at the beginning of file transfer. Then, let $S_k^{(n)}$, $k \in \mathcal{K}$, denote the *state* of the transmission process between the AP and the wireless node at the beginning of any successful or unsuccessful transmission sequence. With some abuse of notation let $S^{(n)}$ denote the collection $\bigcup_{k\in\mathcal{K}}\{S_k^{(n)}\}$ of all states corresponding to a given n. Now, from one transmission sequence to the next, the state of the process evolves from $S_k^{(n)}$ to either $S_j^{(n)}$ (in case of unsuccessful transmission) or $S_l^{(n+1)}$ (in case of successful transmission) for some

$j, l \in \mathcal{K}$. Figure-5.3 shows a single step in such a Markov chain evolution for $K = 3$. For any general K the state transition matrix T, for the complete Markov chain until all D FCPs have been successfully received, can be formulated as,

$$T = \begin{array}{c} \\ 0 \\ 1 \\ \vdots \\ D-1 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & \cdots & & D \end{array} \\ \begin{bmatrix} T_f & T_s & & \mathbf{0} & \\ & T_f & T_s & & \\ & & \ddots & \ddots & \\ & \mathbf{0} & & T_f & T_s \end{bmatrix}_{D\times(D+1)} \end{array}$$

where,

$$T_f = \begin{bmatrix} t_{1,1}^{(f)} & \cdots & t_{1,K}^{(f)} \\ \vdots & \ddots & \vdots \\ t_{K,1}^{(f)} & \cdots & t_{K,K}^{(f)} \end{bmatrix}_{K\times K} \quad \text{and} \quad T_s = \begin{bmatrix} t_{1,1}^{(s)} & \cdots & t_{1,K}^{(s)} \\ \vdots & \ddots & \vdots \\ t_{K,1}^{(s)} & \cdots & t_{K,K}^{(s)} \end{bmatrix}_{K\times K}$$

where,

$$t_{k,j}^{(f)} = ((1-\gamma)p_f(k) + \gamma)\, h_{k,j} \quad \& \quad t_{k,l}^{(s)} = (1-\gamma)p_s(k)h_{k,l}.$$

In the preceding two formulae, $t_{k,j}^{(f)}$ denotes the probability that the state of the process evolves from $S_k^{(n)}$ to $S_j^{(n)}$ due to unsuccessful transmission, $t_{k,l}^{(s)}$ denotes the probability that the state evolves from $S_k^{(n)}$ to $S_l^{(n+1)}$ due to successful transmission, $p_s(k)$ denotes the probability that the channel conditions during a transmission sequence remain suitable for a successful transmission, $p_f(k)$ denotes the probability that the channel conditions during a transmission sequence lead to a failure of the MAC frame transmission and γ denotes the probability of collision between any two RTS frames from different wireless nodes or AP. We will soon observe at the end of this sub-section how the two probabilities $p_s(k)$ and $p_f(k)$ are a function of some $k \in \mathcal{K}$.

In order to see why a collision probability γ comes into picture, note the following. Even though we focus only on downlink and that AP is the only node transmitting FCPs to all other wireless nodes, by the description of our FCT protocol, each wireless node also transmits a 'STOP' message packet to the AP at successful reception of D FCPs. This uplink transmission of 'STOP' messages though occurring sporadically must be taken into consideration because the usual RTS/CTS channel reservation mechanism is launched before transmitting MAC frames comprising these 'STOP' message packets. Since different wireless nodes perceive different channel conditions, each one of them may receive the last D^{th} FCP and in turn transmit the 'STOP' message to AP at different time instants, while the AP may still be transmitting FCPs to other nodes. Then, inevitably there is a possibility for RTS frames from the AP and different wireless nodes to collide, as in a standard WLAN cell scenario with bi-directional traffic (see for e.g., [KAMG05b]).

Remark: Note that collisions due to the 'STOP' message can be negligible for large file sizes. Hence, modeling of 'STOP' message could be ignored for relatively large values of

N_p. However, it is well known that the vast majority of connections on Internet transfer files of size less than 10 KBs (or, 7 packets with MSS of 1460 bytes) (see [ea01] and references there in). Therefore, since vast majority of traffic comprises data downloads of less than 7 packets, modeling of 'STOP' message can not be ignored.

In order to derive γ, we again assume the decoupling approximation in [Bia00b] and follow the same approach as in [KAMG05b]. The decoupling approximation says that the collision probability is constant and independent, regardless of the number of retransmission attempts made by a node or the AP (see [Bia00b, KAMG05b] for details). With this approximation γ is given by, $\gamma = 1 - (1-\beta)^{N-1}$, where β is the long run average back-off rate (in *attempts per slot*). For our case it is given by, $\beta = 1/E[T_B]$. Note that γ here might be slightly over-estimated. However, we shall see in Section-5.5.2 that this leads to only in-significant deviation in the results.

Note that the collection $\bigcup_{k\in\mathcal{K}}\{S_k^{(D)}\}$ corresponding to $n = D$ forms a set of absorption states. When all D FCPs have been successfully transmitted the FCP transmission process terminates in one of the states $\{S_l^{(D)},\ l \in \mathcal{K}\}$. Thus for the purpose of computing the stationary probabilities, $\{\pi_l,\, l \in \mathcal{K}\}$, we may consider only the absorption states and ignore all other states which would lead to a computational complexity of only $O(K^2)$ instead of $O(K^2D)$.

For the purpose of modeling the wireless channel medium we consider a standard two state Markov channel error model. Such a model is depicted in Figure-5.4. Across two consecutive 802.11 slots, the channel may transit between a *good* and a *bad* state as shown in Figure-5.4, where p_g is the probability that the channel remains in good state, $1 - p_g$ is the probability of transiting from good to bad state, p_b is the probability of remaining in the bad state and $1 - p_b$ is the probability of transiting from bad to good state. In such a model $p_g + p_b \leq 1$ usually. However, as a standard simplification we will assume $p_g + p_b = 1$ (see [KS04b, KS04a] and references there in). Note that, if the channel ever goes into bad state for even a single slot during the transmission of *any* of the RTS, CTS, MAC or ACK frames, the corresponding frame is received erroneously and considered lost. Such a situation will lead to an unsuccessful MAC frame transmission sequence or a renewal cycle with zero reward. Now, let T_R, T_C and T_A denote the times (in slots) required to transmit an RTS, CTS and ACK frame, respectively. Also, let d be the size (in bits) of a single FCP or MAC frame, which are assumed to be of identical size as mentioned earlier. Then, with the channel model described in the foregoing discussion, we can now explicitly define the probabilities $p_s(k)$ and $p_f(k)$ for some $k \in \mathcal{K}$ as,

$$p_s(k) = p_g^{T_R+T_C+\left\lceil \frac{d}{r_k}\right\rceil+T_A}, p_f(k) = p_f^{(R)} + p_f^{(C)} + p_f^{(D)}(k) + p_f^{(A)}(k), \tag{5.1}$$

$$\text{where,} \qquad p_f^{(R)} = 1 - p_g^{T_R}, \quad p_f^{(C)} = p_g^{T_R}\left(1 - p_g^{T_C}\right),$$

$$p_f^{(D)}(k) = p_g^{T_R+T_C}\left(1 - p_g^{\left\lceil \frac{d}{r_k}\right\rceil}\right), p_f^{(A)}(k) = p_g^{T_R+T_C+\left\lceil \frac{d}{r_k}\right\rceil}\left(1 - p_g^{T_A}\right). \tag{5.2}$$

5.4 Mean Downlink Throughput

We now derive an explicit expression for the mean downlink throughput achieved by a single node. Recall the renewal reward process formulation of the MAC frame transmission sequences from the AP to a given node (Section-5.3.4). Let $L_i \stackrel{\Delta}{=} t_i - t_{i-1}$ denote the length (in slots) of i^{th} renewal cycle. Applying the renewal reward theorem (Theorem D.15 in [KMK04]) to our formulation, we can state that the mean downlink throughput, Θ, from AP to any wireless node is given by,

$$\Theta = \frac{\lim_{m\to\infty} \frac{1}{m}\sum_{i=1}^{m} W_i}{\lim_{m\to\infty} \frac{1}{m}\sum_{i=1}^{m} L_i} = \frac{E[W]}{E[L]}, \tag{5.3}$$

where, $E[W]$ is the expected reward over a renewal cycle in terms of number of bits of data transmitted and $E[L]$ is the expected length of a renewal cycle.

In order to compute the throughput we first compute $E[L]$ as follows. Using Equations 5.1 and 5.2, define the following entities,

$$p_s = E_{\mathcal{K}}[p_s(k)] = p_g^{T_R+T_C+T_A} E_{\mathcal{K}}[p_g^{\left\lceil \frac{d}{r_k} \right\rceil}], \tag{5.4}$$

$$p_f^{(D)} = E_{\mathcal{K}}[p_f^{(D)}(k)] = p_g^{T_R+T_C} \left(1 - E_{\mathcal{K}}[p_g^{\left\lceil \frac{d}{r_k} \right\rceil}]\right), \tag{5.5}$$

$$p_f^{(A)} = E_{\mathcal{K}}[p_f^{(A)}(k)] = p_g^{T_R+T_C} E_{\mathcal{K}}[p_g^{\left\lceil \frac{d}{r_k} \right\rceil}] \left(1 - p_g^{T_A}\right), \tag{5.6}$$

where, $E_{\mathcal{K}}[\cdot]$ is the expectation operator over the index set $\mathcal{K}$. This expectation may be computed using the stationary probabilities $\{\pi_l,\, l \in \mathcal{K}\}$ mentioned earlier in Section 5.3.4. Now, from Figure-5.1 we can see that $E[L]$ can be derived as,

$$\begin{aligned} E[L] &= \mathbb{E}(\text{length of renewal cycle}) \\ &= (1-\gamma)\left[p_s\left(E_{\mathcal{K}}\left[\left\lceil \frac{d}{r_k} \right\rceil\right] + T_{ov}\right) + p_f^{(R)} T_{ov}^{(R)} + p_f^{(C)} T_{ov}^{(C)}\right] \\ &+ (1-\gamma)\left[p_f^{(D)}\left(E_{\mathcal{K}}\left[\left\lceil \frac{d}{r_k} \right\rceil\right] + T_{ov}^{(D)}\right) + p_f^{(A)}\left(E_{\mathcal{K}}\left[\left\lceil \frac{d}{r_k} \right\rceil\right] + T_{ov}^{(A)}\right)\right] + \gamma T_{ov}^{(R)}, \end{aligned} \tag{5.7}$$

where,

$$T_{ov}^{(R)} = T_{DIFS} + E[T_B] + T_R + T_{SIFS} = \hat{T}_{ov}^{(R)} + E[T_B], \tag{5.8}$$

$$T_{ov}^{(C)} = T_{DIFS} + E[T_B] + T_R + T_C + T_{SIFS} = \hat{T}_{ov}^{(C)} + E[T_B], \tag{5.9}$$

$$T_{ov}^{(D)} = T_{DIFS} + E[T_B] + T_R + T_C + 3T_{SIFS} = \hat{T}_{ov}^{(D)} + E[T_B], \tag{5.10}$$

$$T_{ov} = T_{ov}^{(A)} = T_{DIFS} + E[T_B] + T_R + T_C + T_A + 3T_{SIFS} = \hat{T}_{ov}^{(A)} + E[T_B], \tag{5.11}$$

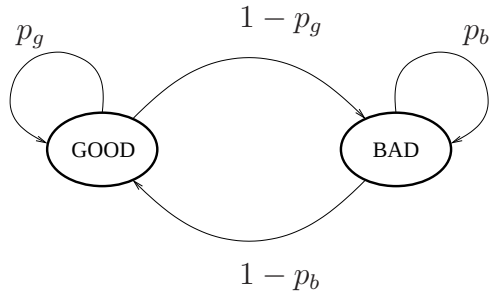


Figure 5.4: Two state Markov channel error model

In the preceding formulae, T_{DIFS} and T_{SIFS} denote the DIFS and SIFS times (in slots), respectively. $E[T_B]$ denotes the expected length of back-off duration as defined earlier in Section-5.3.1. $T_{ov}^{(\cdot)}$ denotes the contribution from MAC layer entities to the overhead caused by an *unsuccessful* transmission of the corresponding RTS, CTS, MAC data or ACK frame. T_{ov} denotes the same for a *successful* transmission of the MAC data frame. An extra term of T_{SIFS} has been included in Equations 5.8 and 5.10 since in order to detect the preceding frame's loss, the AP must wait (for SIFS duration) till the time instant when the wireless node starts to send the next frame in the sequence.

Next, for computing $E[W]$ observe that an FCP of size d bits (or an original un-coded packet of size $\frac{d}{1+\epsilon}$ bits) is received by a wireless node only when the corresponding MAC frame is successfully transmitted over a renewal cycle. From this observation the expected reward over a renewal cycle is given by,

$$E[W] = \frac{(1-\gamma)\, p_s\, d}{1+\epsilon}. \tag{5.12}$$

Plugging Equations 5.7 and 5.12 in Equation 5.3 one can now compute the mean downlink throughput.

5.5 Validation of the Model

The goal of this section is to validate our model and expression for mean downlink throughput with *ns2* simulations.

5.5.1 Implementation Details

MATLAB software was used for all numerical computations and values used for various system parameters are as follows: 802.11 slot = $20\mu s$, $T_R = 4.5$ slots, $T_C = 2.8$ slots, $T_A = 2.8$ slots, $T_{DIFS} = 2.5$ slots, $T_{SIFS} = 0.5$ slots, $CW = 64$ slots, $E[T_B] = 31.5$ slots, $d = 1024$ bytes, $K = 3$, $r_1 = 6$ Mbits/s, $r_2 = 12$ Mbits/s, $r_3 = 24$ Mbits/s. For *ns2* simulations we broadly considered two different cases. One of the cases corresponds to the real back-off process as per 802.11 specifications, where the value of contention window CW grows exponentially (when loss occurs) and the back-off interval is selected uniformly

randomly. This is the case implemented by default in `MAC/802_11` class module in *ns2*. The other case is based on our assumptions on the back-off process detailed previously in Section-5.3.1. In order to correctly simulate the scenarios corresponding to our model based on these assumptions, we modified the `MAC/802_11` class module in *ns2* and fixed the value of variables corresponding to CW such that it remains constant at 64 slots (i.e., $\beta = 0.03$). Additionally, we modified the code so that the random back-off interval is sampled from a geometric distribution with mean $E[T_B] = \frac{CW-1}{2}$ instead of uniform distribution. Specifically, we used *ns2* version 2.29 for our simulations and modified the following files to incorporate changes described in the foregoing discussion: `mac/mac-802_11.{h,cc}`, `gen/ns_tcl.cc`, `tools/rng.h` and `tools/random.h`. The WLAN topology considered for simulations was identical to the one proposed for modeling analysis. Mobility of wireless nodes was not introduced in the simulation scenarios. The configuration of AP node was modified to implement the Adaptive Auto Rate Fallback (AARF) rate control algorithm and its parameters used in all our simulation scenarios are identical to those specified in Table-2 in [LMT04]. It can be easily shown that these parameters for $K = 3$ give rise to the following PHY rate transition matrix that we used in our simulations,

$$H = \begin{bmatrix} 0.63\,(1-p_g) & 0.37\,p_g & 0 \\ 0.63\,(1-p_g) & 0.0925\,p_g/2 & 0.2775\,p_g/2 \\ 0 & 0.63\,(1-p_g) & 0.37\,p_g \end{bmatrix}.$$

The FCT protocol defined previously in Section-5.2 was implemented by modifying the `Agent/UDP` class module in `apps/udp.h` and `apps/udp.cc` files in *ns2*. To be specific, it was modified to support a new `advanceby()` public function (similar to the one present in `Agent/TCP` class module) and a new `sendSTOP()` private function. The `advanceby()` function can be called in a TCL script to let a `MAC/802_11` node (or AP) to send dummy fountain coded packets at a pre-specified constant bit rate. The `sendSTOP()` function is called in the `recv()` function to automatically send a 'STOP' message to the sender as soon as a pre-specified number of FCPs have been successfully received at receiver side. In turn, on receipt of this 'STOP' message the sender terminates its current file session.

The wireless medium in all simulation scenarios was modeled by a two-ray ground radio propagation model and all nodes were configured to use a single antenna. Similar to the analytical model, a two state Markov channel error model was used on a per slot basis for the simulation scenarios. Different values for the probability p_g of good channel state for different scenarios shall be explicitly specified later. Each simulation result was obtained by executing the simulation scenario for about 100 runs in order to gather sufficient statistics for calculation of mean values.

5.5.2 Simulating with Modified Back-off Process

We compare here the results of our analysis with those of simulations based on the modified back-off process as per our assumptions in Section-5.3.1.

Figure-5.5 shows mean downlink throughputs from AP to a single node for a file of

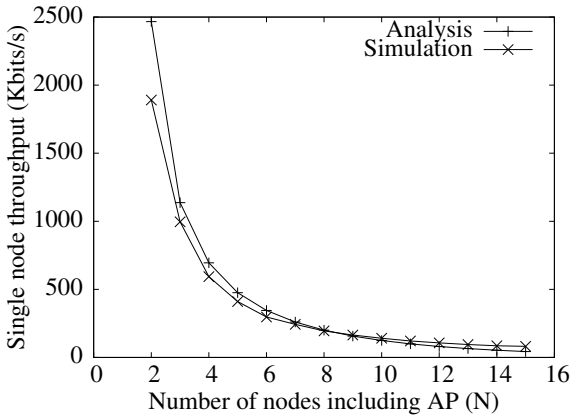


Figure 5.5: *Modified* back-off ($D = 5$)

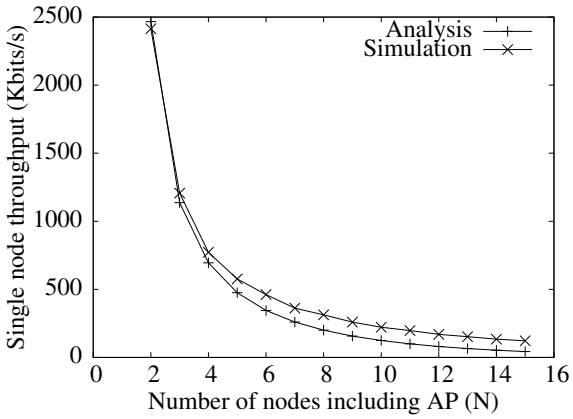


Figure 5.6: *Real* back-off ($D = 5$)

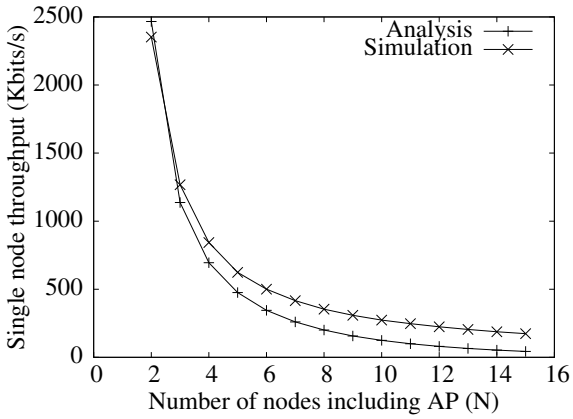


Figure 5.7: *Modified* back-off ($D = 100$)

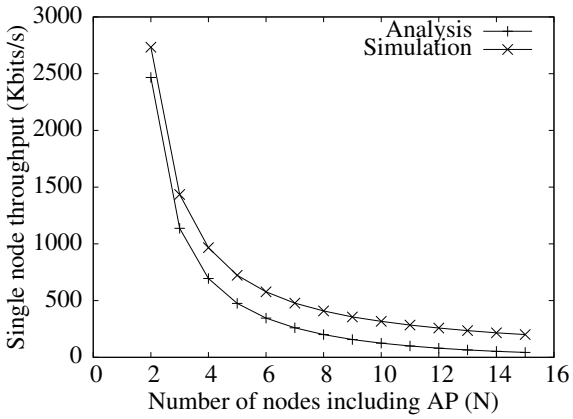


Figure 5.8: *Real* back-off ($D = 100$)

size 3850 bytes ($N_p \approx 3.85$ packets), a fountain code with design parameter $\epsilon = 0.3$, $K = 3$, $\beta = 0.03$, $d = 1500$ bytes and a wireless channel for which $p_g = 0.99$. With a fountain code of design parameter $\epsilon = 0.3$ a file of size ≈ 3850 bytes can be encoded such that the wireless node must receive at least approximately $D = 5$ FCPs in order to decode the original file. Also note that $p_g = 0.99$ does not necessarily represent highly favorable channel conditions since state transitions in the Markov channel error model occur on a per slot basis. In other words, the channel must continue to remain in good state for a few tens of slots successively, for the MAC transmission to be successful.

In Figure-5.5 we can observe a very good match between the simulation and analysis results. Figure-5.7 shows mean downlink throughputs for another scenario of $D = 100$ FCPs, $\epsilon = 0.3$, $K = 3$, $\beta = 0.03$, $d = 1500$ bytes and $p_g = 0.99$. We again observe a good match of the results from simulations with those from analysis. For higher values of N the throughput results tend to deviate marginally from each other. This deviation may be explained as follows. In our analysis the collision probability, γ, is slightly over-estimated. It is calculated assuming that the RTS frame from AP (for FCP transmission) may collide with the RTS frame from any of the wireless nodes (for 'STOP' message transmission)

during *any* MAC frame transmission sequence. However, in simulations, according to the FCT protocol's design, a 'STOP' message is transmitted by a wireless node only after the *last* D^{th} FCP from AP has been successfully received. Thus, in reality (simulations), since the AP node is the only node trying to gain access to the channel during most of the file transmission duration, fewer collisions of RTS frames occur. This clearly explains the marginally higher throughput obtained from simulations as compared to the analysis.

5.5.3 Simulating with Real Back-off Process

Now, we compare the results of our analysis with those of simulations based on the real back-off process as per 802.11 specifications. The goal of this comparison is to demonstrate that though our assumptions of Section-5.3.1 are essential for modeling purposes, they are actually insignificant in terms of the throughput performance metric that we have been investigating.

In Figure 5.6, mean downlink throughputs for a scenario identical to that of Figure 5.5 (i.e., $D = 5$ FCPs, $\epsilon = 0.3$, $K = 3$, $\beta = 0.03$, $d = 1500$ bytes and $p_g = 0.99$) are plotted. The only difference being that the simulation results here are based on the real back-off process. It is easy to observe that throughput metric in the two different figures exhibits *nearly* the same performance variation characteristics between the analysis and simulation results. Figure 5.8 confirms this observation for another scenario identical to that of Figure 5.7 (i.e., $D = 100$ FCPs, $\epsilon = 0.3$, $K = 3$, $\beta = 0.03$, $d = 1500$ bytes and $p_g = 0.99$).

Before concluding this section we would like to mention that apart from the scenarios discussed here, we have carried out numerical analysis and simulations for additional scenarios, such as for different values of D, ϵ, K, β, d and p_g. In all these additional scenarios, similar observations were made as those discussed here. It was also observed in all simulation results that the AP achieved nearly equal downlink throughputs for each node in the WLAN cell. Some of these additional scenarios are presented in the following section. As a conclusion to this section, we may state that the simulation results presented here demonstrate that our modeling analysis captures fairly accurately the performance of the FCT protocol in a WLAN last-hop link. Moreover, our assumptions on the back-off process do not significantly affect the analytical results that we have investigated here.

5.6 Performance Analysis

Having validated the model, we now proceed towards analyzing performance of the obtained throughput metric w.r.t. the various system parameters. Unless otherwise stated, numerical values for the parameters used in this section are identical to those specified in Section 5.5. Plugging Equations 5.7 & 5.12 in 5.3 and re-arranging we get,

$$\Theta(\beta) = \frac{p_s\, d}{(1+\epsilon)\left[(p_s + p_f^{(A)})(\bar{T}_D + \hat{T}_{ov}^{(A)}) + p_f^{(R)}\hat{T}_{ov}^{(R)} + p_f^{(C)}\hat{T}_{ov}^{(C)} + p_f^{(D)}(\bar{T}_D + \hat{T}_{ov}^{(D)}) + f(\beta)\right]}, \tag{5.13}$$

where,

$$f(\beta) = (p_s + p_f)\frac{1}{\beta} + \frac{1-(1-\beta)^{N-1}}{(1-\beta)^{N-1}}(\hat{T}_{ov}^{(R)} + \frac{1}{\beta}),\ \bar{T}_D = E_{\mathcal{K}}\left[\left\lceil \frac{d}{r_k} \right\rceil\right],$$

where, $p_f = E_{\mathcal{K}}[p_f(k)]$. For $\Theta(\beta)$ to be maximum, $f(\beta)$ should be minimum. Therefore, taking the derivative of $f(\beta)$ and equating it to 0 we get the following simple quadratic equation in β after simplification,

$$(N-1)\,\hat{T}_{ov}^{(R)}\ \beta^2 + N\ \beta - 1 = 0, \tag{5.14}$$

where we have used the fact that $p_s + p_f = E_{\mathcal{K}}[p_s(k)] + E_{\mathcal{K}}[p_f(k)] = 1$ (since $p_s(k) + p_f(k) = 1$) along with the following approximation,

$$(1-\beta)^N \approx 1 - N\beta + \frac{N(N-1)}{2}\beta^2,$$

since $\beta \ll 1$ for $N > 2$. Solving Equation 5.14 we get the optimal β^* as,

$$\beta^* \approx \frac{\sqrt{N^2 + 4(N-1)\hat{T}_{ov}^{(R)}} - N}{2(N-1)\hat{T}_{ov}^{(R)}}. \tag{5.15}$$

Equation 5.15 gives the approximate optimal value of back-off rate β (and hence CW) that should be employed by the AP to obtain maximum throughput performance for a given network setting (i.e., N and $\hat{T}_{ov}^{(R)}$). This clearly highlights the considerable theoretical importance of Equation 5.15 and it is to be additionally noted that it is independent of the probabilities $p_s(k)$ and $p_f(k)$.

In Figure 5.9, single node throughput, $\Theta(\beta)$, is shown for different values of N as computed from Equation 5.13. Note that the maximum throughput is highly dependent on N. The curves vary steeply for values of β less than β^* and are relatively smooth otherwise, indicating that throughput performance may be penalized if the value of β employed by AP is even negligibly less than β^*. Figure 5.9 strengthens the theoretical importance of Equation 5.15 since even a slight deviation from optimal β^* (and hence CW) may cause significant performance degradation. Table 5.1 shows a good match between the approximate and accurate values of β^* obtained from Equation 5.15 and Figure 5.9, respectively.

Figure 5.10 shows a plot of throughput as a function of the fountain code parameter, ϵ, for different N and the corresponding optimal β^*. For a given fountain code, lower is the value of ϵ, higher is its efficiency. But higher efficiency comes at the cost of a more complex coding-decoding algorithm. In Figure 5.10 it is seen that employing a fountain code with higher efficiency may result in better throughput performance *only* for low values of N. This observation can be exploited to *dynamically* select an appropriate fountain code based on an estimate of N by the AP.

Figure 5.11 shows similar characteristics as those observed in Figure 5.10, but for dependance on channel conditions (i.e., p_g). Selecting a fountain code with higher efficiency

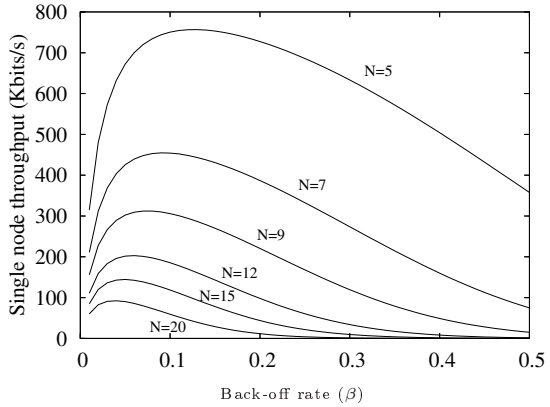


Figure 5.9: $D = 20$, $K = 3$, $\epsilon = 0.3$, $d = 1024$ bytes and $p_g = 0.99$

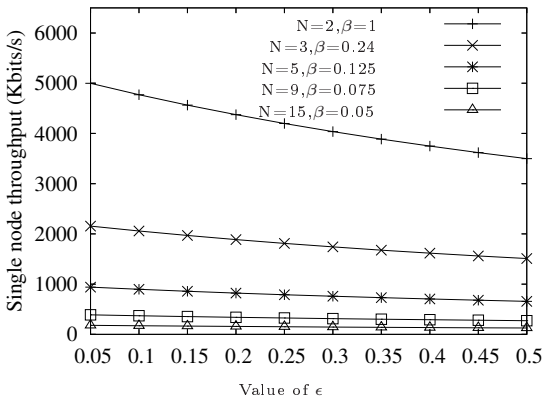


Figure 5.10: $D = 20$, $K = 3$, $d = 1024$ bytes and $p_g = 0.99$

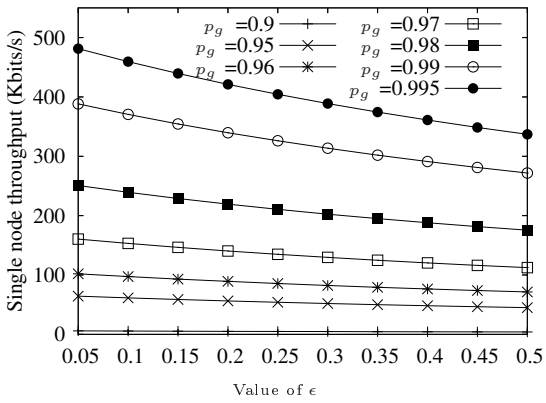


Figure 5.11: $N = 9$, $\beta = 0.075$, $D = 20$, $K = 3$ and $d = 1024$ bytes

N	5	7	9	12	15	20
Approximate	0.1174	0.0903	0.0743	0.0592	0.0495	0.0391
Accurate	0.1250	0.0901	0.0748	0.0601	0.0507	0.0402

Table 5.1: Comparison of Approximate and Accurate values of β^*

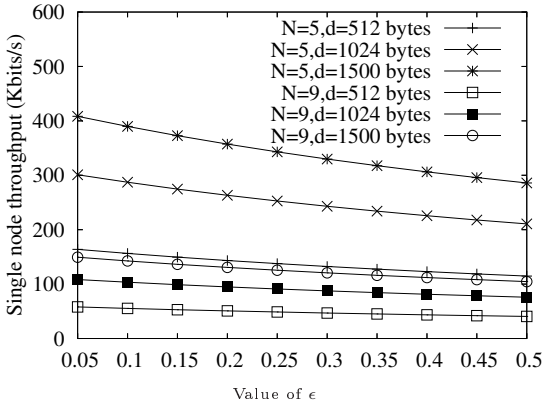


Figure 5.12: $D = 20$, $K = 3$, $\beta(5) = 0.125$, $\beta(9) = 0.075$ and $p_g = 0.99$

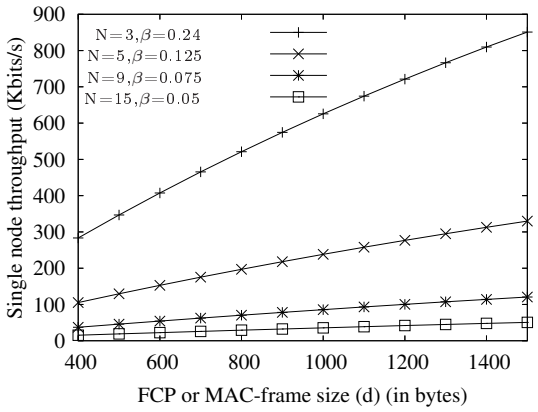


Figure 5.13: $D = 20$, $K = 3$, $\epsilon = 0.3$ and $p_g = 0.99$

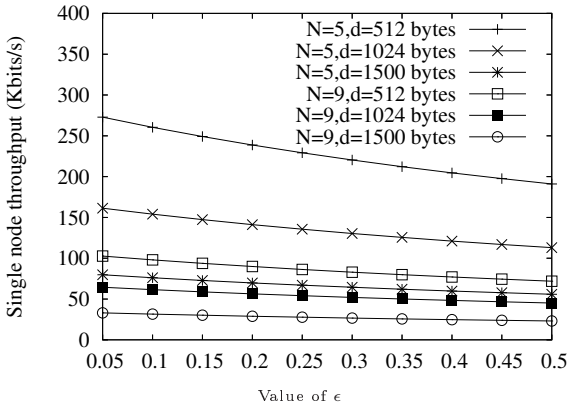


Figure 5.14: $D = 20$, $K = 3$, $\beta(5) = 0.125$, $\beta(9) = 0.075$and $p_g = 0.95$

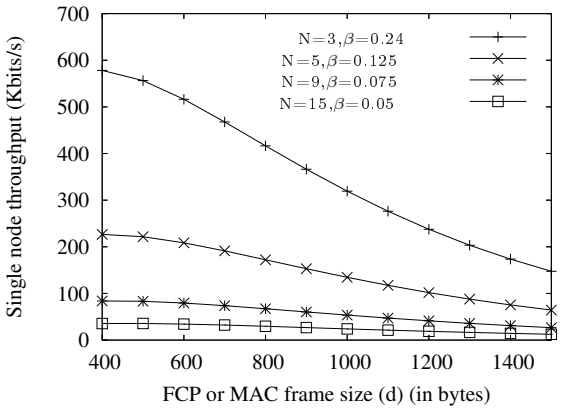


Figure 5.15: $D = 20$, $K = 3$, $\epsilon = 0.3$ and $p_g = 0.95$

can be beneficial in terms of throughput *only* when good channel conditions prevail (high values of p_g). If channel conditions are bad, throughput performance is insensitive to the choice of the fountain code (i.e., ϵ). Again, this observation can be exploited to dynamically select an appropriate fountain code based on an estimate of the channel conditions by AP.

At last, let us look at the dependance of throughput on the FCP size (or MAC-frame size; see Section 5.3.4). As an example for $N = 5$ and 9, Figure 5.12 illustrates that for any given fountain code and number of nodes N, increasing the FCP size, d, results in an increase in the throughput. Figure 5.13 confirms this illustration for a particular value of $\epsilon = 0.3$ and different values of $N = 3, 5, 9, 15$. However, this observation is valid only for fairly good channel conditions with $p_g = 0.99$. In Figures 5.14 and 5.15 we observe that for slightly bad channel conditions with $p_g = 0.95$ the trend reverses as compared to Figures 5.12 and 5.13. For any given fountain code and number of nodes N, increasing the FCP size, d, now results in a decrease in the throughput. We may thus finally conclude that during good channel conditions FCPs of largest possible size must be transmitted and during bad

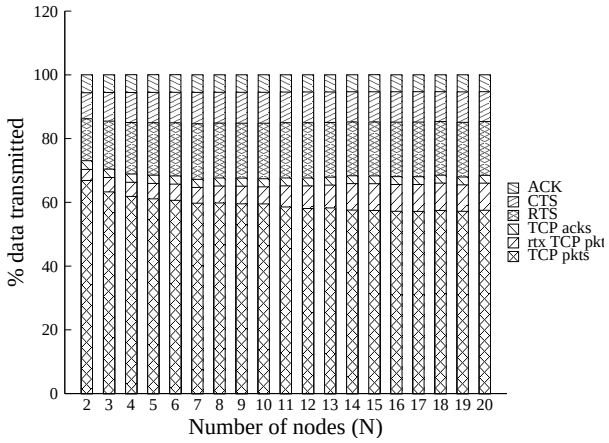


Figure 5.16: Percentage (%) of data with TCP, $p_g = 0.99$

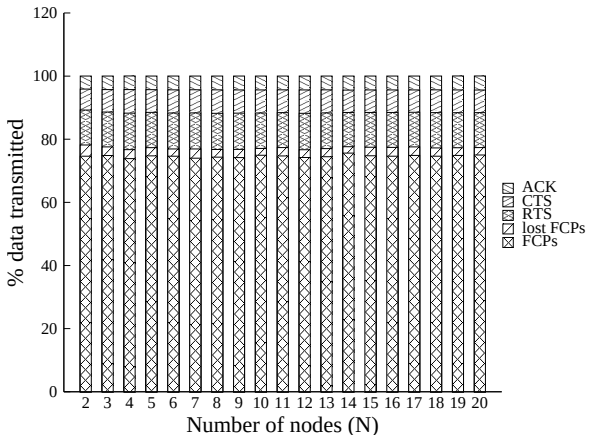


Figure 5.17: Percentage (%) of data with FCT, $\epsilon = 0.3$, $p_g = 0.99$

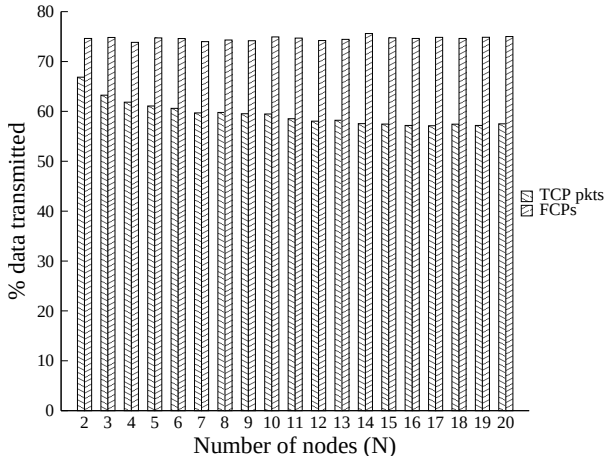


Figure 5.18: Comparison of TCP packets and FCPs

channel conditions it is preferable to transmit FCPs of smallest possible size.

5.7 Comparison of FCT with TCP

As per our discussion in Section 5.1, the FCT protocol may be a better alternative to TCP in an 802.11 WLAN cell since it abandons the need for TCP ack like traffic on reverse link. In the forthcoming discussion we explicitly quantify this performance improvement in terms of percentage of useful data transmitted. Keeping the real back-off and using settings similar to the ones used in Section 5.5, we performed *ns2* simulations to measure the percentage contribution by un-coded N_p *useful* data packets, successfully transmitted in a WLAN cell, among all kinds of packets and MAC frames. This percentage measure has been carried out for downlink transfers using TCP and FCT (with $\epsilon = 0.3$), separately. Figures 5.16 and 5.17 show simulation results for a long file transfer that lasts up to 3 hours over a fairly good wireless channel with $p_g = 0.99$. In the figures, % of all kinds of packets and

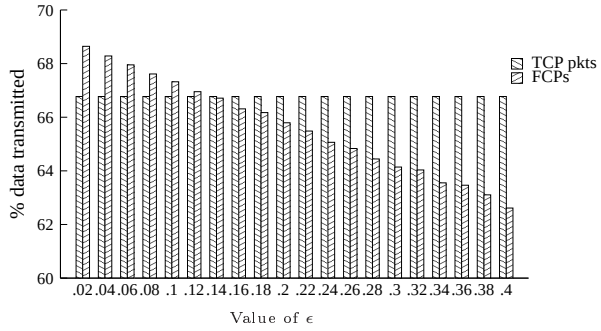


Figure 5.19: % of data, $N = 2$, $p_g = 0.9$

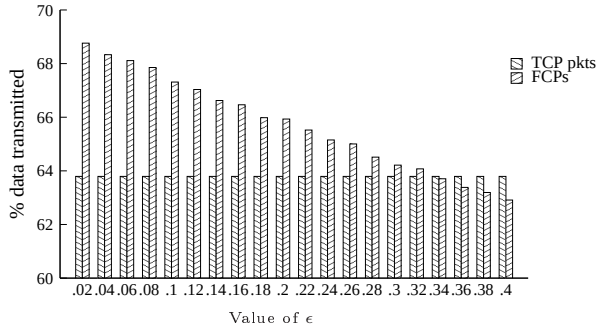


Figure 5.20: % of data, $N = 3$, $p_g = 0.9$

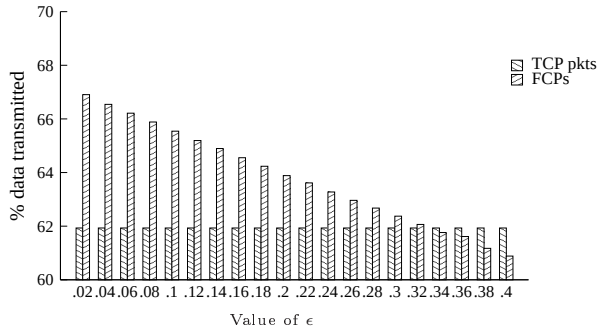


Figure 5.21: % of data, $N = 4$, $p_g = 0.9$

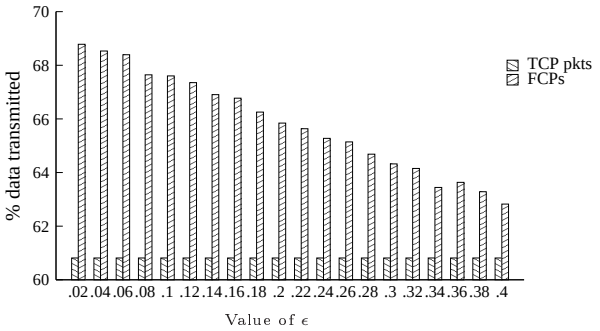


Figure 5.22: % of data, $N = 5$, $p_g = 0.9$

MAC frames is plotted against the number of nodes N (including AP). Figure 5.18 shows a comparison between the percentages of useful data packets for TCP and FCT. It can be clearly observed that FCT outperforms TCP. Moreover, the performance improvement increases, though marginally, with increasing number of nodes. We may thus infer that for a suitable design parameter (ϵ) of a fountain code and fairly good channel characteristics ($p_g = 0.99$), there is greater incentive to switch to FCT when number of nodes in a WLAN cell are relatively high. However, this may not be the case when channel conditions are bad and we discuss this issue next.

Figures 5.19, 5.20, 5.21 and 5.22 show for $N = 2$, 3, 4 and 5, respectively, the percentage of file data successfully transmitted using TCP and different FCTs represented by varying values of ϵ. The wireless channel characteristics for these results pertain to poor channel conditions, i.e., $p_g = 0.9$. Consider for e.g., an FCT with $\epsilon = 0.2$. We observe in the figures that for $N = 2$ TCP performs better than the FCT, where as for values of N greater than 2 TCP performs worse than the FCT. Compare this FCT with another FCT having $\epsilon = 0.4$. We now observe that TCP performs better than this new FCT ($\epsilon = 0.4$) for $N = 2$,3 and 4 and vice-versa holds true for $N = 5$. For yet another FCT with $\epsilon = 0.1$, we observe that it performs better than TCP for all values of $N = 2$, 3, 4 and 5.

The foregoing discussion helps us to deduce that an FCT may perform better or worse than TCP depending on its design parameter ϵ, the number of nodes N and the wireless channel characteristics. This trade-off between the performance of an FCT (compared to TCP) and the extra ϵN_p redundant packets, can be intuitively understood to be related

to the underlying wireless channel characteristics and the collision probability γ which is a function of N. This observation may help in designing a dynamic *protocol selection* algorithm that can optimally and/or adaptively select one of the TCP and FCT protocols for downlink transmission in a WLAN.

5.8 Conclusion

We have proposed a Markovian stochastic model to analyze the performance of a simple Fountain Codes based Transport (FCT) in an 802.11 WLAN setting where the AP is allowed to employ a generic rate control algorithm. The results of this study clearly demonstrate that the AP must use an optimal value of the backoff-rate, β, for achieving maximum downlink throughput performance. Moreover, we have seen that choosing a fountain code with higher efficiency (lower value of ϵ) may improve the throughput performance only when number of wireless nodes (N) are low or when good channel conditions prevail. Next, for a given fountain code and estimate of N, during good channel conditions the size of FCP must be chosen as large as possible and during bad channel conditions it must be chosen as small as possible. Finally, through a brief simulation study we have illustrated that FCT may perform better or worse than TCP depending on the fountain code parameter, ϵ, the number of nodes in a WLAN cell and the wireless channel conditions.

Chapter 6

New Cross-Layer Channel Switching Policy for TCP Transmission in 3G UMTS

In 3G UMTS, two main transport channels have been provided at the layer-2 (MAC) for downlink data transmission: a common FACH channel and a dedicated DCH channel. The performance of TCP in UMTS depends much on the channel switching policy used. In this chapter, we first propose and analyze three new basic threshold-based channel switching policies for UMTS that we name as QS (Queue Size), FS (Flow Size) and QSFS (QS & FS combined) policy. These policies significantly improve over a 'modified threshold policy' in [PAAD03] by about 17% in response time metrics. We further propose and evaluate a new improved switching policy that we call FS-DCH (*at-least* flow-size threshold on DCH) policy. This policy is biased towards short TCP flows[1] of few packets. It is thus a *cross-layer* policy that improves the performance of TCP by giving priority to the initial few packets of a flow on the fast DCH channel. Extensive simulation results show that FS-DCH policy improves over others by about 30% to 36% in response time metrics for a particular case.

Note: The material in this chapter has appeared in [KBAK08]. This work was sponsored by France Telecom R&D.

[1] A *flow* is defined as a burst of packets in a TCP connection.

6.1 Introduction

Keeping in pace with the increasing demand from users for access to information and services on public and private networks, the third generation (3G) Universal Mobile Telecommunication System (UMTS) has been designed to offer services such as high speed Internet access, high quality image and video exchange and global roaming. Data traffic in UMTS has been classified broadly into four different classes, namely–conversational, streaming, interactive (e.g., web browsing) and background (e.g., email) classes. The bulk of data in streaming and interactive transmissions is carried over the *downlink* from UTRAN (UMTS Terrestrial Radio Access Network) to a UE (User Equipment). Data generated in the higher layers of UTRAN is carried over the air interface to the UEs via the downlink transport channels, which are mapped in the physical layer to different physical channels. There are two types of layer-2 downlink transport channels that have been provided in UMTS: *dedicated* channels and *common* channels. A common channel is a resource shared between all or a group of users in a cell, where as a dedicated channel is a resource identified by a certain code on a certain frequency and is reserved for a single user only. The only dedicated channel is termed as DCH and one of the six common transport channels that is mainly used for packet data on the downlink is the FACH channel [HT01]. The number of DCH channels in a UMTS cell is *interference limited*. If a new user's connection cannot be admitted into the cell (this is decided by an appropriate interference based CAC or connection admission control), it must wait until a DCH channel is released by the already connected users or until when interference conditions become suitable for this new user to be allocated a new DCH channel. Being a dedicated channel, DCH guarantees higher data rates but the set-up time for DCH is significant (of the order of 250ms [PAAD03, HT01]). On the other hand, the common channel FACH inherently guarantees lower data rates but its set-up time is less. According to the WCDMA (Wideband-CDMA) specifications detailed by the 3GPP group, for a particular user, long flows with large amount of packets can be transmitted on the user dedicated DCH channel and short flows of few packets can be transmitted on the common FACH channel which is shared by all users. However, the 3GPP specifications do not provide any standardization of such a channel selection/switching policy. A network operator is free to choose its own proprietary channel switching policy.

6.1.1 Main Contributions

In this chapter, we propose some new *basic* channel switching policies for packet data transmission on the downlink of a single UMTS cell. In Section 6.5 we observe that our new switching policies improve on the 'modified threshold policy' in [PAAD03] by around 17% in response time metrics. Thereafter, based on some observations about the DCH and FACH channel characteristics and the need for distinction of long and short TCP flows, we further propose another new *cross-layer* channel switching policy, which is our main contribution in this work. To the best of our knowledge, ours is the first attempt to propose such a cross-layer channel switching policy for UMTS downlink that is based on diffrentiation between long and short TCP flows. All the new policies are in accordance with the current WCDMA

specifications and we evaluate their performance in terms of response time and slowdown metrics using simulations.

6.1.2 Synopsis

We start in Section 6.2 by defining the basic threshold-based channel switching policies. We name them as QS (Queue Size) policy, FS (Flow Size) policy and QSFS (QS & FS combined) policy. In all these policies a new flow or connection starts on the common FACH channel. In Section 6.3, we observe that in the basic policies the switching delay for connections switching from FACH to DCH and vice-versa is not very significant as compared to the transmission time of packets on the FACH channel, given the fact that FACH is a low bandwidth channel with high priority signaling traffic on it. We also argue that it is advantageous for short flows to have small response times. This observation and argument motivates us to propose the design of a new cross-layer policy that we call FS-DCH (*at-least* flow-size threshold on DCH) policy in which we try to achieve better response time and slowdown for short flows. In Section 6.4 we describe the network model and simulation set-up that we have used for performance evaluation of all the policies. Section 6.5 leads to discussion on the various observations that can be made from simulation graphs obtained. We finally conclude in Section 6.6.
Remark: Note that we could have directly described only the *cross-layer* FS-DCH policy. However, we first explain the *basic* QS, FS and QSFS policies in order to be able to lay down a basis for motivating and progressively designing the cross-layer FS-DCH policy. Without first describing the basic policies it would have been difficult to motivate the design of the cross-layer policy.

6.1.3 Related Work

Most of the existing channel switching policies are very simple, timer and threshold based policies and do not involve any complex or cross-layer switching criteria. Queue size threshold based policies have been proposed in [PAAD03] in which a new connection is initially allocated to FACH. On indication that the current flow of the connection might be long (i.e., a long buffer queue for that source is observed), then beyond some upper threshold, the Packet Scheduler in UMTS tries to allocate a DCH to that connection (if one is available). While on DCH, when the queue size of the connection falls below another lower threshold, the connection is switched back to FACH. The authors in [PAAD03] also present a modified threshold policy, in which, while a connection is on DCH, if its queue size falls below a lower threshold, a timer is started and the connection remains on DCH. If there are no arrivals during the timer period, the connection is switched back to FACH. The timer is used to let the TCP acknowledgements (ACKs) reach the sender and release new packets. In [AS03], the switching policy switches connections from FACH to DCH when the number of packets transmitted (i.e., flow size) for a given user on FACH exceeds a threshold. The choice of the threshold depends on the load on FACH and other QoS conditions. In [RFQ03], a switching policy based on bandwidth demand has been proposed. A connection

is switched from FACH to DCH if its bandwidth demand exceeds a threshold and remains on FACH otherwise. The channel switching schemes in [OKIO02] work with blocking and unblocking packets present in the RLC (Radio Link Control) and MAC sub-layers and different schemes propose to transmit the unblocked packets on either common or dedicated channels, differently.

6.2 Basic Channel Switching Policies

We first propose three new basic threshold-based channel switching policies. In all these policies, the FACH channel is served with either a PS or a LAS[2] scheduling mechanism and the DCH channel is implemented as *Priority scheduling* with priority given to connections having maximum queue lengths. Before we discuss in detail about the three channel switching policies, we define below the notations used in their formal definitions:

- Let $Q(i)$ denote the queue length of a connection i at the UMTS base station (NodeB).
- Let T_h and T_l ($T_h \geq T_l$) denote two thresholds on the queue length $Q(i)$ when the connection is on FACH and DCH channels, respectively.
- Let $f(i)$ denote the cumulative flow size (i.e., number of packets transmitted) over the FACH and DCH channels, for the current flow of a connection i.
- Let 's' denote a threshold on the cumulative flow size $f(i)$ of the current flow.
- In all the policies described in this chapter, a connection starts on FACH by default and then if a DCH is available, it is switched to DCH depending on different thresholds. If a DCH is not available then a switching request r_i corresponding to this connection i is added to a request set so that later when a DCH is available, connection i will be switched to DCH. Let R denote this request set.
- Let $W(i)$ denote the total time for which a request r_i of connection i remains unserved. Alternatively, it denotes the total time for which a connection i has been waiting to be switched to DCH since its request r_i to switch to DCH was added to R.
- Let N_{dch} denote the total number of DCH channels in a single UMTS cell.
- Let U_{dch} denote the total number of DCH channels that have been allocated or currently in use in the UMTS cell. Note that $U_{dch} \leq N_{dch}$.

6.2.1 QS Policy

In the QS (Queue Size) policy with parameter T_h, a new connection i starts on the FACH channel and waits for its queue length to exceed an upper threshold T_h before switching to DCH. If there is no DCH channel available then a request r_i for this connection to switch

[2]LAS: Scheduling based on *Least Attained Service*

to DCH is made. For a connection j on DCH when its queue length drops below the lower threshold T_l, a timer is started for T_{out} seconds. If there are packet arrivals during the timer period, the timer is reset. When the timer expires, if the queue length of connection j is still below the lower threshold and another set of connections on FACH are attempting to switch to DCH and no more free DCH channels are available, the connection j switches back to FACH. Once this connection switches to FACH after a switch delay (of around 250ms [PAAD03, HT01]), a connection having the *maximum* queue length among the set of connections on FACH that were attempting to switch to DCH, is switched to DCH. In this way we give priority to the connections with the maximum queue lengths while switching from FACH to DCH. This is what we mean by Priority scheduling on the DCH channel. PS+Priority then implies that, FACH uses PS scheduling mechanism and DCH uses Priority scheduling. This PS+Priority queue system is the essential difference between our new basic QS, FS and QSFS policies and the policies proposed in [PAAD03] which use PS+FCFS queueing. We will see later in Section 6.5 that our new policies significantly improve over the 'modified threshold policy' in [PAAD03] by around 17% in response time metrics. This leads to the conclusion that PS+Priority queueing system is the main feature due to which our new policies improve over the modified threshold policy in [PAAD03]. The QS policy can be formally summarized and defined as follows:

QS policy: The QS (Queue Size) policy is characterized by the following set of rules:

- A connection i starts on FACH by default. It switches to DCH if $Q(i) > T_h$ & $U_{dch} < N_{dch}$.

  If $Q(i) > T_h$ & $U_{dch} = N_{dch}$ then r_i is added to R.

- If connection j is on DCH then if $Q(j) < T_l$, a timer is started for duration T_{out} seconds. If there are packet arrivals during the timer period, the timer is reset. When the timer expires and still $Q(j) < T_l$, then, if (1) $U_{dch} = N_{dch}$ & $R \neq \phi$ then connection j switches to FACH and connection i with $r_i \in R$ switches to DCH, where connection i is chosen such that,

  $$r_i = \arg\max_{r_k \in R} Q(k),$$

  else, (2) the connection j remains on DCH and another timer of duration T_{out} seconds is started.

In the above definition, once connection i switches to DCH successfully, r_i is deleted from the request set R.

Motivation behind QS policy: The main motivation behind QS policy is to treat short flows and long flows differently. The size of a flow can be estimated by its queue size.

Short flows will not exceed a sufficient upper threshold T_h on the queue size and will get served on FACH. Thus, the idea is to avoid switching cost for short flows as the cost may be more or comparable to the service requirement of the short flows. Large-sized or long flows on the other hand will see their buffer queue build-up and will be switched to DCH in the above defined policy. An important advantage of this policy is that using only local information (i.e., queue size) which is easily available, implicit queue size based scheduling can be implemented in a scalable (with number of users) fashion.

6.2.2 FS Policy

In the FS (Flow Size) policy with parameter 's', the Packet Scheduler waits for the number of packets served for the current flow of a connection on FACH to exceed a threshold 's' before switching it to DCH. A connection on DCH switches back to FACH according to the same rule as in QS policy. The FS policy can be formally defined as follows:

FS policy: The FS (Flow Size) policy is similar to the QS policy except for the fact that a flow size threshold 's' is used instead of the queue size threshold T_h on FACH. It is thus characterized by the following set of rules:

- A connection i starts on FACH by default. It switches to DCH if $f(i) > s$ & $U_{dch} < N_{dch}$.

  If $f(i) > s$ & $U_{dch} = N_{dch}$ then r_i is added to R.

- If connection j is on DCH then it follows the same rule as in QS policy. When connection j switches to FACH successfully, $f(j)$ is set to 0.

The FS policy is similar to QS policy except for the fact that the flow size is directly computed from the number of packets served. A flow gets threshold amount of service on FACH, exceeding which the flow is termed as a long flow and switched to DCH. The policy is scalable with number of users as the size of a flow can be computed locally.

6.2.3 QSFS Policy

In QSFS (QS & FS combined) policy a connection on FACH switches to DCH when conditions of both QS and FS policy are satisfied. A connection on DCH switches back to FACH according to the same rule as in QS policy. The QSFS policy can be formally defined as follows:

QSFS policy:

- A connection i starts on FACH by default. It switches to DCH if $Q(i) > T_h$ & $f(i) > s$ & $U_{dch} < N_{dch}$.

  If $Q(i) > T_h$ & $f(i) > s$ & $U_{dch} = N_{dch}$ then r_i is added to R.

- If connection j is on DCH then it follows the same rule as in QS policy. When connection j switches to FACH successfully, $f(j)$ is set to 0.

We defer the performance evaluation through simulations of the above mentioned policies to Section 6.5.

6.3 Designing a New Cross-Layer Channel Switching Policy

Most of the software applications running over UTRAN use TCP as the transmission protocol. TCP reacts to congestion and losses either by drastically reducing its congestion window size after a timeout, or with some fluidity through fast retransmit procedures. For short flows with small number of packets, a loss of one of the last few packets is often detected only after a timeout, due to insufficient NACKs received by the sender. Thus timeouts of short flows are not very effective in reducing network congestion and one of the most important aspects on the downlink channel is to sustain efficient TCP performance by preventing timeouts of short flows and congestion in buffer queues [AABB04]. For example, in peer-to-peer file exchanges two users exchange a small number of packets (generating short flows) before one of them downloads a long heavy data file. Same is true for FTP and HTTP web browsing traffic where packet exchanges between applications running across UTRAN and a UE consist either entirely of short flows (if caching is enabled in the browser) or of short flows followed by a long file transfer (if caching is not enabled). Similarly, short flows are also generated by conversational voice packet transfers (not streaming voice) where maximum acceptable end-to-end delay according to the human perception is around 400 ms. Thus from user ergonomics point of view, it would seem advantageous to minimize the transfer times of short flows by giving them priority over long flows and serving them on a faster link [AABB04]. This motivates us to design a cross-layer channel switching policy in which the initial packets of a TCP flow are given priority on a fast link. If this flow turns out to be a long flow then it can be afforded to serve this flow on a slow link, since slight increase in transfer time of a long flow would be insignificant. However, if this long flow builds up a very large queue length on the slow link, then it would again have to be switched back to the fast link.

In all the existing and basic channel switching policies discussed previously in Sections 6.1.3 and 6.2, respectively, a new flow of a connection always starts on the slow FACH

channel and waits until some threshold parameter has been attained, before switching to the fast DCH channel. Short data bursts of say less than 10 packets may take a long time (on slow FACH) to surpass any threshold parameter or they may never surpass it at all (due to insufficient number of packets). Moreover, such short data bursts would be transmitted during the initial TCP slow-start phase which could further lengthen their time to surpass any threshold parameter. On the other hand, long flows with a large number of packets will most probably surpass the thresholds and get a chance to be transmitted on the fast DCH channel. Thus there is a possibility that short flows in their entirety will suffer high transmission times on the slow FACH channel, where as for long flows even though their initial few packets are transmitted on the FACH channel, their overall transmission time may improve since most of their (remaining) packets are transmitted on the DCH channel. This intuition can be further strengthened by some concrete calculations that follow.

Let us take a closer look on the FACH channel. The FACH channel has a very low set-up time, usually has a capacity of around 33 kbps and has a high priority signal traffic (from a constant bit rate (CBR) source) [HT01] running on it apart from the data packets. The CBR source transmits signal traffic at the rate of around 24 kbps. So a short data burst of say 10 packets of 1 kbyte each will take approximately 8.88 seconds (or 2.42 seconds in the best case when CBR traffic is absent) to be transmitted on the FACH channel. Now let us consider the DCH channel. The DCH channel has a capacity of around 384 kbps. There is a set-up time of around 250 ms [PAAD03, HT01] for the DCH channel which is much higher than the set-up time of the FACH channel. So unlike the mechanism used in existing and basic switching policies, if a connection starts on FACH and switches to DCH immediately without waiting to attain any thresholds, a 10 kbytes burst will get transmitted in approximately $0.25+10\times8/384 = 0.25+0.208 = 0.458$ seconds. This significantly reduces the transmission time by a factor of about 20 in the presence of CBR traffic and about 5 in its absence. Thus, switching a new flow to DCH as soon as it starts can be beneficial for short data bursts which would have otherwise suffered high transmission times on the slow FACH channel. This clearly illustrates that the existing and basic policies discussed previously in Sections 6.1.3 and 6.2, respectively, suffer from a major drawback. The drawback being that a new flow is allowed to transmit initially on slow FACH for a long time (by the threshold mechanism) before it gets a chance to be transmitted on the fast DCH.

The above argument gives us the motivation to design a cross-layer channel switching policy in which the initial few packets of a new TCP flow of a connection on FACH are given priority on the fast DCH channel by switching the connection from FACH to DCH *as soon as possible*. If this new flow is a short flow then it will be entirely served on DCH thus ensuring minimum transfer times for short flows, as explained with the help of some calculations in the previous paragraph. Otherwise if this flow turns out to be a long flow, then later if the buffer queue length of the associated connection falls below a threshold T_l, the connection is either *preempted* and switched back to FACH to allow other new flows on FACH to switch to DCH, or the connection remains on DCH and then ultimately times out (in the absence of packet arrivals during an inactivity timer period) and is switched to FACH indicating the end of current flow on the connection. Thereafter, any new packet

arrivals on this timed out connection on FACH will be termed as a new flow. Thus at any given instant there are either *new* flows on FACH attempting to switch to DCH, or there are *old* flows on FACH (which may also be long with a high probability) which have already transmitted their initial few packets (say at least first 's' packets) on DCH. If the buffer queue length of the connections with old flows surpasses the threshold T_h, then they attempt to switch to DCH again in order to minimize the use of FACH channel, since it is a very slow channel that can cause significant increase in transmission times.

FS-DCH policy:

- A connection i starts on FACH by default. It switches to DCH if (1) $f(i) \leq s$ & $U_{dch} < N_{dch}$ or (2) $f(i) > s$ & $Q(i) > T_h$ & $U_{dch} < N_{dch}$.

  If (1) $f(i) \leq s$ & $U_{dch} = N_{dch}$ or (2) $f(i) > s$ & $Q(i) > T_h$ & $U_{dch} = N_{dch}$ then r_i is added to R.

  In this rule, the condition (1) causes a new connection starting on FACH to attempt to switch to DCH as soon as possible.

- If connection j is on DCH and $Q(j) < T_l$ then

  (a) if $f(j) \leq s$, then it follows the same rule as in QS policy. When connection j switches to FACH successfully, $f(j)$ is set to 0.

  (b) if $f(j) > s$, then

  if (1) $U_{dch} = N_{dch}$ & $R \neq \phi$, connection j is *preempted* and it switches to FACH and connection i with $r_i \in R$ switches to DCH, where connection i is chosen such that $f(i) \leq s$ (its a new flow) and

  $$r_i = \arg\max_{r_k \in R} W(k).$$

  If there is no such connection that satisfies the condition $f(i) \leq s$ then connection l is chosen such that $f(l) > s$ (its an old flow) and

  $$r_l = \arg\max_{r_k \in R} Q(k),$$

  else, (2) it follows the same rule as in QS policy.

Note that in our new policy described above, a new connection must always necessarily start transmitting on the common FACH channel, since the number of DCH channels are interference limited and a DCH may not always be available to be allocated for a new connection. When a connection i on FACH attempts to switch to DCH and if no DCH

channel is available, a request r_i to switch to DCH is pushed into a request set R and this request is served when a DCH channel is available later.

We call the strategy of allowing a new flow to transmit at least its first 's' packets on DCH as the *first 's' on DCH* mechanism and it is one of the two key features of our new improved switching policy. The other key feature is the use of *dual-level priority switching* mechanism. This mechanism works as follows. If more than one connections on FACH are candidates (i.e., they have requested to switch to DCH) to be switched to a single available DCH channel, then the dual-level priority switching mechanism chooses only one connection among all connections with new flows, on a *first-come first-served* (FCFS) basis, to be switched to DCH. In the absence of connections with new flows, the connection with the *maximum queue length* among all connections with old flows, is switched to DCH. We term our cross-layer channel switching policy as FS-DCH (*at-least* flow-size threshold on DCH) policy and it can be formally summarized and defined as shown on the previous page. In the definition, once connection i (or l) switches to DCH successfully, r_i (or r_l) is deleted from the request set R. We defer the performance evaluation through simulations of the FS-DCH policy to Section 6.5.

6.4 UMTS Network Model & Simulation Setup

In this section we describe the UMTS network model that we use for performance evaluation of the various aforementioned policies through simulations. The model described here is very similar to the one in [PAAD03]. We consider a network model with N_{tcp} TCP sources which need to send data to mobile receivers. We assume a single cell scenario with one NodeB base station and several mobile stations which act as destinations for TCP traffic. The TCP sources are assumed to be connected to the base station of the cell with a high speed (5mbps, 30ms) link. The base station can transmit data from a single TCP source on either DCH or FACH, at any given time. There is one FACH and N_{dch} DCH channels in the system. The FACH is a time division multiplexed channel. In addition to any TCP connections which may be present on a FACH, there is signaling traffic which must be transmitted on the FACH. The signaling traffic has priority over the TCP connections. During the silence periods of the signaling traffic, data from one or more TCP connections can be transmitted on the FACH. Data from the TCP connections is assumed to be transmitted on the FACH with a PS or LAS service mechanism. If all the DCHs have a TCP connection allocated, a connection on DCH should be first switched to FACH before a connection from FACH can be switched on to a particular DCH. This means that a switch can take up to 500ms (if there is already a TCP connection configured on the DCH and if we consider the connection release time to be the same as the connection set-up time).

In the model we assume that there exists a queue corresponding to each TCP connection in the NodeB base station. The base station is hence able to track both the queue length and the number of packets served (flow size) for each connection. During the switching time from one channel to another, no packets from the queue of the TCP connection being switched can be transmitted. While a connection is switching from one channel to

another, the ACKs of a TCP connection traverse the original channel until the switch is completed.

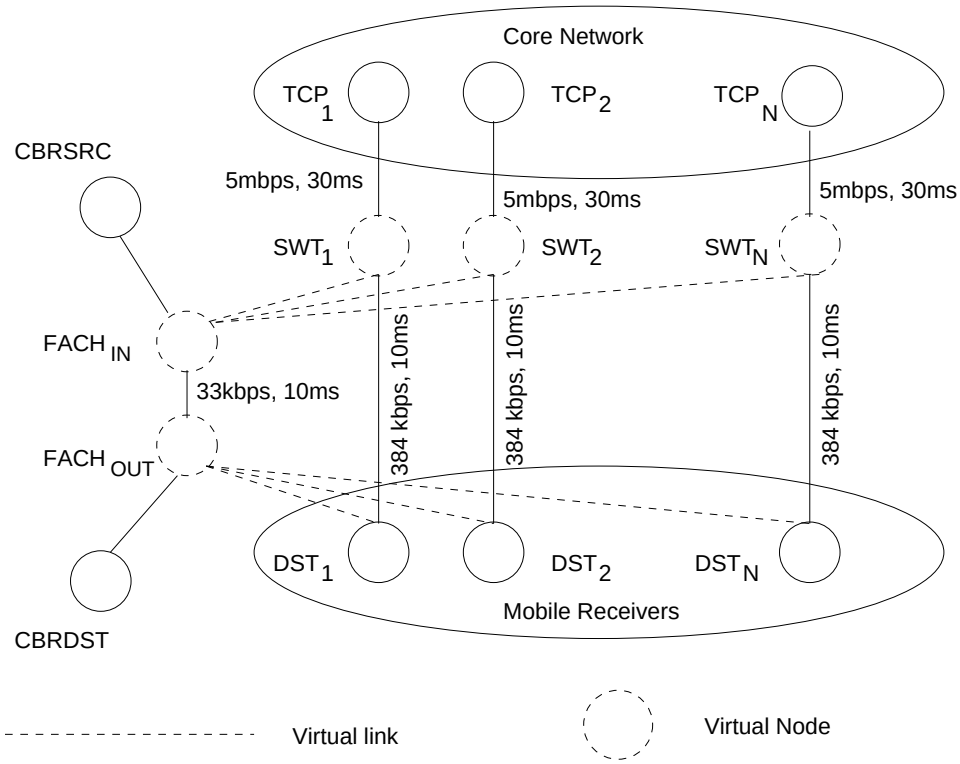


Figure 6.1: Simulation Setup

The simulation setup for the above described network model is presented in Figure 6.1. Each TCP source node TCP_i is connected to a routing node called Switch (SWT_i). SWT_i is present inside the NodeB base station and can be connected either to the $FACH_{IN}$ or directly to the TCP destination via the DCH. The SWT_i node has been introduced to simplify the simulations and may not be present inside a real NodeB base station. The $FACH_{IN}$ is another virtual node which simulates either the PS or LAS service discipline taking place on the FACH. In the PS discipline, the node $FACH_{IN}$ gives priority to the traffic from $CBRSRC$ while serving the packets from the SWT_i's (only those which are currently not transmitting on DCH) in a round-robin manner. We note that there are no queues at $FACH_{IN}$ and all the packets are either queued at SWT_i or at the $CBRSRC$. The $CBRSRC$ simulates a constant bit rate source of signaling/control traffic. It generates packets at rate R_{sig} and is assumed to be present within the NodeB. Even though we model the destination of the signaling traffic ($CBRDST$) as another node different from the mobile destinations DST_i, we note that it does not affect the simulations as simultaneous transfer of data and control packets to the same mobile receiver is indeed possible in UMTS when different channels are used. The links $SWT_i - FACH_{IN}$ are virtual links within the base station and thus have zero delay. Note that the data from SWT_i to DST_i can take two different routes i.e., $SWT_i - FACH_{IN} - FACH_{OUT} - DST_i$ (via FACH) or simply

$SWT_i - DST_i$ (via DCH). At any given time only one of the above two routes can be active for a given connection. Although in the simulation scenario we have as many DCH links as TCP source nodes, the simulation allows us to activate not more than N_{dch} DCH channels at a time, which may be chosen strictly smaller than the number of TCP sources (N_{tcp}). In the simulations we switch a connection from FACH to DCH by changing the cost of the links and recomputing the routes. This is done as follows. Initially, the cost of direct path from the Switch to the TCP destination is set to 10 and the cost of all other links to 1. Hence, the traffic gets routed through the FACH. When a switch is required, the cost of DCH is set to 0.5 and the routes are recomputed. This activates the DCH and the traffic gets routed on the DCH.

6.4.1 Limitations and Assumptions

The layer 2 in UTRAN consists of two sub-layers: MAC layer and RLC (Radio Link Control) layer. As described previously, the physical layer (layer 1) offers services to the MAC layer via transport channels of two types: dedicated channels and common channels. The MAC layer in turn offers services to the RLC layer above it through logical channels. The different logical channels are mapped to the transport channels in the MAC layer. The two most important *logical entities* in MAC layer are MAC-c/sh and MAC-d. The MAC-c/sh entity handles data for the common and shared channels, where as the MAC-d entity is responsible for handling data for the dedicated channels. However, the execution of switching between common and dedicated channels is also performed by the MAC-d entity in UTRAN (in the serving RNC) based on a switching decision derived by the channel switching algorithm that resides in the RRC (Radio Resource Controller) [HT01].

Data packets or SDUs (Service Data Units) arriving from upper layers are segmented into smaller data packets or PDUs (Protocol Data Units) by the RLC layer and PDUs are then forwarded to the MAC layer. In our network model used to carry out the simulations for performance evaluation of various switching policies, we do not consider the segmentation of SDUs into PDUs. In other words, we do not model the RLC layer since the main focus of this work is to investigate the channel switching mechanism. We thus model only the MAC-d entity in the MAC layer. We also do not take care of packet loss, mobility and handovers, since considering them would highly complicate the model and it is beyond the scope of this work.

6.4.2 Simulation Parameters

We use ns-2 [ns2] in order to simulate the various switching policies for performance evaluation. The simulation parameters used are described below:

- We consider the number of dedicated channels, $N_{dch} = 1$ and the number of TCP sources, $N_{tcp} = 2$ and 3.
- The duration of simulations is taken to be $200,000$ secs. in order to reach stationarity and each simulation scenario is averaged over 10 runs.

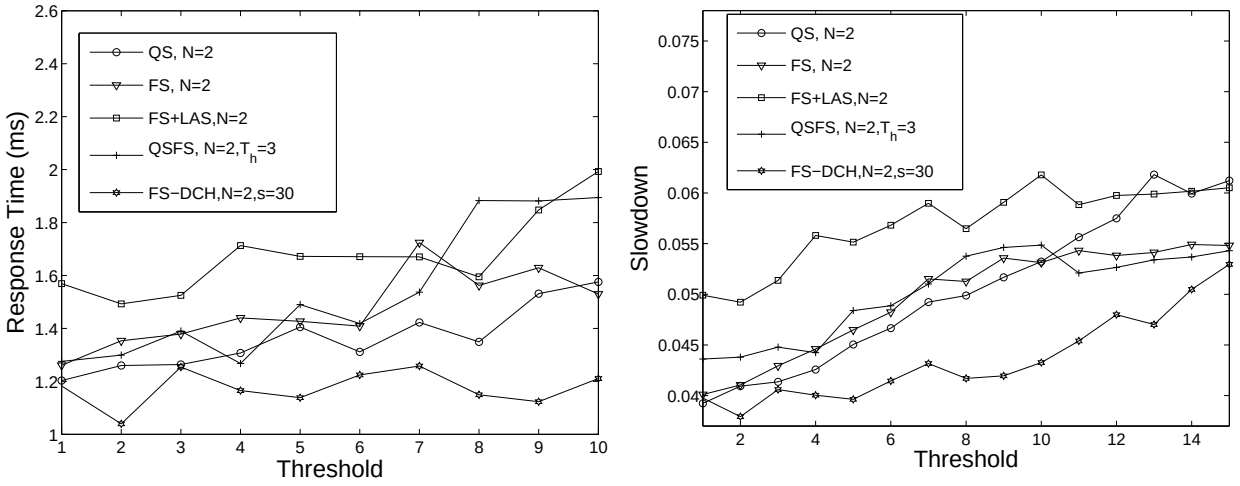
- The transmission rates for FACH and DCH channels are considered to be 33 kbps and 384 kbps, respectively.
- The switching cost D_{sw} (in terms of time) between FACH and DCH channels and vice-versa is 250ms each [PAAD03, HT01].
- We consider the signaling traffic source (non TCP traffic source) that uses the FACH, to be a constant bit rate CBR source with rate $R_{sig} = 24$ kbps. It sends a 1 kbyte packet at an interval of 1/3s and has a non preemptive priority over TCP traffic.
- The TCP connection traffic is modeled is as follows: In a TCP connection, data arrives in bursts. The number of packets in a burst has a Pareto distribution and the shape parameter is taken to be $k = 1.1$. The average file size is taken to be $FS_{avg} = 30$ kbytes. A TCP connection alternates between "ON" and "OFF" states. The ON state is comprised of several bursts and no packets are transmitted during the OFF state. In the ON state, the inter-arrival time between successive bursts is exponentially distributed with mean $T_{ON} = 0.3$s. At the end of each burst in ON state, the connection goes into OFF state with probability $P_{OFF} = 0.33$. It remains in the OFF state for an exponentially distributed duration with mean $T_{OFF} = 5$s before it goes back into ON state again.
- The value of T_l (lower threshold on DCH) is taken as 1 and the packet size as 280 bytes.

6.5 Performance Evaluation of Policies

In this section, we analyze the results obtained from an extensive set of simulations of the various channel switching policies that we have discussed until now. We study PS scheduling of TCP sources on the FACH channel for QS, FS, QSFS and FS-DCH policies. In addition to this we also study LAS scheduling on FACH channel for FS policy specifically. LAS scheduling can also be studied with other policies, but since LAS looks at the number of served packets, which relates to the flow size, FS policy is the most appropriate one to study with LAS scheduling.

In Figures 6.2-6.3, we compare different policies in terms of response time and slowdown metrics as a function of the thresholds s or T_h, as the case may be. The x-axis in all graphs for QS policy denotes threshold T_h, for FS policy it denotes threshold s, for QSFS policy it denotes threshold s and for FS-DCH policy it denotes threshold T_h. The response time is calculated as the total average time required to completely transmit a burst. By completely transmitting a burst, we mean the time until a TCP ACK for the last packet of a burst sent, is received at the sender side. Slowdown is defined as the response time divided by the average burst size. In other words, for an average burst size of x, if $T(x)$ is its response time then the slowdown $S(x)$ is defined as $\frac{T(x)}{x}$.

It should be noted here that the plot for each policy in Figures 6.2-6.3 has been obtained by averaging simulation results gathered over 10 runs with each run of duration



(a) (b)

Figure 6.2: Comparison of different policies in terms of response time and slowdown metrics for $N_{tcp} = 2$, $FS_{avg} = 30$ kbytes and $N_{dch} = 1$.

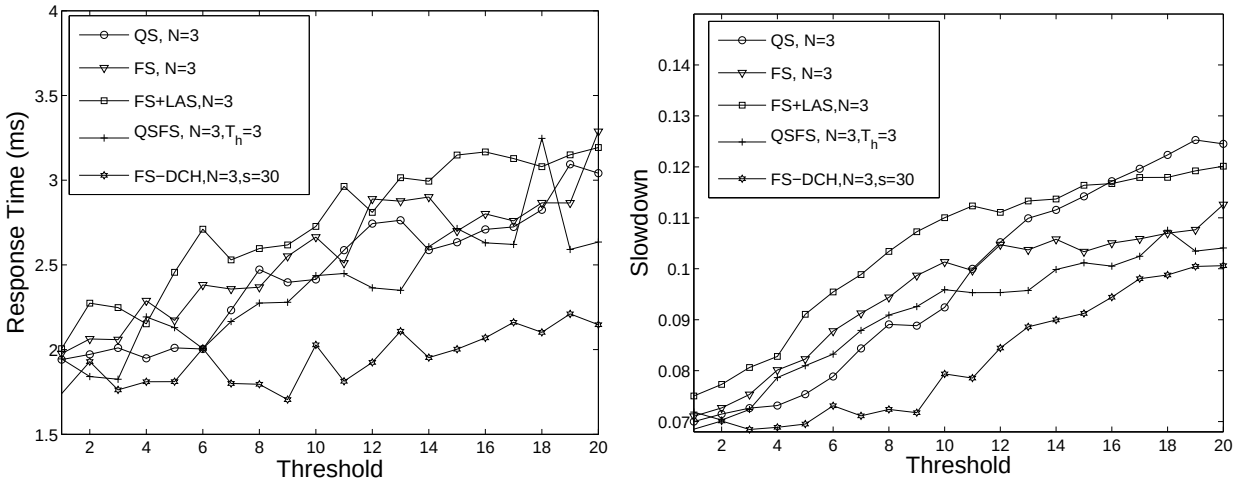
200,000 secs. We still obtain not so smooth plots inspite of such an averaging exercise. The reason for this may be attributed to the bursty nature of the traffic generated by TCP.

If we compare the simulation results of our new basic QS, FS and QSFS policies in Figures 6.2-6.3 with results of the 'modified threshold policy' proposed in [PAAD03], we can easily observe that our new switching policies improve on the 'modified threshold policy' by around 17% in terms of response time.

In Figure 6.2(a), we observe that FS-DCH outperforms all other policies in terms of response time, where as FS+LAS scheme has the highest response time. The other three schemes have fairly comparable response times. The average improvement in response time achieved by FS-DCH over all other policies is around 30%. Within the range of threshold values shown, we observe an increasing trend in response time under all policies except for FS-DCH. The QS policy performs slightly better than the FS policy in minimizing mean response time.

Under QS, FS and QSFS policies, at higher values of T_h an increase in the response time is observed because a higher value of T_h implies more time is spent in the FACH. The FACH is a low bandwidth channel which has high priority signaling traffic on it. This results in low average bandwidth being shared amongst the TCP connections due to the following reason. For a TCP connection, the switch to DCH is based on its current buffer size which in turn depends on its current congestion window size. The congestion window size is incremented whenever an ACK is received by the sender. When a TCP connection is on a low bandwidth link, the window builds up slowly due to greater delay in receiving an ACK. This slow buildup of the window size results in slow buildup of the current buffer size. As the value of T_h is increased, a TCP connection has to spend more time on the slow FACH, resulting in a higher response time.

The comparison of average slowdown in Figure 6.2(b) shows that the slowdown metric



(a) (b)

Figure 6.3: Comparison of different policies in terms of response time and slowdown metrics for $N_{tcp} = 3$, $FS_{avg} = 30$ kbytes and $N_{dch} = 1$.

fairly follows the same trend as that of average response time in Figure 6.2(a). FS-LAS has the highest slowdown and FS-DCH has the lowest. Other policies perform almost the same except that performance of QS worsens for higher values of the threshold.

In Figure 6.3, we plot the average response time and slowdown for $N_{tcp} = 3$. It can be easily seen that FS-DCH again performs the best in terms of both response time and slowdown and all other policies perform comparably among themselves. The average improvement in response time achieved by FS-DCH over all other policies is around 36%.

From the above discussions it can be concluded that the cross-layer FS-DCH policy is better than all other policies for $N_{tcp} = 2$ & 3 and $N_{dch} = 1$. In other simulation results which are not presented here due to brevity, we have observed that FS-DCH policy continues to perform better than other policies when the value of N_{tcp} is increased in proportion to the value of N_{dch}, i.e., for example, $N_{tcp} = 4$ and 6 for $N_{dch} = 2$, etc.

6.6 Conclusion

In this chapter, we have proposed several *scalable* channel switching policies for packet data transmission on UMTS downlink. Simulation results show that our new basic switching policies QS, FS and QSFS improve on the 'modified threshold policy' in [PAAD03] by around 17% in response time metrics. We have further proposed a new and improved *cross-layer* channel switching policy that we call FS-DCH policy. FS-DCH is a biased policy that improves the performance of TCP flows by giving priority to short flows on the fast DCH channel. Results obtained from extensive simulations show that FS-DCH performs better than the basic QS, FS and QSFS policies. For example for $N_{tcp} = 2$ and 3, FS-DCH gives a significant average improvement of 30% and 36%, respectively, over all other policies in terms of response time.

Chapter 7

Capacity Optimizing Hop Distance in a MANET with Power Control

In a dense multi-hop network of mobile nodes capable of applying adaptive power control, we consider the problem of finding the optimal hop distance that maximizes a certain throughput measure in *bit-metres/sec*, subject to average network power constraints. The mobility of nodes is restricted to a circular periphery area centered at the nominal location of nodes. We incorporate only randomly varying path-loss characteristics of channel gain due to the random motion of nodes, excluding any multi-path fading or shadowing effects. Computation of the throughput metric in such a scenario leads us to compute the probability density function of random distance between points in two circles. Using numerical analysis we discover that choosing the nearest node as next hop is *not always* optimal. Optimal throughput performance is also attained at non-trivial hop distances depending on the available average network power.

Note: The material in this chapter has appeared in [KVKA06]. This work was sponsored by "The Indo-French Centre for the Promotion of Advanced Research" (CEFIPRA).

7.1 Introduction

In this chapter, we study the optimal next hop distance that maximizes the system *end-to-end flow throughput* in a mobile multi-hop wireless network environment subject to a network average power constraint. In our investigation we assume a spatially dense layout of nodes and we incorporate channel gain due to path-loss caused by the mobility of nodes. It is assumed that during exchange of control packets, a transmitter node can estimate channel gain and employ adaptive power control, modulation and coding. We consider a *periphery* limited mobility scenario in which nodes are restricted to move in their own local, approximately circular, periphery. For the calculation of the average throughput with path-loss, this kind of a mobility model leads us to compute the probability density function (PDF) of random distance between two nodes inside their circular periphery. Computation of this PDF constitutes a problem in geometric probability and to the best of our knowledge the derivation of PDF of random distance between points in each of two circles has never been investigated before. This is thus the first main contribution of this chapter. The second main contribution of this chapter is that, to the best of our knowledge, ours is the first attempt to derive a throughput maximizing optimal hop distance in a dense ad hoc network environment with such a restricted mobility model.

7.2 Network and Mobility Model

Consider a dense multi-hop wireless network where the nominal locations of mobile nodes are densely distributed. A contention based distributed channel access mechanism such as the CSMA/CA based DCF is employed by nodes for transmission scheduling. It is assumed that all nodes use the same contention mechanism with identical parameters and every node has packets to be transmitted at all times. For control signalling (such as RTS/CTS in IEEE 802.11), nodes do not employ power control and hence use constant power. We assume a "single cell" situation in which control packets are heard by all nodes in the network and after a node wins a contention, it reserves the channel exclusively for itself. Even if it is possible for a source to transmit packets directly to its far away destination in a single hop (by using high enough transmission power), we assume that the source instead routes packets over shorter hops taking advantage of smaller transmit power consumption in the latter case. It is further assumed that, during control signalling, a transmitter can measure the channel attenuation and apply adaptive power control, coding and modulation on a per transmission basis.

We assume a *periphery* limited mobility scenario in which every node is restricted to move within an approximate circular periphery centered at its nominal location. Inside their confined areas, the movement of nodes is in accordance with *any* arbitrary mobility model that generates a uniform spatial distribution of nodes in steady-state. Figure 7.1 shows such a scenario. We could approximate the periphery by a square or any other shape but a circle is a more natural choice. For the sake of clarity, the magnified box shows only non-overlapping periphery nodes, but in general any two nodes that are moving sufficiently close to each other can have overlapping peripheries, as shown in the left box. We define

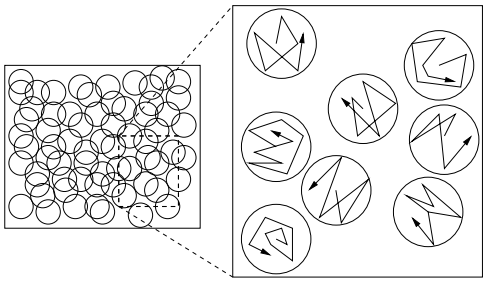


Figure 7.1: *Periphery* limited mobility model

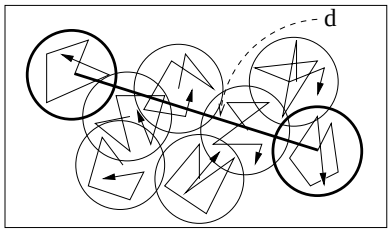


Figure 7.2: Consecutive relay nodes in a route do not overlap

periphery hop distance d as the distance between the centers of the periphery circles of two adjacent *relay* nodes that constitute a route (see Figure 7.2). We assume that peripheries of consecutive or adjacent relay nodes constituting a route, do not overlap (Figure 7.2). To be more clear on this, even though peripheries of *any* two nodes in general that are moving sufficiently close to each other, *can overlap*, peripheries of adjacent *relay* nodes that constitute a route are assumed to be *non-overlapping*. Thus any *feasible* periphery hop distance d satisfies $d > 2a$ where a is the radius of the circular periphery. This is a fairly practical assumption since for instance in the security guard MANET example that follows, two guards constituting a route need not have their assigned security zones overlapping. This kind of a mobility model can be readily applied to various situations. For example, a MANET formed by security guards who are restricted to move in their assigned security zones during a public event. A similar MANET formed by people across different rooms of a building restricted to move in their rooms. Soldiers in a battlefield moving inside their own units, a group of sensor robots moving in a mine field or nuclear establishment restricted to their confined areas, etc. are other examples. With such a mobility model, our goal is to obtain an *optimal* periphery hop distance d^* that maximizes an end-to-end flow throughput measure. In the rest of the chapter, when we say that the distance between two nodes is d, we always actually mean that the distance between the centers of the periphery circles of the two nodes is d. Our model incorporates only randomly varying path-loss characteristics of channel gain and does not include any multi-path fading or shadowing effects. If the channel gain measured by a transmitter is denoted by h and $P(h)$ denotes the corresponding power control applied, the channel capacity by Shannon's formula is then given by

$$C(h) = Wlog_2\left(1 + \frac{P(h)h\alpha}{\sigma^2}\right)$$

where, W is the RF bandwidth, σ^2 is AWGN power and α is a constant depending on the minimum distance between periphery centers of two nodes, $2a$.

7.3 Background and Problem Objectives

With multi-hop communication between a source and its destination taking place in such a scenario, there is an inherent tradeoff between employing shortest path routing (in terms of number of hops) with high transmission power and many small hops with low transmission power. Shortest path routing (long hop distance routing) with high transmission power provides low end-to-end packet delays but at the same time can easily result in fast depletion of battery energy. Instead, the use of smaller hops reduces power consumption, but increases the total number of packets to be relayed in the network. These arguments clearly illustrate the need for an optimal power control policy combined with an optimal hop distance choice. Such an optimization problem for fixed multi-hop networks has been studied in [RKA] in which the authors obtain an optimal power control and hop distance that maximizes an end-to-end flow throughput measure in bit-metres/sec, under an average network power constraint. In this work we use a slightly modified version of the results obtained for fixed networks in [RKA]. Our network model is similar to the model used in [RKA] augmented by our mobility model described in Section 7.2. For the sake of clarity and completeness, we briefly re-visit the derivation of results in [RKA] to deduce a slightly modified version for our use.

Objective function with end-to-end flow throughput: Consider a source and its destination that are distance D apart and engage intermediate relay nodes for multi-hop communication. By the dense node layout assumption, we can always find a multi-hop path between the source and its destination such that the periphery centers of the relay nodes lie on the straight line connecting them. We assume that the periphery hop distances between consecutive relay nodes are all equal to d metres. Only one transmitter can successfully transmit packets at any given time (see Section 7.2). Let $\Theta(d)$ denote the long-term, time average rate of packet transmissions over the entire network. Each packet transmitted in the network is counted in $\Theta(d)$ and when a packet is relayed, it is counted as many times as it is forwarded. We consider the *fixed transmission time* case [RKA] along with channel gain due to variable path-loss caused by mobility of nodes. By fixed transmission time it is meant that, upon winning channel access, a node is allowed to transmit only for a fixed amount of time T irrespective of the channel conditions. If power control is applied during a transmission opportunity then the node will be able to transmit only $L(h) := C(h)T$ amounts of data, where h is the channel attenuation and $C(h)$ is the available Shannon capacity as described before. Averaging the data sent during a transmission opportunity over random path-loss, the long-term data transmission rate in the network, obtained by approximate analysis in [KAMG05a] (also see [Bia00a]) would be given by,

$$\Theta(d) = \frac{p_s \int_h L(h)\, g_H(h)\, dh}{p_i T_i + p_c T_c + p_s\,(T_o + T)},$$

where p_i is the probability that the contention period goes idle, p_s is the probability that there is a successful transmission, p_c is the probability that there is a collision, T_i is the average idle period during the contention mechanism, T_o and T_c are fixed overheads associated with a successful transmission and a collision, respectively and $g_H(h)$ is the probability density function (PDF) of random path-loss h.

Now, if we suppose that all source-destination pairs are distance D apart then there are approximately $\frac{D}{d}$ hops for a pair. As discussed before, each end-to-end packet is counted as many times as it is relayed, then with $\Theta(d)$ as the total packet rate of the network, the end-to-end aggregate flow throughput is given by

$$\frac{\Theta(d)}{\frac{D}{d}} = \frac{\Theta(d)}{D} d.$$

However each packet is transmitted across distance D. Thus the long-term end-to-end flow throughput in the system in *bit-metres/sec* is given by

$$\frac{\Theta(d)}{\frac{D}{d}} \times D = \Theta(d)\, d.$$

The objective function that is to be maximized can therefore be defined as

$$\phi(P(h), d) \stackrel{\Delta}{=} \Theta(d)\, d,$$

subject to an average network power constraint that will be detailed in the following discussion. Now, denote the random distance separation between consecutive relay nodes by l. Then for a fixed d, the randomly varying path-loss from transmission to transmission is due to the randomly changing quantity l. The variation in power control $P(h)$ from transmission to transmission will thus depend only on l for a fixed d. Maximizing $\phi(P(h), d)$ can therefore be isolated into first maximizing over the (deterministic) function $P(\cdot)$ for a fixed d and then maximizing over d. Consequently we seek to solve the following optimization problem

$$\max_{d} \max_{P(\cdot): E[P(h)] \leq \bar{P}} \phi(P(h), d) \quad \text{, or}$$

$$\max_{d} \max_{P(\cdot): E[P(h)] \leq \bar{P}} \frac{p_s \left(\int_h L(h)\, g_H(h)\, dh \right) d}{p_i T_i + p_c T_c + p_s \left(T_o + T \right)} \tag{7.1}$$

Consider the 'Channel Left Idle when Bad' case [RKA] when the channel is left idle for T seconds (and not relinquished for a new contention mechanism cycle to begin) if the power $P(h)$ allocated by the transmitter is 0 for any channel state h. We see that the denominator of the objective function $\phi(P(h), d)$ in Equation 7.1 does not depend on $P(h)$ and d. Thus our maximization problem of Equation 7.1 reduces to maximizing $\phi(P(h), d)$ given by

$$\begin{aligned} B_\phi \cdot d \cdot \int_h L(h) g_H(h)\, dh \quad &= \\ &\frac{B_\phi W T d}{\ln 2} \int_h \ln \left(1 + \frac{P(h) h \alpha}{\sigma^2} \right) g_H(h)\, dh \end{aligned} \tag{7.2}$$

subject to an average network power constraint given by

$$\frac{p_i E_i + p_c E_c + p_s(E_o + \int_h P(h) g_H(h)\, dh\, T)}{p_i T_i + p_c T_c + p_s(T_o + T)} \leq \bar{P}$$

where

$$B_\phi = \frac{p_s}{p_i T_i + p_c T_c + p_s(T_o + T)}$$

is a constant and E_i, E_c and E_o are average energy overheads due to idle time, collision and transmission respectively. As the denominator of the power constraint, as well, is a constant independent of $P(h)$ and d, the above power constraint can be rewritten as

$$\int_h P(h) g_H(h)\, dh \;\leq\; \bar{P}',$$

where $\bar{P}' = (\bar{P}(p_i T_i + p_c T_c + p_s(T_o + T)) - (p_i E_i + p_c E_c + p_s E_o))/p_s$ is now different from the previous $\bar{P}$. This is however a well known optimization problem that has a water-pouring form solution [GV97] given by

$$P(h) \;=\; \left(\frac{B_\phi W T d}{\lambda \ln 2} - \frac{\sigma^2}{h\alpha}\right)^+,$$

where λ is obtained from

$$\int_h P(h) g_H(h)\, dh \;=\; \bar{P}'.$$

From the above equation, note that such a water-pouring form solution assigns a large transmission power when the channel is good (large h) and no power at all when the channel is bad (small h). For computing λ and hence $P(h)$, we determine the PDF $g_H(h)$ of path-loss in the following section. Once $P(h)$ is known we determine the optimal periphery hop distance d^* in Section 7.5.

7.4 Obtaining the path-loss density

In the network model described before in Section 7.2, we assume a spatially dense layout of nodes with peripheries of arbitrary nodes that move sufficiently close to each other, overlapping. However, according to the feasibility of periphery hop distance assumption $d > 2a$, the circular peripheries of consecutive or adjacent relay nodes, constituting a route in a multi-hop connection, do not overlap (see Section 7.2 for details). For simplicity, let the radius of circular periphery of all nodes be same and equal to a metres. Now, consider two circles $\mathcal{C}_1$ and $\mathcal{C}_2$ of radius a centered at $(0, 0)$ and $(d, 0)$ with two consecutive relay nodes of a route, moving according to the mobility model described before, one in each one of them (Figure 7.3). Packets are transmitted between this transmitter-receiver pair after

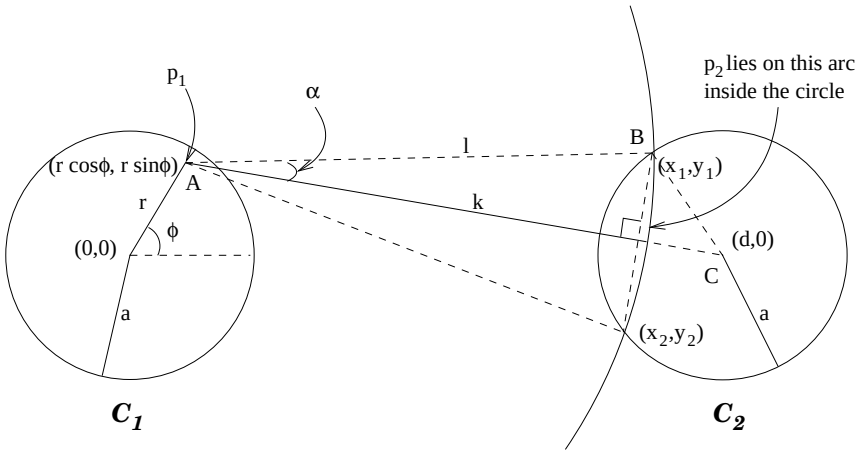


Figure 7.3: Random distance between two consecutive relay nodes moving inside their circular peripheries

the transmitter wins a transmission opportunity at the end of a contention attempt. At the next transmission opportunity, either only the transmitter or only the receiver or both or none would have moved to another random point(s) in the circle(s). Thus each time a transmission takes place on a transmitter-receiver link, it can be assumed that the end point positions of that link are sampled independently randomly from within their peripheries. There is no correlation between positions in between transmission opportunities. Then, from transmission to transmission, the path-loss between a transmitter-receiver pair will vary randomly owing to the random distance separation phenomenon between the two nodes. In the following part of this section we first compute $f_L(l)$ the PDF of random distance between any two random points in two circles and then later compute the PDF of random path-loss $g_H(h)$ from $f_L(l)$.

7.4.1 PDF of random distance between two circles

Let the distance between two randomly chosen points p_1 and p_2 in each of the two circles $\mathcal{C}_1$ and $\mathcal{C}_2$ shown in Figure 7.3 be denoted by l. Then the probability density for this random distance l can be computed as

$$f_L(l) = \frac{\int_{\mathcal{C}_1} d\vec{p}_1 \int_{\mathcal{C}_2} \delta(|\vec{p}_1 - \vec{p}_2| - l)\, d\vec{p}_2}{\int_{\mathcal{C}_1} d\vec{p}_1 \int_{\mathcal{C}_2} d\vec{p}_2} \tag{7.3}$$

where $\delta(\cdot)$ is Dirac's delta function. For a fixed l, the term $\int_{\mathcal{C}_2} \delta(|\vec{p}_1 - \vec{p}_2| - l)\, d\vec{p}_2$ in the numerator of the equation above, represents the length of an arc of the circumference of a circle of radius l centered at $\vec{p}_1$, that lies inside $\mathcal{C}_2$. Referring to the geometry shown in Figure 7.3, if k is the distance from the center of the circle of radius l, centered at $p_1 = (r\cos\phi, r\sin\phi)$, to the line joining points of its intersection with $\mathcal{C}_2$, then the length

of the arc inside $\mathcal{C}_2$ is given by $2\,l\,\alpha$, where

$$\alpha = \arccos\left(\frac{k}{l}\right).$$

Using polar coordinates and $d\vec{p}_1 = r\,dr\,d\phi$, the entire numerator can thus be written as

$$2\,l\int_{\mathcal{C}_1} \arccos\left(\frac{k}{l}\right)\, r\; dr\, d\phi.$$

The denominator is simply the product of the areas of $\mathcal{C}_1$ and $\mathcal{C}_2$ given by $\pi^2 a^4$. For computing k we proceed as follows. Consider the two circles centered at $(r\cos\phi, r\sin\phi)$ and $(d, 0)$ with radii l and a, respectively. Denote the difference between their x coordinates as $e = d - r\cos\phi$, difference between their y coordinates as $f = 0 - r\sin\phi$ and distance between their centers by $p = \sqrt{e^2 + f^2}$. Now in $\triangle$ABC, the cosine formula for triangles gives $a^2 = l^2 + p^2 - 2\,l\,p\cos\alpha$. But we also have $\cos\alpha = \frac{k}{l}$ and this gives the distance between center of first circle and line joining points of their intersection as

$$k = \frac{p^2 + l^2 - a^2}{2p}.$$

Note that l can vary as $d - 2a \leq l \leq d + 2a$. Now, from Equation 7.3, the PDF $f_L(l)$ can be written as

$$f_L(l) \quad = \quad \frac{2\,l}{\pi^2 a^4}\iint_{{\mathcal{C}_1}^o} r\arccos\left[\frac{d^2 + r^2 - 2\,d\,r\cos\phi + l^2 - a^2}{2l\sqrt{d^2 + r^2 - 2\,d\,r\cos\phi}}\right] dr d\phi \tag{7.4}$$

where ${\mathcal{C}_1}^o$ is a sub-region of the circle $\mathcal{C}_1$ given by

$$\cos\phi \geq \frac{d^2 + r^2 - (l+a)^2}{2dr}$$

for $d - 2a \leq l < d$ and

$$\cos\phi \leq \frac{d^2 + r^2 - (l-a)^2}{2dr}$$

for $d < l \leq d + 2a$. The sub-regions are derived using the bounds $l - a \leq p$ and $l + a \geq p$ for the two circles to intersect. We have not been able to integrate $f_L(l)$ to obtain a closed form expression and hence we will pursue numerical analysis in Section 7.5.

7.4.2 PDF of path-loss as transformation of $f_L(l)$

The path-loss h for a transmission distance l is given by $h = \frac{1}{l^\eta}$ where η is the path-loss exponent. Since l is randomly changing due to mobility of nodes, the transmissions

encounter random path-loss. For mathematical convenience let h be defined as $h := \left(\frac{d}{l}\right)^{\eta}$, then the path-loss PDF $g_H(h)$ can be computed as

$$g_H(h) = f_L\left(\frac{d}{h^{\frac{1}{\eta}}}\right) \left|\frac{-d}{\eta \cdot h^{1+\frac{1}{\eta}}}\right|$$

From Equation 7.4, $g_H(h)$ is thus given by

$$\begin{aligned} g_H(h) \quad &= \quad \frac{2}{\pi^2 a^4} \times \frac{d^2}{\eta \cdot h^{1+\frac{2}{\eta}}} \times \\ &\iint_{\mathcal{C}_1{}^o} r \arccos\left[\frac{d^2 + r^2 - 2\,d\,r\cos\phi + \frac{d^2}{h^{\frac{2}{\eta}}} - a^2}{2\,\frac{d}{h^{\frac{1}{\eta}}}\sqrt{d^2 + r^2 - 2\,d\,r\cos\phi}}\right] dr d\phi \end{aligned} \tag{7.5}$$

where $\mathcal{C}_1{}^o$ is the transformed region

$$\cos\phi \geq \frac{d^2 + r^2 - \left(\frac{d}{h^{1/\eta}} + a\right)^2}{2dr}$$

for $1 < h \leq \left(\frac{d}{d-2a}\right)^{\eta}$ and

$$\cos\phi \leq \frac{d^2 + r^2 - \left(\frac{d}{h^{1/\eta}} - a\right)^2}{2dr}$$

for $\left(\frac{d}{d+2a}\right)^{\eta} \leq h < 1$.

7.5 Optimal Hop Distance by Numerical Analysis

Having obtained the PDF of path-loss in the previous section, we now obtain the optimal periphery hop distance d^* with the help of numerical analysis, since we have not been able to symbolically integrate $g_H(h)$ to obtain a closed form expression. With $h = \left(\frac{d}{l}\right)^{\eta}$ defined in the previous section, the water-pouring form optimal power control becomes

$$P(h) \;=\; \left(\frac{B_\phi\, W\, T\, d}{\lambda \ln 2} - \frac{\sigma^2 d^\eta}{h\alpha}\right)^+ .$$

The expression for $\phi(P(h), d)$ from Equation 7.2 then translates to

$$\begin{aligned} \frac{B_\phi\, W\, T\, d}{\ln 2} \int_h \ln\left(1 + \frac{P(h) h\alpha}{\sigma^2 d^\eta}\right) g_H(h)\, dh \quad &= \\ \frac{B_\phi\, W\, T\, d}{\ln 2} \int_h \ln\left(\frac{h\alpha B_\phi W T}{\sigma^2 d^{\eta-1}\, \lambda \ln 2}\right) g_H(h)\, dh & \end{aligned} \tag{7.6}$$

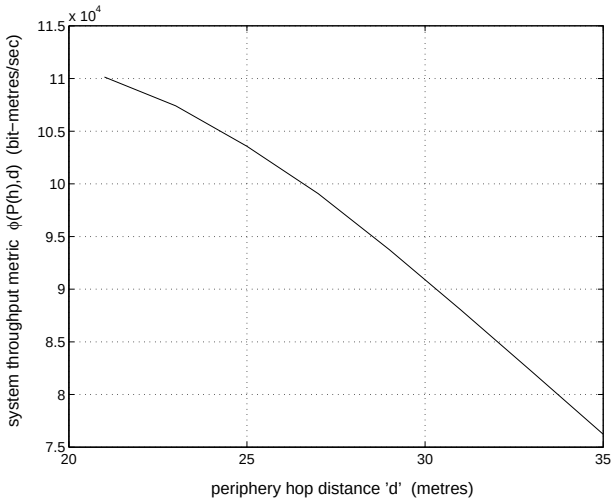


Figure 7.4: $\phi(P(h), d)$ for $\bar{P}$=0.01 watts

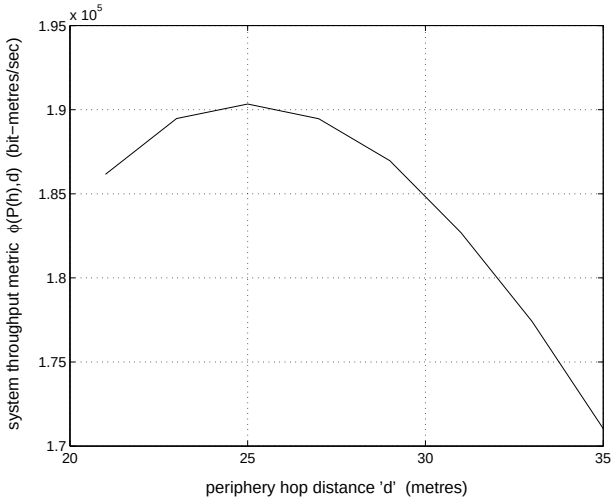


Figure 7.5: $\phi(P(h), d)$ for $\bar{P}$=0.05 watts

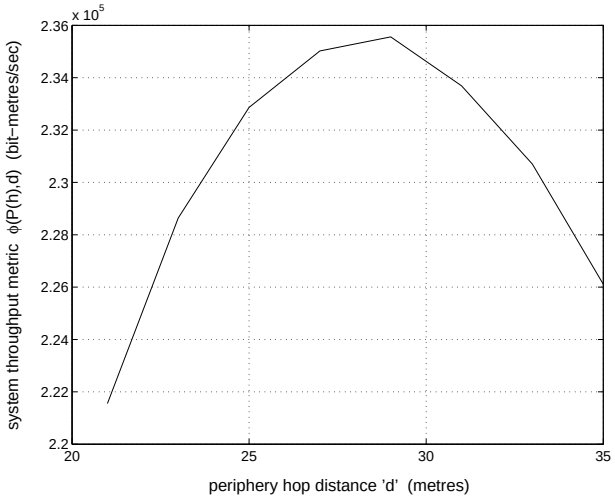


Figure 7.6: $\phi(P(h), d)$ for $\bar{P}$=0.1 watts

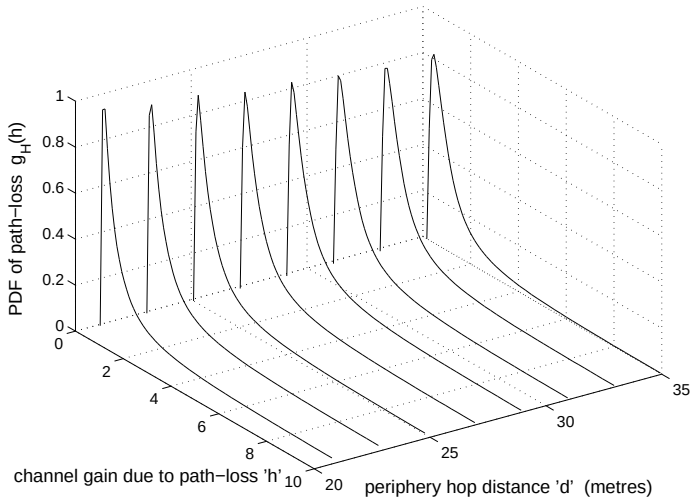


Figure 7.7: Path-loss distr. $g_H(h)$ for $\bar{P}$=0.1 watts

In order to obtain the optimal periphery hop distance d^* for an IEEE 802.11 standard scenario, in Figures 7.4-7.6 we plot $\phi(P(h), d)$ against d. In IEEE 802.11, time is divided into discrete standardised time intervals called *slots* (e.g., $1slot = 20\mu s$ for 802.11b). The various parameters that we consider for our numerical analysis are $W = 20MHz$ (typical channel bandwidth), $T = 300slots$ (example fixed transmission time), $\sigma^2 = 2.208 \times 10^{-13}watts$ (typical noise power from kTB formula), $\eta = 4$ (typical path-loss exponent for an obstructed path within a building), $\{T_i, T_o, T_c\} = \{1, 90.8, 17\}slots$ (for 802.11b), $a = 10m$ and different values of $\bar{P}$. Typical values of $p_i = 0.0271$, $p_s = 0.0981$ and $p_c = 0.8748$ are used for a network of 1000 (802.11b) nodes. For computational simplification we assume $p_iE_i + p_cE_c + p_sE_o = 7.14 \times 10^{-6}joules$ which is a practical assumption for a network of 1000 nodes.

Figure 7.4 shows that, the optimal periphery hop distance, for an average network power of $\bar{P} = 0.01watts$, is the distance to the *nearest* node. In other words, the throughput metric $\phi(P(h), d)$ is maximized while choosing the smallest possible periphery hop distance. However, similar Figures 7.5 and 7.6 for $\bar{P} = 0.05watts$ and $\bar{P} = 0.1watts$ respectively, show that *non-trivial* optimal periphery hop distances are obtained and they increase with increasing average network power. An interesting shift of $\phi(P(h), d)$ towards concavity is observed, thus yielding non-trivial optimal periphery hop distances of $d^* = 25m$ and $d^* \approx 29m$, respectively. Note that even though d^* is theoretically obtained from a continuous set of feasible values, due to our dense node layout assumption, we should always be able to find a next hop whose periphery center is at-least approximately, d^* metres from the periphery center of transmitting node.

Figures 7.7 and 7.8 show PDF of path-loss $g_H(h)$ and optimal power control $P(h)$ for $\bar{P} = 0.1watts$ and different values of d. In both these figures the path-loss h is given by $h = \left(\frac{d}{l}\right)^{\eta}$. Note that the peaks of the path-loss PDF for different d are attained at values of $h < 1$. They are not attained at $h = 1$ and hence $l = d$ is not the maximum likelihood random distance between points in two different circles whose centers are separated by

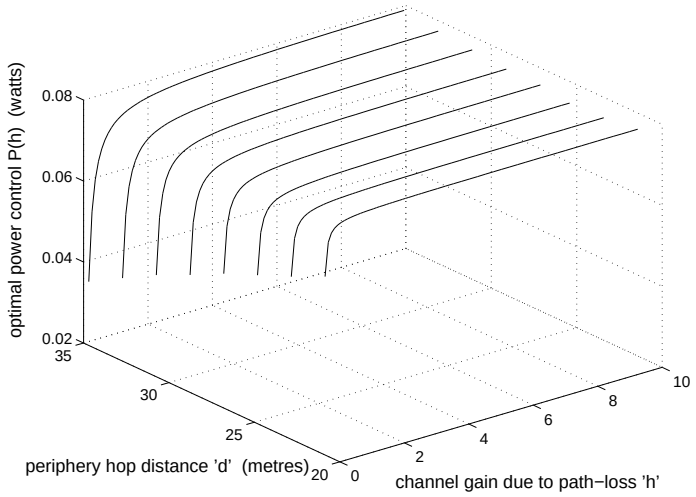


Figure 7.8: Power control $P(h)$ for $\bar{P}$=0.1 watts

distance d. Also observe that the optimal power control $P(h)$ in Figure 7.8 assigns very little or almost no power for bad channel conditions (small h) and most of the power is assigned for good channel conditions (large h). Note the increase in optimal throughput value with increasing $\bar{P}$ in Figures 7.4-7.6.

7.6 Conclusion

In this chapter we have shown that neither shortest path routing (in terms of number of hops; longest hop distance routing), nor smallest hop distance routing may be optimal for a dense mobile ad hoc network in which nodes follow a periphery restricted mobility model. With low average network power, a bit-metres/sec throughput metric may be maximized at the trivial choice of nearest node, with non-overlapping periphery, as the next hop. However, with higher amounts of average network power, we obtain optimal throughput performance at non-trivial periphery hop distances.

Chapter 8

Route Lifetime based Optimal Hop Selection in VANETs on Highway

The main goal of this chapter is to better understand the routing dynamics in Vehicular ad hoc networks (VANETs) that are a special class of Mobile ad hoc networks (MANETs) but exhibit very different behavior from them. We consider the problem of optimal next hop selection in a route between two vehicles for a simple scenario of VANETs on a fast track highway. For a given choice of number of hops between the source and destination, we seek the optimal choice of next hop based on its speed and inter-node distance so as to maximize the expected *route lifetime*. Our analytical model accounts for the randomly changing speeds of nodes (vehicles) over time and hence the optimal choice depends on the dynamics of the stochastic process corresponding to the speed of nodes. Under a Markovian assumption on the process of speed of nodes, we show that the optimal choice of speeds is such that the lifetimes of adjacent links are as close as possible. Explicit expressions for optimal choice of next hop node's speed and inter-node distance are obtained for certain fast track highway scenarios of interest. A monotone variation property of the speed of relay nodes under the optimal policy is proved. These properties have been confirmed with simulations. The optimal policies and their structures can assist in enhancing the performance of existing VANET routing protocols.

Note: The material in this chapter has appeared in [KKA06b]. This work was sponsored by "The Indo-French Centre for the Promotion of Advanced Research" (CEFIPRA).

8.1 Introduction

Vehicular ad hoc networks (VANETs) are being researched upon and developed by many organizations [LH04, ea05a, fle, now, carb, cara, ove, pre, GK00]. Vehicles equipped with communication devices can form VANETs for tasks such as inter-vehicle collision avoidance, road-accident notification, traffic situation update, coordinated driving systems or simply inter-vehicle voice communication. Like Mobile ad hoc networks (MANETs), VANETs do not rely on any fixed infrastructure and instead depend on intermediate *relay* nodes for route establishment protocols and data transmission. However, VANETs tend to exhibit a drastically different behavior from the usual MANETs [BEH04]. High speeds of vehicles, mobility constraints on a straight road and driver behavior are some factors due to which VANETs possess very different characteristics from the typical MANET models. Broadly speaking, four such characteristics have been identified as rapid topology changes, frequent fragmentation of the network, small effective network diameter and limited temporal and functional redundancy [BEH04]. Due to this fundamental behavioral difference between MANETs and VANETs, topology-based routing protocols developed for the former cannot be directly used in the latter. Topology-based protocols are the table-driven *proactive* protocols and on-demand *reactive* protocols [RT99]. For example authors in [JBW05] have shown that TORA (an on-demand protocol) is completely unsuitable for VANETs. Instead, position-based routing protocols such as LAR, DREAM or GPSR [KV98, BCSW98, KK00] that require a-priori knowledge of vehicle's geographic location (from a GPS service) could be used for VANETs for faster route discovery and improved performance. But, position-based routing protocols suffer from geographic routing failures due to presence of *topology holes* [THR03] and authors in [THR03] propose *spatially aware routing* for VANETs to overcome this drawback. The optimality of spatially aware routing however has not been proved and the routing algorithm could be further enhanced in order to improve performance.

A routing protocol usually has three main functions: route discovery, route selection (among various candidate routes discovered) and route maintenance. Once a route from a source to its destination has been discovered and selected, route maintenance must be carried out, in order to track link failures (due to movement of relay nodes) and perform route re-discovery. Route maintenance and re-discovery are expensive in signalling and computation, and hence it is desirable to choose the optimal route comprising links with maximum possible lifetimes during the route selection phase. In this chapter, we propose some optimal route selection criteria from an analytical viewpoint for the simple scenario of a VANET on a straight line highway. Our optimal route selection criteria consist of the optimal choice of next hop based on *maximum route lifetime*. The proposed criteria are not competitors to the route selection methods in any of the existing routing protocols, but rather can be used in conjunction with them. Our goal is not to propose a complete VANET routing protocol along with its implementation aspects and this is outside the scope of our work. We rather focus on gaining an analytical insight into the route lifetime dynamics of a VANET by considering only a simple scenario. Our observations about the structural characteristics of the optimal policies can assist in enhancing the performance of any of the existing VANET routing protocols mentioned earlier. Since the scope of this work is

limited to gaining an analytical viewpoint, we will not address any issues related to *how* the analytical results can be used to enhance the performance of a VANET routing protocol. We claim that our analytical results are simple and can be easily incorporated in the design of a new or re-engineering of an existing VANET routing protocol.

As discussed earlier, route selection in VANETs can be very different from that in MANETs and designing a routing protocol for VANETs can be very complex due to the rapidly changing topology and frequent link breakdowns. In our model, we introduce certain simplifying assumptions (along with their justification) as compared to a real life scenario in order to gain an analytical insight into the dynamics of vehicle mobility and route lifetime in VANETs. The assumptions are well justified with citations and without these assumptions the route lifetime dynamics may be very hard to study and can not be easily understood. For instance, a VANET in city traffic scenario can be very hard to model and our analysis does not hold for this case.

8.1.1 Synopsis and Preview of Contributions

In Section 8.2, we first discuss about the motion dynamics of a set of nodes constituting a route in a VANET and define the model that is used to track the dynamics of the changing inter-node distances and node connectivity. The lifetime or connectivity of a route between a source and a destination may depend on various parameters such as the speeds of intermediate relay nodes, their maximum range, etc. In Section 8.3, we therefore motivate the choice of some of these variable parameters for our optimization objective of maximizing the route lifetime. This is followed by the formulation of an objective function consisting of expected lifetime of links constituting a route (Section 8.4). Our goal is to optimize this objective function so as to obtain an optimal choice of intermediate relay nodes between a source and its destination such that the route lifetime, which is defined as the least of the expected link lifetimes, is maximized. In Section 8.5, an attempt to analytically determine the link lifetimes is made using a Markov chain model for the speeds of the vehicles that are randomly changing lanes on a straight line highway. Explicit expressions for the link lifetimes and either the optimal relay node (next hop) selection policies or some structural intuitions about them are obtained for certain scenarios that are of most concern in a fast track highway traffic situation. In Section 8.6, we compare the derived structural characteristics of the optimal policies with simulation results obtained from a VANET simulator and it is observed that the analytically obtained structural results are consistent with the simulation results.

The contributions of this work are twofold. Firstly, the heuristics and structural characteristics of the next hop selection policies developed in this work can assist in better understanding the dynamics of route lifetime in VANETs. Secondly, the results can serve in enhancing the performance of existing routing protocols for VANETs.

8.2 System Dynamics and Model

The system model that is used to track the dynamics of nodes or vehicles moving on a highway is described here followed by the justification of certain simplifying assumptions.

We consider VANETs on a straight line highway in which a vehicle can establish connectivity only with other vehicles traveling in the *same* direction of its motion. In other words, we consider ad hoc networks formed by only those vehicles that are moving on the

same side of a highway and not in opposite directions. Also note a key point here that we are considering only *fast track* highways with vehicles moving at high speeds and not normal roads with slower moving traffic. We also assume that the traffic flow is smooth and there is no congestion on the highway. Particularly, there are no localized congestion spots and vehicles in all lanes of the highway travel smoothly. In such a scenario, we may safely assume the following model for the speed of nodes. Consider vehicles (nodes) traveling on an infinitely long straight highway with L lanes and moving in the same direction. Each lane i has an associated speed limit s_i. Assume that in a given lane, all the nodes travel with a speed corresponding to the speed limit of that lane. In other words, it is assumed that all nodes move on the highway with a discrete set of speeds which consists of the speed limits of each lane. We follow the convention that $s_1 < s_2 < \ldots < s_L$. When a node transits to an adjacent lane due to driver's natural behavior, it then travels with the speed associated with the new lane.

Remark: It is to be noted here that our analysis works for *any* given discrete set of speeds; the idea of associating a speed to a lane has been introduced only to conceptually visualize the system. In particular, one could also consider a highway with no defined lanes and then all vehicles progressively change speeds (according to some random process) among the set $\{s_1, \ldots, s_L\}$, with all vehicles following this speed change process independently of each other.

8.2.1 Dynamics of Individual Nodes

The process of changing speeds of any individual node due to lane change on the highway is assumed to be an independent, stationary and ergodic stochastic process. We are thus also implicitly assuming that the vehicles do not leave the highway. It is assumed as well that the vehicles do not change their direction of motion since we consider VANETs formed by only those vehicles that are traveling on the same side of highway in the same direction. In this work, we restrict ourselves to the case where the changing speed of any node can be modeled as an irreducible, aperiodic Markov process, taking a finite set of constant values $\{s_1, s_2, \ldots, s_L\}$. We assume that a node continues to move in lane i with an associated speed s_i, $1 \leq i \leq L$ for an exponential amount of time before changing its lane or its speed, equivalently. This time is exponentially distributed with rate μ_i and we denote that a node in lane i transits to another lane j with probability $P_{i,j}$ with $P_{i,i} = 0$. Note that one could have also considered the case $P_{i,i} \neq 0$ but, that would be equivalent to choosing a higher value for μ_i. Even though our analysis holds for generic transition probabilities $P_{i,j}$, we assume the following natural structure on node transitions in our highway scenario: from *state* (or, lane) i, a node can transit only to the states $(i-1) \vee 1$ or $(i+1) \wedge L$. Clearly, from state 1 a node can transit only to state 2 and from state L the only possible transition is to state $L-1$.

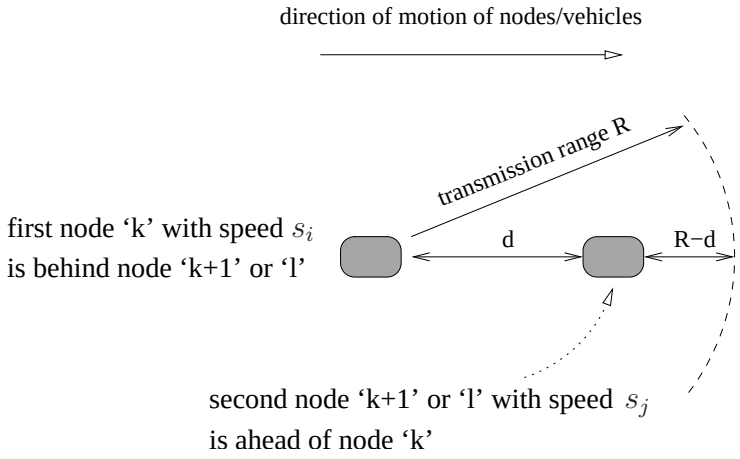


Figure 8.1: Two successive nodes constituting a route path

8.2.2 Placement of Nodes

We assume that node spread-out along the highway is dense in the sense that in a sufficiently small neighborhood of any point on a lane we can always find at least one node on the *same lane*. This is like assuming that the transmission range, R, of a node is significantly large as compared to the distances between two successive nodes in any lane. Most of the results in this work can be extended to the case where we assume that the existence of a node at any point on a lane is itself a stochastic process. However, since we are more interested in the *structural dynamics* of next hop selection, we will assume that this stochastic process is a constant process, i.e., there is always a node at any given point on any lane with the density of this constant process being different for each lane (it will depend on the associated speed s_i). It is also assumed that the width of the lanes on the highway is negligible when compared to the transmission range of mobile nodes along the *length* of highway. We call this assumption as the *straight line communication* assumption.

8.2.3 Evolution of Inter-node Distances and Node Connectivity

Consider any two nodes k and l (node l is ahead of node k; see Figure 8.1) moving in any two lanes i and j with both the nodes moving in the same direction. Assume that at time t the two nodes have speeds $v_k(t)$ and $v_l(t)$, respectively. Since the two nodes are moving and also have their speeds changing with time due to lane change, the distance between these nodes will also vary with time. Let us denote the distance of node l from node k (measured in the direction of motion) at time t as $\tilde{d}_{kl}(t)$. Assume that node k is the source of transmissions meant for node l. We say that a direct link or single hop route exists between nodes k and l as long as $0 \leq \tilde{d}_{kl}(t) \leq R$, where R is the maximum possible transmission range of a node, i.e., a node can successfully transmit up to any range $\leq R$.

The distance between any two adjacent nodes k and $k+1$ (node $k+1$ is ahead of node k) constituting a route, denoted simply by $\tilde{d}_k(t)$, forms a stochastic process that begins with

an initial value of $\tilde{d}_k(0) = d_k$ (or d in general) and whose evolution over time, $\tilde{d}_k(t)$, depends on the initial speeds of the two nodes. We assume that two successive nodes k and $k+1$ in a route remain connected *only* until when $\tilde{d}_k(t)$ takes a value outside the interval $[0, R]$ for the first time (see Figure 8.1). In other words, we consider any two nodes k and l to be connected if node l lies within the maximum transmission range of node k *only* in the direction of motion and not otherwise.

Assume there are M relay nodes in a route between the source and its destination, with the source being the 0^{th} node and the destination as the $(M+1)^{th}$ node. Let v_0 and v_{M+1} be the velocities of the source and destination nodes and let D be the distance between them. For a given value of M, let $d_k, 0 \leq k \leq M$, be the distance between nodes k and $k+1$ (see Figure 8.2). Based on the straight-line communication assumption mentioned in the previous sub-section, we impose that $\sum_{k=0}^{M} d_k = D$ so that the last hop distance $d_M = D - \sum_{k=0}^{M-1} d_k$. For a non-broken route formed by nodes $0, 1, 2, \ldots, M+1$, we require that $0 \leq d_k \leq R$ and let $v_k, 0 \leq k \leq M+1$, be the velocity of the k^{th} node with v_0 and v_{M+1} known in advance. Note that v_ks may take any one of the set of constant values $\{s_1, \ldots, s_L\}$ and there are L^M different possible values that the vector $\underline{v} = (v_1, \ldots, v_M)$ can take.

8.2.4 Justification of Assumptions in the Model

The scope of our model is limited to fast track highways since a lot of commercial applications are expected to be deployed for VANETs on highways. Moreover, only limited focus by the research community has gone into the study of vehicular mobility on fast track highways. One such study has been done in [ea05a]. As for the mobility of nodes, *any* general mobility model can be considered for the movement of vehicles on highway. We limit our scope to the Markovian mobility model which has also been considered by other researchers [NAG04]. The straight line communication assumption in Section 8.2.2 is well justified since widths of highway lanes are usually of the order of 4 meters where as the typical maximum transmission range of a vehicular wireless device is around 200 meters [ea06b]. Also note that since the communication devices operate on car battery which is recharged by the vehicle engine, battery-life of nodes is not an issue in our model.

8.3 Variable Parameters

Consider two tagged nodes, a source and a destination moving in any two (possibly same) lanes, traveling in the same direction. At time 0, these nodes are assumed to be distance D apart. If D is large enough then these nodes may not be able to communicate with each other directly due to their restricted transmission range. Even though the transmitting entities in such networks are not severely battery power constrained, interference considerations dictate an upper bound on the transmit power used and this in turn restricts the transmission range of nodes. Intermediate relay nodes are thus required for the two tagged nodes to form a VANET. However, more than one option (vehicles in front of the source node moving with

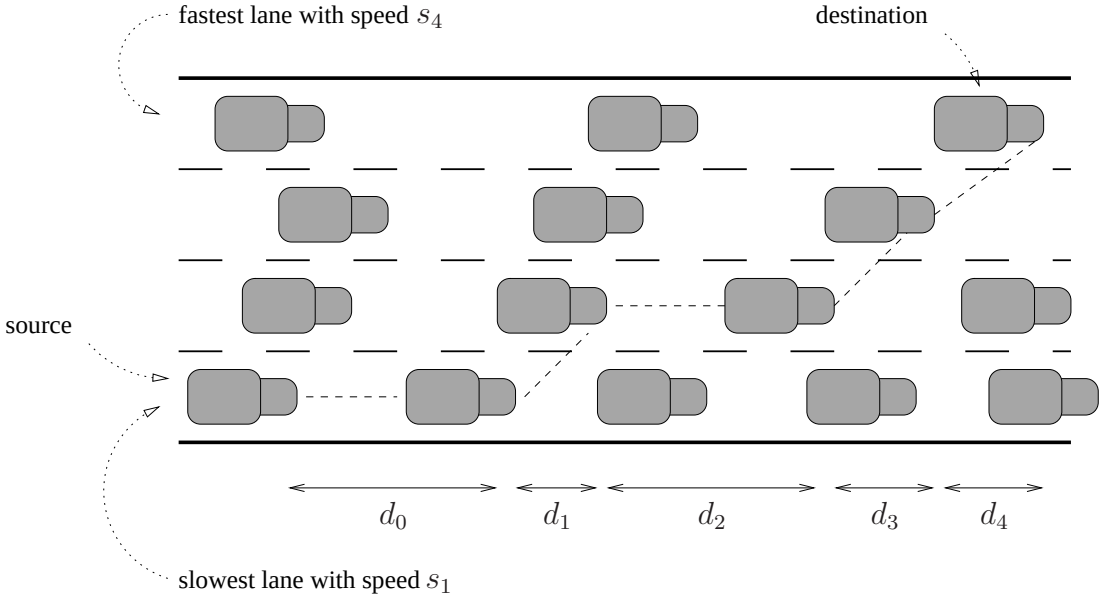


Figure 8.2: Example fast track highway scenario for $M = 4$ and $L = 4$. The links of a route are shown by dashed lines (figure not to scale).

identical or different speeds in the same lane or adjacent lanes, respectively) for the choice of next hop may be available. How would one decide whether to choose the vehicle in the same lane or in adjacent lanes as the next hop. In this work, we address the problem of coming up with an *optimal* choice of next hop (relay node) such that the associated link lifetimes and hence the route lifetime (defined as the least of all link lifetimes) is maximized. In Section 8.2 we have already discussed the constraints under which this decision should be made. Here we emphasize on the fact that making such a decision may not be as simple as it seems at first. An evident reason being that the underlying state space over which the route lifetime has to be optimized is composed of different variable parameters, each representing a component parameter of the overall state or variable space.

Most of the existing routing protocols for MANETs (including spatially aware routing for VANETs) rely on the shortest path (in terms of number of hops between source and destination) approach. The distance covered by a hop in a VANET however is bounded above by the maximum transmission range R which is assumed to be constant. In a dense network (detailed earlier in Section 8.2.2), minimizing the number of hops would result (with a high probability) in a route path in which the distance covered by each hop is around R. This would indeed guarantee a minimum number of hops and this approach though works well in wired networks also, it has the following drawback for VANETs. Due to high mobility and large speeds of nodes in a VANET, the lifetime or connectivity of a route path obtained on the basis of shortest path routing (in terms of number of hops) may be very small. On the other hand, an arbitrarily large number of relay nodes can not be chosen because of the excessive end-to-end packet delivery delay caused and since with increasing number of relay nodes, frequent link failures may occur which may result in very poor network performance.

It is clear now that there is a trade-off between the route lifetime and network perfor-

mance with increasing number of relay nodes and an optimal choice on the number of relay nodes is needed. It is also clear that for a given set of inter-node distances, the route lifetime critically depends on the speeds of the relay nodes. Based on this discussion, following are the possible variable parameters that should be considered for our optimization problem of maximizing the route lifetime:

1. *Optimization over Number of Relay Nodes:* Owing to a constraint on the range over which a node's transmission can be successfully decoded, the number of relay nodes must be more than a certain minimum that depends on the initial distance D between the two tagged nodes. However, one can not choose an arbitrarily large number of relay nodes and an optimal choice on the number of relay nodes is thus necessary.

2. *Optimization over Inter-node Distances:* For a given number of relay nodes, an optimal choice of the inter-node distances is essential to establish a route with maximum lifetime.

3. *Optimization over Speeds of the Intermediate Nodes:* Once the number of relay nodes and the inter-node distances are fixed, we need to decide from which lanes the intermediate relay nodes are to be chosen. In other words, we need to decide what speed the next hop relay node must possess for a maximum possible link lifetime.

In the present work, we assume that nodes (vehicles) are equipped with a GPS receiver and we also assume that the optimal number of relay nodes and the speeds of the source and destination nodes are known in advance. Avoiding relatively large values for number of relay nodes, a roughly optimal choice on number of relay nodes can be fairly approximated from the knowledge of transmission range R and position of source and destination nodes obtained from the GPS receiver. Approximate speeds of source and destination nodes can also be obtained from a GPS service. Modern day GPS receivers are indeed capable of estimating the vehicle's speed. Given this information, we are interested in obtaining the optimal inter-node distances and optimal speeds of relay nodes that result in a maximum possible route lifetime.

8.4 Problem Formulation

In our model described in Section 8.2 we assume a dense vehicular traffic scenario on the highway. Due to this assumption multiple candidate routes may exist for choosing an optimal route. If multiple candidate routes are available then we want to choose the route with the *maximum* lifetime. We are given that there are $M+2$ nodes, indexed $0, 1, \ldots, M+1$, constituting a route (see Figure 8.2). Node 0 is the source node and node $M+1$ is the destination node. Now consider any two successive nodes k and $k+1$ in the route that are distance d_k (or d in general) apart at time zero. Assume also that at time zero, node k is in lane i and node $k+1$ is in lane j such that $\tilde{d}_k(0) = d_k$, $v_k(0) = s_i$ and $v_{k+1}(0) = s_j$. Let $T(d_k, v_k, v_{k+1})$ be the expected time after which the link between these two nodes breaks

(see Section 8.2.3). We refer to the quantity $T(d_k, v_k, v_{k+1})$ as the *link lifetime* of the link between the successive nodes k and $k+1$ in a route.

For a route comprised of $M+1$ links, our problem is to find an optimal inter-node distance assignment denoted by $\underline{d}^* = (d_0, \ldots, d_{M-1})$, and an optimal speed assignment, denoted by $\underline{v}^* = (v_1, \ldots, v_M)$, to the M relay nodes such that maximum route lifetime is attained. We thus seek the optimal distance vector $\underline{d}^*$ and speed vector $\underline{v}^*$ such that the route lifetime defined as *least* of the link lifetimes is maximized. Our optimization problem is therefore the following,

$$\underset{\underline{v},\underline{d}}{\text{Maximize}} \quad \underset{k=0..M}{\text{Min}} \quad T(d_k, v_k, v_{k+1}) \ . \tag{8.1}$$

Instead of solving the above problem directly we further look at a different, parameterized set of objective functions to be optimized. These new objective functions will coincide with the original one of Equation 8.1 when the parameter takes a special value. Solution to any one of these parameterized form of optimization problems will then finally provide us with the solution to the problem of Equation 8.1. To define the parameterized objective functions, we first state the following simple lemma, which we prove here for the sake of completeness.

Lemma 1 *For any* finite dimensional *vector* $\underline{x}$ *with positive elements* x_k, $1 \leq k \leq n$, *if* $||\underline{x}||_\alpha$ *denotes the* $l_\alpha-$*norm, i.e.,*

$$||\underline{x}||_\alpha = \left[\sum_{1 \leq k \leq n} x_k^\alpha \right]^{\frac{1}{\alpha}},$$

and $||\underline{x}||_\infty$ *denotes its* $l_\infty-$*norm, i.e.,*

$$||\underline{x}||_\infty = \max_{1 \leq k \leq n} \{x_k\},$$

then

$$\lim_{\alpha \to \infty} ||\underline{x}||_\alpha = ||\underline{x}||_\infty.$$

Proof. Let, for the given vector $\underline{x}$,

$$k^* \stackrel{\Delta}{=} \arg \max_{1 \leq k \leq n} x_k.$$

Then, for any α,

$$||\underline{x}||_\infty = x_{k^*} \leq ||\underline{x}||_\alpha,$$

and also,

$$||\underline{x}||_\alpha \leq n^{\frac{1}{\alpha}} ||\underline{x}||_\infty.$$

This completes the proof. ■

Theorem 1 *The optimization problem of Equation 8.1 has the same optimizer as that of any of the following optimization problems,*

$$\underset{\underline{v},\underline{d}}{Minimize} \left[\sum_{k=0}^{M} (T(d_k, v_k, v_{k+1}))^{-\alpha}\right]^{\frac{1}{\alpha}}, \tag{8.2}$$

$$\underset{\underline{v},\underline{d}}{Maximize} \left[\sum_{k=0}^{M} (T(d_k, v_k, v_{k+1}))^{-\alpha}\right]^{\frac{-1}{\alpha}}, \tag{8.3}$$

as $\alpha \rightarrow \infty$.

Proof. Firstly, it is obvious to see the equivalence of optimization problems of Equations 8.2 and 8.3, since their objectives are inverse of each other. Now, the optimization problem of Equation 8.1 clearly has the same solution as that of the problem,

$$\underset{\underline{v},\underline{d}}{\text{Minimize}} \quad \underset{k=0..M}{\text{Max}} \quad \frac{1}{T(d_k, v_k, v_{k+1})}. \tag{8.4}$$

Now, for any integer $\alpha > 0$, we can compute the $l_\alpha-$norm of an M-dimensional vector whose k^{th} element is $1/T(d_k, v_k, v_{k+1})$. The $l_\alpha-$norm of this vector, for any given values of v_ks is,

$$\left[\sum_{k=0}^{M} (T(d_k, v_k, v_{k+1}))^{-\alpha}\right]^{\frac{1}{\alpha}}.$$

Since $1/T(d_k, v_k, v_{k+1})$s are strictly positive and bounded quantities, we can invoke Lemma 1 to conclude with the statement of the present theorem. ■

In fact, we can say something more about the relation between the two optimization problems of Equations 8.1 and 8.2 in the following theorem.

Theorem 2 *There exists a* finite α^* *such that the maximizer of optimization problem of Equation 8.1 is identical to that of Equation 8.2* for all values of $\alpha > \alpha^*$.

Proof. Fix a vector $\underline{x}$ with elements $x_k, 1 \leq k \leq n$. Then, from Lemma 1 we know that $\lim_{\alpha\rightarrow\infty} ||\underline{x}||_\alpha = \max_{1\leq k\leq n} x_k$. Now, form a vector $\underline{y}$ whose k^{th} element y_k is the k^{th} maximum among the elements of $\underline{x}$ (so that $y_1 = \max_{1\leq k\leq n} x_k = ||\underline{x}||_\infty$). Since the number of elements in $\underline{x}$ is n, which is finite, the difference $y_1 - y_2 > 0$ (assuming that no two elements of $\underline{x}$ are equal; the case where some of the elements of $\underline{x}$ are equal can also be easily considered.). Since $\lim_{\alpha\rightarrow\infty} ||\underline{x}||_\alpha = \lim_{\alpha\rightarrow\infty} ||\underline{y}||_\alpha \rightarrow y_1$, for any $\epsilon > 0$ there exists a finite $\alpha^*_\epsilon(\underline{x})$ such that , $y_1 - ||\underline{x}||_\alpha < \epsilon$ for all $\alpha > \alpha^*_\epsilon(\underline{x})$.

Now, since the set of possible values of the speed vector $\underline{v}$ over which optimization is carried out is finite (of cardinality L^M), then for a given combination of inter-node distances $\underline{d} = (d_0, \ldots, d_M)$ we can define,

$$\delta \triangleq \min_{s_1\leq v_k, v_l\leq s_L,} \text{POS}(|T(d_k, v_k, v_{k+1}) - T(d_l, v_l, v_{l+1})|),$$

where,

$$\text{POS}(|x-y|) = \begin{cases} |x-y| & \text{if } |x-y| > 0, \\ |x| & \text{otherwise.} \end{cases}$$

Then, $\alpha^* \triangleq \max_{\underline{x}} \alpha^*_\delta(\underline{x}) < \infty$ is the finite quantity that we were seeking. ■

Theorem 2 ensures that there is no discontinuity in the solution of the optimization problem of Equation 8.2 with respect to the solution of Equation 8.1, as $\alpha \to \infty$. Working with the objective function of Equations 8.2 or 8.3 thus has an advantage that we can work with any *finite* value of α, and if this solution is independent of α, we will have obtained the solution to the optimization problem of Equation 8.1. Also note that for numerical computation purposes, the computational complexity of the Min function in Equation 8.1 is $O(M \log M)$, where as that of the summation terms in Equations 8.2 and 8.3 is $O(M)$. Hence, the different forms of objective functions in Theorem 1 could be given preference over that in Equation 8.1 and can be used as per convenience.

8.5 Determining Expected Lifetimes and Optimal Solution

Having done with the problem formulation, here we seek to obtain explicit expressions for the link lifetimes, to be able to explicitly define the objective function of either Equation 8.1 or Equations 8.2 and 8.3. We study the expected lifetime of the connection between two nodes that are d distance apart at time 0 and have speeds s_i and s_j, respectively. We use the notation that a *pair* of nodes k and l is in *state* s_{ij} when node k is in lane i with associated speed s_i and node l is in lane j with associated speed s_j. Here onwards, along with $T(d, v_k, v_l)$, we will also use the notation $T(d, s_{ij})$ for the link lifetimes of any two nodes, interchangeably. With some abuse of notation we use the same notation for the state s_{ij} and the relative speeds between the two nodes $s_{ij} \triangleq s_j - s_i$, interchangeably. Consider a pair of successive nodes k and $k+1$ (or l) forming a link in a route as shown earlier in Figure 8.1. If the second node is within the range R of the first node then using the *straight line communication* assumption mentioned earlier in Section 8.2.2, the expected remaining link lifetime is given by $T(d, s_{ij})$ and we state the following theorem,

Theorem 3 *$T(d, s_{ij})$ satisfies the following renewal-type recursions,*

$$\begin{aligned} s_{ij} > 0 \quad T(d, s_{ij}) = e^{-(\mu_i+\mu_j)\frac{R-d}{s_{ij}}} \frac{R-d}{s_{ij}} + \int_0^{\frac{R-d}{s_{ij}}} (\mu_i+\mu_j) e^{-(\mu_i+\mu_j)u} \Big[u + \\ \sum_l P_{i,l} \frac{\mu_i}{\mu_i+\mu_j} T(d + s_{ij}u, s_{lj}) + \sum_l P_{j,l} \frac{\mu_j}{\mu_i+\mu_j} T(d + s_{ij}u, s_{il}) \Big] du, \end{aligned} \tag{8.5}$$

$$s_{ij} < 0 \quad T(d, s_{ij}) = e^{-(\mu_i+\mu_j)\frac{d}{|s_{ij}|}} \frac{d}{|s_{ij}|} + \int_0^{\frac{d}{|s_{ij}|}} (\mu_i + \mu_j) e^{-(\mu_i+\mu_j)u} \Big[u + \sum_l P_{i,l} \frac{\mu_i}{\mu_i + \mu_j} T(d - |s_{ij}|u, s_{lj}) + \sum_l P_{j,l} \frac{\mu_j}{\mu_i + \mu_j} T(d - |s_{ij}|u, s_{il}) \Big] du, \tag{8.6}$$

$$s_{ij} = 0 \quad T(d, s_{ij}) = \int_0^{\infty} (\mu_i + \mu_j) e^{-(\mu_i+\mu_j)u} \Big[u + \sum_l P_{i,l} \frac{\mu_i}{\mu_i + \mu_j} T(d, s_{lj}) + \sum_l P_{j,l} \frac{\mu_j}{\mu_i + \mu_j} T(d, s_{il}) \Big] du. \tag{8.7}$$

Proof. Since a node continues to stay in lane i for an exponentially distributed amount of time with mean $\frac{1}{\mu_i}$, it follows that a pair of nodes, remains in state s_{ij} for an exponential amount of time with mean $\frac{1}{\mu_i+\mu_j}$ before one of them transits to any of the adjacent lanes. Then for $s_{ij} > 0$, the probability that the second node escapes from the range of the first node before any of them changes lane is given by $e^{-(\mu_i+\mu_j)\frac{R-d}{s_{ij}}}$ and $e^{-(\mu_i+\mu_j)\frac{R-d}{s_{ij}}} \frac{R-d}{s_{ij}}$ is the corresponding expected time to escape. If any one of the nodes makes a transition to another lane before the second node escapes from first node's range, then $P_{i,l}\frac{\mu_i}{\mu_i+\mu_j}$ is the probability that the first node made a transition to lane l and $P_{j,l}\frac{\mu_j}{\mu_i+\mu_j}$ is the probability that the second node made a transition to lane l. Hence with $(\mu_i + \mu_j)e^{-(\mu_i+\mu_j)u}$ as the exponential distribution function and $T(d+s_{ij}u, s_{lj})$ and $T(d+s_{ij}u, s_{il})$ being the remaining expected lifetimes of the first and second nodes respectively, after time u has elapsed, we get Equation 8.5 above. Now, note again that we consider two nodes to be connected if the second node, $k+1$ or l, lies within the maximum transmission range of the first node k, *only* in the direction of motion and not otherwise (see Section 8.2.3). Then for $s_{ij} < 0$, the probability that the first node k overtakes (and thus the link breaks; see Section 8.2.3) the second node, $k+1$ or l, before any of them makes a transition is given by $e^{-(\mu_i+\mu_j)\frac{d}{|s_{ij}|}}$ and $e^{-(\mu_i+\mu_j)\frac{d}{|s_{ij}|}} \frac{d}{|s_{ij}|}$ is the corresponding expected time to overtake. The rest of the terms in Equation 8.6 are obtained with similar arguments as for Equation 8.5 except that the remaining expected lifetimes after the event that one of the nodes transited after time u is given by $T(d - |s_{ij}|u, s_{lj})$ and $T(d - |s_{ij}|u, s_{il})$ for the first and second nodes respectively. Equation 8.7 is a straight forward derivation from either Equation 8.5 or 8.6. ■

Instead of solving the system of Equations 8.5, 8.6 and 8.7 explicitly in its most general form, which is a very complex exercise, we solve it only for some special cases. The main reason for considering only these special cases is that these are the only cases which are of relevance in a fast track highway scenario (see Section 8.2) and solutions for cases other than these can not be applied to fast track highways. Another interesting aspect of considering these special cases is that the results that we obtain for these cases constitute a simple form and provide important insights into the structure of the corresponding optimal distance and

speed policies. Later with the help of simulations we will attempt to validate the obtained structures.

We specified earlier in Section 8.2 that we consider only fast track highways with no congestion spots. In such a situation we assume that drivers do not change lanes too frequently and that a node remains in the same lane for a relatively longer period of time as compared to the lifetime of the link(s) formed by this node.

In the following sub-sections we attempt to solve the link lifetime recursion equations for particular cases of $L = 2$ and $L \geq 3$. The case $L = 1$ is trivial because there is no breakdown of routes, since all nodes are always traveling with the same speed s_1. Firstly, we consider the case $L = 2$ and obtain explicit expressions for the quantities $T(d, s_{ij})$s assuming $\mu_1 = \mu_2$. We then solve the optimization problem of Equation 8.1 directly for $M = 1$ and $\frac{R}{s_{12}} < \frac{1}{2\mu}$. For values of $M > 1$, the global optimization problem can be solved by splitting it into several optimization problems each one of them optimizing over a pair of two adjacent links (i.e., $M = 1$). The solution of these split problems can then be combined to obtain solution to the global optimization problem (for $M > 1$) after taking care of certain coupling issues related to adjacent pairs of links. Second, we consider the case with general values of $L \geq 3$ and $\frac{1}{\mu_i} >> \frac{R}{s_i}$ so that a node remains in lane i for a very long period as compared to the lifetime of a link. For this case we derive only the optimal speed assignment policy, develop some structural heuristics about the optimal speed vector solution to the problem of Equation 8.2 and derive some interesting properties of the optimal speed vector solution to the problem of Equation 8.1 and . Both these cases provide important guidelines on optimally choosing the inter-node distances and speed of next hop.

8.5.1 Case $L = 2$

Consider the case where the number of lanes is $L = 2$. There are only two possible speeds s_1 and s_2 in this case with $s_2 > s_1$. At any time t, let the source have speed $v_0(t)$ and destination have speed $v_{M+1}(t)$. Recall that the processes $\{v_0(t)\}$ and $\{v_{M+1}(t)\}$ are assumed to be independent Markov processes over the state space $\{s_1, s_2\}$. The infinitesimal generator matrix is then given by,

$$\begin{array}{c} \\ s_1 \\ s_2 \end{array} \begin{array}{c} \begin{array}{cc} s_1 & s_2 \end{array} \\ \begin{bmatrix} -\mu_1 & \mu_1 \\ \mu_2 & -\mu_2 \end{bmatrix} \end{array}$$

Here μ_i is the rate of the exponentially distributed sojourn time when the process $\{v_0(t)\}$ (or, $\{v_{M+1}(t)\}$) is in state s_i. We state the following lemma which is easy to prove,

Lemma 2 *If $\mu_1 = \mu_2 = \mu$ then,*

1. *The process of the speed of destination node with respect to the source node, i.e., $\{v_0(t) - v_{M+1}(t)\}$ forms an irreducible periodic Markov process over (finite) state space $\{0, s_{12}, s_{21}\}$ with the mean sojourn time in any state being exponentially distributed with rate 2μ.*

2. The state transition probability matrix is of the form,

$$\begin{array}{c} \\ s_{12} \\ 0 \\ s_{21} \end{array} \begin{array}{c} \begin{array}{ccc} s_{12} & 0 & s_{21} \end{array} \\ \begin{bmatrix} 0 & 1 & 0 \\ 0.5 & 0 & 0.5 \\ 0 & 1 & 0 \end{bmatrix} \end{array}$$

In words, from the states with non-zero relative speed, transition is always to the one with a relative speed of 0 *and from the state with relative speed* 0*, the transition is to either of the other two states, each with probability* 0.5*.*

An important consequence of the observation of Lemma 2 is that the function $T(d, v_k, v_l)$ depends on v_k and v_l only via $v_l - v_k$ with $v_l - v_k \in \{0, s_{12}, s_{21}\}$. We will see later that the observation of Lemma 2 also helps us to compute the function $T(d, 0)$ directly via a simple application of Wald's lemma [Gal98, Chapter 7] without solving any integral equation for $T(d, 0)$. We have the following recursions for $T(d, s_{12})$ and $T(d, s_{21})$ from Equations 8.5 and 8.6,

$$T(d, s_{12}) = e^{-2\mu\frac{(R-d)}{s}}\frac{R-d}{s} + \int_0^{\frac{R-d}{s}} (u + T(d + su, 0))2\mu e^{-2\mu u}du, \quad \text{for } s_{12} > 0, s = s_2 - s_1, \tag{8.8}$$

$$T(d, s_{21}) = e^{-2\mu\frac{d}{s}}\frac{d}{s} + \int_0^{\frac{d}{s}} (u + T(d - su, 0))2\mu e^{-2\mu u}du, \quad \text{for } s_{21} < 0, s = s_2 - s_1. \tag{8.9}$$

For obtaining $T(d, 0)$ we follow the approach of *random walks*. Recall that $T(d, 0)$ is the expected time for which the distance between the two nodes remains in the interval $[0, R]$, starting with distance d apart and 0 relative speed. Clearly, the distance between the nodes can change only when the relative speed between the two nodes is non-zero. The periods of zero and non-zero relative speed alternate and the instants of the beginning of zero relative speed form renewal instants for the relative speed process.

Consider a particle starting at point d. As in random walks, in each time unit the particle moves to either left or right (each with probability $\frac{1}{2}$) and moves by an exponentially distributed amount. The mean of the jump size is $\frac{1}{m}$ where $m = 2\mu$. Let $S_n, n \geq 1$ be the position of particle just after n^{th} jump. It is then seen that,

$$S_n = d + \sum_{i=1}^{n} X_i,$$

where $|X_i|$s are exponentially distributed random variables (with rate m) corresponding to the jump sizes (see Figure 8.3). X_i takes negative and positive values with probability $\frac{1}{2}$ each. Let N be the random variable corresponding to the number of jumps required by the

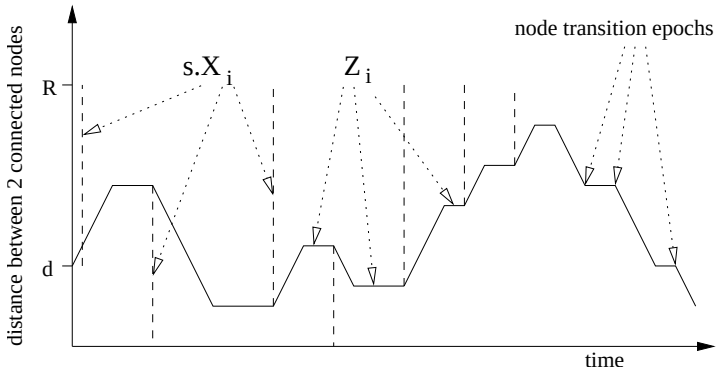


Figure 8.3: Random walk model for 2 successive nodes in a route

particle to exit the interval $[0, R]$ with $R > d$. Let q be the probability that the position of particle becomes greater than R. The treatment of [Gal98, Chapter 7] can then be used to show that, since $|X_i|$s are independent and identically distributed,

$$E\sum_{i=1}^{N}|X_i| = E[N]E[|X_1|] \qquad \text{and} \qquad E[(S_N - d)^2] = E[N]E[|X_1|^2].$$

To compute $E\sum_{i=1}^{N}|X_i|$, we need $E[N]$ which is derived from the second relation above as follows. Since $|X_i|$ are exponentially distributed, we can invoke the memoryless property of exponential distribution to see that,

$$S_N - d = \begin{cases} R - d + Y & w.p.\ q \\ -d - Y & w.p.\ 1-q \end{cases}, \tag{8.10}$$

where Y is an exponentially distributed random variable with rate m. Hence,

$$\begin{aligned} E[(S_N - d)^2] &= E[N]E[|X_1|^2] \\ &= qE[(R - d + Y)^2] + (1-q)E[(d+Y)^2] \\ &= (d^2 + E[Y^2] + 2dE[Y]) + q(R-2d)[R + 2E[Y]]. \end{aligned}$$

From the above expression, since $E[Y] = E[X_1] = \frac{1}{m}$, we can obtain $E[N]$ if we know q. We now obtain q using the fact that [Gal98],

$$E[S_N - d] = E\sum_{i=1}^{N} X_i = E[N]E[X_1] = 0.$$

Now, using the possible values of $S_N - d$ mentioned in Equation 8.10,

$$E[S_N - d] = 0 = q(R - d + E[Y]) + (1-q)(-d - E[Y]),$$

hence, $q = \frac{d+E[Y]}{R+2E[Y]} = \frac{md+1}{mR+2}$, where we have used the fact that $E[Y] = \frac{1}{m}$. From this value of q we get (using the fact that $E[Y] = \frac{1}{m}$ and $E[X_1^2] = \frac{2}{m^2}$),

$$E[N] = ((R-2d)(d+\frac{1}{m}) + d^2 + \frac{2}{m^2} + 2\frac{d}{m})\frac{m^2}{2}.$$

Assuming a normalized speed of $s = 1$ with out loss of generality, it is seen that,

$$T(d,0) = E[\sum_{i=1}^{N}(Z_i + |X_i|) - (\sum_{i=1}^{N} X_i - (R-d))I_{\{R-d<\sum_{i=1}^{N} X_i\}} - (-d - \sum_{i=1}^{N} X_i)I_{\{-d>\sum_{i=1}^{N} X_i\}}],$$

where Z_is are also exponentially distributed random variables with rate m and they correspond to the time when the distance between the two nodes does not change because of zero relative speed (see Figure 8.3). Using the memoryless property of exponential distribution, we see that if,

$$I_{\{R-d<\sum_{i=1}^{N} X_i\}} = 1,$$

then $\sum_{i=1}^{N} X_i - (R-d)$ is (independent and) exponentially distributed with rate m. Similarly if,

$$I_{\{-d>\sum_{i=1}^{N} X_i\}} = 1,$$

then $(-d - \sum_{i=1}^{N} X_i)$ is exponentially distributed with rate m. Also,

$$E[I_{\{R-d<\sum_{i=1}^{N} X_i\}}] = q = 1 - E[I_{\{-d>\sum_{i=1}^{N} X_i\}}].$$

Hence,

$$T(d,0) = \frac{2E[N]}{m} - \frac{1}{m} = (R-d)md + R + \frac{1}{m}.$$

We can thus write explicit expressions for the link lifetimes from Equations 8.8 and 8.9 as,

$$T(d,1) = md(R-d) + 2(R-d) \qquad \text{and} \qquad T(d,-1) = md(R-d) + 2d.$$

8.5.1.1 Optimal speed vector solution to the optimization problem of Equation 8.1 for the case of $\frac{R}{s} < \frac{1}{m}$

We consider the case where $\frac{R}{s} < \frac{1}{m}$. This scenario is of relevance since in a fast track highway traffic, a node remains in its lane for an average time greater than the lifetime of the link formed by this node and its next hop. Assuming $s = 1$ with out loss of generality, it is easy to see that for this case,

$$T(d,1) \leq T(d,0), \quad d \leq R \qquad \text{and} \qquad T(d,-1) \leq T(d,0), \quad d \leq R.$$

Now, let the distance between the source and destination be D such that $R < D < 2R$. Thus one needs at least two hops or equivalently one intermediate relay node for communication. Let the number of intermediate relay nodes be $M = 1$. Also, let the speed of destination with respect to the source be $s = 1$ (i.e., $s_{ij} > 0$). Here we find the optimal speed assignment for a fixed inter-node distance assignment and then later in the next paragraph, we optimize over inter-node distances. So, for a given distance d between the source and the intermediate node, the decision is to be made on the speed v of the only intermediate relay node. Let the expected lifetime of the link between source and relay node be $L_1(v)$ and that of the link between relay node and destination be $L_2(v)$. The values of these quantities are then given by the following table,

v	$L_1(v)$	$L_2(v)$
s_1	$T(d,0)$	$T(D-d,1)$
s_2	$T(d,1)$	$T(D-d,0)$

Now,

$$T(D-d,0) - T(d,0) = m(D-R)(2d-D)$$

and

$$T(D-d,1) - T(d,1) = (m(D-R)+2)(2d-D).$$

Hence from Equation 8.1, for $d > \frac{D}{2}$,

$$\arg \max_{v \in \{s_1, s_2\}} (L_1(v) \wedge L_2(v)) = s_1$$

and for $d < \frac{D}{2}$,

$$\arg \max_{v \in \{s_1, s_2\}} (L_1(v) \wedge L_2(v)) = s_2.$$

Thus, we see that by the solution to the optimization problem of Equation 8.1, for $s_{ij} > 0$ *the speed of the intermediate node should be the same as the speed of the farther node, i.e.,* s_1 *for* $d > \frac{D}{2}$ *and* s_2 *for* $d < \frac{D}{2}$. Similarly, it is easy to derive that when the source node has speed s_2 and destination node has speed s_1 (i.e., $s_{ij} < 0$) *the speed of the intermediate node should be the same as the speed of the nearer node, i.e.,* s_2 *for* $d > \frac{D}{2}$ *and* s_1 *for* $d < \frac{D}{2}$.

8.5.1.2 Optimal distance vector solution to the optimization problem of Equation 8.1 for the case of $\frac{R}{s} < \frac{1}{m}$

As earlier, let the distance between the source and destination be D such that $R < D < 2R$. Let the number of intermediate relay nodes be $M = 1$ and without loss of generality, let the speed of destination with respect to the source be normalized with $s = 1$. Then for $d > \frac{D}{2}$, it has been shown in the previous paragraph that the optimal speed selection is s_1. Now, it can be shown after simple algebra that for $T(d,0) < T(D-d,1)$ to hold we must have,

$$d > \frac{D(m(D-R)+2) - R + \frac{1}{m}}{2(m(D-R)+2)}. \tag{8.11}$$

Let us denote the RHS of Inequality 8.11 by K. Now, if m is such that $K < R$ (and $\frac{D}{2} < K$) then for $d < K$ we have $\min(T(d,0), T(D-d,1)) = T(D-d,1)$. For obtaining optimal d^* we differentiate $T(D-d,1)$ w.r.t. d and equate it to zero, from which we get,

$$d^* = D - \left(\frac{R}{2} - \frac{1}{m}\right).$$

For $d > K$ we have $\min(T(d,0), T(D-d,1)) = T(d,0)$. For obtaining optimal d^* we differentiate $T(d,0)$ w.r.t. d and equate it to zero to get,

$$d^* = \frac{R}{2}.$$

For $d < \frac{D}{2}$, it has been shown in the previous paragraph that the optimal speed selection is s_2. It can be shown after simple algebra that for $T(d,1) < T(D-d,0)$ to hold we must have,

$$d > \frac{m(D-R)D + R - \frac{1}{m}}{2(m(D-R)+2)}. \tag{8.12}$$

Denote the RHS of Inequality 8.12 by K'. Now, if m is such that $K' > D-R$ (and $K' < \frac{D}{2}$) then for $d > K'$ we have $\min(T(d,1), T(D-d,0)) = T(d,1)$. For obtaining optimal d^* we differentiate $T(d,1)$ w.r.t. d and equate it to zero. We thus get,

$$d^* = \frac{R}{2} - \frac{1}{m}.$$

For $d < K'$ we have $\min(T(d,1), T(D-d,0)) = T(D-d,0)$. For obtaining optimal d^* we differentiate $T(D-d,0)$ w.r.t. d and equate it to zero and get,

$$d^* = D - \frac{R}{2}.$$

8.5.2 Case $L \geq 3$

8.5.2.1 Some properties of the solution to the optimization problem of Equation 8.2 with $\frac{R}{s_i} << \frac{1}{\mu_i}$

Here we derive some structural properties of the solution to the optimization problem of Equation 8.2 for the particular case of interest when $\frac{R}{s_i} << \frac{1}{\mu_i}$ so that a node stays in its lane for a time much greater than its link lifetimes. Assume any value of $L \geq 3$ and consider the link lifetime dynamics of two nodes in lanes i and j that are separated by an initial distance $d < R$. It can be easily seen that for $i \neq j$ and $\frac{R}{s_i} << \frac{1}{\mu_i}$, Equations 8.5 and 8.6 can be rewritten as,

$$T(d, s_{ij}) = \frac{R-d}{s_{ij}} \quad \forall\ s_{ij} > 0 \qquad \text{and} \qquad T(d, s_{ij}) = \frac{d}{s_{ij}} \quad \forall\ s_{ij} < 0, \tag{8.13}$$

respectively. If both the nodes are initially in the same lane, then the distance between these two nodes remains constant till the instant when any one of them changes lanes, so that $\forall s_{ii} = 0$,

$$T(d, s_{ii}) = \frac{1}{2\mu_i} + \sum_{j \neq i} \frac{P_{i,j}}{2}(T(d, s_{ij}) + T(d, s_{ji})).$$

Now consider a route consisting of M intermediate nodes so that the source and destination nodes have speeds v_0 and v_{M+1}, respectively and let the distance vector $\underline{d} = (d_0, \ldots, d_M)$ be fixed. For obtaining the speed vector $\underline{v} = (v_1, \ldots, v_M)$ that maximizes the route lifetime, we consider minimizing the objective function of Equation 8.2. The objective function of Equation 8.2 for any given value of α is given by,

$$\left[\sum_{k=0}^{M} \left[\frac{1}{T(d_k, v_k, v_{k+1})}\right]^{\alpha}\right]^{\frac{1}{\alpha}}.$$

Define $f_k(x, y) = \frac{1}{T(d_k, v_k, v_l)}$ such that $x = v_k$ and $y = v_l$. Note that the sub-index k in $f_k(x, y)$ is due to the fact that $T(d_k, v_k, v_l)$ can be different depending on the value of v_k, as seen in Equations 8.13. Clearly, if it is allowed to choose an intermediate relay node with any arbitrary continuum speed x (thus not restricting to the discrete set of speeds $s_i, 1 \leq i \leq L$), the following condition should be satisfied for an optimal speed assignment to this relay node,

$$\frac{d}{dx}\left[(f_{k-1}(v_{k-1}, x))^{\alpha} + (f_k(x, v_{k+1}))^{\alpha}\right]^{\frac{1}{\alpha}} = 0.$$

This implies in particular that,

$$\frac{f_{k-1}(v_{k-1}, x)}{f_k(x, v_{k+1})} = \left[-\frac{df_k(x, v_{k+1})}{df_{k-1}(v_{k-1}, x)}\right]^{\frac{1}{\alpha-1}}.$$

It is easy to show that $\frac{df_k(x, v_{k+1})}{df_{k-1}(v_{k-1}, x)} < 0$. Taking $\alpha \to \infty$, we see that we need,

$$\frac{f_{k-1}(v_{k-1}, x)}{f_k(x, v_{k+1})} = 1,$$

implying that the lifetimes of adjacent links should be *equalized* in order to optimize the objective function of Equation 8.2. Note that this is only a necessary condition and not a sufficient one, i.e., not all configurations that result in equal lifetimes of adjacent links will be the solution of the optimization problem under consideration. However, *any solution of the optimization problem will satisfy the above mentioned property.* This property also holds for the case where the speeds of the relay nodes are restricted to a finite discrete set such as $\{s_1, \ldots, s_L\}$. In such a case however, it is obvious that exact equalization of the lifetimes of adjacent links will not be achieved due to the lack of the choice of continuum set of speeds for the relay nodes. We discuss this issue next.

Let $\underline{v}^* = (v_1^*, \ldots, v_M^*)$ be an optimal speed assignment of relay nodes for the problem of Equation 8.2, for given values of M, v_0, v_{M+1} and fixed $\underline{d}$. Assume that $\underline{v}^*$ is the optimal speed assignment for all values of $\alpha > \alpha^*$ (so that the optimal policy remains unchanged, as proved in Theorem 2). If the speed of an l^{th} intermediate node of an optimal route is changed from v_l^* to some fixed value v_l, then for all values of $\alpha > \alpha^*$ we obtain the following relation after some simple algebra,

$$f_{l-1}(v_{l-1}^*, v_l^*) \leq \left[f_{l-1}(v_{l-1}^*, v_l)^\alpha + f_l(v_l, v_{l+1}^*)^\alpha - f_l(v_l^*, v_{l+1}^*)^\alpha\right]^{\frac{1}{\alpha}}.$$

Restricting ourselves to only odd values of $\alpha > \alpha^*$, we note that the right hand side of the above relation is the $l_\alpha-$norm of the three-dimensional vector $(f_{l-1}(v_{l-1}^*, v_l), f_l(v_l, v_{l+1}^*), -f_l(v_l^*, v_{l+1}^*))$. Since the third element of this vector is non-positive and the relation is valid for all values of $\alpha > \alpha^*$, the right hand side of the relation above converges to $\max(f_{l-1}(v_{l-1}^*, v_l), f_l(v_l, v_{l+1}^*))$. This is valid since Lemma 1 is independent of whether the limit is achieved for only odd values of α. Hence for any allowed choice of v_l we get,

$$f_{l-1}(v_{l-1}^*, v_l^*) \leq \max(f_{l-1}(v_{l-1}^*, v_l), f_l(v_l, v_{l+1}^*)).$$

Similarly we get,

$$f_l(v_l^*, v_{l+1}^*) \leq \max(f_{l-1}(v_{l-1}^*, v_l), f_l(v_l, v_{l+1}^*)).$$

In particular,

$$f_{l-1}(v_{l-1}^*, v_l^*) \leq \min_{v_l} \max(f_{l-1}(v_{l-1}^*, v_l), f_l(v_l, v_{l+1}^*)).$$

Since $f_{l-1}(\cdot,\cdot)$ and $f_l(\cdot,\cdot)$ are convex functions of v_l, their point-wise maximum is also a convex function of v_l. Hence, the minimum of the point-wise maximum is attained at one of the intersection points of $f_{l-1}(\cdot,\cdot)$ and $f_l(\cdot,\cdot)$ where they attain equal values. This is indeed the case in our situation because the two functions under consideration are actually piecewise linear and intersect at at least one point so that one of these points of intersection forms the minimum.

Moreover, it is easy to show in the present case that the minimizer under consideration will lie between v_{l-1}^* and v_{l+1}^*. If there is no other allowed speed in between these two speeds, then the minimizer is one of these two itself. This observation further simplifies the search for an optimizer since now the speed of the intermediate relay node is known to be lying between the speeds of the two end nodes $l-1$ and $l+1$. This *monotone* transition of the speeds of intermediate relay nodes in an optimal policy is confirmed by our simulation study in Section 8.6.

8.5.2.2 Generic formula for the choice of optimal speed of relay nodes when $\frac{R}{s_i} << \frac{1}{\mu_i}$

Here we derive a generic formula for the choice of optimal speed of a relay node (solution to the optimization problem of Equation 8.1) for the particular case of interest when $\frac{R}{s_i} << \frac{1}{\mu_i}$

so that a node stays in its lane for a time much greater than its link lifetimes. Assume any value of $L \geq 3$ and consider the link lifetime dynamics of two nodes in lanes i and j that are separated by an initial distance $d \leq R$. As earlier, it can be shown that for $i \neq j$ and $\frac{R}{s_i} << \frac{1}{\mu_i}$, Equations 8.5 and 8.6 can be rewritten as,

$$T(d, s_{ij}) = \frac{R-d}{s_{ij}} \quad \forall\ s_{ij} > 0 \qquad \text{and} \qquad T(d, s_{ij}) = \frac{d}{s_{ij}} \quad \forall\ s_{ij} < 0.$$

Deviating slightly from the earlier notational convention, let the speed of source and destination nodes be s_S and s_D and let them be separated by a distance D. For a 2-hop communication we have $M = 1$. Now, if we assume continuum set of relay node speeds, then for a fixed distance vector the relay node speeds should be such that the link lifetimes of both links are equal (as seen in the previous paragraph). Therefore, if s denotes the continuum speed of the relay node and $R < D < 2R$ then from $\frac{R-d}{s-s_S} = \frac{R-D+d}{s_D-s}$ we get,

$$s = \frac{s_D(R-d) + s_S(R-D+d)}{2R-D}. \tag{8.14}$$

This shows that the relay node's optimal speed is a *convex combination* of speeds of source and destination for a two hop route. In particular, at $d = R$ we have $s = s_S$ and at $d = D-R$ we have $s = s_D$. To approximate this continuum speed s with one of the available discrete speeds $\{s_1, \ldots, s_L\}$, we take the following approach. Let s_i be the best approximation to s and let the expected lifetimes of the two links be denoted by $L_1(v)$ and $L_2(v)$, where v is the speed of the relay node.

Case $s < s_i$ If $s < s_i$ then s can either be approximated by s_i or s_{i-1}. For the choice of s_i we have $L_1(s_i) = \frac{R-d}{s_i-s_S}$, $L_2(s_i) = \frac{R-D+d}{s_D-s_i}$ and $L_1(s_i) < L_2(s_i)$. Similarly, we also have $L_1(s_{i-1}) > L_2(s_{i-1})$. Therefore, for s_i to satisfy the optimality of Equation 8.1 we must have $L_1(s_i) > L_2(s_{i-1})$ which results in the following condition on d,

$$d < \frac{R(s_D - s_{i-1}) + (D-R)(s_i - s_S)}{s_D - s_{i-1} + s_i - s_S}.$$

Case $s > s_i$ As in the previous case, with $s > s_i$, s can be approximated by s_{i+1} or s_i. For the choice of s_{i+1} we have $L_1(s_{i+1}) < L_2(s_{i+1})$ and for s_i we have $L_1(s_i) > L_2(s_i)$. Now, for s_i to satisfy the optimality of Equation 8.1 we should have $L_1(s_{i+1}) < L_2(s_i)$ which gives the bound,

$$d > \frac{R(s_D - s_i) + (D-R)(s_{i+1} - s_S)}{s_D - s_i + s_{i+1} - s_S}.$$

Combining the two aforementioned cases and generalizing for any $L \geq 3$, following is a generic formula for the choice of optimal speed of a relay node. Choose s_i as the speed of the intermediate node, if the initial distance, d, between the source and the relay

node lies between the upper and lower bounds obtained above. In other words, the relay node's optimal speed, s, given by Equation 8.14 should be approximated by a discrete speed $s_i \in \{s_1, \ldots, s_L\}$ if d is such that,

$$d \in \left[\frac{R(s_D - s_i) + (D-R)(s_{(i+1)\wedge L} - s_S)}{s_D - s_i + s_{(i+1)\wedge L} - s_S}, \frac{R(s_D - s_{(i-1)\vee 1}) + (D-R)(s_i - s_S)}{s_D - s_{(i-1)\vee 1} + s_i - s_S} \right].$$

Note that here $s_S < s_D$ and s_S and s_D can take any values from $s_1, \ldots, s_L$. For $M = 1$, if we assume continuum set of intermediate node speeds as earlier, then for a fixed distance vector, the intermediate node speeds should be such that the link lifetimes of both links are equal (as seen in the previous paragraph). This implies (it can be shown after some algebra) that the optimal link lifetimes are independent of the choice of inter-node distances, thus implying a *non-unique solution* for the choice of relay node speeds.

8.6 Simulation Study of a VANET

In order to validate the analysis, we have developed a VANET simulator in Java. With this simulator we study and validate only the structural characteristics of the optimal speed assignment policies assuming a fixed inter-node distance assignment. Due to the limitations of this simulator, we do not study the optimal inter-node distance solution. The simulator is based on the model and assumptions proposed in Section 8.2 and is implemented such that the nodes move in their lanes in a discrete time space. A node in lane i transits to any of the adjacent lanes at the beginning of a time slot of length 0.1 seconds and the transition takes place with probability $1 - p_i$. Given that a node in lane i transits, the transition is to lane j with the same lane transition probability, $P_{i,j}$, which was mentioned earlier in Section 8.2. For our simulations we consider the probabilities $p_1 = \cdots = p_L = p$ to be identical for all the lanes. The probability p is related to μ_i by the relation $\frac{1}{1-p} = \frac{0.1}{\mu_i}$ and for $\frac{R}{s_i} << \frac{1}{\mu_i}$ it is equivalently said that $p \to 1$.

The simulator computes the expected link lifetimes of all possible links by exhaustively simulating over all possible speed assignments $\underline{v}$ of the relay nodes for a given scenario of M relay nodes, L lanes, the inter-node distance vector $\underline{d}$, speeds of source and destination v_0 and v_{M+1}, transmission range R, source and destination separation D and the probability p. Once an exhaustive set of link lifetimes for all possible values of $\underline{v}$ is obtained by employing this brut-force method, either of the objective functions of Equation 8.1 or 8.2 is applied over this set to obtain an optimal speed assignment policy.

8.6.1 Simulation scenarios

A car battery operated mobile device has a typical transmission range of around 200 meters. We therefore consider the possible space of inter-node distances in a VANET to vary from 140 to 200 meters and transmission range, R, of 200 meters is considered for all the simulation scenarios. It has been shown in a previous work [GTB99] that large number of hops in an ad hoc network can significantly degrade the TCP throughput performance. Based on this

result, we consider the number of hops $(M+1)$ to vary from 2 to 7 only and the distance between the source and destination nodes is varied from 800 to 1200 meters. The value of p is taken to be, 0.995 when $\frac{R}{s_i} << \frac{1}{\mu_i}$ does not hold and 0.99999 when $\frac{R}{s_i} << \frac{1}{\mu_i}$ does hold (see Appendix). We perform simulations for the number of lanes, L, varying from 2 to 6 and unless explicitly stated in the discussion on the simulation results, the associated speeds are taken as shown in the table that follows,

l	1	2	3	4	5	6
$s_l\ (m/s)$	14	17	22	30	42	55
$\approx s_l\ (km/hr)$	50	60	80	110	150	200

Simulations were carried out for a large set of fast track highway scenarios for which heuristics have been developed in the previous sections using analytical models. In the following part of this section we discuss the various scenarios that were simulated and compare their results with the structural results and heuristics obtained analytically.

1. *Existence of α^* of Theorem 2*: In Figure 8.4 we consider the scenario $L=4$, $M=4$, $v_0=s_1$, $v_{M+1}=s_3$, $p=0.995$, $D=800m$ and $\underline{d}=(162,153,155,158,172)$. We plot the various optimal policies obtained using both the objective functions of Equation 8.1 and 8.2 and we clearly see that the optimal policies obtained by Equation 8.2 coincide with that obtained by Equation 8.1 for $\alpha \geq 40$. Figure 8.5 shows similar characteristics for another scenario of $L=5$, $M=5$, $v_0=s_1$, $v_{M+1}=s_5$, $p=0.99$, $D=1000m$ and $\underline{d}=(168,162,167,166,163,174)$ with the optimal policies coinciding for $\alpha \geq 100$. This validates our Theorem 2 on the existence of α^*.

2. *Structure of optimal policy for $L=2$ (Section 8.5.1.1)*: Figure 8.6 shows plots of optimal policies obtained from Equation 8.1 for $L=2$, $M=1$, $p=0.995$, $D=300m$ and $\underline{d}=(158,142)$. Figure 8.7 shows a similar plot for $L=2$, $M=5$, $p=0.995$, $D=1000m$ and $\underline{d}=(162,164,165,161,170,178)$. The former figure clearly illustrates that under optimality, an intermediate node is assigned the speed of the farther node for $s_{ij}>0$ and that of the nearer node for $s_{ij}<0$. The latter figure shows that a next hop in an optimal route goes across lanes of different speeds (i.e., a lane transition for the choice of next hop occurs) when the expected remaining lifetimes (not shown here) are the maximum possible.

3. *Lifetime Equalization over continuum set of speeds for $L \geq 3$ and $\frac{R}{s_i} << \frac{1}{\mu_i}$ (Section 8.5.2)*: In Figure 8.8 we consider the scenario $L=3$, $M=1$, $v_0=s_3=22m/s$, $v_2=s_1=14m/s$, $p=0.99999$, $D=300m$ and $\underline{d}=(143,157)$. In order to be able to validate the equalizing structure obtained in Section 8.5.2 over a continuum set of intermediate node speeds, we vary the speed associated with lane 2 from $14m/s$ to $30m/s$ in small steps of $1m/s$ and plot the link lifetimes for some of these speeds of lane 2 separately. This allows the only intermediate node 1 to be assigned one of the quasi-continuum set of speeds for the optimization problem of Equation 8.2.

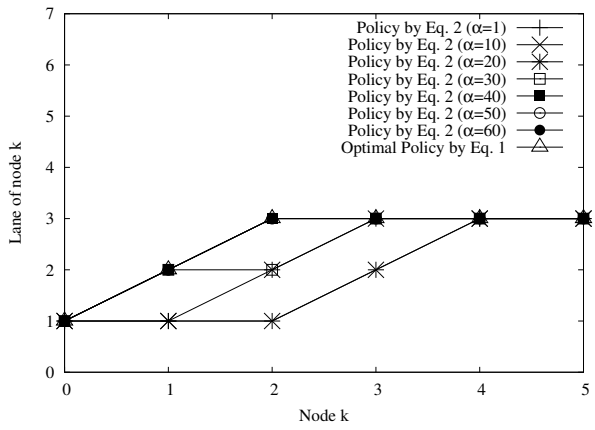


Figure 8.4: Policies showing existence of α^* of Theorem 2

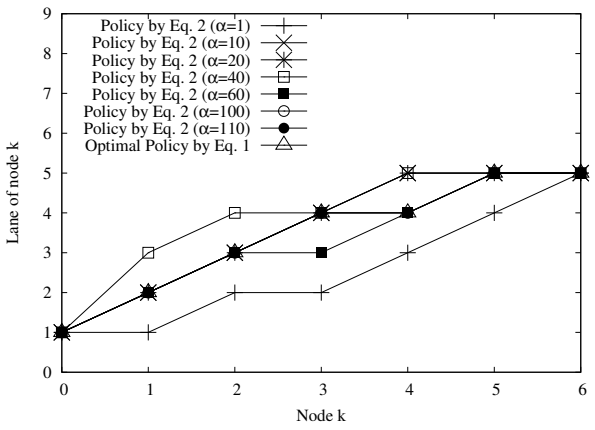


Figure 8.5: Policies showing existence of α^* of Theorem 2

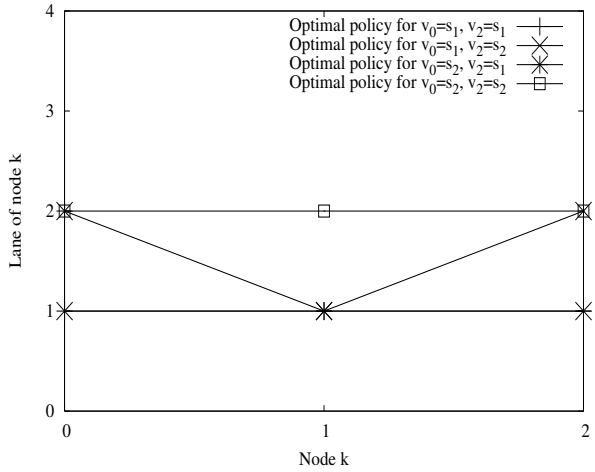


Figure 8.6: Structure of the optimal policy for $L = 2$

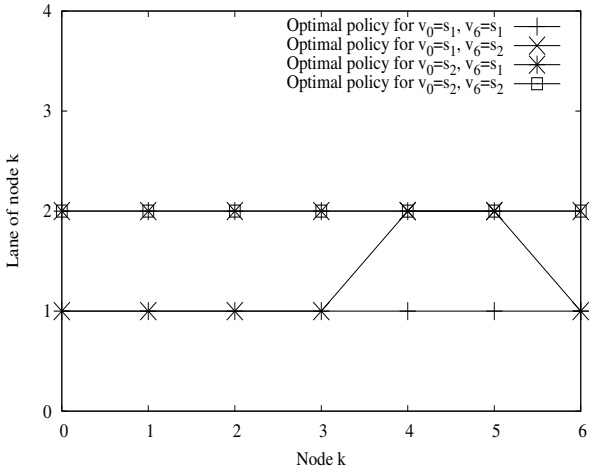


Figure 8.7: Structure of the optimal policy for $L = 2$

It is seen in the figure that under optimality, for varying values of v_1, the optimal lifetimes of the links between node 0 and 1 and node 1 and 2 are different. However at $v_1 = 23m/s$ the optimal lifetimes of the two adjacent links are almost equal thus confirming our result obtained in Section 8.5.2 that the lifetimes of adjacent links should be equalized in order to optimize the objective function of Equation 8.2. In fact, it can be observed that we obtain the maximum of the least of the two lifetimes for speed $v_1 = 23m/s$ and the optimal lifetimes obtained for other values of v_1 are not truly optimal because of the unavailability of the choice of speed $23m/s$ in those scenarios.

4. *Link Lifetime Equalization under the optimal policy for* $L \geq 3$ *and* $\frac{R}{s_i} << \frac{1}{\mu_i}$: To strengthen our validation of the lifetime equalization structure discussed in the previous note, in Figure 8.9, we plot the link lifetimes for a subset of the all possible speed

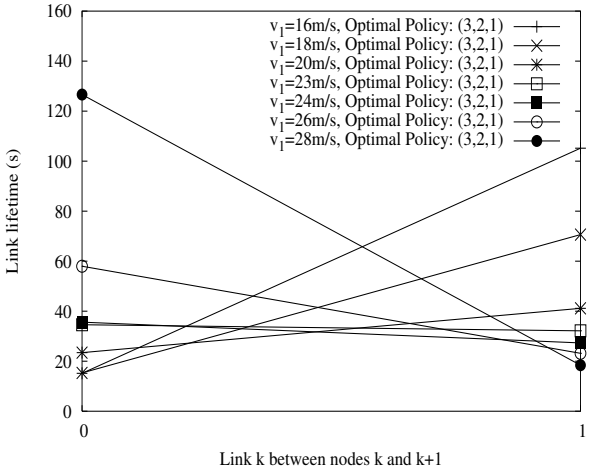


Figure 8.8: Link lifetime equalization over continuum of speeds

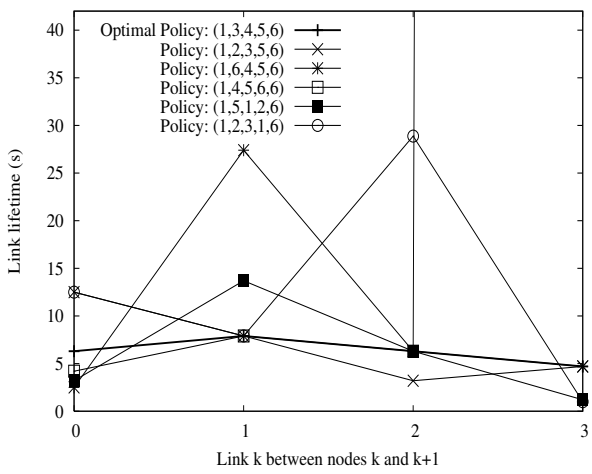


Figure 8.9: Link lifetime equalization under the optimal policy

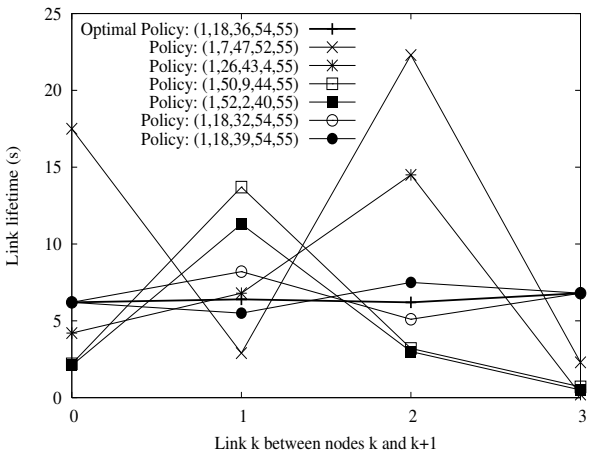


Figure 8.10: Link lifetime equalization under the optimal policy

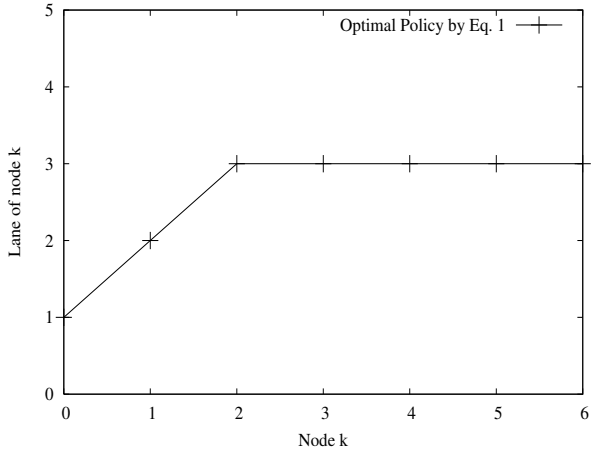


Figure 8.11: Monotone speed transitions for $L \geq 3$ and $\frac{R}{s_i} << \frac{1}{\mu_i}$

combinations (including the optimal combination) for a given particular scenario. It is clearly seen in this figure that at the optimal policy the link lifetimes are equalized and the maximum of the least lifetimes is obtained at the optimal policy, by definition. Figure 8.10 illustrates the same result for another set of given scenario. Figure 8.9 uses the scenario $L = 6$, $M = 3$, $v_0 = 14m/s$, $v_4 = 30m/s$, $p = 0.99999$, $D = 700m$ and $\underline{d} = (160, 175, 180, 185)$. Figure 8.10 uses the scenario $L = 55$, $M = 3$, $v_0 = 14m/s$, $v_4 = 30m/s$, $p = 0.99999$, $D = 700m$ and $\underline{d} = (169, 166, 167, 198)$.

5. *Monotone speed transitions for $L \geq 3$ and $\frac{R}{s_i} << \frac{1}{\mu_i}$ (Section 8.5.2)* : The phenomenon of monotone transition of speeds from a given node to any one of the next hops in a route, is observed in all of the figures discussed until now. Specifically, Figure 8.11 shows the optimal policy assignment for the scenario $L = 5$, $M = 5$, $v_0 = s_1$,

$v_6 = s_3$, $p = 0.99999$, $D = 1000m$ and $\underline{d} = (168, 162, 167, 166, 163, 174)$ where all the intermediate nodes have speeds v_i such that $v_0 \leq v_i \leq v_6$. This validates the corresponding analytical result stated in Section 8.5.2.

8.7 Conclusion

Designing efficient routing protocols for VANETs is quite a challenging task owing to the fast speed of nodes and mobility constraints on the movement of nodes. An attempt has been made in this chapter to help accomplish this task better. Under some simplifying assumptions, the analysis of this work has established that the solution to the optimization problem under consideration tends to equalize the lifetimes of adjacent links in a route. Moreover, there is a monotone variation of the speeds of intermediate relay nodes under the optimal policy. Explicit expressions for optimal choice of next hop based on its speed and inter-node distance have been obtained for certain fast track highway scenarios of interest. These solution structures have also been confirmed with simulations using a VANET simulator.

Even though our results are based on some simplifying assumptions and are restricted to selected scenarios, they can certainly be helpful in acquiring an intuition about the underlying dynamics of route connectivity in VANETs on highway. Moreover, the explicit solutions and structures obtained in this work can be of considerable practical interest as they reduce the space over which an existing VANET routing algorithm would search for the optimal routing policy.

8.7.1 Generalizing the equalization structural property

The structural property related to 'equalization' of adjacent link lifetimes derived in Section-8.5.2 can in fact be generalized to any path optimization problem in wireless access networks. In the following chapter, we shall describe some reasons behind this generalization and discuss the derivation of this structural property in detail.

Appendix: When $\frac{R}{s_i} << \frac{1}{\mu_i}$ does not hold, a value of $p = 0.995$ is well justified for a fast track highway. This is because in discrete time the node transition process has a geometric distribution and for a slot size of 0.1 seconds the probability that a node transits to another lane in 1 second comes to about 0.0048, which is quite low. The mean time between two successive transitions by a node is about 20 seconds and this is fairly justified for a fast track highway. On the other hand, when $\frac{R}{s_i} << \frac{1}{\mu_i}$ does hold, a value of $p = 0.99999$ leads to a mean time of 10^5 seconds between two successive transitions by a node. We thus have a situation where the node stays in its lane for a time much greater than its link lifetimes.

Chapter 9

On Solutions to Path Optimization Problems in Random Access Networks

The inherent nature of the physical setup and transmission mechanism in wireless ad hoc networks with random channel access, results in correlation between the *link metrics* of adjacent links, when considering path optimization problems. In this chapter, we identify a special structure inherent to the solution of Dynamic Programming (DP) problem arising in such an optimization over paths. According to this structure, the optimal policy tries to *equalize* the link metrics of adjacent links in a multi-hop route. We validate this structural property with a VANET simulator.

Note: The material in this chapter has appeared in [KKA06a]. This work was sponsored by "The Indo-French Centre for the Promotion of Advanced Research" (CEFIPRA).

9.1 Introduction

The performance characteristics of either wired or wireless networks are measured and analysed using various kinds of *metrics*. These metrics usually signify certain performance attributes pertaining to the transmission links in the network. We can therefore think of a *link metric* that defines the performance measure of transmission links in the network. This link metric could either be some kind of a *cost* like link delay or it could as well be some kind of a *reward* utility like link lifetime, link rate, etc. Also see [DPZ04b, DPZ04a] for some examples of such metric. In wire-line networks, a link metric usually depends on certain performance attributes of only one of the end nodes (transmitting node). For example the expected delay over a link depends on the queueing delay of only the transmitting node. Whereas, in random access wireless networks, a link metric may depend on attributes associated with both the end nodes. For example, the link rate and link lifetime in a wireless ad hoc network depends on the power levels or battery energy of both the end nodes. For sure, in both wired and wireless networks, a link metric could as well be some independent entity by itself, e.g., propagation delay and channel gain due to multi-path fading.

Due to dependence on both the end nodes, link metrics of two adjacent links in a multi-hop wireless network may depend on attributes of the common node between them and hence they may be correlated. With another point of view, in wireless networks, attributes of multiple nodes may collectively determine properties of a given link, as opposed to wired networks where properties of a link are determined by attributes of only one terminal node. This fundamental difference between wired and wireless networks requires considering significantly different approaches when finding an *optimal* path between a source node and its destination. It is to be noted that the above discussion is valid only for the choice of those link metric that satisfy the properties discussed in the previous paragraph.

All routing algorithms aim to achieve *path optimization*, and most of these algorithms use the Dynamic Programming (DP) technique to compute *optimal paths*. In wired networks, whether the decision is based on a link metric or attributes of nodes, the complexity of finding an optimal path is the same. However, in wireless networks the situation may be very different, especially when the link metric depends on the attributes of both end nodes as discussed before. In wireless networks that have rapidly changing network topologies and require frequent discovery of new routes, getting some insight into the structural properties of the optimal path can help in significantly reducing the time complexity of route rediscovery. This is especially beneficial when nodes are energy-limited devices. Our main contributions in this chapter are

1. Identify basic difference between path optimization in wired and wireless networks.

2. When the link metric depends on attributes of both the end nodes (i.e., the wireless case), we identify in Section 9.2.2 a special structural property that is satisfied by the solution of the DP problem of Section 9.2. This structural property, which says that the optimal policy tends to *equalize* the link metrics of two successive links, results in significant reduction in the computational complexity of the DP algorithm. In Section 9.3, we validate this structural property by considering a specific example of

routing in a vehicular ad hoc network (VANET).

3. Yet another interesting result obtained in this work is that an optimization problem of a Max-Min form can be transformed to a linear optimization problem (Theorem 6).

9.2 The Basic Problem Formulation

In multi-hop wireless networks, one can divide the problem of finding an optimal path between a source and its destination, into two sub problems:

1. For a given number of intermediate relay nodes, find the *best path* i.e., the optimal choice of relay nodes that optimizes a certain objective function comprising of link metrics, and

2. Having solved the above problem, find the optimal *number* of relay nodes.

In the present work we are interested in the first optimization problem mentioned above. Since the source and destination nodes are known in advance and thus fixed, we first present the basic problem formulation in a general and natural framework of dynamic programming with *fixed initial and terminal states*. Let l_i denote an attribute associated with node i. An example of such an attribute is the (inverse of) average transmit queue occupancy of a node. Let the link metric of a link between nodes i and j be denoted by $H(l_i, l_j)$ for some function $H(\cdot, \cdot)$. In the rest of the chapter, attributes of a node will also be referred to as *node metric*. Now, for a given number of intermediate relay nodes M, the problem is to find a vector of indices of the relay nodes $(i_1, \ldots, i_M)$ that minimizes

$$\sum_{j=1}^{M+1} H(l_{i_{j-1}}, l_{i_j}), \tag{9.1}$$

where the source node has index i_0 and its destination has index i_{M+1}. This kind of an objective function arises for example when we want to minimize the path-sum delay in a route. Before proceeding further, we mention here that the above optimization problem is general enough for also studying the Max-Min form class of problems (this equivalence is shown later), where the objective is to maximize

$$\min_{j=1,\ldots,M+1} T(l_{i_{j-1}}, l_{i_j}).$$

This kind of an objective function may arise, say when we want to maximize (a critical reward) the least of the lifetimes of links constituting a route. Here we use the notation $T(\cdot, \cdot)$ for link metric instead of the one used in Equation 9.1, for notational convenience. By applying an appropriate transformation to the link metric $H(\cdot, \cdot)$, this will also allow us to solve the problem of Equation 9.1 in an indirect manner, by using the solution of yet another set of problems that are equivalent to the above Max-Min problem.

In the rest of this section, we first establish the equivalence of the optimization problem of Equation 9.1 to the Max-Min optimization problem, via yet another set of optimization problems equivalent to the Max-Min problem. Then later in Section 9.2.2 we derive a special structural property that is satisfied by the solutions of all these equivalent problems.

9.2.1 Equivalence of Equation 9.1 problem to the Max-Min problem

The Max-Min optimization problem can be detailed as follows. Let the node metric be allowed to vary over a finite set of values of cardinality L. We are given the number of intermediate nodes M in the route path, for a particular choice of a source node and its destination. We want to find the node metrics $l_{i_j},\ 1 \leq j \leq M$ of the M intermediate relay nodes such that we obtain a maximum value of (a critical reward) the least of the link metrics, i.e., we seek a vector $l_{i_j},\ 1 \leq j \leq M$ that solves the following optimization problem,

$$\underset{1 \leq l_{i_j} \leq L, j=0,\ldots,M}{\text{Maximize}} \quad \underset{j=0,\ldots,M}{\text{Minimum}} \ T(l_{i_j}, l_{i_{j+1}}) \ . \tag{9.2}$$

This problem can be rewritten as finding a vector $l_{i_j},\ 1 \leq j \leq M$ that solves the following optimization problem (the only change is that of notation, since here by node i we mean i^{th} node in the optimal path),

$$\underset{1 \leq l_i \leq L, i=0,\ldots,M}{\text{Maximize}} \quad \underset{i=0,\ldots,M}{\text{Minimum}} \ T(l_i, l_{i+1}) \ . \tag{9.3}$$

Instead of solving the above problem directly, we further look at a different, parameterized set of objective functions to be optimized. These new objective functions will coincide with the original one of Equation 9.3 when the parameter takes a special value. Solution to any one of these parameterized form of optimization problems will then finally provide us with the solution to the problem of Equation 9.1 via the solution to the Max-Min problem of Equation 9.3. We first restate the following simple lemma which has been proved earlier in Section-8.4, Chapter-8.

Lemma 3 *For any* finite dimensional *vector* $\underline{x}$ *with positive elements* $x_i,\ 1 \leq i \leq n$, *if* $||\underline{x}||_\alpha$ *is the* $l_\alpha-$*norm, i.e.,* $||\underline{x}||_\alpha = \left[\sum_{1 \leq i \leq n} x_i^\alpha\right]^{\frac{1}{\alpha}}$, *and* $||\underline{x}||_\infty$ *denote its* $l_\infty-$*norm, i.e.,* $||\underline{x}||_\infty = \max_{1 \leq i \leq n}\{x_i\}$, *then*

$$\lim_{\alpha \to \infty} ||\underline{x}||_\alpha = ||\underline{x}||_\infty.$$

Theorem 4 *The optimization problem of Equation 9.3 has the same optimizer as that of*

any of the following optimization problems,

$$\underset{1\leq l_i\leq L, i=1,\ldots,M}{Minimize} \left[\sum_{j=0}^{M} (T(l_j, l_{j+1}))^{-\alpha}\right]^{\frac{1}{\alpha}}, \tag{9.4}$$

$$\underset{1\leq l_i\leq L, i=1,\ldots,M}{Maximize} \left[\sum_{j=0}^{M} (T(l_j, l_{j+1}))^{-\alpha}\right]^{\frac{-1}{\alpha}}, \tag{9.5}$$

as $\alpha \rightarrow \infty$.

Proof: This proof follows in the same line as that of Theorem-1 in Chapter-8 with some difference in notation. The optimization problem of Equation 9.3 clearly has the same solution as that of the problem,

$$\underset{1\leq l_i\leq L, i=0,\ldots,M}{\text{Minimize}} \quad \underset{i=0,\ldots,M}{\text{Maximum}} \quad \frac{1}{T(l_i, l_{i+1})}.$$

Now, for any integer $\alpha > 0$, we can compute the $l_\alpha-$norm of an M-dimensional vector whose i^{th} element is $1/T(l_i, l_{i+1})$. The $l_\alpha-$norm of this vector, for any given values of l_i's is

$$\left[\sum_{i=0}^{M} (T(l_i, l_{i+1}))^{-\alpha}\right]^{\frac{1}{\alpha}}.$$

Since $1/T(l_i, l_{i+1})$'s are strictly positive and bounded quantities, we can invoke Lemma 3 to conclude the equivalence of Equations 9.3 and 9.4. Then equivalence of Equations 9.4 and 9.5 is straightforward. •

In fact, we can say something more about the relation between the two optimization problems of Equations 9.3 and 9.4. This additional property actually leads us to our main result of this section (Theorem 6).

Theorem 5 *There exists a* finite $\alpha^*(T)$*, such that the maximizer of optimization problem of Equation 9.3 is* same *as that of either of Equation 9.4 or 9.5,* for all values of $\alpha \geq \alpha^*(T)$.

Proof: This proof follows in the same line as that of Theorem-2 in Chapter-8 with some difference in notation. Fix a vector $\underline{x}$ with elements $x_i, 1 \leq i \leq n$. Then, from Lemma 3 we know that $\lim_{\alpha\rightarrow\infty} ||\underline{x}||_\alpha = \max_{1\leq i\leq n} x_i$. Now, form a vector $\underline{y}$ whose i^{th} element y_i is the i^{th} maximum among the elements of $\underline{x}$ (so that $y_1 = \max_{1\leq i\leq n} x_i = ||\underline{x}||_\infty$). Since the number of elements in $\underline{x}$ is n, which is finite, the difference $y_1 - y_2 > 0$ (assuming that no two elements of $\underline{x}$ are equal; the case where some of the elements of $\underline{x}$ are equal can also be easily considered.). Since $\lim_{\alpha\rightarrow\infty} ||\underline{x}||_\alpha = \lim_{\alpha\rightarrow\infty} ||\underline{y}||_\alpha \rightarrow y_1$, for any $\epsilon > 0$ there exists a finite $\alpha^*_\epsilon(\underline{x}, T)$ such that , $y_1 - ||\underline{x}||_\alpha < \epsilon$ for all $\alpha > \alpha^*_\epsilon(\underline{x}, T)$.

Now, since the set of possible values of the node metric over which optimization is carried out is finite (of cardinality L), we can define,

$$\delta \stackrel{\Delta}{=} \min_{l_i, l_j} \mathrm{POS}(|T(l_i, l_{i+1}) - T(l_j, l_{j+1})|),$$

where,

$$\mathrm{POS}(|x-y|) = \begin{cases} |x-y| & \text{if } |x-y| > 0, \\ |x| & \text{otherwise} \end{cases}$$

Then, $\alpha^*(T) \stackrel{\Delta}{=} \max_{\underline{x}} \alpha^*_\delta(\underline{x}, T) < \infty$ is the finite quantity that we were seeking. •

Theorem 5 ensures that there is no discontinuity in the solution to the optimization problem of Equation 9.4 or 9.5 as $\alpha \to \infty$. Working with the objective function of Equation 9.4 or 9.5 thus has an advantage that we can work with any *finite* value of α and, if this solution is independent of α, we will have obtained the solution to the optimization problem of Equation 9.3. Also, the different forms of objective functions in Theorem 4 can be used as per convenience depending on whether the link metric is a reward or cost.

Further, the result of Theorem 5 can be used to transform the problem of Equation 9.1 to that of a Max-Min problem:

Theorem 6 *The minimizer of Equation 9.1 is same as the optimizer of Equation 9.3 with* $T(\cdot,\cdot) = H(\cdot,\cdot)^{-K}$ *for any* K *satisfying,*

$$K\alpha^*(H^{-K}) \leq 1, \tag{9.6}$$

where $\alpha^*(H)$ *is the same as in Theorem 5.*

Proof: We know from Theorem 5 that the optimizer of problem

$$\underset{1 \leq l_i \leq L, i=0,\ldots,M}{\text{Maximize}} \quad \underset{i=0,\ldots,M}{\text{Minimum}} \quad (H(l_i, l_{i+1}))^{-K}$$

is the same as the minimizer of Equation 9.4 (with α replaced by β for notational convenience),

$$\sum_{j=1}^{M+1} (H(l_{i_{j-1}}, l_{i_j}))^{K\beta}$$

for all $\beta \geq \alpha^*(H^{-K})$. If for some K, it turns out that $K\beta = 1$ for some $\beta \geq \alpha^*(H^{-K})$ then the second problem above is that of minimizing Equation 9.1. A value of K gives rise to existence of such a β iff $K\alpha^*(H^{-K}) \leq 1$. •

Theorem 6 and 4 say that for any problem of the kind in Equation 9.1 we can solve a transformed problem of Equation 9.3 and vice versa. However, existence of such a value of K is not entirely obvious and needs a proof that follows. We first state the following lemma.

Lemma 4 $\alpha^*(H^{-K}) \to_{K \to 0} 1$.

Proof: With $\alpha = 1$, as $K \to 0$, the solution set (the set of optimizers) of Equation 9.4 converges to that of Equation 9.3 because the objective function converges to a constant. •

Theorem 7 *There exists a K that satisfies Equation 9.6.*

Proof: The proof follows from Lemma 4. •

Now, we can use a circular at first sight argument: In order to study the structural property of the minimizer of Equation 9.1 (with a link metric function $H(\cdot,\cdot)$), we solve the equivalent Max-Min problem as in Theorem 6 (with a link metric function $H(\cdot,\cdot)^{-K}$). This Max-Min problem is in turn solved using the approach of Theorem 4, i.e., by solving problem of Equation 9.4, again with a link metric function $H(\cdot,\cdot)^{-K}$ *and with* $\alpha \to \infty$.

9.2.2 Structural property at optimal point

Consider two nodes, the source node has metric l_0 and its destination node has metric l_{M+1}. Thus, we need to find a set of node metrics $l_i, 1 \leq i \leq M$ such that objective function of Equation 9.4 is maximized. The constraint is that the set of possible values of l_i is finite and of cardinality L. We see that the objective function for any given value of α, for some appropriate function $T(\cdot,\cdot)$, is given by

$$\left[\sum_{j=0}^{M}\left[\frac{1}{T(l_j,l_{j+1})}\right]^{\alpha}\right]^{\frac{1}{\alpha}}.$$

Let $f_i(x,y) := \frac{1}{T(x,y)}$ be the link metric of i^{th} link, if $l_i = x$ and $l_{i+1} = y$. Clearly, if one is allowed to choose any value for the intermediate node's attributes (thus not restricting to the finite set of cardinality L), we would require the metric of node i to be a value x such that

$$\frac{d}{dx}\left[(f_{i-1}(l_{i-1},x))^{\alpha} + (f_i(x,l_{i+1}))^{\alpha}\right] = 0.$$

This implies, in particular, that

$$\frac{f_{i-1}(l_{i-1},x)}{f_i(x,l_{i+1})} = \left[-\frac{df_i(x,l_{i+1})}{df_{i-1}(l_{i-1},x)}\right]^{\frac{1}{\alpha-1}}.$$

Taking $\alpha \to \infty$, we see that we need

$$\frac{f_{i-1}(l_{i-1},x)}{f_i(x,l_{i+1})} = 1,$$

implying that the link metrics of successive links should be *equalized* in order to optimize the objective function of Equation 9.4. Note that this is only a necessary condition and not a sufficient one, i.e., not all configurations that result in identical link metrics of successive links will be a solution to the optimization problem under consideration. However, *any*

solution to the optimization problem will satisfy the above mentioned property. The above structure also carries over to the case where the allowed values of metrics of the intermediate nodes are restricted to a finite discrete set. However in that case, as it is obvious, exact equalization of the link metric of successive links will not be achieved owing to the lack of choice of metrics for the intermediate nodes. In Section 9.3 we consider an example network, where owing to a finite set of possibilities, the optimal policy has to settle down for a convex combination, instead of an exact equalization. It is important to note here that, owing to Theorem 6, the above structural result holds good for *both* the optimization problems of Equation 9.1 and Equation 9.3, as well.

9.3 Validation of Equalization property using a VANET simulator

In order to validate the equalization of link metrics property derived in the previous section, we have developed a vehicular mobile ad hoc network (VANET) simulator. The simulator simulates nodes (vehicles) moving on a highway in their respective lanes and the objective is to find the best path that maximizes the least of the *link lifetimes* of a route between any given source-destination pair of vehicular nodes.

Each lane is characterized by an associated speed and every node moving in a given lane travels with this same speed that is associated with its lane. Thus all vehicles moving in the same lane have identical speeds. The vehicles can change lanes and hence speeds, randomly over time. Each node is assumed to have a motion that is independent of the other nodes. To be more precise, we assume that the stochastic process corresponding to the changing speed of a node, is a Markov chain over the set of different speeds associated with different lanes on the highway. The time is slotted and at the end of each time slot each node makes a decision on whether to change its lane or not. This decision is made independently of any past decisions. If a node decides to change its lane, it moves to either of its adjacent lanes according to some probability distribution. Thus a node continues to move in a particular lane for a random amount of time that is geometrically distributed. We assume that a node in lane i, $1 \leq i \leq L$, transits to any of the adjacent lanes at the beginning of a time slot with probability $1 - p_i$.

We assume that such a mobile ad hoc network formed by vehicles moving on a highway consists of a dense layout of nodes, so that there is at least one node inside any sufficiently small area on any given lane. The problem that we address is to find the best path that maximizes the *lifetime* of a route between any given source-destination pair of nodes that can not communicate directly, thus requiring relaying by intermediate nodes. We assume that the speed (or, the lane) of the source and its destination and the initial distance D between them (at the time of route setup) are given. Also given is M, the number of intermediate relay nodes and the hop distances $d_i, 1 \leq i \leq M$ between these relay nodes at the time of route setup; thus the initial distance (d_{M+1}) between the last relay node and the destination node is fixed since D is given. The distances between these M relay nodes can change with time if they are moving on lanes with different speeds. Hence the problem is

to find the speeds of the intermediate nodes so as to maximize the time until when any one of the links constituting the whole route, *first* breaks down, i.e., maximize the *route lifetime* or in other words maximize the minimum of expected link lifetimes which is the link metric here. Therefore, if $v_i, 0 \leq i \leq M+1$ is the initial speed of i^{th} node in the path (0 and $M+1$ indices are for source and destination nodes respectively), and if $T(d_i, v_{i-1}, v_i)$ denotes the expected time until when the nodes $i-1$ and i can communicate, then the problem becomes

$$\text{Maximize}_{v_1,\ldots,v_M} \ \text{Minimum}_{1\leq i\leq M+1} T(d_i, v_{i-1}, v_i).$$

With respect to our basic problem of Section 9.2, the link metric here is $T(\cdot,\cdot,\cdot)$ and the node attribute or node metric here is the speed v_i of a node. Thus, for a given value of the hop distances, the above problem is identical to that of Equation 9.3 in Section 9.2.

In the simulation results presented here, the time slot length has been taken as 0.1 seconds. For our simulations we consider that the probabilities $p_1 = \cdots = p_L = p$ are identical for all lanes. The set of possible lane speeds is $s_i,\ 1 \leq i \leq L$ and hence $v_j \in \{s_i,\ 1 \leq i \leq L\}$. We assume that $s_1 < s_2 < \cdots < s_L$. The simulator computes the expected link lifetimes of all possible links by exhaustively simulating over all possible speed assignments $\underline{v} = \{v_1, \cdots, v_M\}$ to the relay nodes for a given scenario of M relay nodes, L lanes, the hop distance vector $\underline{d} = \{d_1, \cdots, d_M\}$, speeds of source and destination v_0 and v_{M+1}, wireless transmission range R, source and destination separation D and the probability p. Once an exhaustive set of link lifetimes for all possible values of $\underline{v}$ is obtained by employing this brut force method, either of the objective functions of Equation 9.3 or 9.4 is applied over this set to obtain an optimal speed assignment policy resulting in the choice of a best or optimal path.

9.3.1 Simulation scenarios

A car battery operated mobile device has a typical transmission range of around 200 meters. We therefore consider hop distances in a VANET to vary from 140 to 200 meters and a wireless transmission range of 200 meters is considered for all the simulation scenarios. It has been shown in a previous work [GTB99] that large number of hops in an ad hoc network can significantly degrade the TCP throughput performance. Based on this result, we consider the number of hops $(M+1)$ to vary from 2 to 7 only and the distance between the source and destination nodes is varied from 800 to 1200 meters. We perform simulations for the number of lanes L varying from 2 to 6 and unless explicitly stated in the discussion on the simulation results, the associated speeds are chosen as shown in the table that follows,

l	1	2	3	4	5	6
$s_l\ (m/s)$	14	17	22	30	42	55
$\approx s_l\ (km/hr)$	50	60	80	110	150	200

Simulations were carried out for a large set of real life scenarios. In the following part of this section we discuss the various scenarios that were simulated and compare their results with the link metric equalization property that we obtained in this chapter earlier.

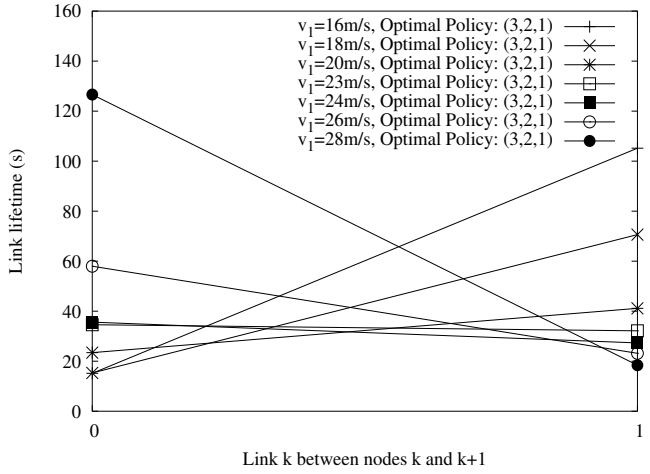


Figure 9.1: Link lifetime equalization over continuum of speeds

9.3.1.1 Lifetime Equalization over continuum set of speeds for small transition probabilities

In Figure 9.1 we plot the results of the scenario $L = 3$, $M = 1$, $v_0 = s_3 = 22m/s$, $v_2 = s_1 = 14m/s$, $p = 0.99999$, $D = 300m$ and $\underline{d} = (143, 157)$. In order to be able to validate the equalizing structure obtained earlier over a continuum set of intermediate node speeds, we vary the speed associated with lane 2 from $14m/s$ to $30m/s$ in small steps of $2m/s$ and plot the link lifetimes for each such speed of lane 2 separately. This allows the only intermediate node 1 to be assigned with one of the quasi-continuum set of speeds for the optimization problem of Equation 9.3 or 9.4 equivalently. It can be seen in the figure that under optimality, for varying values of v_1, the optimal lifetimes of the links between node 0 and 1 and node 1 and 2 are different. However for $v_1 = 23m/s$ the optimal lifetimes of the two adjacent links are almost equal (the line joining them is almost flat) thus confirming our result obtained in Section 9.2.2 that the lifetimes of adjacent links should be equalized in order to optimize the objective function of Equation 9.4. In fact, it can be observed that we obtain the maximum of the least of the two link lifetimes for speed $v_1 = 23m/s$ and the optimal lifetimes obtained for other values of v_1 are not truly optimal because of the unavailability of the choice of speed $23m/s$ in those cases.

9.3.1.2 Lifetime Equalization under the optimal policy for small transition probabilities

To strengthen our validation of the link metric equalization structure discussed in the previous note, in Figure 9.2, we plot the link lifetimes for a particular subset of the all possible speed combinations (including the optimal combination) for a given scenario. Figure 9.3 illustrates the same result for another scenario. It is clearly seen in these figures that at the optimal policy the link lifetimes are equalized and the maximum of the least link lifetimes is obtained at the optimal policy, by definition. Figure 9.2 uses the scenario $L = 6$, $M = 3$,

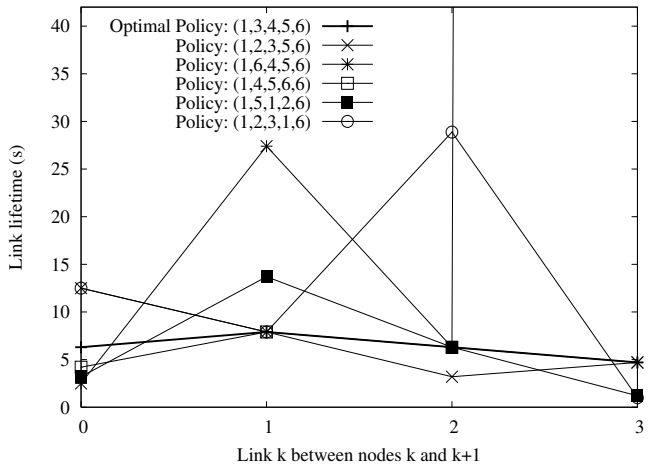


Figure 9.2: Link lifetime equalization under optimal policy

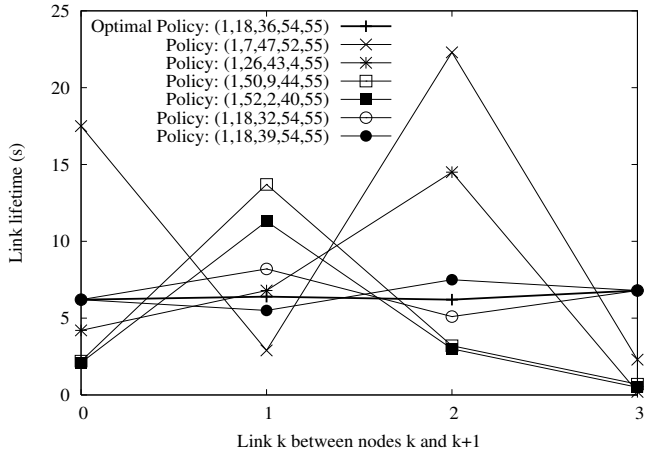


Figure 9.3: Link lifetime equalization under optimal policy

$v_0 = 14m/s$, $v_4 = 30m/s$, $p = 0.99999$, $D = 700m$ and $\underline{d} = (160, 175, 180, 185)$. Figure 9.3 uses the scenario $L = 55$, $M = 3$, $v_0 = 14m/s$, $v_4 = 30m/s$, $p = 0.99999$, $D = 700m$ and $\underline{d} = (169, 166, 167, 198)$.

9.4 Conclusion

We have considered path optimization problems in wireless networks where a link metric depends on attributes associated with the end nodes. We relate these path optimization problems to a Max-Min form class of problems and show that the optimal or best path satisfies the property that the link metrics of consecutive links in a multi-hop route are *equalized*. This property has been validated with a VANET simulator for the route lifetime maximization problem.

Chapter 10

Performance Modeling for Workload Dependent Parameters and Real Time Operation

In this chapter, we are concerned with performance modeling of transaction-based distributed software applications deployed over an arbitrary computer network architecture. AMBIENCE, which is a research prototype tool that has been developed at IBM Research, makes use of the powerful *Inferencing* technique that allows one to generate a workload-independent service time and CPU overhead based performance model. In this work, we extend the Inferencing technique for generating arriving *workload dependent* service time and CPU overhead based performance models. We call this extended form of Inferencing as *Enhanced Inferencing*. Implementation of this Enhanced Inferencing in AMBIENCE shows significant improvement in performance model fitting and approximation. We further present an innovative approach of using Kalman filtering with Inferencing like performance models in order to be able to pursue *real time* performance modeling of production environment computer application systems that face fast changing *non-stationary* arriving workloads. This real time performance model not only works reliabily for *non-stationary* workloads, but also incorporates *workload dependent* model parameters.

Note: The material in this chapter has appeared in [KZ, KTZ]. This work was carried out at IBM T. J. Watson Research Center.

10.1 Introduction

Performance modeling is a crucial step in capacity planning of computer network systems. Performance models of complex software and hardware network architectures can be very helpful in accurately predicting their performance for varying data traffic patterns and workloads. In this chapter, we are concerned with performance modeling of transaction-based distributed software applications deployed over an arbitrary computer network architecture. Transaction based applications usually comprise data traffic of type 'request/response' pair, hitting a complex network of 'server' entities that comprise a software application and physical computer machines. For example, an e-business application workload is composed of requests to and responses from a network of e-business servers. Consider for example an enterprise online shopping scenario. This interaction includes authentication transactions such as login, and business transactions such as browsing a catalog, searching for products, adding items to a shopping cart, proceeding to check out, etc. Each of these transactions may use the e-business server's resources differently.

Performance modeling becomes more challenging if the workload (i.e., number of jobs or transactions handled per unit of time) for an IT system deployed in production environment is *non-stationary* in nature and has fast changing characteristics. The workload may have peak/off-peak volume, daily/weekly patterns, which can be volatile. Moreover, state of the art transaction-based software applications are quite complex in design. Due to this, the end-to-end transactions may incur *variable service times* depending on the:

1. total arriving workload (reasons detailed in Section-10.3).

2. time dependent overheads as a function of the software application uptime (applications running since long time may have increased memory-page access times and tedious storage patterns in memory due to fragmentation, etc.)

3. complex modular design of software applications: variable and application context/state sensitive function call sequences between various modules and databases (if any) invoked by the same transaction job.

4. fluctuating overheads caused by the host operating system environment.

5. configuration of application deployment environment.

It is thus a challenging task to assess an IT system's capability of delivering end-to-end performance assurance across the entire IT environment, given the variety of system architectures, numerous applications with different functions, and the vast diversity in user behavior.

Performance models based on traditional queueing theory require parameters such as service times and network delays of different transaction classes and CPU overheads of different computer machines, in a given network, to compute performance metrics such as average transaction response times, average number of jobs/transactions waiting to be processed in a buffer queue, etc. There are other existing techniques that can make use

of simulations and manual caliberations to compute similar performance metrics [ZLR+03]. However, none of these techniques can be practically applied if service time, network delay and overhead parameters are not known. This is often the case for an arbitrary software application deployed over an arbitrary distributed network architecture.

AMBIENCE [ZXSI02, LXMZ03], which is a research prototype tool that has been developed at IBM Research, makes use of the powerful *Inferencing* technique that allows one to generate a service time, network delay and CPU overhead based performance model of an arbitrary computer network architecture. Inferencing allows one to compute the service time and overhead model parameters from readily available measurement data on end-to-end response times, CPU utilizations and workload arrival rates. It however assumes that the arriving workload traffic is *stationary* in nature or measurements have been gathered over a stationary time window. It further assumes *constant* service times and CPU overheads, *independent of the arriving workload*. However, as discussed earlier, workload arriving at an IT system in production environment may be non-stationary in nature due to diversity in user behavior. Moreover, due to design complexities of state of the art software applications, end-to-end transactions may incur variable service times. Therefore, Inferencing may not work *reliabily* if either arriving transactional workload is non-stationary or design complexity of software application causes workload dependency in service times and CPU overheads.

10.1.1 Main Results and Contribution

In this work, we extend the Inferencing technique for generating *workload-dependent* service time and CPU overhead based performance models. We call this extended form of Inferencing as *Enhanced Inferencing*. We further present an innovative approach of using Kalman filtering with Inferencing like performance models in order to be able to pursue *real time* performance modeling of online and production environment computer network systems that face fast changing non-stationary arriving workloads. This real time performance model not only works reliabily for *non-stationary* workloads, but also incorporates *workload dependent* model parameters.

10.2 Background

Consider a single transaction class hitting a single server. A transaction class refers to a grouping of transactions with similar characteristics. For example the transaction classes may include transactions for searching, buying, logging on, etc. The buying transaction class may include transactions that are responsible for purchasing a particular product and so on. Different transaction classes typically have different service requirements. From queueing theory we know that for an open model based on $M/M/1/\infty$ queue with Processor Sharing (PS) service mechanism, we have,

$$R = \frac{s}{1-u}, \tag{10.1}$$

where,

$$u = \lambda\, s. \tag{10.2}$$

In the above formulae, s is the service time of the given job/transaction class, R is its corresponding response time, λ is the arrival rate of all such jobs and u is the server utilization. The above equations may be repeatedly applied to a system that includes multiple machines (servers) and multiple classes of transactions.

10.2.1 Notation

All entities, $\tilde{\cdot}$, with a tilde on their top are actual *average* measurement values that can be obtained from a real network. Let there be K transaction classes and M physical machines in a given network. Let $\mathcal{C}$ denote the index set of all transaction classes that may have different service requirements or different SLA targets. Let $\mathcal{L}$ denote the index set of all physical machines that are part of a given network. Let the rate at which total workload of a class $c \in \mathcal{C}$ transaction arrives from an external source into the network (through any machine) be denoted by $\tilde{\gamma}^c$. Let the rate at which workload of a class $c \in \mathcal{C}$ transaction arrives at machine $i \in \mathcal{L}$ from within the network be denoted by λ_i^c. Note that λ_i^c denotes the *effective* workload of a class c arriving at machine i, where as $\tilde{\gamma}^c$ denotes the total workload of a class c being generated by a source outside the network. Denote $\boldsymbol{\lambda^c} = [\lambda_1^c, \lambda_2^c, \ldots, \lambda_M^c]$. Also, in the vector $\boldsymbol{\gamma^c} = [0, 0, \ldots, \tilde{\gamma}^c, \ldots, 0]$, $\tilde{\gamma}^c$ is located at some jth position ($j \in \mathcal{L}$) when class c transactions enter the network through machine j. Then one may compute the effective arrival workloads by solving the open Jackson network traffic balance equations as,

$$\forall c \in \mathcal{C}, \qquad \boldsymbol{\lambda^c} = \boldsymbol{\gamma^c}(I - P^c), \tag{10.3}$$

where, $P^c = [P_{ij}^c]$ is the traffic routing probability matrix, i.e., probability that a class c flow leaves a machine i and goes to machine j is given by P_{ij}^c. We assume that P^c is well defined for a given network architecture, i.e., the values P_{ij}^c are given.

10.2.2 Inferencing for workload independent model parameters

Here we briefly describe the Inferencing technique from [ZXSI02, LXMZ03] that allows one to generate a service time, network delay and CPU overhead based performance model. The service time, network delay and overhead model parameters are assumed to be constant and independent of the arriving workload.

Consider the case of stationary workload or assume that measurements are taken from a network when the arriving workload stays within a given stationary regime. Let s_i^c denote the service time of a class c transaction at machine i, d_{net}^c denote the total network delay incured by a class c transaction and o_i^{cpu} denote the total CPU overhead at machine i. Then we may apply the queueing theory based formulae described earlier in Equations-10.1 and 10.2 to obtain the service times, s_i^c, using the inferencing technique as follows. We first write down the expressions for response time, R^c, of traffic class c and utilization, u_i, of machine i as,

$$\forall c \in \mathcal{C}, \qquad \sum_{i \in L} \alpha_i^c \frac{s_i^c}{1 - \tilde{u}_i} + d_{net}^c = R^c \tag{10.4}$$

$$\forall i \in \mathcal{L}, \qquad \sum_{c \in \mathcal{C}} \frac{\lambda_i^c}{P_i} s_i^c + o_i^{cpu} = u_i \tag{10.5}$$

where, $\alpha_i^c = \frac{\lambda_i^c}{\tilde{\gamma}^c}$ and P_i is the total number of processors (CPUs) in machine i. One may then seek to minimize the sum of squares of relative errors between the analytical entities given by above two Equations 10.4 and 10.5 and their corresponding mean measurement values obtained from the real network. The corresponding optimization problem would then comprise a quadratic objective function in the set of variables $\{s_i^c, d_{net}^c, o_i^{cpu}, e^c, e_i | c \in \mathcal{C}, i \in \mathcal{L}\}$,

$$\min \sum_{c \in \mathcal{C}} \left(\frac{e^c}{\tilde{R}^c} \right)^2 + \sum_{i \in \mathcal{L}} \left(\frac{e_i}{\tilde{u}_i} \right)^2 \tag{10.6}$$

with the following set of linear constraints,

$$\forall c \in \mathcal{C}, \qquad R^c + e^c = \sum_{i \in L} \alpha_i^c \frac{s_i^c}{1 - \tilde{u}_i} + d_{net}^c + e^c = \tilde{R}^c \tag{10.7}$$

$$\forall i \in \mathcal{L}, \qquad u_i + e_i = \sum_{c \in \mathcal{C}} \frac{\lambda_i^c}{P_i} s_i^c + o_i^{cpu} + e_i = \tilde{u}_i \tag{10.8}$$

The solution to this optimization problem will give us the service times, s_i^c, network delays, d_{net}^c, and CPU overheads, o_i^{cpu}, required as part of the solution to the Inferencing problem.

10.2.3 Conducting Experiments

In the previous sub-section, we assumed that measurements are taken from a network when the arriving workload stays within a given stationary regime. These measurements would include average response times $\tilde{R}^c$ for each class $c \in \mathcal{C}$, average CPU utilizations $\tilde{u}_i$ for each machine $i \in \mathcal{L}$ and average arrival rates λ_i^c computed from $\boldsymbol{\gamma^c}$ for each class $c \in \mathcal{C}$. A single *experiment* can thus be carried out for a given application or computer network system which would result in such measurements to construct a single set of constraints given by Equations-10.7 and 10.8. The Inferencing optimization problem can then be solved along-with these single set of constraints to obtain the solution for model parameters. In a similar manner, multiple experiments can be conducted at varying workloads to formulate multiple set of constraints given by Equations-10.7 and 10.8. Each of these experiments should be conducted to compute average measurement values over the stationary regime corresponding to the arriving workload during that experiment. The Inferencing optimization problem can then again be solved these multiple set of constraints to obtain the solution for model

parameters. This time however, the solution would be representative of the constant service times across varying workloads for which the multiple experiments were conducted. In other words, using multiple experiments can give a better fit for the service time solution as compared to using a single experiment.

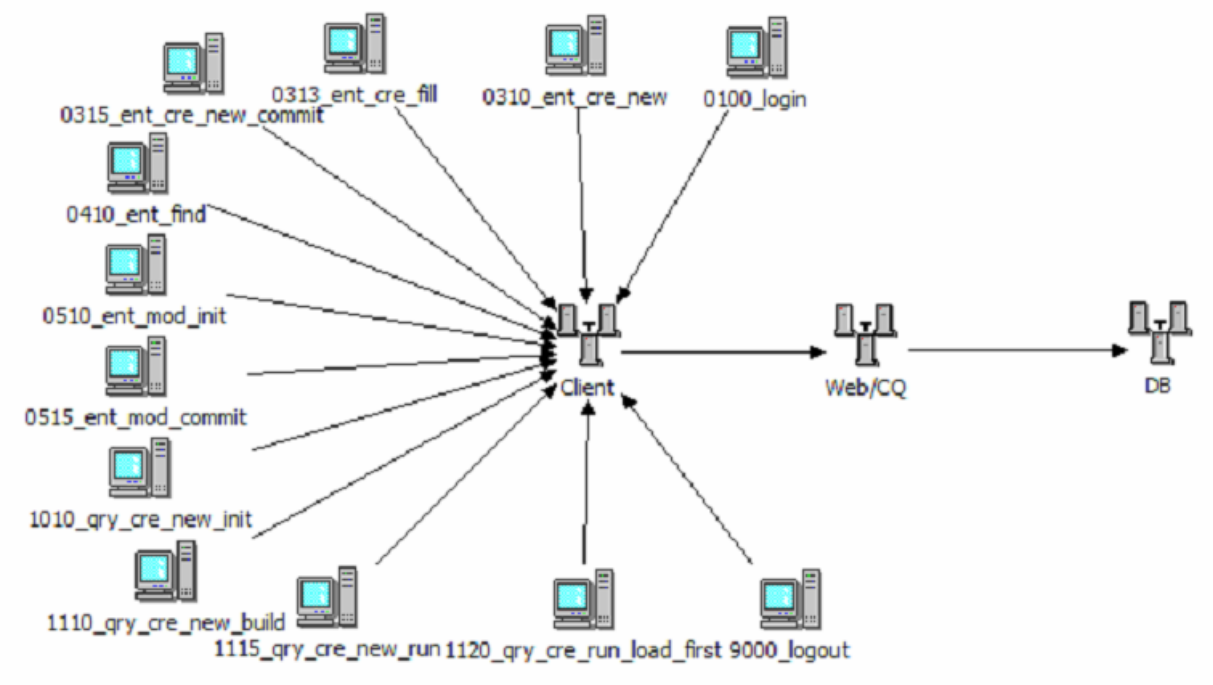


Figure 10.1: Example Application and Network topology

10.2.4 Inferencing Example

Consider an example application and network topology as shown in Figure-10.1. The 12 icons on left represent 12 different transaction classes that are generated by the client machine "Client". These transactions first hit the webserver and ClearQuest applications installed on the "Web/CQ" machine and then eventually hit the database installed on "DB" machine. We are interested in generating a performance model of this system using AMBIENCE tool which uses the Inferencing algorithm. For this, 6 different experiments were carried out at 6 different transactional workloads: 370 sessions/hour, 633 sessions/hour, 813 sessions/hour, 896 sessions/hour and 925 sessions/hour. Measurements obtained from these experiments were input into AMBIENCE and the performance model was built. Figure-10.2 shows the CPU utilization scaling chart. The dots in the chart represent average utilization measurements for different machines at different workloads. The lines in the chart are utilization predictions generated by the model obtained using the Inferencing algorithm. On x-axis "Load Scale" represents ratio between the transactional workload for different experiments and the base workload. The experiment with smallest workload is considered as the base workload experiment. Thus, for the 370 sessions/hour experiment the Load Scale is 1.0. In Figure-10.2 we see that the model predicted utilization is linear in variation and

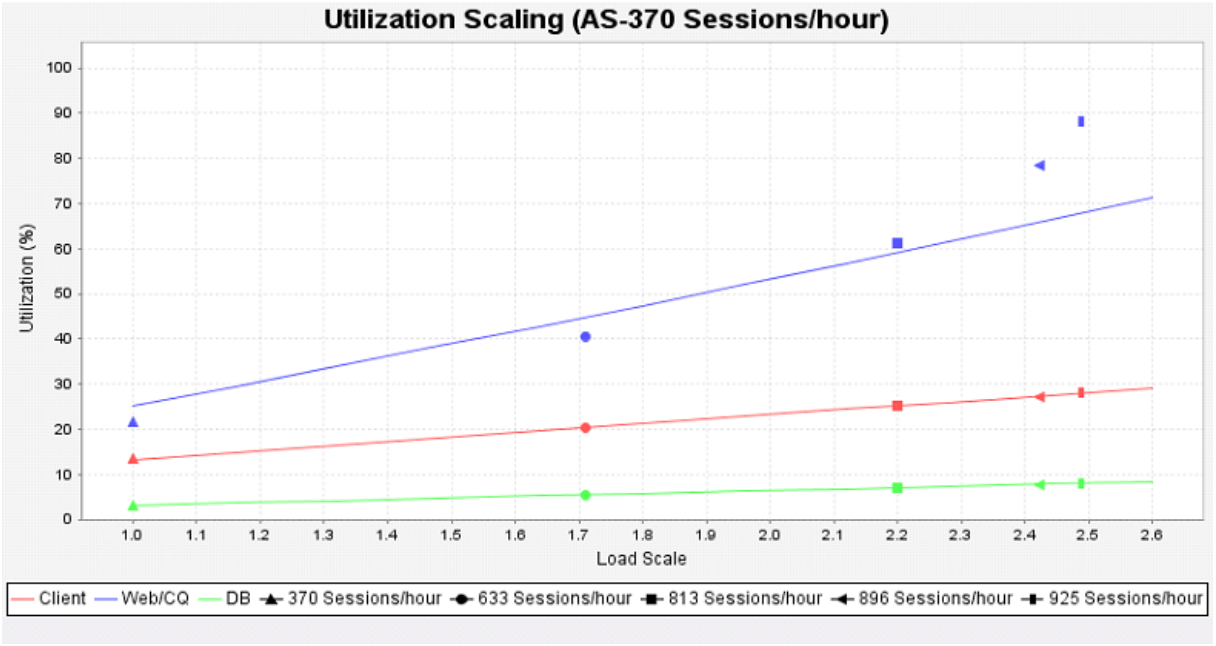


Figure 10.2: Utilization Scaling with Inferencing

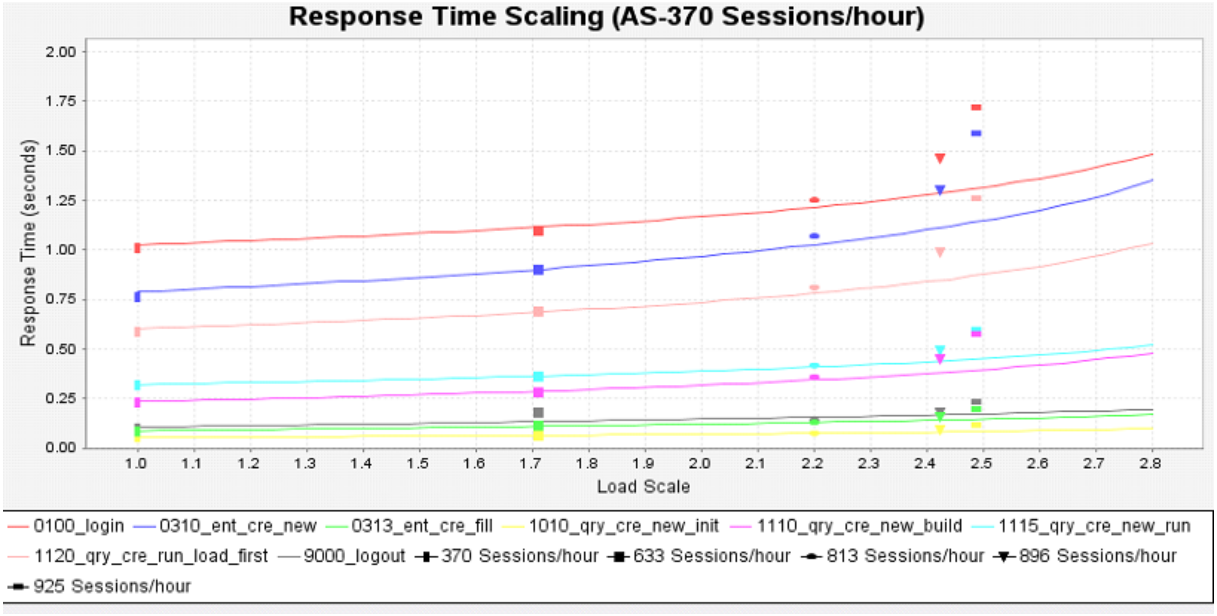


Figure 10.3: Response Time Scaling with Inferencing

matches well with the measured utilization for "Client" and "DB" machines. This indicates that the service times at these two machines are indeed workload independent and their inferred values fit well in the model. However, for "Web/CQ" machine we see that the model predicted and measured utilizations do not match well. Since the variation in measured utilizations is not linear it may indicate that the actual service time may be workload dependent. Fig.-10.3 shows the resp. time scaling chart. Again we see that model predicted and measured response times match well for some transactions, but not for others.

10.3 Enhanced Inferencing for workload dependent model parameters

As discussed in Section-10.1, complex design of state of the art software applications may lead to variable service times depending on the total arriving workload. This is because computer machines may incur extra processing overheads when mean transactional workload is high as compared to when the mean transactional workload is low.

In a multi-tasking operating system environment, higher average transactional workload may cause increased context-switching of various application modules and components in the CPU and memory. Over a given period of time the CPU may end up spending more time in context-switching than processing the actual transaction jobs. This may lead to a slower transaction processing rate by the CPU and increased service times for a given transaction class. In simple words, it may lead to a slower server.

Also, higher transaction workload hitting the NIC (Network Interface Card) would force the NIC to generate higher number of interrupts at the CPU for data transfers between the NIC and main memory. This would cause fewer CPU cycles being allocated to the processing of actual transaction jobs and the CPU would spend more time in serving the interrupts. Service times of transactions would be further affected due to this phenomenon.

Therefore, it is clear from the experiment results in Section-10.2.4 and the above discussion that we must consider service times and CPU overheads as some function of the arriving workload and not constants. In other words, service time for a particular transaction class at a given machine and total CPU overhead in a particular machine may be more at higher mean workloads as compared to at lower mean workloads. We therefore extend the Inferencing technique to incorporate arriving workload dependent service times and CPU overheads and call it *Enhanced Inferencing*.

Empirically, it has been observed that the rate at which service times increase with increasing arriving workload does not follow a common trend. It rather depends on various factors listed in Section-10.1 earlier, i.e., software application uptime, complex modular design of software applications, fluctuating overheads caused by the host operating system environment, configuration of application deployment environment, etc. It has been observed that the rate of increase of service times and CPU overheads is usually either a polynomial, exponential or logarithmic function of the arriving workload. Based on this discussion and observations, intuitively it would then make sense to consider the service times and CPU overheads as increasing polynomial, exponential or logarithmic functions of the total arriving workload. Moreover, we have observed that the *sum* arriving workload over all transaction classes works as a fairly good approximation in order to incorporate dependence of service times and CPU overheads on the complete workload vector. This motivates us to consider the service times and CPU overheads as polynomial functions of either the simple sum workload, the exponential of sum workload or the logarithmic of sum workload. However, note that this dependency on arriving workload may also be characterized as, service times and CPU overheads being functions of any arbitrary or generic representation of the arriving workload. In other words, the service times and CPU overheads may be expressed as $s_i^c(f(\{\lambda_i^c | c \in \mathcal{C}\}))$ and $o_i^{cpu}(g(\{\lambda_i^c | c \in \mathcal{C}\}))$, respectively, for *any* given functions

f and g. Here, as an example we consider these functions to be $f(\{\lambda_i^c | c \in \mathcal{C}\}) = \sum_c \lambda_i^c$ and $g(\{\lambda_i^c | c \in \mathcal{C}\}) = \sum_c \lambda_i^c$. Therefore, as an example, assume that for a given transaction class c the service time at machine i may be expressed in one of the following ways,

$$s_i^c(\sum_c \lambda_i^c) = a_{0,i}^c + a_{1,i}^c \sum_c \lambda_i^c + a_{2,i}^c(\sum_c \lambda_i^c)^2 + \cdots + a_{n,i}^c(\sum_c \lambda_i^c)^n \tag{10.9}$$

$$s_i^c(\sum_c \lambda_i^c) = a_{0,i}^c + a_{1,i}^c e^{\sum_c \lambda_i^c} + a_{2,i}^c(e^{\sum_c \lambda_i^c})^2 + \cdots + a_{n,i}^c(e^{\sum_c \lambda_i^c})^n \tag{10.10}$$

$$s_i^c(\sum_c \lambda_i^c) = a_{0,i}^c + a_{1,i}^c \log \sum_c \lambda_i^c + \cdots + a_{n,i}^c(\log \sum_c \lambda_i^c)^n \tag{10.11}$$

where, n is the degree of the chosen polynomial. Note that one may choose either one of the formulations in Equation 10.9, 10.10 or 10.11 for each computer machine i, separately. However, service times of all transaction classes in a given machine are assumed to have identical formulation. Similarly, assume that the total CPU overhead at machine i may be expressed in one of the following ways,

$$o_i^{cpu}(\sum_c \lambda_i^c) = b_{0,i} + b_{1,i} \sum_c \lambda_i^c + b_{2,i}(\sum_c \lambda_i^c)^2 + \cdots + b_{m,i}(\sum_c \lambda_i^c)^m \tag{10.12}$$

$$o_i^{cpu}(\sum_c \lambda_i^c) = b_{0,i} + b_{1,i} e^{\sum_c \lambda_i^c} + b_{2,i}(e^{\sum_c \lambda_i^c})^2 + \cdots + b_{m,i}(e^{\sum_c \lambda_i^c})^m \tag{10.13}$$

$$o_i^{cpu}(\sum_c \lambda_i^c) = b_{0,i} + b_{1,i} \log \sum_c \lambda_i^c + \cdots + b_{m,i}(\log \sum_c \lambda_i^c)^m \tag{10.14}$$

where, m is the degree of the chosen polynomial. Again, either one of the formulations in Equation 10.12, 10.13 or 10.14 may be chosen for each computer machine i, separately. We may now rewrite Equations 10.4 and 10.5 as,

$$\forall c \in \mathcal{C}, \qquad \sum_{i \in L} \alpha_i^c \frac{s_i^c(\sum_c \lambda_i^c)}{1 - \tilde{u}_i} + d_{net}^c = R^c \tag{10.15}$$

$$\forall i \in \mathcal{L}, \qquad \sum_{c \in \mathcal{C}} \frac{\lambda_i^c}{P_i} s_i^c(\sum_c \lambda_i^c) + o_i^{cpu}(\sum_c \lambda_i^c) = u_i \tag{10.16}$$

The optimization problem would then again comprise a quadratic objective function, but this time in the set of variables $\{a_{p,i}^c, b_{q,i}, d_{net}^c, e^c, e_i | p \in \{0, 1, \ldots, n\}, q \in \{0, 1, \ldots, m\}, c \in \mathcal{C}, i \in \mathcal{L}\}$,

$$\min \sum_{c \in \mathcal{C}} \left(\frac{e^c}{\tilde{R}^c}\right)^2 + \sum_{i \in \mathcal{L}} \left(\frac{e_i}{\tilde{u}_i}\right)^2 \tag{10.17}$$

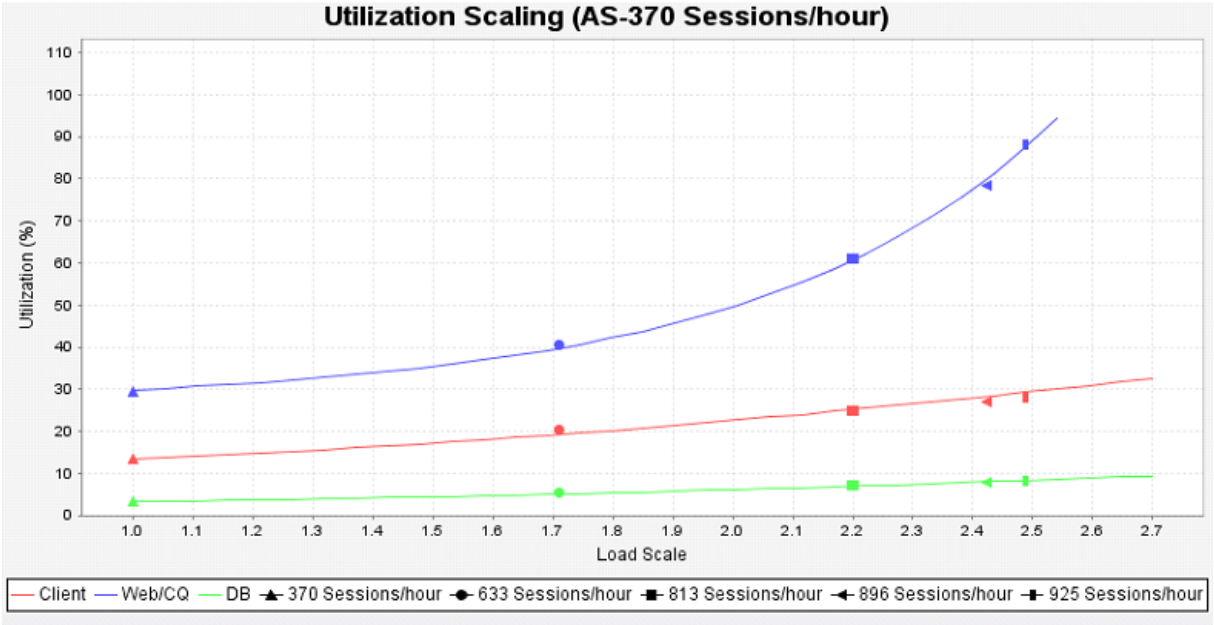


Figure 10.4: Utilization Scaling with Enhanced Inferencing

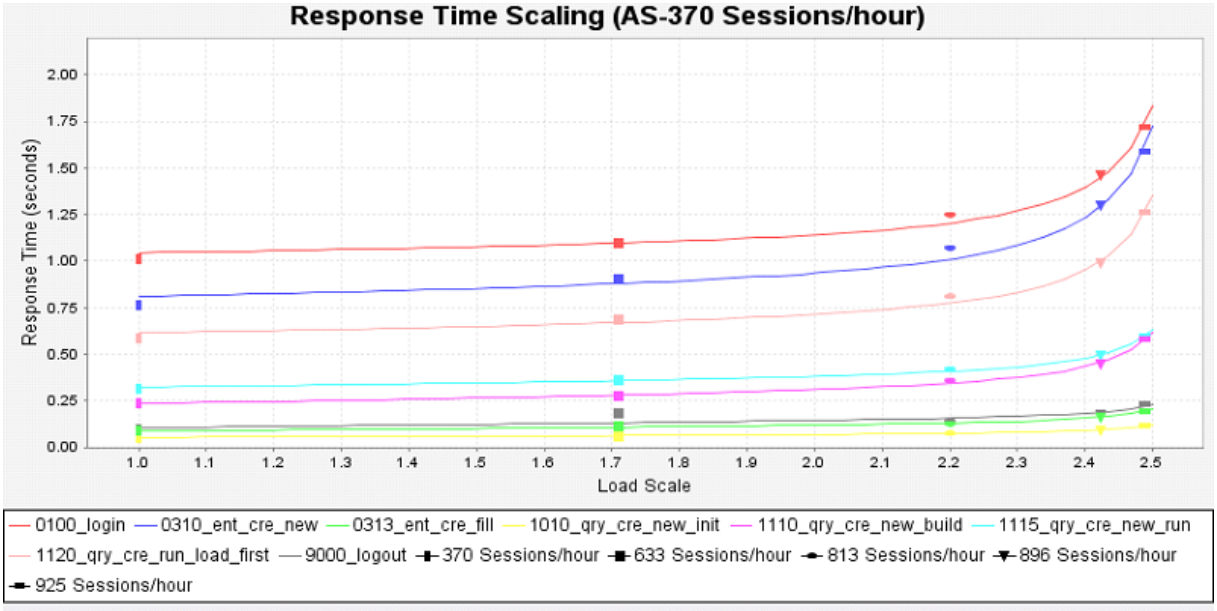


Figure 10.5: Response Time Scaling with Enhanced Inferencing

with the following set of linear constraints,

$$\forall c \in \mathcal{C}, \qquad R^c + e^c = \sum_{i \in L} \alpha_i^c \frac{s_i^c (\sum_c \lambda_i^c)}{1 - \tilde{u}_i} + d_{net}^c + e^c = \tilde{R}^c \tag{10.18}$$

$$\forall i \in \mathcal{L}, \qquad u_i + e_i = \sum_{c \in \mathcal{C}} \frac{\lambda_i^c}{P_i} s_i^c (\sum_c \lambda_i^c) + o_i^{cpu} (\sum_c \lambda_i^c) + e_i = \tilde{u}_i \tag{10.19}$$

The solution to this optimization problem will give us the service time and CPU overhead polynomial coefficients, $a^c_{p,i}$ and $b_{q,i}$, and network delays, d^c_{net}, required as part of the solution to the *Enhanced Inferencing* problem. Note that these coefficient values would be constant resulting in purely sum arriving workload dependent service time and CPU overhead formulations. Also note that Inferencing described in Section 10.2.2 is only a particular case of the Enhanced Inferencing technique described here with constant zero degree polynomials.

10.3.1 Example revisited with Enhanced Inferencing

AMBIENCE tool was modified to implement the Enhanced Inferencing algorithm described above in Section-10.3. The service times and CPU overheads are now represented by the polynomial coefficient vectors instead of constant values. We regenerated a performance model for the example discussed previously in Section-10.2.4 using the modified AMBIENCE tool that now uses the Enhanced Inferencing algorithm. Figure-10.4 shows the CPU utilization scaling chart. We see that the model predicted utilization is now non-linear in variation for the "Web/CQ" machine and matches well with the measured utilization. For the "Client" and "DB" machines they match well just like before. Figure-10.5 shows the response time scaling chart for different transactions. This time we see that model predicted and measured response times match well for all the transactions. This improved performance model was represented by the service times and CPU overheads being degree 1 polynomial functions of the exponential of sum workload. Thus we see that with Enhanced Inferencing based on workload-dependent parameters, AMBIENCE shows significant improvement in performance model fitting.

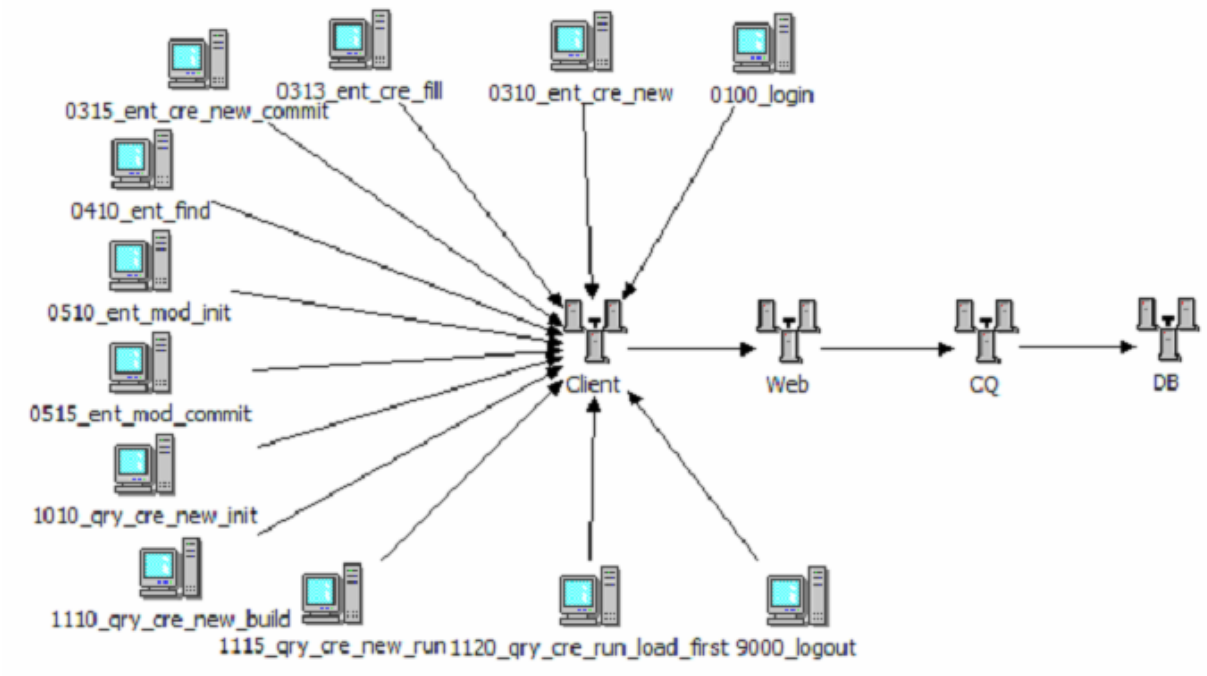


Figure 10.6: Modified Application and Network topology

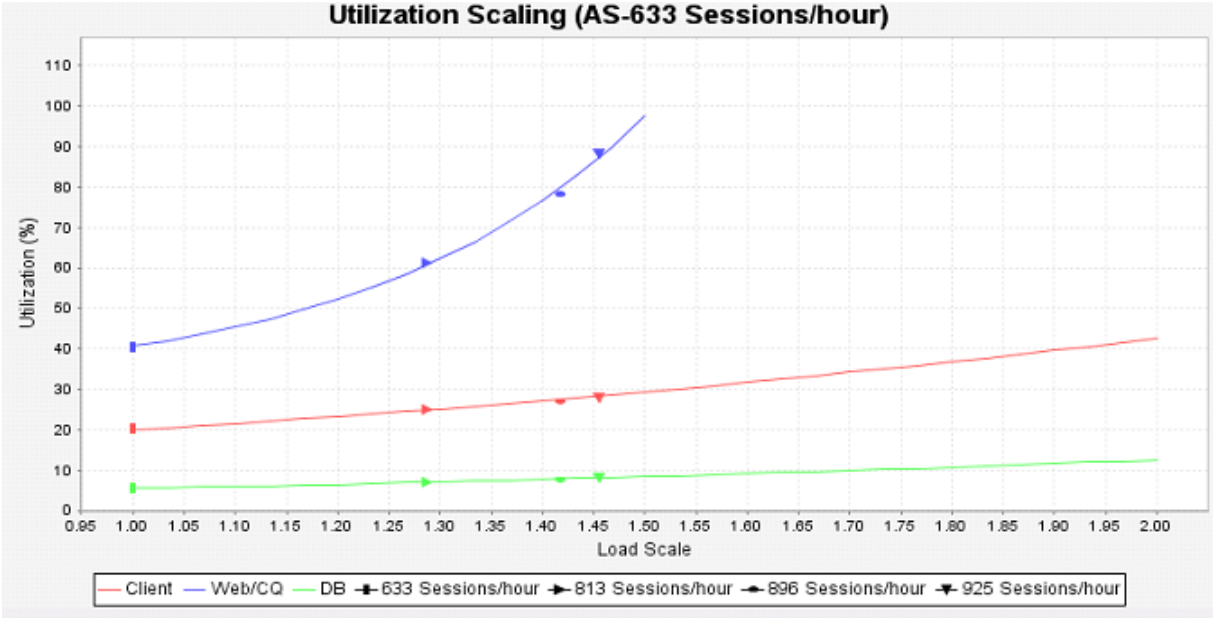


Figure 10.7: Utilization Scaling for "Web/CQ" machine in original topology

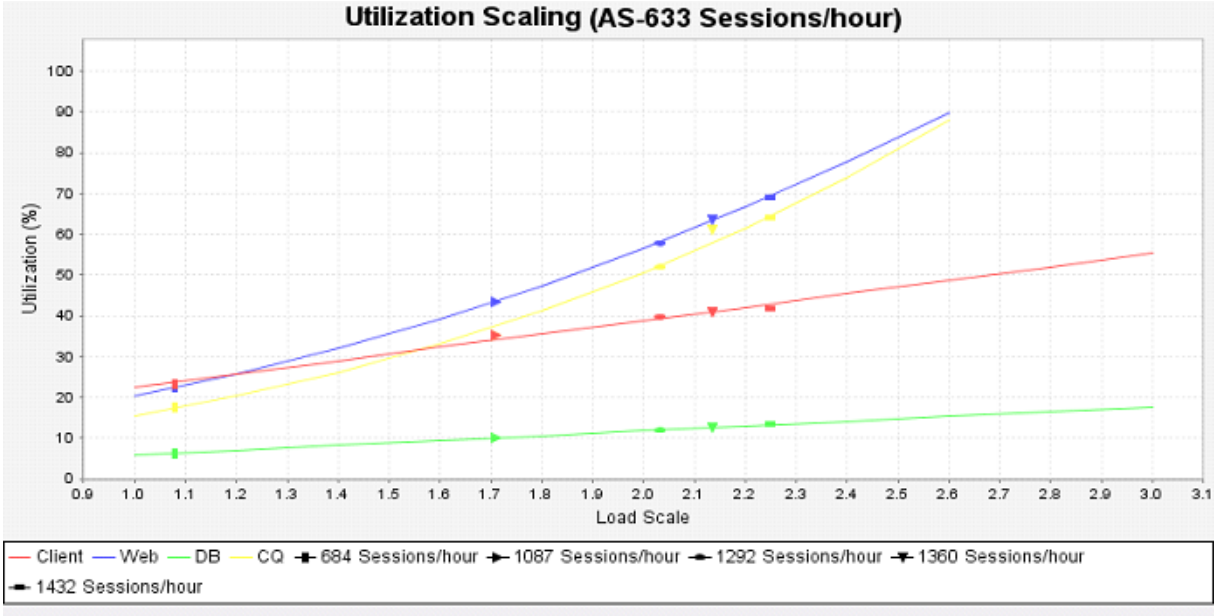


Figure 10.8: Utilization Scaling for "Web" and "CQ" machines in modified topology

10.3.2 Another case study with Enhanced Inferencing

Here we present another case study in relation to the example studied in Sections-10.2.4 and 10.3.1. Similar to the topology shown in Figure-10.1, Figure-10.6 shows a modified topology with the webserver and ClearQuest applications installed on different machines.

Figures-10.7 and 10.8 show CPU utilization scaling charts corresponding to performance models for Figures-10.1 and Figure-10.6, separately. As discussed already in Section-10.3.1, in Figure-10.7 the overall utilization for "Web/CQ" machine scales exponentially.

The CPU overhead is infered to be a degree 1 polynomial in exponential of sum workload. It is explicitly given by: $0.2966 + 0.00267exp$(sum arrival workload). Also we see that the "Web/CQ" machine reaches 100% utilization bottleneck at about 950 sessions/hour.

In Figure-10.8, the overall utilizations for "Web" and "CQ" machines scale quadratically. The CPU overheads are infered to be degree 2 polynomials in simple sum workload. They are explicitly given by: $0.07851 + 0.008728$(sum arrival workload)^2 and $0.01251 + 0.000992$(sum arrival workload)^2 for the "Web" and "CQ" machines, respectively. Also note that each of the "Web" and "CQ" machines reach 100% utilization bottleneck at about 1680 sessions/hour.

Thus, we have seen here that separating the webserver and ClearQuest applications and installing them on two different machines instead of a single one, can increase the capacity of the system to process transaction workload from about 950 sessions/hour to 1680 sessions/hour. Moreover, we can explicitly quantify the CPU overhead scaling variations as shown by the formulae above.

10.4 Real Time Performance Modeling

As discussed in Section-10.1, Inferencing assumes arriving workload to be stationary and builds a performance model based on workload-independent service times and overhead parameters. With Enhanced Inferencing based on workload-dependent parameters, AMBIENCE shows significant improvement in performance model fitting and approximation. It generates a more accurate model that better captures the real performance characteristics. However, Enhanced Inferencing still assumes arriving workload to be stationary. Real world transactional workload on the other hand may not always be stationary due to peak/off-peak volumes, daily/weekly patterns and diversity in user behavior. If arriving transactional workload has fast changing non-stationary characteristics, it may be difficult to compute *average* measurement values of the arriving workload, response times and CPU utilizations that are representative of a sufficiently long measurement time window. Moreover, since both Inferencing and Enhanced Inferencing solve a one-step optimization problem using average measurement values, they can only be used in an off-line manner when average measurement values can be computed from an available time series of measurement data. These techniques can not be used to perform *real time* tracking of service time, network delay and CPU overhead model parameter estimates. Real time tracking of model parameters can be beneficial in many ways,

1. Facilitates autonomic and real time operation of performance modeling and prediction which can be beneficial for time sensitive applications like financial applications, military and space applications, etc.

2. Creates a dynamic performance model that can adapt to abrupt and unexpected changes in arriving workload or other system parameters.

3. Allows real time resource partitioning and management of computer systems.

Online tracking of a performance model requires something more than just solving a one-step optimization problem based on mean measurement values gathered over a limited time window. Kalman filter provides the required framework for estimating model parameters in a real-time fashion. Kalman filter is a minimum mean-square error (MMSE) estimator that estimates the state of a dynamic system from a series of incomplete and *noisy* measurements. Until now we have seen that both Inferencing techniques *minimize the square of error in mean measurement values*. However, Kalman filter *minimizes the mean of the square of the estimation error* and operates by propagating both the mean and covariance of estimates through time. This makes Kalman filter more robust. Kalman filter may be extended to the even more robust version: the H^{∞} filter (also called a minimax filter) which minimizes the *worst-case* estimation error, i.e., it minimizes not just the mean of the square of the estimation error (second moment), but also higher order moments of the estimation error. Moreover, the H^{∞} filter may be augmented by an H^{∞}-optimal controller that may be employed for robust management of resources such as computer hardware (provisioning) and thread pool, in order to provide the required QoS or SLA targets. In this work we focus only on the use of Kalman fiter and we shall show here how it can be used in conjunction with the queueing theory based performance models to compute a series of real time estimates of service time and network delay model parameters.

Let us treat a computer network system as a dynamical system. As before, measurements on arriving workload, response times and CPU utilizations can be easily gathered say at a sampling interval of T seconds, to produce time series measurement data. Instead of working with mean values of this time series data that we have been doing until now, we shall make use of the whole time series itself as an input to the Kalman filter. Workload arriving to the computer system may have fast changing non-stationary characteristics and service times may be workload dependent when considering a long duration of time (few minutes to several hours). Assume that T is small enough (few seconds) so that the arriving workload can be considered stationary during this sampling interval. Then the stationary queueing theory based performance models hold during this sampling interval. If we consider the *state* of the system to comprise service time and network delay then Kalman filter can be used to compute a time series estimate of this state that would be inherently workload dependent. The reason being that unlike in Inferencing we are not estimating a mean value of service time or network delay that is valid for the entire duration over which the mean arriving workload was computed. Here, each estimate of the state at each sampling interval is valid for that interval itself and depends on the arriving workload measured during that interval.

In this work we assume that the reader is aware of the general theory of Kalman filtering. The reader is referred to [Sim06] for the same. We shall first consider a simple case with only single machine and single traffic class before demonstrating the effectiveness of this approach for single machine and multiple traffic classes. Extending this approach to multiple machines and multiple traffic classes for a generic architecture is straight forward and part of our ongoing work.

10.4.1 Single machine and single traffic class

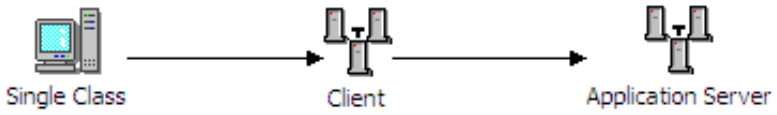


Figure 10.9: Simple scenario with single transaction class and machine

Consider a simple scenario with only one physical machine and a single traffic class as shown in Figure-10.9. For this simple system we shall consider the state of the system to comprise only service time, s. The server utilization u is also included in the state, but only because it facilitates the overall formulation of Kalman filter. With the notation described in Section 10.2, define the system state $\boldsymbol{x}$ as follows,

$$\boldsymbol{x} = \begin{bmatrix} u \\ s \end{bmatrix}, \tag{10.20}$$

and define the measurement model $h(\cdot)$ from Section 10.2 as follows,

$$\boldsymbol{z} = h(\boldsymbol{x}) = \begin{bmatrix} \lambda \\ R \end{bmatrix} = \begin{bmatrix} u/s \\ s/(1-u) \end{bmatrix}. \tag{10.21}$$

If each sampling interval is denoted by k then we may assume the following dynamics for the state evolution,

$$\boldsymbol{x_k} = \boldsymbol{F_k x_{k-1}} + \boldsymbol{w_k},$$

or,

$$\begin{bmatrix} u_k \\ s_k \end{bmatrix} = \begin{bmatrix} 1 & \lambda_k - \lambda_{k-1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{k-1} \\ s_{k-1} \end{bmatrix} + \boldsymbol{w_k},$$

where, F_k is the state transition model which is applied to the previous state $\boldsymbol{x_{k-1}}$ and $\boldsymbol{w_k}$ is the process noise which is assumed to be drawn from a zero mean multivariate normal distribution with covariance $\boldsymbol{Q_k}$, i.e.,

$$\boldsymbol{w_k} \sim N(0, \boldsymbol{Q_k}).$$

The iterative measurement model is taken to be,

$$\boldsymbol{z_k} = \boldsymbol{H_k x_k} + \boldsymbol{v_k},$$

where, $\boldsymbol{H_k}$ is the observation model which maps the true state space into the observed space and $\boldsymbol{v_k}$ is the observation noise which is assumed to be zero mean Gaussian white noise with covariance $\boldsymbol{R_k}$, i.e.,

$$\boldsymbol{v_k} \sim N(0, \boldsymbol{R_k}).$$

Since the measurement model in Equation 10.21 is non-linear in terms of system state parameters, we must use the 'Extended' version of the Kalman filter [Sim06]. Then, the corresponding Jacobian matrix of the measurement model is given by,

$$\boldsymbol{H} = \frac{\partial h}{\partial \boldsymbol{x}} = \begin{bmatrix} 1/s & -u/s^2 \\ s/(1-u)^2 & 1/(1-u) \end{bmatrix}$$

and $\boldsymbol{H_k}$ can be computed as,

$$\boldsymbol{H_k} = \left[\frac{\partial h}{\partial \boldsymbol{x}}\right]_{\hat{\boldsymbol{x}}_{k|k-1}}.$$

Now, we may use the standard Extended Kalman filtering theory [Sim06] to track the system state over time. One of the major advantages of Kalman filter is that it is a recursive estimator. This means that only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state. In the following Extended Kalman filtering algorithm, the notation $\hat{\boldsymbol{x}}_{n|m}$ represents the estimate of $\boldsymbol{x}$ at time n given observations up to and including time m. The state of the filter is represented by two variables:

1. $\hat{\boldsymbol{x}}_{k|k}$, estimate of state at time k given observations up to and including time k.

2. $\boldsymbol{P}_{k|k}$, error covariance matrix (a measure of estimated accuracy of the state estimate).

The Kalman filter algorithm has two distinct phases: Predict and Update. The predict phase uses the state estimate from the previous timestep to produce an estimate of the state at the current timestep. In the update phase, measurement information at the current timestep is used to refine this prediction to arrive at a new, more accurate state estimate, again for the current timestep. These two phases are given as,

Predict:

$$\boldsymbol{\hat{x}_{k|k-1}} = \boldsymbol{F_k}\boldsymbol{\hat{x}_{k-1|k-1}}$$

$$\boldsymbol{P_{k|k-1}} = \boldsymbol{F_k}\boldsymbol{P_{k-1|k-1}}\boldsymbol{F_k^T} + \boldsymbol{Q_k}$$

Update:

$$\boldsymbol{\tilde{y}_k} = \boldsymbol{z_k} - \boldsymbol{h}(\boldsymbol{\hat{x}_{k|k-1}})$$

$$\boldsymbol{S_k} = \boldsymbol{H_k}\boldsymbol{P_{k|k-1}}\boldsymbol{H_k^T} + \boldsymbol{R_k}$$

$$\boldsymbol{K_k} = \boldsymbol{P_{k|k-1}}\boldsymbol{H_k^T}\boldsymbol{S_k^{-1}}$$

$$\boldsymbol{\hat{x}_{k|k}} = \boldsymbol{\hat{x}_{k|k-1}} + \boldsymbol{K_k}\boldsymbol{\tilde{y}_k}$$

$$\boldsymbol{P_{k|k}} = (\boldsymbol{I} - \boldsymbol{K_k}\boldsymbol{H_k})\boldsymbol{P_{k|k-1}}$$

10.4.1.1 Implementation Results

We implemented the above described Kalman filtering algorithm in a stand alone application written in Java. Measurements were gathered for the system shown in Figure-10.9 and input as time series data to the Kalman filter. Figure-10.10 shows the time series data for both measurements and the estimated service time and utilization. We see that the service time is workload dependent and increases with the increase in measured arrival rate of workload. Recall from previous discussion in Section-10.4.1 that the utilization u was included in the state only to facilitate the overall formulation of Kalman filter. In Figure-10.10 we can see that the estimated utilization matches well with the measured utilization.

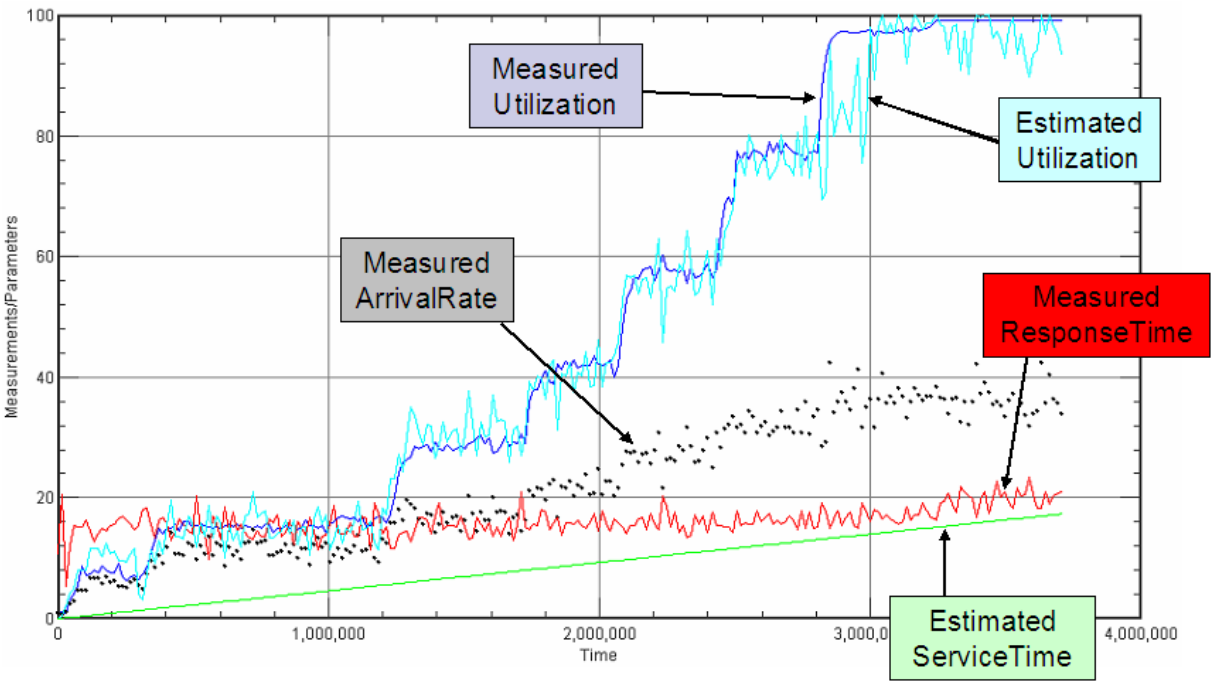


Figure 10.10: Simple case of single transaction class and machine

10.4.2 Single machine and multiple traffic classes

We shall now consider the case of a single physical machine and multiple workload traffic classes. Here we shall consider an example for 3 classes as shown in Figure-10.11.

For 3 classes indexed as class a, b and c and a single machine, based on Equations 10.4 and 10.5 we may define the system state $\boldsymbol{x}$ as,

$$\boldsymbol{x} = \begin{bmatrix} s^a & s^b & s^c & d^a & d^b & d^c \end{bmatrix}^T, \tag{10.22}$$

where, s^a, s^b and s^c are service times at the only machine for classes a, b and c, respectively, and d^a, d^b and d^c are network delays for the three classes. The measurement model $h(\cdot)$ is

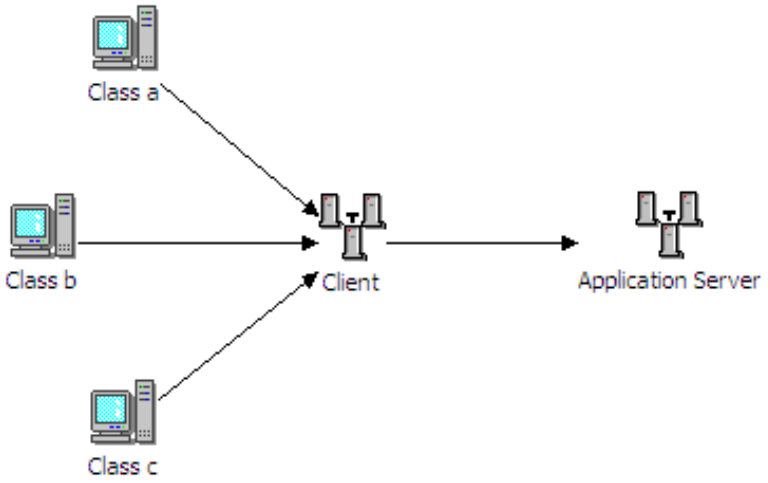


Figure 10.11: Three transaction classes and single machine

defined as,

$$\boldsymbol{z} = h(\boldsymbol{x}) = \begin{bmatrix} R^a \\ R^b \\ R^c \\ u \end{bmatrix} = \begin{bmatrix} \frac{s^a}{1-u} + d^a \\ \frac{s^b}{1-u} + d^b \\ \frac{s^c}{1-u} + d^c \\ \frac{1}{P}(\lambda^a s^a + \lambda^b s^b + \lambda^c s^c) \end{bmatrix}. \tag{10.23}$$

The state evolution model is taken as,

$$\boldsymbol{x_k} = \boldsymbol{F_k x_{k-1}} + \boldsymbol{w_k},$$

with $\boldsymbol{F_k} = I$ and the measurement model is taken to be the same as before,

$$\boldsymbol{z_k} = \boldsymbol{H_k x_k} + \boldsymbol{v_k}.$$

The corresponding Jacobian matrix of the measurement model is given by,

$$\boldsymbol{H} = \frac{\partial h}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{1}{1-u} & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{1}{1-u} & 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{1-u} & 0 & 0 & 1 \\ \frac{\lambda^a}{P} & \frac{\lambda^b}{P} & \frac{\lambda^c}{P} & 0 & 0 & 0 \end{bmatrix}$$

and $\boldsymbol{H_k}$ may again be computed as, $\boldsymbol{H_k} = \left[\frac{\partial h}{\partial \boldsymbol{x}}\right]_{\hat{\boldsymbol{x}}_{k|k-1}}$. Now, we may again use the Extended Kalman filtering algorithm described previously in Section-10.4.1 to track the system state given by Equation-10.22 over time.

10.4.2.1 Implementation Results

Previously in Section-10.4.1 the application of Figure-10.9 was designed in such a way that the service time was workload dependent and the generated workload varied slowly. For

the case being discussed here we designed a sample application which employed a periodic workload generator and the service times and network delays were workload independent. Figures-10.12 and 10.13 show gathered time series data for the workload arrival rate and response time measurements, respectively, for the three different classes a, b and c. Note that workload arrival rate is highly non-stationary over a small window of time of few seconds. Also notice the intentional switch of workload and service time parameters, twice, between classes a and c. The filter should be easily able to detect this switch of parameters which demonstrates the power and effectiveness of our approach as compared to Inferencing like techniques. Inferencing like techniques can not be easily used to detect such switches since they work with measurement values averaged over the entire run and not the entire time series measurement data itself. Figure 10.14 shows CPU utilization measurements for the only application server machine.

These measurements were input to the Kalman filter and time series estimates of service times and network delays for all 3 classes were obtained. Figures-10.15 and 10.17 show these service time and network delay estimates, respectively, represented by the thin lines. The thick flat lines represent the average expected service times and network delays that were programmed in the sample application. The estimates are workload independent as expected (i.e., do not change with periodically changing workload), but vary around the average expected values. This variation may be attributed to the queueing effect. Also notice that the filter was easily able to detect the switch of parameters between classes a and c both the times. The service time estimates however show some spiky behavior around the switch instants. Another observation to be noted is that the estimates converge slowly to their new values just after the switch. The filter is slow in responding to abrupt switches in the model parameters.

We augmented the Kalman filter with a *trick* algorithm in order to make the filter respond more swiftly to abrupt switches in model parameters. We shall not mention details about this trick algorithm here as we are in the process of improving it further and patenting it. Figures-10.16 and 10.18 show service time and network delay estimates with the augmented Kalman filter. We clearly see that the filter now converges faster to the new values just after the switch.

10.5 Conclusion

In this chapter, we extended the Inferencing technique that assumes arriving workload to be stationary and performance model parameters to be workload independent, to propose Enhanced Inferencing which considers model parameters to be workload dependent. Enhanced Inferencing demonstrated significant improvement in performance model fitting over Inferencing. We then proposed an innovative approach of using Kalman filtering with Inferencing like performance models in order to be able to pursue *real time* performance modeling for non-stationary arriving workloads. This real time performance model not only works reliabily for non-stationary workloads, but also incorporates workload dependent model parameters.

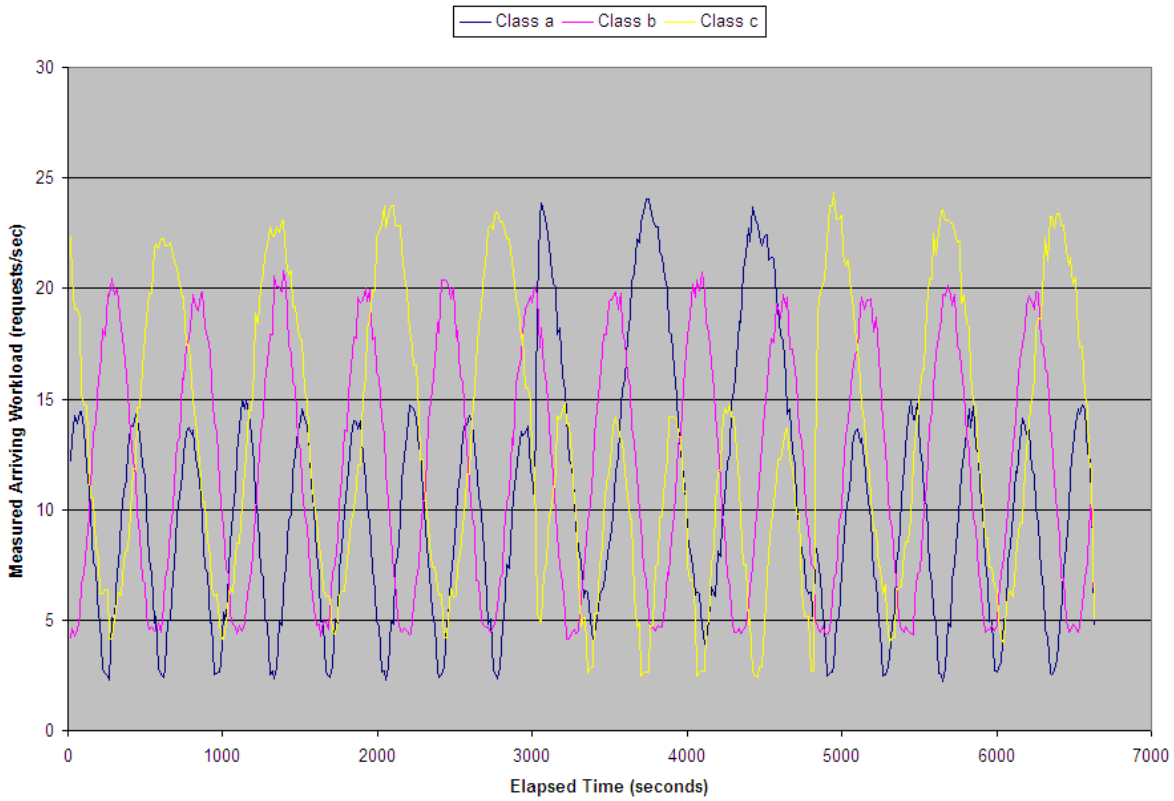


Figure 10.12: Workload for three transaction classes and single machine scenario

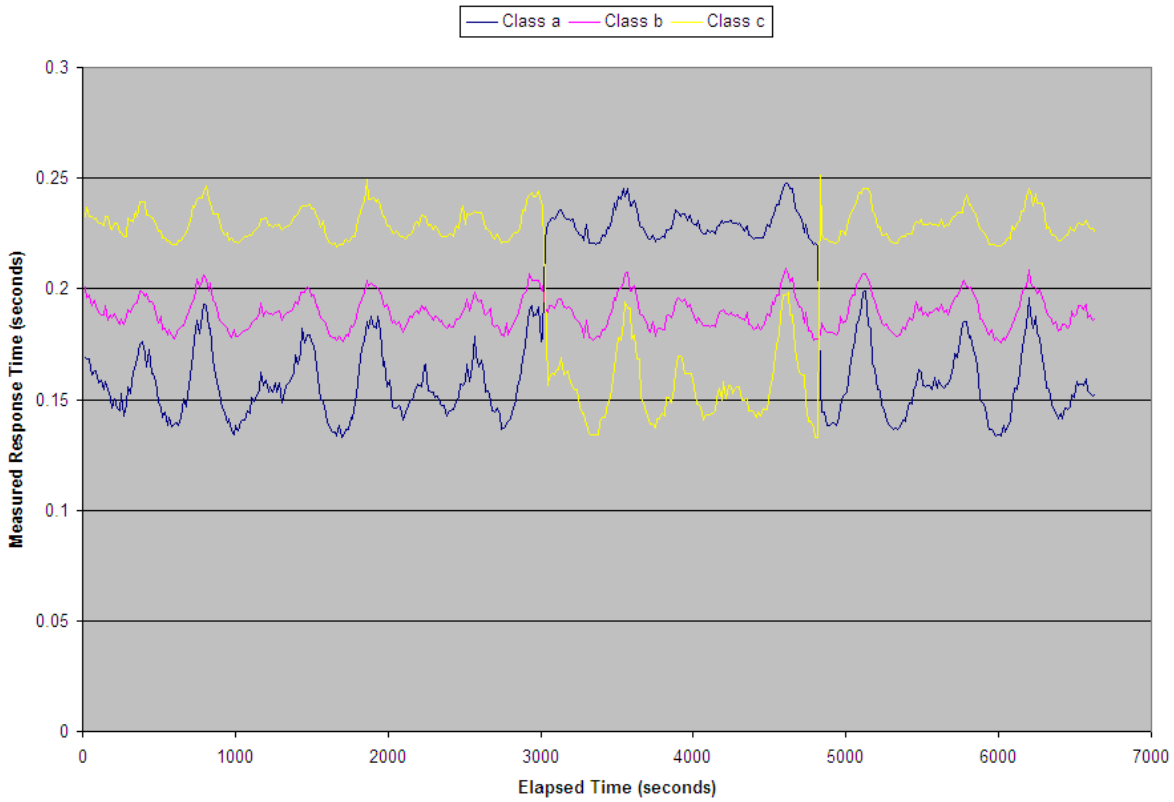


Figure 10.13: Response Times for three transaction classes and single machine scenario

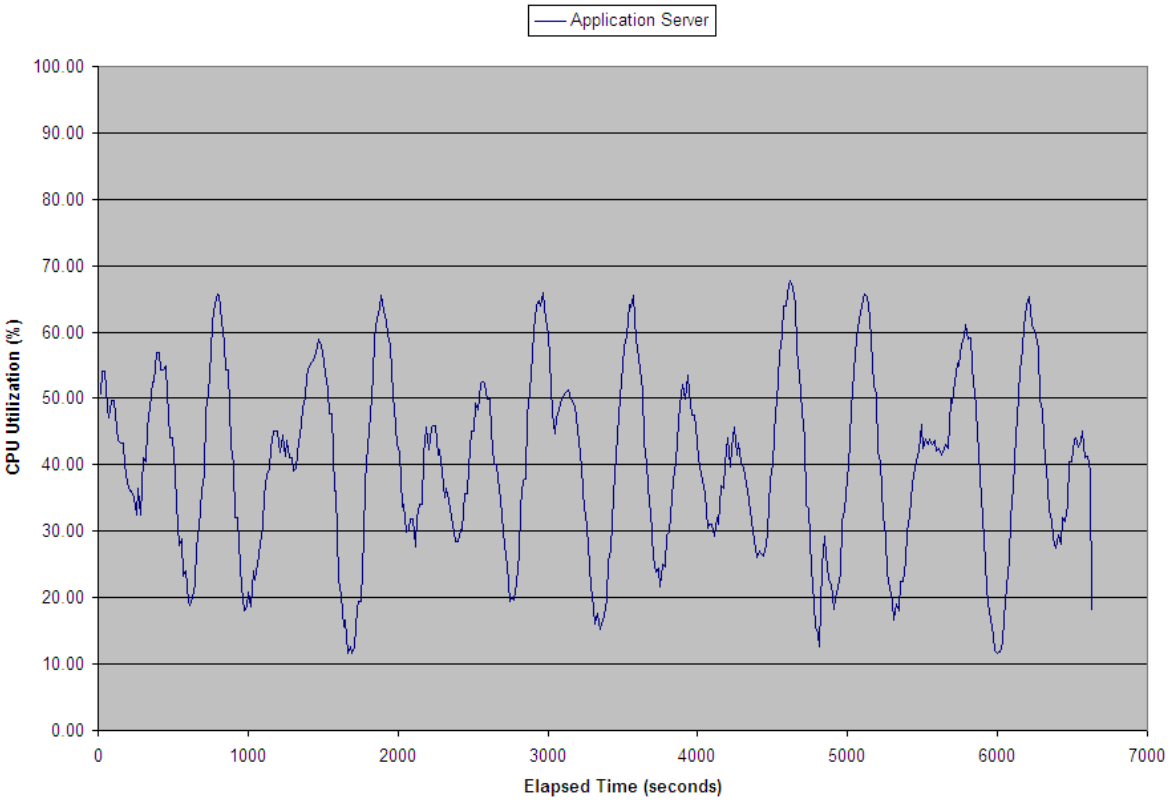


Figure 10.14: CPU utilization for three transaction classes and single machine scenario

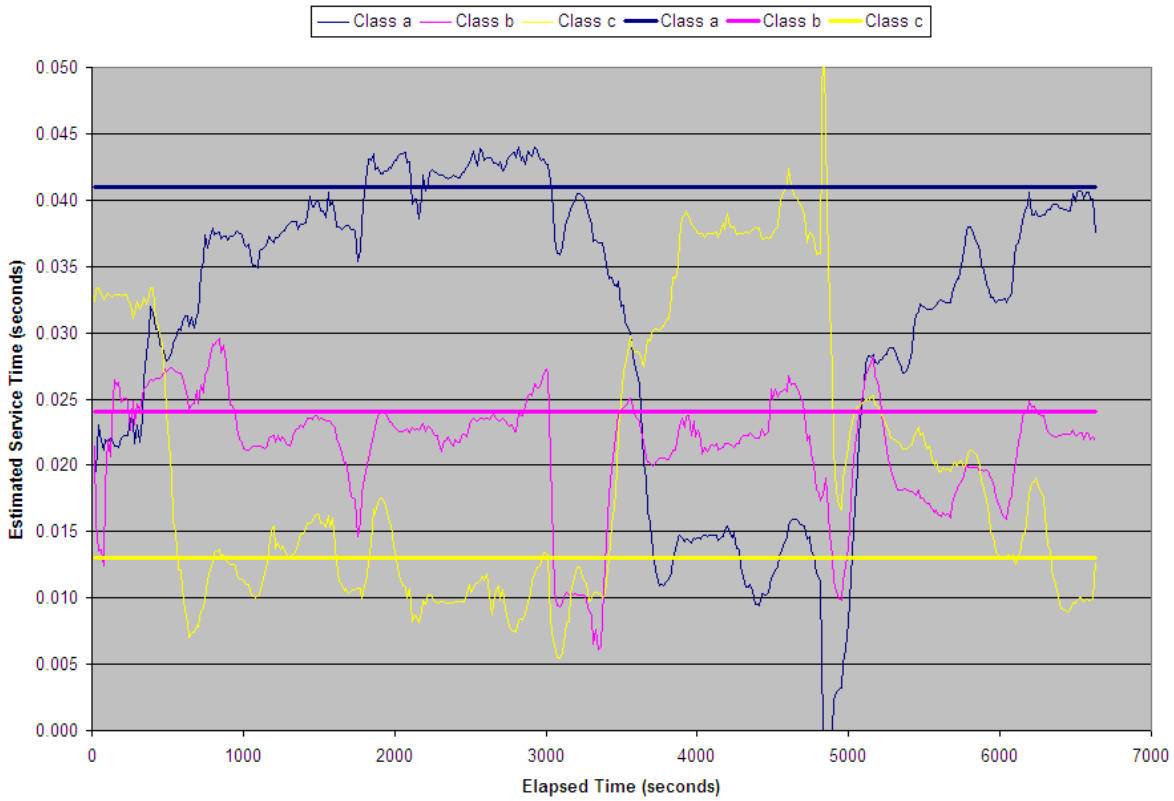


Figure 10.15: Service time estimates for the three classes without trick algorithm

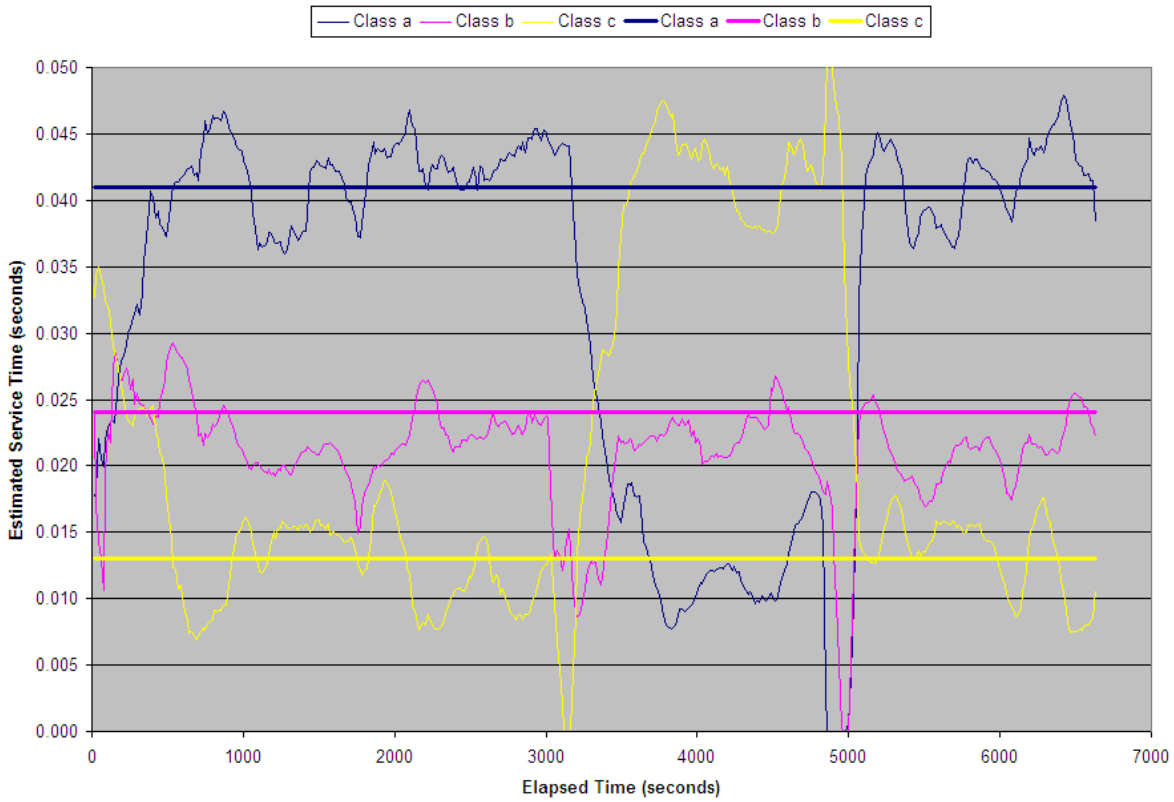


Figure 10.16: Service time estimates for the three classes with trick algorithm

Figure 10.17: Delay estimates for the three classes without trick algorithm

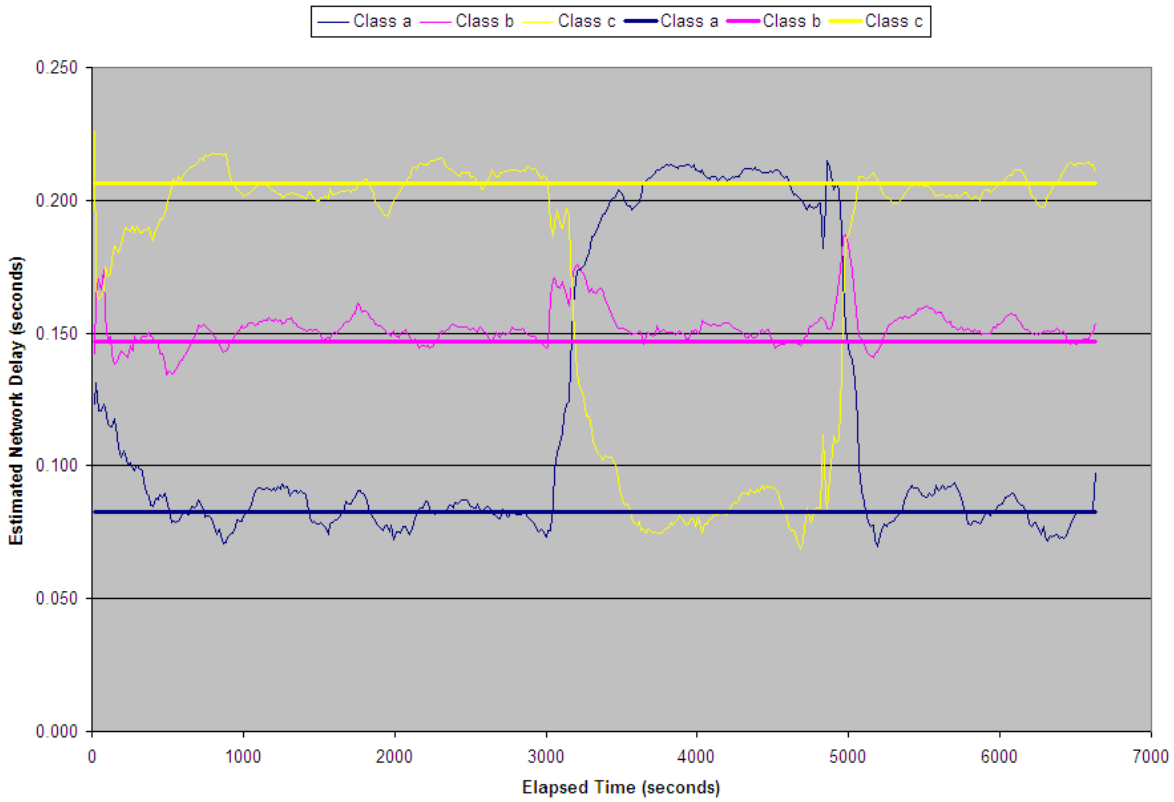


Figure 10.18: Delay estimates for the three classes with trick algorithm

Chapter 11

Conclusion

The main focus of this thesis has been various optimization and control problems related to wireless access and ad hoc networks. In the last chapter however, we presented 'real time' performance modeling of wired, online computer network systems.

In Chapter-2, we studied globally optimal user-network association in an integrated 802.11 WLAN and 3G UMTS hybrid cell. The main results were in obtaining of a stationary optimal policy possessing both 'mobile-greedy' and 'load-balancing' properties with a neither convex nor concave type switching curve structure. Threshold type and symmetric switching curves were observed for the analogous homogenous network cases.

In Chapter-3, we studied individually optimal user-network association in an integrated WLAN and UMTS hybrid cell. The main result was in obtaining of the Nash equilibrium achieving association policy which was observed to possess a descending staircase curve structure.

In Chapter-4, we studied cooperative and non-cooperative control in a single cell 802.11 WLAN. Comparison between the two control mechanisms revealed that network performance in the non-cooperative game scenario is inefficient as compared to the cooperative scenario.

In Chapter-5, we proposed a Markovian stochastic framework to model the performance of a simple Fountain Codes based Transport (FCT) protocol in a single cell IEEE 802.11 WLAN. A detailed performance analysis study was carried out to provide insights into the choice of various system parameters that can lead to optimal network performance. We also presented a brief comparison between the performance of FCT and TCP through simulations.

In Chapter-6, we first proposed and analyzed three new basic threshold-based channel switching policies for UMTS. We further proposed and evaluated a new improved cross-layer switching policy that we called FS-DCH (*at-least* flow-size threshold on DCH) policy. Extensive simulation results demonstrated that FS-DCH policy improves over others by

about 30% to 36% in response time metrics for a particular case.

In Chapter-7, in a dense multi-hop network of mobile nodes capable of applying adaptive power control, we considered the problem of finding the optimal hop distance that maximizes a certain throughput measure in *bit-metres/sec*, subject to average network power constraints. Using numerical analysis we discovered that choosing the nearest node as next hop is not always optimal. Optimal throughput performance can also be attained at non-trivial hop distances depending on the available average network power.

The main goal of Chapter-8 was to better understand the routing dynamics in VANETs that are a special class of MANETs but exhibit very different behavior from them. We considered the problem of optimal next hop selection in a route between two vehicles for a simple scenario of VANETs on a fast track highway. Explicit expressions for optimal choice of next hop node's speed and inter-node distance were obtained for certain fast track highway scenarios of interest. A monotone variation property of the speed of relay nodes under the optimal policy was proved. The optimal policies and their structures can assist in enhancing the performance of existing VANET routing protocols.

In Chapter-9, we identified a special structure inherent to the solution of Dynamic Programming (DP) problem arising in optimization over paths. According to this structure, the optimal policy tries to equalize the link metrics of adjacent links in a multi-hop route.

In Chapter-10, we extended the Inferencing technique for generating arriving workload dependent service time and CPU overhead based performance models. We called this extended form of Inferencing as Enhanced Inferencing. Implementation of this Enhanced Inferencing in AMBIENCE demonstrated significant improvement in performance model fitting and approximation. We further presented an innovative approach of using Kalman filtering with Inferencing like performance models in order to be able to pursue real time performance modeling of production environment computer application systems that face fast changing non-stationary arriving workloads.

Appendix A

Présentation des Travaux de Thèse

11.1 Introduction

Les réseaux sans-fil de communication et informatique ont connu une croissance exponentielle au cours de la dernière décennie. Les téléphones cellulaires et dispositifs de connectivité sans-fil sont devenus un outil essentiel et une partie de la vie quotidienne dans de nombreux pays. Alors que des technologies de téléphone cellulaire de la troisième génération (3G) telles que 'Universal Mobile Telecommunications System' (UMTS) ont récemment été déployés dans certains pays, les réseaux locaux sans-fil (WLAN) tels que ceux basés sur la norme IEEE 802.11 ont été complétant ou remplaçant les réseaux câblés dans de nombreux maisons, les entreprises et sur les campus depuis quelques années. De nombreux industriels et organismes gouvernementaux ont également été impliqués dans le développement des réseaux ad hoc des véhicules (VANETs), afin de permettre les communications entre véhicules en proximité et entre les véhicules et l'infrastructure réseau à proximité de routiers.

La croissance explosive des systèmes sans-fil en même temps que la prolifération des appareils sans-fil tels que les téléphones 3G, ordinateurs portables WiFi et de dispositifs sans-fil de véhicules, suggère un bel avenir pour les réseaux sans-fil, en tant que systèmes ad hoc tout seuls et dans le cadre de la plus grande infrastructure de réseau. Toutefois, de nombreux défis techniques restent encore à la conception des réseaux sans-fil robuste qui peut fournir les performances nécessaires pour appuyer les nouvelles applications. Ces défis techniques ont donné lieu à de nombreux problèmes de recherche qui ont attiré beaucoup d'attention dans la communauté de la recherche. Cette thèse traite certains de ces problèmes de recherche et propose plusieurs nouveaux résultats qui peuvent être utilisés pour améliorer les performances des systèmes et de la technologie sans-fil existants. En plus de

systèmes sans-fil, nous étudions et présentons également une approche innovatrice pour la modélisation de la performance des systèmes réseaux informatiques en *temps réel*.

Les réseaux sans-fil peuvent être classés en deux catégories, à savoir, réseaux *d'accès* sans-fil et réseaux *ad hoc* sans-fil. Les réseaux d'accès sans-fil fournissent généralement la connectivité à l'infrastructure de réseau câblé par l'intermédiaire du moyen sans-fil. Exemples de réseaux d'accès sans-fil inclurent des systèmes cellulaire tels que 2G GSM, 3G UMTS, etc. et des systèmes WiFi tels que les réseaux locaux sans-fil 802.11 WLANs. Les réseaux ad hoc sans-fil d'autre part, sont des réseaux sans-fil décentralisés qui peut être installé et démonté dynamiquement. Les réseaux ad hoc sans-fil peut être classés comme les réseaux ad hoc des mobiles (MANET), les réseaux maillés sans-fil et réseaux de capteurs sans-fil. Les VANETs sont un type particulier de MANET qui présentent un comportement radicalement différent de l'habituel MANET. Dans cette thèse, nous allons limiter notre attention à l'étude des technologies d'accès 3G UMTS et 802.11 WLAN et les réseaux ad hoc des mobiles et des véhicules.

Avec la baisse des coûts de matériel, certains appareils sont maintenant disponibles sur le marché qui fournissent à la fois WiFi et connectivité cellulaire à des prix abordables. Ces appareils peuvent détecter la présence de l'une ou l'autre des deux réseaux et de passer entre eux, lorsqu'ils sont disponibles. Avec de plus en plus d'utilisateurs abonnés à des services à large bande sans-fil, ces dispositifs leur permettent d'avoir accès aux hot-spot WLAN et réseaux cellulaires UMTS à la fois. Ainsi, les opérateurs de réseaux cherchent à offrir une connectivité sans soudure et omniprésent à des utilisateurs mobiles par le biais des réseaux *hybrides* 802.11 WLAN et 3G UMTS intégré. Ces réseaux hybrides sont de plus en plus étudié par les opérateurs de réseaux comme un moyen convergent de fournir la connectivité pour les applications de la voix et des données en même temps. Dans cette thèse, nous avons deux chapitres consacrés à l'étude d'association optimale d'utilisateur et réseau dans une cellule hybride de WLAN et UMTS.

Contrôle de la puissance et taux PHY sont deux mécanismes qui sont essentiels à la réalisation d'un fonctionnement efficace d'un WLAN. Contrôle de la puissance et taux PHY sont également très souvent utilisés pour fournir la qualité de service (QoS) et les deux sont utiles pour l'obtention d'un canal radio à faible taux d'erreur binaire (BER). La spécification IEEE 802.11 pour les appareils mobiles permet la personnalisation de certains paramètres de fonctionnement critique comme taux PHY et taille de données de couche MAC et de tout fabricant ou même l'utilisateur peut adaptativement sélectionner le taux PHY de transmission de données pour atteindre un taux contrôlé. À cette fin, plusieurs algorithmes d'auto-sélection des taux [KL97, HVB01, GCNC01, AKKD01, QCJS03] ont été proposées dans la littérature et la plupart d'entre eux permettent les noeuds WLAN d'sélectionner leurs taux PHY dans une manière non-coopératif, par définition. Dans cette thèse, nous avons un chapitre sur des études d'inefficacité de l'IEEE 802.11 MAC protocole en vertu de contrôle non-coopératif de la puissance et le taux.

'Fountain Codes' sont des 'rateless erasure codes' et offrent un avenir très prometteur pour améliorer les techniques existantes de transmission de paquets des données. Type spécifique de fountain codes tels que les codes Luby Transform (LT), les codes Raptor, etc., peuvent être utilisés par un expéditeur pour générer des paquets codés à partir des

paquets sources de données *sur la volée* et le nombre de ces paquets codés peuvent être potentiellement illimité. Si le fichier d'origine à côté expéditeur comprend N_p paquets, puis par décodage de n'importe quel $N_p(1+\epsilon)$ (un peu plus de N_p) fontaine codé paquets reçu, le récepteur peut récupérer l'ensemble du fichier avec une probabilité $1-\delta$. La probabilité d'échec à décoder le fichier, δ, est délimitée par au-dessus de $\delta \leq 2^{-\epsilon N_p}$ et dépend de la *degré de distribution* utilisés pour coder les paquets à côté expéditeur. Un protocole de transport basée sur les 'Fountain Codes' (FCT), repose sur un paradigme autre que ce du omniprésent TCP. Elle supprime la nécessité d'un mécanisme de rétroaction inverse, habituellement essentiel pour assurer la fiabilité dans transmission de paquets de données. Absence d'un mécanisme de rétroaction inverse peut sensiblement améliorer les performances des réseaux avec des canaux sans-fil demi-duplex (comme 802.11 WLANs), où les collisions entre transmissions avant et arrière de données MAC contribuent de manière significative à la dégradation des performances. Dans cette thèse, nous avons un chapitre consacré à la modélisation et l'analyze de la performance d'un simple protocole FCT dans une seule cellule IEEE 802.11 WLAN.

Il existe deux types de couche-2 canaux descendante de transport qui ont été fournis en UMTS: *canaux dédiés* et *canaux commun*. Un canal commun est une ressources partagées entre tous ou un groupe d'utilisateurs dans une cellule, alors que un canal dédié est une ressource identifiée par un code sur un certaine fréquence et est réservé à un seul utilisateur. La seule canal dédié est appelée DCH et l'une des six canaux communes de transport qui est principalement utilisée pour la transmission des paquets de données sur le lien descendant est le canal FACH [HT01]. Selon les spécifications WCDMA (Wideband-CDMA) précisées par le groupe 3GPP, pour un utilisateur en particulier, les flux long avec grande nombre de paquets peuvent être transmis sur le canal dédié DCH et les flux courte de quelques paquets peuvent être transmis sur le canal commune FACH qui est partagée par tous les utilisateurs. Toutefois, les spécifications 3GPP ne prévoient pas standardization d'un tel politique de sélection/changement de canal. Un opérateur de réseau est libre de choisir son propre politique de commutation de canal. Dans cette thèse, nous avons un chapitre dans lequel nous proposons une nouvelle *cross-layer* politique de commutation de canal pour transmission TCP en 3G UMTS.

Les réseaux ad hoc des mobiles (MANET) sont composés de noeuds mobiles sans-fil qui peuvent librement et de manière dynamique s'organiser. De cette façon, ils forment des topologies arbitraire et temporaire de réseau ad hoc, permettant aux dispositifs d'interconnecter de façon transparente dans les zones qui n'ont pas de infrastructure préexistante. Les MANET ont été un thème de recherche très populaire depuis la généralisation des périphériques WiFi 802.11. Dans cette thèse, nous étudions la distance optimale du prochain saut pour optimiser le débit du système dans des MANET soumis à des contraintes de puissance moyenne.

Les véhicules équipés de dispositifs de communication peuvent constituer des réseaux ad hoc des véhicules (VANETs) pour des tâches telles que évitement de collision inter-véhiculaire, notification d'accident sur la route, mise à jour de situation du trafic, la coordination des systèmes de conduite ou tout simplement communication vocale inter-véhiculaire. À l'instar de MANETs, VANETs ne reposent pas sur une infrastructure fixe et au lieu dépen-

dent sur des noeuds de relais intermédiaire pour des protocoles de mise en place de route et de la transmission de données. Toutefois, VANETs présentent un comportement radicalement différent des habituels MANETs. Vitesse haute des véhicules, la mobilité contrainte sur un route droit et le comportement du conducteur sont certains facteurs en raison des quels VANETs possèdent des caractéristiques très différentes des typiques MANETs. VANETs sont l'objet de recherches et à la mise au point par de nombreux organizations industriels et gouvernementaux. Dans cette thèse, nous avons un chapitre dédié à l'étude de sélection optimale du prochain saut dans un route entre deux véhicules pour un scénario simple de VANETs sur une autoroute. Il y a un autre chapitre qui traite en détail une structure inhérents à la solution de problème Programmation Dynamique (DP) qui se pose en problèmes d'optmization de route tels que celui considéré pour VANETs.

Modélisation de la performance des systèmes réseaux informatiques est une étape cruciale dans la planification de la capacité des réseaux informatiques. Les modèles de performance des architectures complexes des logiciels peut être très utile pour prédire précisément leurs performances pour différents modes de trafic de données et la charge de travail. Modélisation de la performance devient de plus en plus difficile si la charge de travail (par exemple, le nombre d'emplois ou de transactions traitées par unité de temps) pour un système informatique utilisé dans un environnement de production est non-stationnaire dans la nature et a des caractéristiques d'évolution rapide. En outre, les derniers logicielles fondée sur des transactions sont très complexes dans la conception. De ce fait, les transactions peuvent encourir des temps de service variable dépendant du charge de travail. Dans cette thèse, nous avons un chapitre où l'on étudie et présente une approche novatrice de l'utilization de filtrage de Kalman avec des modèles de performance basée sur la théorie des files d'attente. Cette approche nous permet de poursuivre en *temps réel* la modélisation de la performance des systèmes réseaux informatiques en-ligne qui traitent une charge de travail non-stationnaire dans la nature et ont des temps de service dépendant du charge de travail.

11.2 Organization et Contribution du Thèse

Cette thèse peut être considéré comme divisé en deux parties. La première partie comprend les Chapitres 2-9 qui traitent de divers problèmes de recherche liés à l'optimization et de contrôle dans les réseaux sans-fil. Alors que les Chapitres 2-6 traitent les réseaux d'accès sans-fil, l'objet de Chapitres 7-9 est les réseaux ad hoc sans-fil. La deuxième partie comprend le Chapitre 10 qui présente l'estimation des paramètres du modèle dépendent de la charge de travail et la modélisation de performance en *temps réel* des réseaux informatiques.

En Chapitre-2, nous étudions l'association utilisateur-réseau globalement optimale dans une *cellule hybride* de 802.11 WLAN et 3G UMTS intégré. Le problème d'association est formulé comme une problème de contrôle de routage de connexion utilisant SMDP (semi-Markov Decision Process). Nous avons ensuite résoudre ce problème SMDP en utilisant un modèle de réseau particulier pour les réseaux WLAN et UMTS et avec des récompenses comprenant des composants financière et le débit total. Le modèle de réseau suppose une allocation saturé des ressources dans les réseaux WLAN et UMTS et une seule classe de

qualité de service pour les mobiles à destination d'un emplacement *moyenne* dans la cellule hybride. L'équation de Programmation Dynamique est résolu en utilisant 'Value Iteration' et une politique optimale et stationnaire avec une structure de courbe de commutation du type ni convexe ni concave est obtenu. Des courbes de commutation de type seuil et symétrique sont observées pour les cas analogue de réseaux homogènes.

En Chapitre-3, nous étudions l'association utilisateur-réseau individuellement optimale dans une *cellule hybride* de 802.11 WLAN et 3G UMTS intégré. Le problème d'association est formulé dans un cadre de jeu non-coopératif. Dans la formulation, l'arrivées des mobiles sont supposées suivre le processus de Poisson et chaque mobile considère son temps moyenne du service dans chaque réseau comme le critère de décision pour se connecter soit au réseau WLAN soit au réseau UMTS. Nous cherchons à calculer la politique optimale d'association qui atteint l'équilibre de Nash. Pour cela, nous développons un système générique d'équations linéaires pour estimer le temps de service moyenne d'un mobile. Ce système est alors résolu en supposant un modèle particulier pour les réseaux WLAN et UMTS et nous avons explicitement calculer la politique optimale d'association qui est observé de posséder une structure de courbe escalier-descendant.

L'IEEE 802.11 MAC protocole, DCF (Distributed Coordination Function), permet les noeuds dans un WLAN de choisir un taux PHY pour chaque transmission de donnée. À cette fin, plusieurs algorithmes d'auto-sélection des taux ont été proposées dans la littérature et la plupart d'entre eux permettent les noeuds WLAN d'sélectionner leurs taux PHY dans une manière non-coopératif, par définition. Dans le cadre d'un jeu non-coopératif, chaque noeud va choisir son taux à fin d'optimiser son propre, utilitaire. En Chapitre-4, nous formulons une fonction utilitaire comprenant le débit et les coûts liés au consommation de puissance et tirons des expressions explicites pour le taux optimal en vertu de la sélection de taux coopératif et non-coopératif. Nous considérons des problèmes d'optimization pour le nombre finis de noeuds n et pour la limite $n \to \infty$ et les débit d'un seul noeud correspondant à la taux PHY optimale en vertu de jeu non-coopératif sont comparées avec ceux obtenus dans le cadre d'une sélection coopératif de taux PHY. Les comparaisons montrent que les performances du réseau dans le scénario du jeu non-coopératif est inefficace par rapport à la scénario coopératif.

En Chapitre-5, nous proposons un cadre stochastique de Markov pour modéliser la performance d'une protocole simple du transport basée sur 'Fountain Codes' (FCT) dans une cellule 802.11 WLAN. Notre modèle permet à la point d'accès WLAN d'employer un algorithme générique pour le contrôle de taux pour transmissions de données MAC sur le lien descendant. Utilisant la théorie de renouvellement nous fournissons une expression explicite du débit moyen descendant. *ns2* simulations sont utilisées pour valider notre modèle et le métrique de débit obtenu. Une étude de performance est ensuite effectués pour permettre de mieux comprendre le choix des différents paramètres du système qui peuvent conduire à une performance optimale du réseau. Enfin, nous présentons une brève comparaison entre le performance de FCT et TCP par le biais de simulations.

Dans le 3G UMTS, deux canaux de transport principales ont été fournies au couche-2 (MAC) pour la transmission descendante de données: un canal commun FACH et un canal dédié DCH. La performance de TCP dans l'UMTS dépend beaucoup sur la politique de

commutation de canal utilisé. En Chapitre-6, nous proposons et analysons trois nouveaux politiques de commutation de base de type seuil pour l'UMTS, que nous appelons des politiques QS (taille file d'attente), FS (taille flux) et QSFS (QS & FS combiné). Ces politiques améliorent le temps de réponse dans une manière significative par rapport une 'politique de type seuil modifié' dans [PAAD03] d'environ 17%. Nous proposons et évaluons une nouvelle politique de commutation améliorée que nous appelons politique FS-DCH (*au-moins* seuil de taille de flux sur DCH). Cette politique est biaisée vers les flux court de TCP de quelques paquets. Il s'agit donc d'une politique *cross-layer* qui améliore la performance de TCP en donnant la priorité à les premières quelques paquets d'un flux sur le canal rapide DCH. Vaste résultats de simulation montrent que la politique FS-DCH s'améliore par rapport à d'autres sur 30% à 36% de temps de réponse pour un cas particulier.

En Chapitre-7, dans un réseau multi-saut dense de noeuds mobiles capables d'appliquer le contrôle de puissance adaptative, nous considérons le problème de trouver la distance optimale du saut à maximiser une certaine mesure de débit en *bit-metres/sec*, soumis à des contraintes de puissance moyenne. La mobilité des noeuds est limitée à une zone de périphérie circulaire centrée à la position nominale des noeuds. Nous incorporons seulement les caractéristiques de perte de gain qui varient de façon aléatoire en raison de la aléatoire motion de noeuds, à l'exclusion des effets 'multi-path fading' ou 'shadowing'. Calcul du métrique de débit, dans un tel scénario nous conduit à calculer la fonction de densité de probabilité de la distance aléatoire entre des points en deux cercles. En utilisant l'analyze numérique, nous découvrons que le choix du noeud le plus proche en tant que le prochain saut *n'est pas toujours* optimale. Performance en tant que débit optimal est également atteint à des distances du saut non-trivial en fonction de la disposition de puissance moyenne dans le réseau.

L'objectif principal du chapitre-8 est de mieux comprendre le dynamique de routage dans VANETs qui sont une catégorie spéciale de MANET, mais démontrent un comportement très différents. Nous considérons le problème de sélection optimale du prochain saut dans un route entre deux véhicules pour un scénario simple de VANETs sur une autoroute. Pour un choix donné de nombre de sauts entre la source et destination, nous cherchons le choix optimal de prochain saut basée sur sa vitesse et distance inter-noeud à fin de maximiser la *durée de vie de route*. Notre modèle analytique comptes l'évolution aléatoire des vitesses des noeuds (véhicules) sur temps et, par conséquent, le choix optimal dépend de la dynamique de la processus stochastique correspondant aux vitesses des noeuds. Dans le cadre d'un hypothèse Markovienne sur le processus de la vitesse des noeuds, nous montrons que le choix optimal de la vitesse est telle que la durée de vie des liens à côté est égalisée. Des expressions explicites pour le choix optimal de la vitesse de prochain saut et distance inter-noeud sont obtenus pour certains scénarios d'intérêt sur l'autoroute. Un variation monotone de la vitesse des noeuds relais dans le cadre de la politique optimale est prouvé. Ces propriétés ont été confirmées avec simulations. Les politiques optimales et leurs structures peuvent aider à l'amélioration de la performance des protocoles de routage VANET existantes.

La nature de l'installation et mécanisme de transmission dans les réseaux ad hoc sans-fil avec un accès aléatoire au canal, génère un corrélation entre les *métriques lien* des liens

adjacents, lors de l'examen de problèmes d'optimization du chemin. En Chapitre-9, nous identifions une structure inhérents à la solution de problème de Programmation Dynamique (DP) qui se pose dans un tel problème d'optimization du chemin. Selon cette structure, la politique optimale essaie de *égaliser* le métrique lien de liens adjacents dans un route multi-saut. Nous validons cette structure avec un simulateur VANET.

En Chapitre-10, nous sommes préoccupés par la modélisation de performance de logicielles distribuées basée sur des transactions et déployées sur un architecture arbitraire de réseau informatique. AMBIANCE, qui est un outil prototype de recherche et a été développé à IBM Research, fait usage de la puissante *Inferencing* technique qui permet de générer une modèle de performance basée sur des paramètres *indépendant* du charge de travail. Dans ce travail, nous étendons le technique d'Inferencing pour génèrer une modèle de performance basée sur des paramètres *dépendant* du charge de travail. Nous appelons cette forme étendue d'Inferencing comme *Enhanced Inferencing*. Mise en oeuvre de cette Enhanced Inferencing dans AMBIENCE montre une amélioration notable du modèle de performance. En outre, nous présentons une approche novatrice de l'utilization de filtrage de Kalman avec des modèles de performance du type Inferencing afin d'être en mesure de poursuivre en 'temps réel' la modélisation de la performance des systèmes informatiques en production qui traitent une charge de travail non-stationnaire. Ce modèle de performance en temps réel non seulement fonctionne fiabilement pour les charges de travail *non-stationnaires*, mais intègre également les paramètres du modèle *dépendant du charge de travail*.

Bibliography

[AABB04] K. Avrachenkov, U. Ayesta, P. Brown, and J. Blanton. Differentiation between Short and Long TCP flows: Predictability of the response time. In *Proceedings of Infocom*, Hong Kong, March. 2004.

[AKKD01] S. Agarwal, S.V. Krishnamurthy, R.K. Katz, and S.K. Dao. Distributed power control in Ad-Hoc wireless networks. In *Proc. of IEEE PIMRC*, 2001.

[AKKV05a] E. Altman, A. Kumar, D. Kumar, and R. Venkatesh. Cooperative and Non-Cooperative Control in IEEE 802.11 WLANs. Technical Report RR-5541, INRIA, Sophia-Antipolis, Mar. 2005.

[AKKV05b] Eitan Altman, Anurag Kumar, Dinesh Kumar, and Ramaiyan Venkatesh. Cooperative and non-cooperative control in ieee 802.11 wlans. *Proceedings of 19th International Teletraffic Congress (ITC-19)*, 2005.

[AS98] E. Altman and N. Shimkin. Individual equilibrium and learning in a processor sharing system. *Operations Research*, 46:776–784, 1998.

[AS03] C. Anderson and J. Soderberg. Channel-Type Switching from a Common Channel to a Dedicated Channel based on Common Channel Load. In *US Patent no. US-6,519,461*, 2003.

[BCSW98] S. Basagni, I. Chlamtac, V. Syrotiuk, and B. Woodward. A distance routing effect algorithm for mobility (DREAM). In *Proceedings of MobiCom*, 1998.

[BEH04] J.J. Blum, A. Eskandarian, and L.J. Hoffman. Challenges of Intervehicle Ad Hoc Networks. *IEEE Transactions on Intelligent Transportation Systems*, 5(4), Dec. 2004.

[Bia00a] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, March 2000.

[Bia00b] G. Bianchi. Performance analysis of the ieee 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, March 2000.

[BLM02] J. Byers, M. Luby, and M. Mitzenmacher. A digital fountain approach to the reliable distribution of bulk data. *IEEE Journal on Selected Areas in Communications*, 20(8):1528–1540, 2002.

[Bon04] T. Bonald. A score-based opportunistic scheduler for fading radio channels. In *Proc. of European Wireless*, 2004.

[cara] Car to car communication consortium. *http://www.car-to-car.org*.

[carb] Safe and comfortable driving based upon inter-vehicle communication. *http://www.cartalk2000.net*.

[DPZ04a] R. Draves, J. Padhye, and B. Zill. Comparison of Routing Metrics for Multi-Hop Wireless Networks. In *Proceedings of SIGCOMM*, 2004.

[DPZ04b] R. Draves, J. Padhye, and B. Zill. Routing in Multi-radio, Multi-hop Wireless Mesh Network. In *Proceedings of MOBICOM*, 2004.

[ea97] Lakshman et al. The performance of tcp/ip for networks with high bandwidth-delay products and random loss. *IEEE/ACM Transactions on Networking*, 5(3):336–350, June 1997.

[ea99] Xylomenos et al. TCP and UDP performance over a wireless LAN. In *Proceedings of IEEE Infocom*, March 1999.

[ea00a] Cali et al. Dynamic tuning of the ieee 802.11 protocol to achieve a theoretical throughput limit. *IEEE Transactions on Networking*, 8(6), December 2000.

[ea00b] Lefevre et al. Understanding TCP's Behavior over Wireless Links. In *Symposium on Communications and Vehicular Technology*, October 2000.

[ea01] Sikdar et al. An integrated model for the latency and steady state throughput of tcp connections. *Performance Evaluation Journal*, 46:139–154, 2001.

[ea02] Lehr et al. Wireless Internet Access: 3G vs. WiFi. In *White Paper, MIT*, August 2002.

[ea03] Fu et al. The Impact of Multihop Wireless Channel on TCP Throughput and Loss. In *Proc. of IEEE INFOCOM*, 2003.

[ea05a] Fuessler et al. Studying Vehicle Movements on Highways and their Impact on Ad-Hoc Connectivity. Technical Report TR-2005-003, Department of Mathematics and Computer Science, University of Mannheim, 2005.

[ea05b] Tian et al. Tcp in wireless environments: Problems and solutions. *IEEE (Radio) Communications Magazine*, 43(3):S27–S32, March 2005.

[ea06a] Hu et al. An adaptive p-persistent 802.11 mac scheme to achieve maximum channel throughput and qos provisioning. *IEEE Wireless Communications and Networking Conference (WCNC)*, 2006.

[ea06b] Leinmuller et al. Improved Security in Geographic Ad Hoc Routing through Autonomous Position Verification. In *VANET*, Los Angeles, USA, Sept. 2006.

[FC05] O. E. Falowo and H. A. Chan. AAA and Mobility Management in UMTS-WLAN Interworking. In *Proc. 12th International Conference on Telecommications (ICT)*, Cape Town, May. 2005.

[fle] Fleetnet. *http://www.et2.tu-harburg.de/fleetnet/*.

[Gal98] R. G. Gallager. *Discrete Stochastic Processes*. Kluwer, 1998.

[GCNC01] J. Gomez, A.T. Campbell, M. Naghshineh, and C.Bisdikian. Conserving Transmission Power in Wireless Ad Hoc Networks. In *Proc. of IEEE ICNP*, Nov. 2001.

[GJD+01] M. Gruteser, A. Jain, J. Deng, F. Zhao, and D. Grunwald. Exploiting physical layer power control mechanisms in IEEE 802.11b network interfaces. Technical report, Univ. of Colorado, Boulder, Dec. 2001.

[GK00] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transaction on Information Theory*, 46(2):388–404, March. 2000.

[GTB99] M. Gerla, K. Tang, and R. Bagrodia. TCP Performance in Wireless Multihop networks. In *Proceedings of IEEE WMCSA*, Feb. 1999.

[GV97] A. J. Goldsmith and P. Varaiya. Capacity of fading channels with channel side information. *IEEE Transaction on Information Theory*, 43(6):1986–1992, Nov. 1997.

[HT01] H. Holma and A. Toskala. *WCDMA for UMTS*. Revised Edition, Wiley, 2001.

[HVB01] G. Holland, N. Vaidya, and P. Bahl. A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks. In *Proc. of ACM Mobicom*, July 2001.

[Jas03] M. Jaseemuddin. An Architecture for integrating UMTS and 802.11 WLAN Networks. In *Proc. of ISCC*, Turkey, July 2003.

[JBW05] S. Jaap, M. Bechler, and L. Wolf. Evaluation of routing protocols for vehicular ad hoc networks in city traffic scenarios. In *11th EUNICE Open European Summer School on Networked Applications*, Spain, July. 2005.

[JZS04] S. Jagannathan, M. Zawodniok, and Q. Shang. Distributed power control of cellular networks in the presence of Rayleigh fading channel. In *Proc. of IEEE Infocom*, Hong-Kong, March 2004.

[KA] Dinesh Kumar and Eitan Altman. Modeling & performance analysis of a fountain codes based transport in ieee 802.11 wlans. *Unpublished Manuscript.*

[KAK07a] Dinesh Kumar, Eitan Altman, and Jean-Marc Kelif. Globally optimal user-network association in an 802.11 wlan & 3g umts hybrid cell. *Proceedings of 20th International Teletraffic Congress (ITC-20)*, 2007.

[KAK07b] Dinesh Kumar, Eitan Altman, and Jean-Marc Kelif. User-network association in an 802.11 wlan & 3g umts hybrid cell: Individual optimality. *Proceedings of IEEE Sarnoff Symposium*, 2007.

[KAMG05a] A. Kumar, E. Altman, D. Miorandi, and M. Goyal. New insights from a fixed point analysis of single cell IEEE 802.11 WLANs. In *Proc. of IEEE Infocom*, USA, March 2005.

[KAMG05b] A. Kumar, E. Altman, D. Miorandi, and M. Goyal. New insights from a fixed point analysis of single cell ieee 802.11 wlans. *Proceedings of IEEE Infocom*, March 2005.

[KBAK08] Dinesh Kumar, Dhiman Barman, Eitan Altman, and Jean-Marc Kelif. New cross-layer channel switching policy for tcp transmission on 3g umts downlink. *Proceedings of 7th Wireless Telecommunications Symposium*, 2008.

[KK00] B. Karp and H.T. Kung. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *Proceedings of MobiCom*, 2000.

[KKA06a] Arzad A. Kherani, Dinesh Kumar, and Eitan Altman. A structural property of solutions to path optimization problems in random access networks. *Proceedings of 4th Intl. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 2006.

[KKA06b] Dinesh Kumar, Arzad A. Kherani, and Eitan Altman. Route lifetime based optimal hop selection in vanets on highway: An analytical viewpoint. *Proceedings of IFIP Networking*, 2006.

[KL97] A. Kamerman and L.Monteban. WaveLAN-II: A high-performance wireless LAN for the unlicensed band. *Bell Lab Technical Journal*, pages 118–133, Summer 1997.

[KMK04] A. Kumar, D. Manjunath, and J. Kuri. Communication networking: An analytical approach. *Morgan Kaufman Series in Networking, (an imprint of Elsevier Science)*, May 2004.

[KPR05] R. Kumar, A. Paul, and U. Ramachandran. Fountain broadcast for wireless networks. *Second International Workshop on Networked Sensing Systems (INSS)*, 2005.

[KRPR05] H. Kwon, K. Rho, A. Park, and J. Ryou. Mobility Management for UMTS-WLAN Seamless Handover: within the Framework of Subscriber Authentication. In *Proc. of ISATED Communication, Network and Information Security (CNIS)*, Nov. 2005.

[KS04a] A. A. Kherani and R. Shorey. Performance improvement of tcp with delayed acks in ieee 802.11 wireless lans. *IEEE Wireless Communications and Networking Conference (WCNC)*, March 2004.

[KS04b] A. A. Kherani and R. Shorey. Throughput analysis of tcp in multi-hop wireless networks with ieee 802.11 mac. *IEEE Wireless Communications and Networking Conference (WCNC)*, March 2004.

[KTZ] Dinesh Kumar, Asser Tantawi, and Li Zhang. Performance modeling for workload dependent parameters & real time operation. *Unpublished Manuscript.*

[KV98] Y.B. Ko and N.H. Vaidya. Location-aided routing in mobile ad hoc networks. In *Proceedings of MobiCom*, 1998.

[KVKA06] Dinesh Kumar, Ramaiyan Venkatesh, Anurag Kumar, and Eitan Altman. Capacity optimizing hop distance in a mobile ad hoc network with power control. *Proceedings of 4th Intl. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 2006.

[KZ] Dinesh Kumar and Li Zhang. Method and apparatus for automatic performance modeling with load dependent service times and overheads. *Patent filed in USPTO by IBM Research, September 2008.*

[LFC05] L. Lopez, A. Fernandez, and V. Cholvi. A game theoretic analysis of protocols based on fountain codes. *10th IEEE Symposium on Computers and Communications (ISCC)*, 2005.

[LH04] J. Luo and J.P. Hubaux. A Survey of Inter-Vehicle Communication. Technical report, EPFL, Switzerland, 2004.

[Lip75] S. A. Lippman. Applying a new device in the optimization of exponential queueing systems. *Operations Research*, pages 687–710, 1975.

[LMT04] M. Lacage, M.H. Manshaei, and T. Turletti. Ieee 802.11 rate adaptation: A practical approach. *ACM International Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM)*, October 2004.

[LP05] F. Lebeugle and A. Proutiere. User-level performance in WLAN hotspots. In *Proc. of ITC-19*, Beijing, China, 2005.

[Lub03] M. Luby. Information additive code generator and decoder for communication systems. In *U.S. Patent 6,307,487; 6,373,406; 6,614,366*, 2001-03.

[Lub02] M. Luby. Lt codes. *43rd Annual IEEE Symposium on Foundations of Computer Science*, 2002.

[LXMZ03] Z. Liu, C. H. Xia, P. Momcilovic, and L. Zhang. AMBIENCE: Automatic Model Building using InferEnce. In *Congress MSR03*, Metz, France, Oct. 2003.

[LZ05] C. Liu and C. Zhou. HCRAS: A Novel Hybrid Internetworking Architecture between WLAN and UMTS Cellular Networks. In *Proc. of IEEE CCNC*, Las Vegas, USA, Jan. 2005.

[Mac05] D. J. C. MacKay. Fountain codes. *IEE Proceedings Communications*, 152(6):1062–1068, December 2005.

[Mit04] M. Mitzenmacher. Digital Fountains: A Survey and Look Forward. In *IEEE Information Theory Workshop*, October 2004.

[MKA06] D. Miorandi, A. Kherani, and E. Altman. A Queueing Model for HTTP Traffic over IEEE 802.11 WLANs. *Computer Networks*, 50(1):63–79, Jan. 2006.

[MYLR04] L. Ma, F. Yu, V. C. M. Leung, and T. Randhawa. A New Method to support UMTS/WLAN Vertical Handover using SCTP. *IEEE Wireless Communication*, 11(4):44–51, Aug. 2004.

[NAG04] V. Namboodiri, M. Agarwal, and L. Gao. A Study on the Feasibility of Mobile Gateways for Vehicular Ad-hoc Networks. In *VANET*, Philadelphia, USA, Oct. 2004.

[now] Network on Wheels. *http://www.network-on-wheels.de/*.

[ns2] The Network Simulator - ns-2. *http://www.isi.edu/nsnam/ns*.

[OKIO02] Y. Ohta, K. Kawahara, T. Ikenaga, and Y. Oie. Performance Evaluation of Channel Switching Scheme for Packet Data Transmission in Radio Network Controller. In *Proceedings of Networking*, Berlin, 2002.

[ove] OverDRiVE project. *http://www.ist-overdrive.org*.

[PAAD03] B. J. Prabhu, E. Altman, K. Avrachenkov, and J. A. Dominguez. A Simulation Study of TCP Performance over UMTS Downlink. In *Proceedings of IEEE VTC-Fall*, Orlando, USA, Oct. 2003.

[Pat94] M. Patriksson. The Traffic Assignment Problem: Models and Methods. 1994.

[PK06] K. Premkumar and A. Kumar. Optimal Association of Mobile Wireless Devices with a WLAN-3G Access Network. In *Proc. of IEEE ICC*, June 2006.

[pre] PReVENT: A European program to improve active safety. *http://www.prevent-ip.org*.

[Put94] M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 1994.

[QCJS03] D. Qiao, S. Choi, A. Jain, and K.G. Shin. MiSer: An optimal low-energy transmission strategy for IEEE 802.11a/h. In *Proc. of ACM MobiCom*, Sept. 2003.

[RFQ03] Jr. Roy F. Quick. Random Access Communications Channel for Data Services. In *US Patent no. US-5,673,259*, 2003.

[RKA] Venkatesh Ramaiyan, Anurag Kumar, and Eitan Altman. On the Transport Capacity of a Single Cell, Dense, Multihop WLAN with Fading. In *Unpublished manuscript*.

[Ros65] J. B. Rosen. Existence and Uniqueness of Equilibrium Points for Concave N-person Games. *Econometrica*, 33:153–163, 1965.

[RT99] E. Royer and C. Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, Apr. 1999.

[Sho03] A. Shokrollahi. Raptor codes. *Unpublished manuscript*, 2003.

[Sim06] Dan Simon. Optimal State Estimation: Kalman, H-infinity, and Nonlinear Approaches. 2006.

[SLC03] J. Song, S. Lee, and D. Cho. Hybrid Coupling Scheme for UMTS and Wireless LAN Interworking. In *Proc. of VTC-Fall*, volume 4, pages 2247–2251, Oct. 2003.

[TG05] Godfrey Tan and John Guttag. The 802.11 MAC Protocol Leads to Inefficient Equilibria. In *Proc. of IEEE Infocom*, Miami, USA, March 2005.

[THR03] Jing Tian, Lu Han, and Kurt Rothermel. Spatially Aware Packet Routing for Mobile Ad Hoc Inter-Vehicle Radio Networks. In *IEEE ITSC*, Shanghai, China, Oct. 2003.

[TXA05] Y. Tian, K. Xu, and N. Ansari. Tcp in wireless environments: Problems and solutions. *IEEE (Radio) Communications Magazine*, 43(3):S27–S32, March 2005.

[UD05] M. Usman and J. Dunlop. A testbed for assessment of fountain codes for wireless channels. *European Wireless*, April 2005.

[VGN05] N. Vulic, S. H. Groot, and I. Niemegeers. Common Radio Resource Management for WLAN-UMTS Integration Radio Access Level. In *Proc. of IST Mobile & Wireless Communications Summit*, Germany, June 2005.

[War52] J. G. Wardrop. Some theoretical aspects of road traffic research communication networks. *Proc. Inst. Civ. Eng.*, Part 1-2:325–378, 1952.

[WB01] C. Wu and D. P. Bertsekas. Distributed power control algorithms for wireless networks. *IEEE Transactions on Vehicle Technology*, 50:504–514, Mar. 2001.

[Yan05] S. Yang. Experimental comparison between tcp and digital fountain over high performance network on uk grid. *Project Report, University of Manchester*, March 2005.

[Yat95] R. D. Yates. A framework for uplink power control in cellular radio systems. *IEEE Journal of Selected Areas in Communications*, 13(7):1341–1347, Sept. 1995.

[YK05] F. Yu and V. Krishnamurthy. Efficient Radio Resource Management in Integrated WLAN/CDMA Mobile Networks. *Telecommunication Systems Journal*, 30(1-3):177–192, Nov. 2005.

[ZHZ03] L. Zan, G. Heijenk, and M. E. Zarki. Fair & Power-Efficient Channel-Dependent Scheduling for CDMA Packet Networks. In *Proc. of ICWN*, USA, 2003.

[ZLR+03] L. Zhang, Z. Liu, A. Riabov, M. Schulman, C. Xia, and F. Zhang. A comprehensive toolset for workload characterization, performance modeling, and online control. *Computer Performance Evaluations, Modelling Techniques and Tools (Springer-Verlag LNCS)*, 2794:63–77, 2003.

[ZXSI02] L. Zhang, C.H. Xia, M.S. Squillante, and W.N. Mills III. Workload Service Requirements Analysis: A Queueing Network Optimization Approach. In *10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS)*, 2002.

Résumé

Cette thèse aborde divers problèmes d'optimisation et de contrôle, liés aux réseaux d'accès et ad hoc sans-fil, et de la modélisation des réseaux informatiques. Dans un premier temps, nous considérons les réseaux d'accès sans-fil. Nous étudions d'abord l'association optimale entre un utilisateur et un réseau basé sur l'une des technologies 802.11 ou UMTS 3G. Par la suite, nous étudions le problème de contrôle non-coopératif du taux de communication quand seule la technologie 802.11 est déployée. Dans ce même contexte, nous analysons les performances d'un protocole de transport utilisant des codes fontaine. Quand seule la technologie UMTS est déployée, nous proposons une politique améliorée de commutation de canal pour les canaux descendants. Dans un deuxième temps, nous nous intéressons aux réseaux ad hoc sans-fil. Nous étudions ainsi la longueur du saut optimal qui maximise la capacité des réseaux denses de mobiles, ainsi que le choix optimal du relai suivant dans les réseaux véhiculaires d'autoroute. À l'aide d'outils tels que les processus de décision semi-Markoviens, la théorie des jeux et celle des chaînes de Markov, le théorème de renouvellement et récompense, les techniques inter-couches, le lemme de Wald et le filtre de Kalman, nous obtenons, dans certains cas, les politiques de contrôle optimal et, dans d'autres cas, le choix ou l'estimation des paramètres optimaux du système. Dans un troisième temps, nous nous tournons vers les systèmes de réseaux informatiques en-ligne ayant des caractéristiques non-stationnaires. Nous élaborons une approche novatrice de l'utilisation du filtre de Kalman combinée à des modèles à base de files d'attente, nous permettant ainsi de poursuivre en "temps réel" la performance de ces systèmes. À noter que dans de tels systèmes, ni l'analyse stationnaire, ni celle transitoire de la théorie des files d'attente ne s'appliquent pour obtenir en temps réel l'estimation des paramètres du modèle tels que le temps de service ou le délai.

Mots-clés: modélisation, stochastique, l'estimation, théorie des jeux, la cellule hybride.

Abstract

This thesis deals with various optimization and control problems related to wireless access and ad hoc networks and performance modeling in computer networks. In wireless access networks, we study two different technologies: 802.11 WLAN and 3G UMTS, both stand alone and together. With both of them together, optimal user-network association in a WLAN and UMTS 'hybrid cell' is investigated. In a stand alone single WLAN cell we study non-cooperative PHY rate control and in another problem, performance analysis of a simple Fountain Codes based transport protocol. For a single UMTS cell we propose an improved channel switching policy for the downlink. In wireless ad hoc networks, we study capacity optimizing hop distance in a dense Mobile ad hoc network (MANET) and optimal next hop selection in a Vehicular ad hoc network (VANET) on a highway. Tools such as SMDP (semi-Markov Decision Process), game theory, Markov chains, renewal reward theorem, cross-layer techniques, Wald's lemma and Kalman filtering are employed to derive optimal control policies in some cases, and choice or estimation of optimal system parameters in others. Finally, we present an innovative approach of using Kalman filtering with queueing theory based performance models in order to be able to pursue 'real time' performance modeling of online computer network systems having fast changing non-stationary characteristics. In such systems, neither stationary nor transient analysis from traditional queueing theory can be practically applied to obtain real time estimates of model parameters such as service times and network delays.

Keywords: modeling, stochastic, estimation, game theory, hybrid cell.