ADAPTIVE CONTROL OF REAL-TIME MEDIA APPLICATIONS IN BEST-EFFORT NETWORKS

A Thesis

by

VIVEK KHARIWAL

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

August 2004

Major Subject: Mechanical Engineering

ADAPTIVE CONTROL OF REAL-TIME MEDIA APPLICATIONS IN BEST-EFFORT NETWORKS

A Thesis

by

VIVEK KHARIWAL

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Approved as to style and content by:

Alexander Parlos (Chair of Committee)	Suhada Jayasuriya (Member)
Jennifer Welch (Member)	Dennis O'Neal (Interim Head of Department)

August 2004

Major Subject: Mechanical Engineering

ABSTRACT

Adaptive Control of Real-Time Media Applications In Best-Effort Networks.

(August 2004)

Vivek Khariwal, B.Eng., Bangalore University, Bangalore, India

Chair of Advisory Committee: Dr. Alexander Parlos

Quality of Service (QoS) in real-time media applications can be defined as the ability to guarantee the delivery of packets from source to destination over best-effort networks within some constraints. These constraints defined as the QoS metrics are end-to-end packet delay, delay jitter, throughtput, and packet losses. Transporting real-time media applications over best-effort networks, e.g. the Internet, is an area of current research. Both the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP) have failed to provide the desired QoS. This research aims at developing application-level end-to-end QoS controls to improve the user-perceived quality of real-time media applications over best-effort networks, such as, the public Internet.

In this research an end-to-end packet based approach is developed. The end-to-end packet based approach consists of source buffer, network simulator ns-2, destination buffer, and controller. Unconstrained model predictive control (MPC) methods are implemented by the controller at the application layer. The end-to-end packet based approach uses end-to-end network measurements and predictions as feedback signals. Effectiveness of the developed control methods are examined using Matlab and ns-2. The results demonstrate that sender-based control schemes utilizing UDP at transport layer are effective in providing QoS for real-time media applications transported over best-effort networks. Significant improvements in providing QoS are visible by the reduction of packet losses and the elimination of disruptions during the

playback of real-time media. This is accompanied by either a decrease or increase in the playback start-time.

To my grandparents and parents for all their sacrifices that made me what I am today.

ACKNOWLEDGMENTS

The word ”guru” in Sanskrit means a person who guides people from darkness to light. I express my deep sense of gratitude to Dr. Parlos who has played a similar role in my development during my stay at Texas A&M University. I am also grateful to my colleagues Aninda, Ram, Srikar, Tolis and Dan for many useful discussions and other help. Thanks are due to my friends Nagesh, Sam, Geetha and Chirag. I am also thankful to Dr. Jayasuriya, Dr. Welch and Dr. Langari for serving on my committee. Lastly, I thank my family and my cousins Poonam and Sonesh for all their support and love.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

This thesis presents a solution to improve the Quality of Service (QoS) of real-time media applications while transporting over best-effort networks. A control theoretic framework is developed for empirical modeling and control of real-time media applications over best-effort networks. A packet-level simulation is used for designing and testing the performance of the controllers.

A. Motivation and Objectives

The last few years have witnessed an increasing demand for real-time media applications over best-effort networks, e.g. the Internet. Best-effort service can lead to packet losses, large end-to-end delay and delay jitter that may be detrimental to the user-perceived quality of media applications. This research aims at developing application-level end-to-end QoS control to improve the user perceived quality of real-time media applications over best-effort networks.

The application layer is the highest layer of the protocol stack. It is responsible for supporting network applications. The transport layer is separate and below the application layer and it helps in establishing communication between application processes on different end-hosts. In this research the UDP (User Datagram Protocol) is used, an unreliable transport protocol widely used in media applications.

In this thesis end-to-end network measurements are used as feedback signals. End-to-end system variables are defined as those measured at the source and at the destination rather than from within the network. This in turn is beneficial in

---

The journal model is *IEEE Transactions on Automatic Control.*

implementing the developed system online, without any network infrastructure modifications. Signals like network packet accumulation and packet losses are considered as feedback signals to achieve the desired QoS. Accumulation of a particular flow can be defined as the sum of packets of that flow that are in transit in the network at any given instant of time, or the difference in the cumulative send and arrival flows minus any packet losses. The motivation for choosing the accumulation signal over the end-to-end packet delay signal, as feedback is that it is very difficult to distinguish lost packets from delayed packets at the receiver in real-time, especially when using an unreliable transport protocol like UDP. It is also difficult to associate a value of the delay signal with respect to lost packets when using an unreliable transport protocol. This is not the first time that flow accumulation has been used for feedback. However, it is the first time this signal has been used for application QoS control. Xia et al. used accumulation for developing a congestion control algorithm. In their work, Xia et al. define accumulation as buffered packets of a flow inside network routers [1].

Improvements in the user-perceived QoS delivered by media applications can be accomplished by achieving the following objectives:

- Packet Loss Adaptation: Minimize, regulate or prevent packet losses as they degrade the quality of media perceived by the end user.

- Delay Jitter Adaptation: Minimize or prevent disruptions due to delay jitter during media playback to maintain the QoS perceived by the end-user.

- Delay Adaptation:End-to-end delays between 150 and 400 milliseconds can be acceptable but not ideal. Minimizing the playback start time can be accomplished by delay adaptation.

- BandwidthAdaptation:Match the source send-rate with the network bandwidth available to the flow.

B. Literature Review

1. Review of Media Application Literature

Applications such as FTP, Telnet, and the Web browser, that require reliability and in-order packet delivery use the Transmission Control Protocol (TCP) as their transport protocol [2]. Applications that have minimum transmission rate and require delay guarantees do not use the TCP and have traditionally used the UDP instead. This is because UDP does not utilize algorithm for congestion control, flow control and its not reliable. Floyd and Fall [3] talk about the negative impact of unresponsive flows (flows using UDP) and promote the inclusion of end-to-end congestion control in design of new protocols to be used in best-effort networks. Congestion control and fairness are in the best long-term interest of media applications as well. Hence any control solution that is intended for applications that use UDP should respond to end-to-end congestion and should promote the "TCP-friendly" behavior in order to accomplish its objectives.

Some attempts have been made to arrive at TCP-friendly protocols [4, 5] where congestion control algorithms have been used that result in smaller oscillations than those of the Additive Increase and Multiplicative Decrease (AIMD) found in TCP. This helps to smoothen the bit stream and eliminate the use of large buffers. Dwyer et al. [6] propose a new technique called Heterogeneous Packet Flows (HPF), which provides in-order reliable delivery based on priorities.

Several methods using packet-based best-effort transmission of audio and video have been developed using UDP as the transport protocol [7, 8]. Cyclic UDP uses the notion of rounds and prioritizes packets within rounds and it is effective in the case of stored media [9]. Some papers talk about feedback control based upon changing video encoder parameters [10, 11]. Many of these techniques are receiver-based and rate control is achieved by trying to meet the bandwidth made available by the network. Sender-based controls have also been explored but not for best-effort networks [12, 13]. The control is based on queue lengths, and other network parameters. Sun et al. [14] apply prediction schemes for rate control. Prediction of packet loss probability and round-trip time is used in computing the source send rate.

Recent studies have shown an inclination of researchers towards building systems and architectures that are end-to-end rather than network-centric [15, 16, 17]. The advantage being that any analysis and control method developed for one system can be used by any other system. This approach is adopted in this thesis for platform and protocol independence.

Ohsaki et al. [18] and Doddi [19] have highlighted the importance of end-to-end packet delay dynamics as they directly affects the QoS and congestion control of media applications. Both, Ohsaki et al. and Doddi adopt a black-box approach for modeling a best-effort network. Doddi, in addition to the linear system identification techniques also implements non-linear system identification for network modeling. This approach is useful as end-to-end dynamics are of interest when implementing time-sensitive applications. The above issues stress on the need to develop methods that accurately measure end-to-end network characteristics [20].

2. Review of Control Theory as Applied to Computer Science

Of late, application of control theory to solve networked application problems has been gaining popularity among computer science researchers. Control theory has been used to provide regulation and optimization of computing systems e.g., the Lotus Notes email server [21, 22]. Abdelzaher and Lu [23] has applied control theory for modeling and performance control of an Internet server . Congestion control can also be performed using predictive algorithms [24, 25]. McNamee et al. [26] describes the control challenges that are encountered for multilevel adaptive video streaming. An optimal state prediction approach to predict current states based on available state observations is discussed by Li and Nahrstedt [27]. Most of the methods developed in the literature are based on linear control theory. Computer networks are highly non-linear systems and developing models of best-effort networks from the network architectural details is difficult. Hence, system identification techniques are a natural choice. Morita et al. [28] use system identification combined with classical control theory to come up with a delay-based congestion control mechanism .

Black et al. [29] talk about "infopipes", that can be defined as an abstraction for information flow in a distributed system . The authors state that the goal of infopipes is to simplify the construction of distributed streaming applications. The paper recognizes the need to adapt the traditional systems and signals modeling and control techniques for use in distributed applications. Modeling and control of constant time-delay systems has been successfully achieved in [30, 31, 32]. Concepts related to stability and studies related to bounded time-varying delays have been explored in [33, 34]. Various methods for tuning controllers have been studied [35, 36, 37, 38]. These techniques are mostly concerned with system stability for constant delay system, rather that end-to-end performance.

Application of control theory in the presence of time-varying time delays in best-effort networks has been proposed in [39, 40]. Quet et al. [39] designed and implemented an optimal control system, robust to an uncertain transport delay. Strict conditions are placed on the form of the time-delay to apply such methods. The results presented in this paper are for time-delay systems that do not lead to sequencing errors and where the time-varying time-delay has little effect on buffer level. This may not be a realistic assumption in the case of best-effort networks using transport protocol such as TCP.

Mangan has systematically approached the problem of improving the QoS of media application over best-effort networks using TCP as transport protocol [41]. He develops some continuous-time and discrete-time models of systems with time-varying time delays. The network is modeled in terms of an end-to-end delay that includes the effects of the dynamics of cross-traffic. Various control strategies including linear, reactive and predictive laws are evaluated and compared for their advantages and limitations. Predictive control performs the best. It improves the QoS and aids in congestion control. The challenge of the predictive control scheme is its complex online implementation.

Controllability of time-varying time-delay systems are evaluated in the literature for certain classes of systems [42, 43, 44, 45]. Ekanayake et al. [44] studies the stability of discrete-time systems with an application to networks . Ekanayake et al. identify the problems caused by uncertainties in delay estimation and present models of discrete-time systems with input or output delays.

C. Proposed Solution

A systematic data-driven approach is adopted to address this complex problem of interest. An engineering systems and signals approach is used to classify and model the specific system of interest. The Network Simulator ns-2 [46], and Matlab are the tools employed in this thesis. Firstly, ns-2 is used to develop an end-to-end packet level simulation model that can simulate certain key characteristics of a best-effort network, like the Internet. This network model acts as the test-bed used in this research. Numerous simulations are performed using the developed network

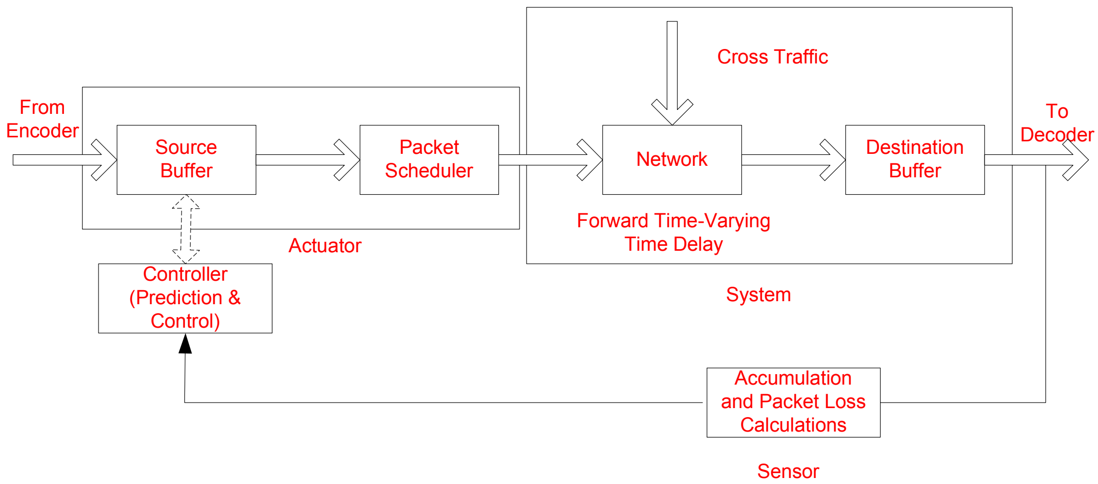

Fig. 1. A Block Diagram of the Proposed System.

model to compare the effectiveness of different control methods upon application level performance. Control efforts based on different control strategies are computed using a combination of ns-2 and Matlab simulations.

Figure 1 shows the proposed system configuration in a system theoretic framework. Accumulation in the network and packet losses are used in an adaptive predictive control algorithm with the objective to control the system. The system is a combination of a Wide Area Network (WAN) such as the Internet, and the destination buffer. Network accumulation, losses, and destination buffer level are measured using a sensor mechanism. Source buffering is required for the implementation of time-varying control effort. The packet scheduler performs the function of a system actuator.

D. Contribution of This Work

This thesis proposes a novel technique for application-level QoS control of real-time media application over best-effort networks. Contributions of the current research work are as follows:

- An end-to-end packet based approach is developed. The feasibility of utilizing UDP to implement a sender-based control method at the application level to improve the QoS for real-time media applications is demonstrated.
- End-to-end empirical predictive modeling of single flow in best-effort networks is demonstrated. Predictive modeling not only helps in achieving QoS for real-time media applications, but also it aids indirectly in congestion avoidance, control, and other challenging issues.
- Model predictive control (MPC) methods are developed based on the end-to-end empirical predictive modeling of network measurements. MPC techniques improve the QoS delivered to the end-user by significantly reducing the packet losses and eliminating the disruptions of the controlled end-to-end flow.

E. Organization of the Thesis

The thesis has been divided into five chapters. Chapter II provides a qualitative discussion of end-to-end measurements in best-effort networks. Major assumptions of the research problem are also discussed. Control algorithms are developed in Chapter III. Empirical predictive modeling of end-to-end best-effort measurements using linear system identification technique is developed in this chapter. Results of the simulations are presented in Chapter IV. Chapter V deals with the thesis summary and provides conclusions and recommendation for future work

CHAPTER II

SIMULATION OF END-TO-END PACKET TRANSPORT USING A NETWORK SIMULATOR : A QUALITATIVE DISCUSSION

This chapter addresses several issues regarding the system of interest. A brief discussion about various available end-to-end network measurements is made. End-to-end is used to describe network measurements measured between the application layer at the source to the application later at the destination. This chapter also lists the assumptions made throughout this research.

A. End-to-End Network Measurements

End-to-end network measurements can be understood from Figure 2. In this thesis ns-2, a network simulator, is used to develop a best-effort packet-switched network simulation model. Different possible end-to-end network measurements that are available and can be used as a potential feedback signal in the proposed solution are considered.

1. Signals Measured at the Source

- Send Rate: Send rate can be defined as number of packets or bytes of data sent from the source per unit time.

- Send Flow: Send flow can be defined as the cumulative amount of packets or bytes of data that have been sent into the network by the source at any given instant of time.

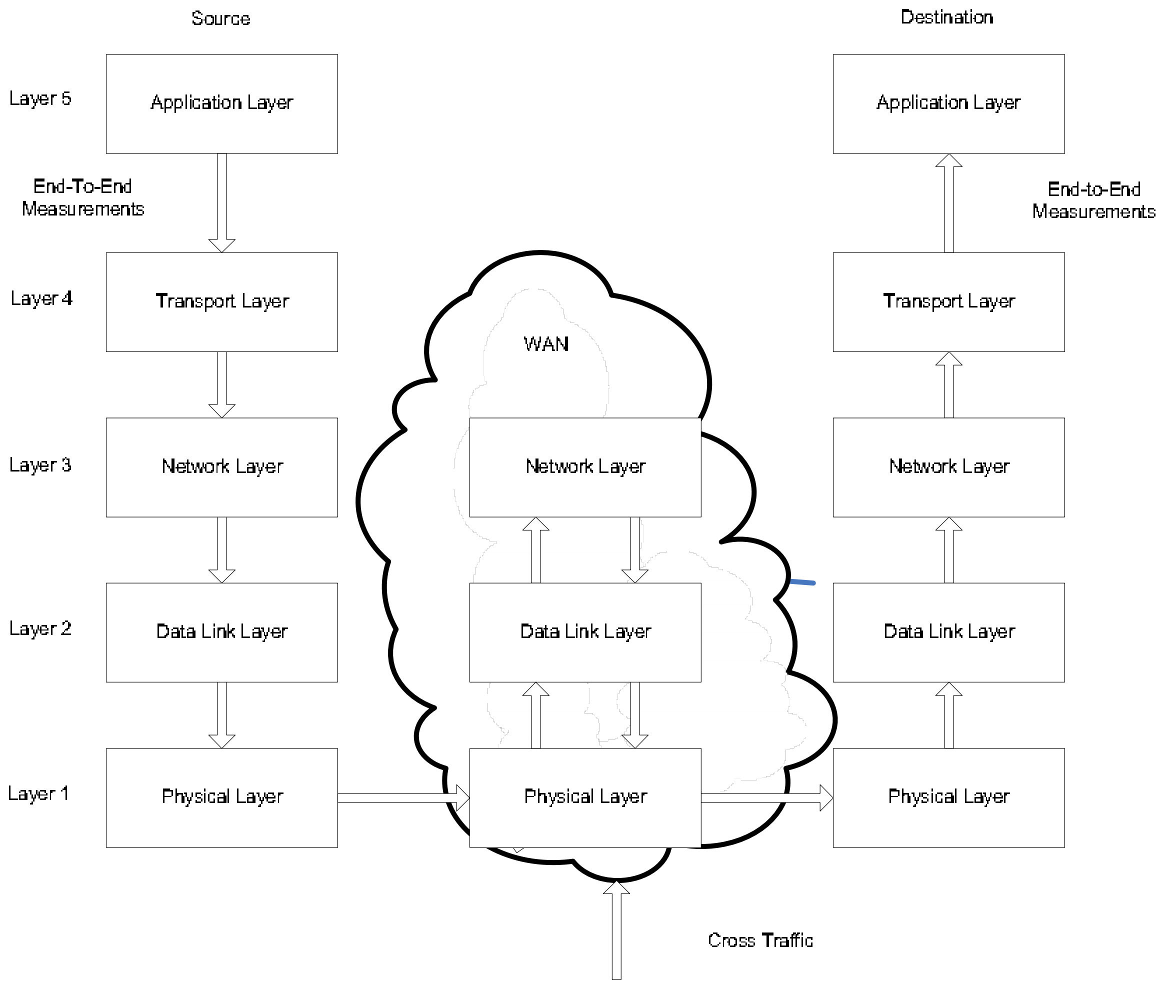

Fig. 2. Schematic Diagram Showing End-to-End Network Concept.

2. Signals Measured at the Destination

- Arrival Rate: Arrival rate can be defined as number of packets or bytes of data that have reached the destination per unit time.
- Arrival Flow: Arrival flow can be defined as the cumulative amount of packets or bytes of data that have reached the destination at any given instant of time.
- End-to-End Delay: End-to-end delay can be defined as the time taken by the packet to travel from the application layer of the source to the application layer of destination. The end-to-end delay has following components [47]:
  - Transmission Delay: Transmission delay is defined as the time taken to transmit all the bits of a packet into the link. It can also be defined as the length of the packet divided by transmission rate of the link. For WANs transmission delays are typically on the order of microseconds.
  - Propagation Delay: Propagation Delay is defined as the time required by a bit to propagate from the beginning of the link to the end of the link. The propagation delay between two nodes is the distance between the two nodes divided by the propagation speed of the medium. In WANs within the US, propagation delays are generally on order of tens of milliseconds.
  - Processing Delay: Processing delay is defined as the time that is required to examine the packets's header and determine where to forward the packet. In WANs, the processing delays are generally on the order of microseconds or less.
  - Queuing Delay: Queuing delay is the time a packet waits in the queue to be transmitted onto the outgoing link. The queuing delay experienced by two successive packets of same flow may vary from milliseconds to

microseconds based on the arriving packets from the other flows, ie the amount of cross-flow

3. Packet Accumulation and Cumulative Packet Losses

In this research, network packet accumulation is used as feedback signal to achieve the desired QoS objective. Accumulation and losses of a particular flow can be defined as the difference in the cumulative send and arrival flows. Accumulation and losses can be expressed as

$$AL(k) = U(k) - Y(k), \tag{2.1}$$

where, $U(k)$ is the send flow, $Y(k)$ is the arrival flow, and k is the discrete time step. Observe, that for a network with packet losses, accumulation and losses $AL(k)$, will have two components at any given instant of time. The first component is the true packet accumulation and second component is the packet losses in the network. For a lossy network, accumulation and losses can be expressed as

$$AL(k) = Acc(k) + L(k) \tag{2.2}$$

where $Acc(k)$ is the true packet accumulation and $L(k)$, network packet losses. These signals are deduced from the accumulation and losses $AL(k)$, by removing the trend from it. Hence it can be said that these signals are just an approximation. Henceforth, the term packet accumulation will be used to signify true packet accumulation in this thesis.

4. End-to-End Delay versus Accumulation

Motivations for choosing the accumulation $Acc(k)$ over end-to-end delay as feedback signal are as follows:

- It is difficult to distinguish between the packets unaccounted for due to late arrival and the packets that were dropped while being transmitted through the network.
- It is difficult to associate a value to the delay signal when a packet is lost and when using an unreliable transport protocol.
- One major advantage of using the accumulation signal instead of the end-to-end delay signal is that the former signal could be used during instances of flow reversal when packet arrive at their destination out-of-order.

B. QoS Metrics for Real-Time Media Applications

In this thesis QoS metrics that constitute the control objectives are as follows :

- Minimize the packet losses: This is a major challenge in improving the QoS of media applications. Reducing losses may also help in minimization of playback start time and elimination of disruption at destination. However, reduction in losses is not always without any penalty. The addition of a source buffer along with the destination buffer will potentially increase the playback start time. The impacts of source buffer and destination buffer are further discussed during the simulation of end-to-end system with rate control.
- Minimize the playback start time: This can be achieved by reducing or completely eliminating the packet losses. Minimization in playback start time can

be achieved only if the controller reduces losses during the initial dead-time. Note that for a lossless network, reduction in playback start time can never be achieved.

- Prevent disruptions during media playback: An initial appropriate amount of destination buffering prevents the disruptions at the destination.

C. Major Assumptions

The major assumptions made in this research in simulating the end-to-end packet transport are as follows:

- A single flow travels only along a single path between a source and a destination. Hence, there is no possibility of out-of-order delivery of packets, i.e no flow reversal.
- The processing, transmission and propagation delays are lumped together while computing the end-to-end delay. while computing the end-to-end delay.
- The time delay associated with the feedback signal from destination to source is ignored. It is assumed that feedback signals are available instantly at the source.
- The time delays associated with the control calculations are ignored.

D. Chapter Summary

This chapter provides some initial qualitative discussions leading to the controller synthesis. Possible end-to-end network measurements are discussed which could be used as feedback signals to control the system of interest. QoS metrics are reiterated along with the major assumptions made in this research work.

CHAPTER III

CONTROLLER DEVELOPMENT

In this chapter feedback control methods for the system of interest are outlined. Desired controller performance criteria are discussed. Adaptive control strategies such as unconstrained model predictive control schemes are developed.

A. Desired Controller Performance Criteria

Performance criteria of the controller are deduced from the QoS metrics outlined. Effectiveness of the application level QoS controllers is evaluated based on these criteria. All of these criteria are related to each other, as the performance with respect to one criterion could impact that of others.

Elimination of the playback disruption of packets at the destination is desired. This can be achieved by regulating the destination buffer within certain bounds and making sure that the destination buffer never empties. This approach of controlling the destination buffer only may eliminate the playback disruptions but may not aid in minimizing the losses.

It is also desirable to minimize playback start time at the destination. This is a difficult objective to achieve. Playback start time could be minimized by reducing packet losses. However, increase in playback start time can not be ruled out even when losses are reduced because the reduction of end-to-end delay and packet losses are conflicting objectives. This is due to the initial buffering of packets at the source and at the destination buffers. Importantly, if the increase in playback start time is within the interactivity range, such an increase might be tolerable.

Finally it is desired to reduce or completely eliminate packet losses. Primarily, packet losses occurs because of congestion. Reducing the packet losses means that the end-to-end application layer QoS control aids in congestion control. Reduction in losses is achieved by sending higher rates during periods of low delays or low congestion and vice versa. Reduction in packet losses eliminates or minimizes the destination buffering. Reduction of packet losses also eliminates the media disruptions and also increases the good-put of the end-to-end flow.

B. Available Control Strategies

First, a brief discussion about the various basic control structures that are available for this problem are presented. In order to evaluate the performance of the developed feedback controller, its performance to that of the uncontrolled case, with no feedback, is considered. This is called open-loop control.

1. Open-Loop Control

Open-loop control is the most basic and simple form of control. Based on a reference signal the control effort is determined. The reference signal being the desired value of the output. In this case, it is the desired playback bit-rate. However, errors in modeling and in the presence of significant disturbances, prevents the desired output being achieved. Open-loop control strategies are least robust and also do not offer any QoS guarantee mechanism. Hence, they are not suitable for transporting real-time media over best-effort networks.

2. Reactive Feedback Control

In case of feedback control the system output is monitored continuously with respect to the reference input. The difference between the two signals also called the error is used as an input to the controller. The Proportional Integral Derivative (PID) controller is an example of a reactive controller. Reactive controller reacts based on the network delayed dynamics of the destination and hence it cannot be used to meet the strict delay and loss guarantees that are required for real-time media application [41].

3. Predictive Control Laws

Predictive control is a technique based on implementing a control action that has been computed based on minimizing some future predicted based on the desired reference trajectory. Predictive controllers intuitively should perform better as they react based on anticipated future information. However, the performance of a predictive controller depends on how well the future system response is predicted. In this thesis predictive control schemes are developed and their performance is compared with an appropriate open loop controller.

4. Linear and Nonlinear Control Laws

The Simplistic structure of alinear controller fails to provide any QoS [41] for real-time media applications. The primary reason for this occurrence is that linear controllers do not account for the non-linear dynamics of the network. In this work a nonlinear controller is developed based on unconstrained model predictive control.

C. Controller Development Based on Unconstrained Model Predictive Control Technique

A linear predictive non-liner control law is proposed in this research. The research deals with development of prediction schemes as well as control schemes based on such predictors. Various feedback signals as mentioned in chapter II are used to provide

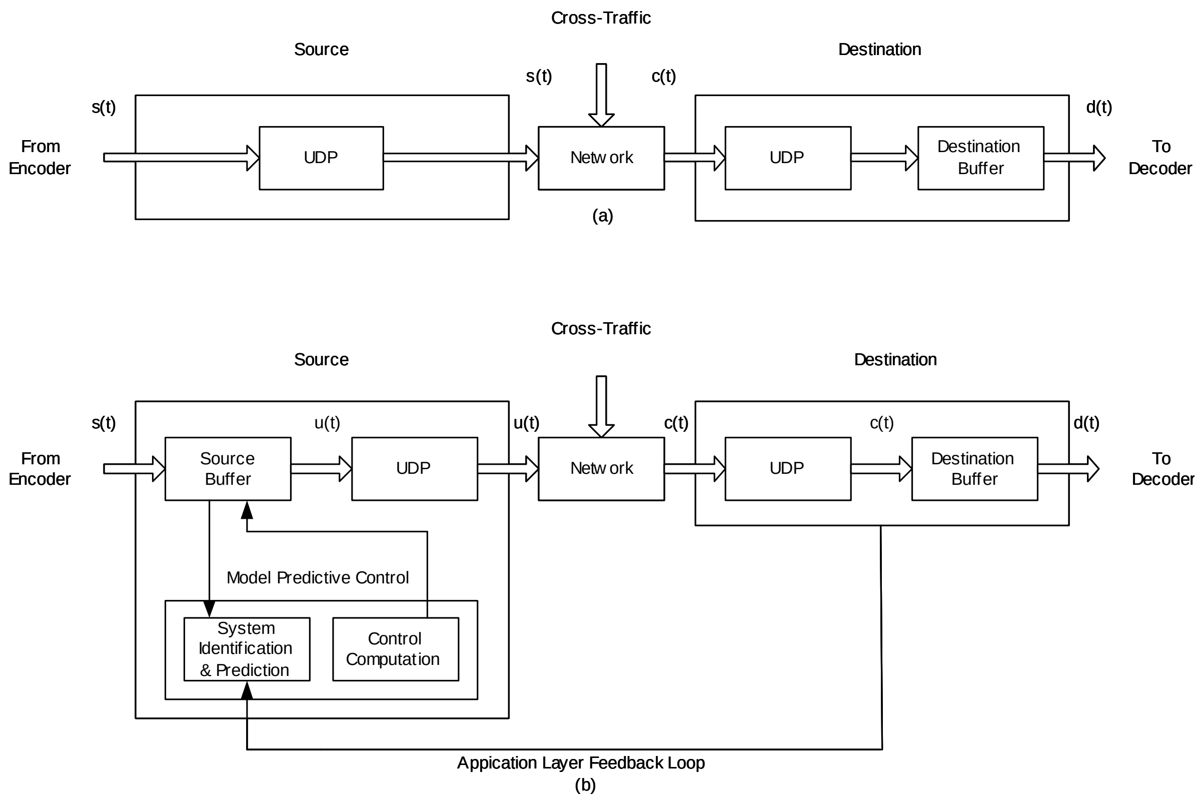

Fig. 3. (a) End-to-End System with No Control and (b) End-to-End System with Application Layer Feedback.

an effective and robust control solution for transporting real-time media applications over best-effort networks.

The specific controller presented is an unconstrained model predictive control (MPC) based on a discrete-time linear model. The basic objective of the MPC control law is to determine a set of controls for a given control horizon that will minimize the sum of the squared deviations of the predicted output from the desired set point. An end-to-end system with no control and an end-to-end system with application layer QoS control using the MPC technique is depicted in Figure 3.

In case of the end-to-end system with no control the presence of a destination buffer is observed. The purpose of this buffer is to build up content so as to eliminate the disruptions when the content is played back. This adds delay to the playback start time of the media application. In case of the system with application layer feedback a source buffer is present in addition to the destination buffer. The purpose of the source buffer is to help the source in implementing a time-varying control effort for the send rate. The total end-to-end delay remains about the same because in the closed-loop case the destination buffer is reducing while the source buffer is increased. Nevertheless buffering at the source rather than at the destination could impact the packet losses. The use of minimal buffering at the destination in the closed loop case is intended to eliminate the playback disruptions.

A brief discussion about various degrees of freedom that are available to tune the end-to-end system with application layer control as shown in Figure 3 is presented followed by the MPC algorithm.

- Predictor: During the development of SISO model, care has to be taken in developing a robust predictor. This is possible if the input-output data that is collected captures all the dynamics of the best-effort networks. Robust predictors have large prediction horizon which aids in accurate control calculations.

- Controller: MPC by itself has two tuning parameters. One is the penalizing factor λ, and the other is the set point w. λ is a more direct and intuitive tuning parameter than factors such as horizon length. Larger λ leads to slower but more robust control. This will be further discussed in the results section, where λ that is chosen for one set of simulation is effective in the case of other simulations too. Set point determines what the nature of future control sequence should be. A good set point helps in achieving the desired QoS objective well.

- Buffers: Source buffer and destination buffer adds two more degrees of freedom to the end-to-end system. Source buffer helps in implementing large time-varying controls where as destination buffer aids in eliminating the disruption at the destination. Both these buffers add some amount of delay to the closed-loop playback start time.

Figure 4 shows basic concept for model predictive control approach.

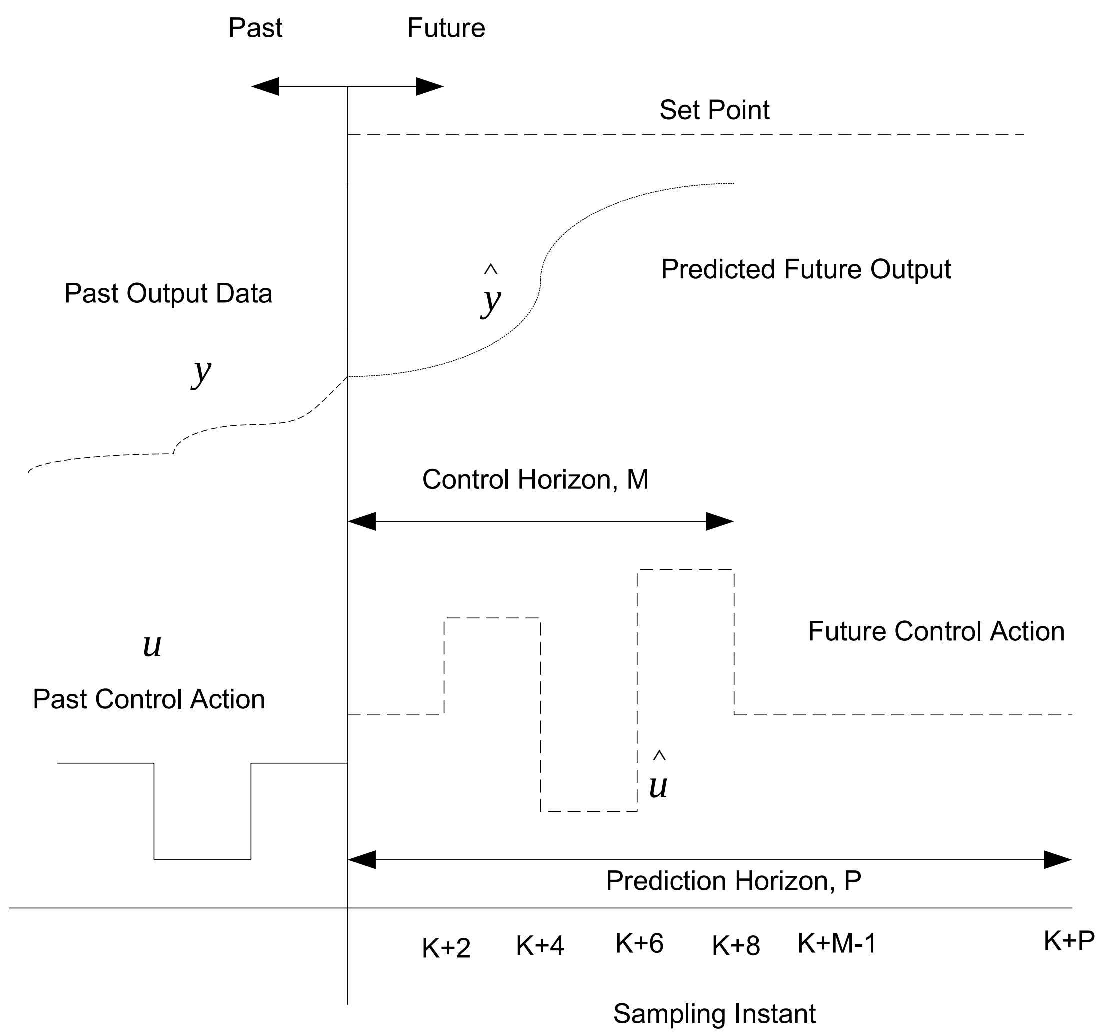

Fig. 4. Basic Concept of the Model Predictive Control Approach.

The MPC approach consists of the following steps.

- Identification of the system. Based on past input and output measurements a linear or even a non liner input-output model is developed which allows computation of future values of the output. In this thesis an AutoRegressive with Exogenous input (ARX) model is used for this purpose.

- Use of the identified model to compute control actions. The identified model is used to predict the system output along a future time horizon. Next, a sequence of control actions is computed along a future time horizon. Finally, a receding horizon strategy is implemented.

The AR in the ARX model refers to the autoregressive part and X to the extra input called the exogenous variable. The general Single-Input Single-Output (SISO) ARX model is represented by the following equation:

$$\begin{gathered} y(k+1) = a_1 y(k) + \ldots + a_{n_a} y(k-n_a+1) + \\ b_1 u(k-d+1) + \ldots + b_{n_b} u(k-d-n_b+2) + e(k+1) \end{gathered} \tag{3.1}$$

where $y(k)$ is the output of the SISO ARX model, $u(k)$ is the input to the ARX model, n_a and n_b are the number of past outputs and number of past inputs respectively, d is the system dead-time and k is the specific sampling instant number. Calculation of coefficients a_i and b_i is the identification process of the system. In this thesis controllers based on identified SISO system is developed. Input to the model is the send rate, $u(t)$, and the output, $y(t)$, is packet accumulation.

In order to develop an unconstrained MPC, output multi-step prediction equations must be derived. From the SISO ARX model represented by the Equation 3.1, the following Single Step Predictor (SSP) of the system output, $\widehat{y}(k+1|k)$, is obtained:

$$\hat{y}(k+1/k) = a_1 y(k) + a_2 y(k-1) + \ldots + a_{na} y(k-n_a+1)$$
$$+ b_1 u(k-d+1) + b_2 u(k-d) + \ldots + b_{n_b} u(k-d-n_b+2) \tag{3.2}$$

Similarly, the p^{th} step-ahead predictor can be written in the following form:

$$\begin{aligned}\hat{y}(k+p/k) = a_1^{(p)} y(k) + a_2^{(p)} y(k-1) + \ldots + a_{na}^{(p)} y(k-na+1) + \\ b_1^{(p)} u(k-d+p) + \ldots + b_{p-1}^{(p)} u(k-d+2) + \\ b_p^{(p)} u(k-d+1) + \ldots + b_{p+n_b}^{(p)} u(k-d-n_b+2)\end{aligned} \tag{3.3}$$

All of the p^{th} step-ahead prediction equations from $P = 1, ..p$ can be combined in matrix form as follows:

$$\begin{aligned}\begin{bmatrix} \hat{y}(k+1/k) \\ \hat{y}(k+2/k) \\ \vdots \\ \hat{y}(k+p/k) \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & \ldots & a_{na-1} & a_{na} \\ a_1^{(2)} & a_2^{(2)} & \ldots & a_{na-1}^{(2)} & a_{na}^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_1^{(p)} & a_2^{(p)} & \ldots & a_{na-1}^{(p)} & a_{na}^{(p)} \end{bmatrix} \begin{bmatrix} y(k) \\ y(k-1) \\ \vdots \\ y(k-n_a+1) \end{bmatrix} \\ + \begin{bmatrix} b_2 & b_3 & \ldots & b_{nb-1} & b_{nb} \\ b_3^{(2)} & b_4^{(2)} & \ldots & b_{nb}^{(2)} & b_{nb+1}^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{p+1}^{(p)} & b_{p+2}^{(p)} & \ldots & b_{nb+p-2}^{(p)} & b_{nb+p-1}^{(p)} \end{bmatrix} \begin{bmatrix} u(k-d) \\ u(k-d-1) \\ \vdots \\ u(k-d-n_b+2) \end{bmatrix} \\ + \begin{bmatrix} b_1 & & & & \\ b_2^{(2)} & b_1^{(2)} & & & \\ \vdots & \vdots & \ddots & & \\ b_{p-1}^{(p-1)} & b_{p-2}^{(p-1)} & \ldots & b_1^{(p-1)} & \\ b_p^{(p)} & b_{p-1}^{(p)} & \ldots & b_2^{(p)} & b_1^{(p)} \end{bmatrix} \begin{bmatrix} u(k-d+1) \\ u(k-d+2) \\ \vdots \\ u(k-d+p) \end{bmatrix}\end{aligned} \tag{3.4}$$

The Equation (3.4) can also be written in the compact form of

$$\hat{\mathbf{y}}_p = \mathbf{A}\mathbf{y}_- + \mathbf{H}_1\mathbf{u}_- + \mathbf{H}_2\mathbf{u}_+ \tag{3.5}$$

where p is the prediction horizon, $\mathbf{u}_+$ the vector of proposed future control actions, $\mathbf{u}_-$ is the vector of past measured control actions, $\hat{\mathbf{y}}_p$, is the vector of the predicted outputs, and $\mathbf{y}\_$ is the vector of the past output measurements

Equation (3.5) can be rewritten as

$$\hat{\mathbf{y}}_p = \mathbf{f} + \mathbf{H}_2\mathbf{u}_+ \tag{3.6}$$

where $\mathbf{f} = \mathbf{A}\mathbf{y}\_ + \mathbf{H}_1\mathbf{u}_-$ consists of only measurements.

The objective of the MPC algorithm is to determine the set of future commands $\mathbf{u}_+$, which are required to drive the predicted output $\hat{\mathbf{y}}_p$ as close to a desired output $\mathbf{y}_{sp}$ (setpoint), in a least-squares sense. If the output is set to be the packet accumulation $Acc(k)$, then the desired output $\mathbf{y}_{sp}$, would be a time-varying value of packet accumulation around the average accumulation of the network in open-loop case. If the output is set to cumulative lost packets $L(k)$, then the desired output $\mathbf{y}_{sp}$, would be zero as we want to eliminate packet losses. The objective function minimizes the error as well as the control effort. If the error vector, that is the difference between the desired and the predicted output is defined as

$$\epsilon = \mathbf{y}_{sp} - \hat{\mathbf{y}}_p. \tag{3.7}$$

Then the cost function can be written as

$$J = \epsilon^T\epsilon + \lambda\mathbf{u}_+^T\mathbf{u}_+, \tag{3.8}$$

where λ is the penalty factor for the variations in the input. Since the MPC is unconstrained, the solution can be obtained by

$$\mathbf{u}_+ = (\mathbf{H}_2^T\mathbf{H}_2 + \lambda)^{-1}\mathbf{H}_2^T(\mathbf{y}_{sp} - \mathbf{f}) \tag{3.9}$$

D. Chapter Summary

This chapter discusses about the various available control strategies. Unconstrained MPC is developed and various parameters that could aid in effective control are discussed. The effectiveness of the control method developed are examined in Chapter IV.

CHAPTER IV

SIMULATION RESULTS

Packet level simulation results are presented in this chapter. The simulation results are obtained utilizing both MATLAB and ns-2. Traces of feedback signals such as packet accumulation are created using ns-2. Control development and control calculations are performed using MATLAB. The effectiveness of the developed control algorithm for specific a network and a specific source send rate is tested for controller robustness purposes.

A. Network Architecture in ns-2 Simulations

The basic network topology that is used for evaluating the effectiveness of the developed controller is shown in Figure 5. The simulation model considered has the following network architecture:

- There are 45 TCP nodes in the network. Each TCP node has about 10 flows which are either ftp or http flows. Also, each node acts as a source as well as a sink. Ftp and http flows sends variable bit rate into the network.
- There are 6 UDP nodes in the network and each node acts as a source and sink. The UDP source sends a constant bit rate into the network.
- There are 2 UDP end-to-end nodes which are being simulated in this research that simulates the real-time media flow. All the other nodes are used to simulate cross-traffic and are not observed.
- The 45 TCP and 6 UDP nodes are connected to a bottleneck link that has 10 mbps bandwidth and a propagation delay of 30ms.

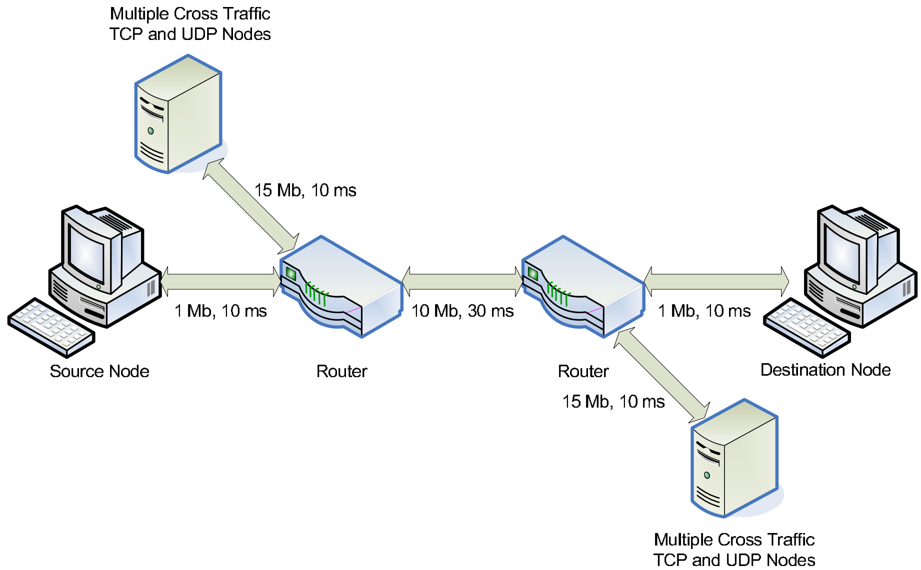

Fig. 5. Network Architecture for ns-2 Simulations.

- Each of the cross traffic nodes that are connected to the bottleneck link has a bandwidth of 15 mbps and a propagation delay of 10 ms.

- Network 1 has 45 TCP nodes and 3 UDP nodes and is the most congested network. Network 2 has 35 TCP and 4 UDP nodes. Network 3 has 25 TCP and 6 UDP nodes with each TCP node having 9 ftp or http flows. Network 4 which is the least congested network has 25 TCP and 6 UDP nodes with each TCP node having 5 ftp or http flows.

Four different network architectures with various send rates are simulated. The network architectures that are simulated have the following characteristics.

- The ratio of the end-to-end flow to that of the total cross traffic flows varies from 1% to 0.25%.

- The traffic mix of packets using UDP as transport layer protocol in simulations is below 10 percent, which is about the same in the Internet.

Its important to note that the networks simulated in this thesis are normalized and used only to demonstrate the feasibility of the developed controller.

In each ns-2 simulation, different flows in cross traffic are started at different times in the first few seconds of the simulation. The application of end-to-end flow sends packets only after a steady state is reached by the cross traffic. The end-to-end flow continues until the very end of the simulations, which lasts for about 15-20 seconds. The network architecture is simulated for various congestion levels by varying the buffer sizes in the routers and by adding or dropping TCP and UDP cross flows. Results of simulations at different send rates with each send rate operating at different levels of network congestion are presented.

B. Effectiveness of the Unconstrained MPC Using the Packet Accumulation Model - Source Send Rate of 100 kbps

Table I summarizes the different networks that are used in simulations. Firstly, the effectiveness of controller that is developed for network 1 with source input rate as 100 kbps is evaluated.

1. Open-Loop Simulations for Network 1

Figures 6, 7, and, 8 present the open-loop simulations for bitrate of 100 kbps for Network 1. In open-loop case no control strategy is implemented. Send rate, arrival rate, destination buffer level, packet loss rate, and cumulative packet losses are shown in Figure 6. Network latency, destination buffer delay, total playback delay and playback rate are shown in Figure 7. Total playback delay is the sum of network delay and the destination buffer delay. In open-loop case no source buffering is performed

Table I. Percentage Packet Losses of Different Networks with 100 kbps as Source Input Rate.

Network	Percentage Packet Losses	Throughput of Cross Flows (Mb)
Network 1	7.95%	17.73
Network 2	5.61%	17.8
Network 3	3.9%	17.82
Network 4	2.02%	17.08

but in control led cases buffering at source will be shown. Playback rate is compared to the source input rate as shown in Figure 7. In all the open-loop cases, minimal buffering at the destination is performed to eliminate the playback disruptions during the playback time. In Figure 8, the cumulative flows along with packets accumulation and cumulative packet losses are depicted. Note that the cumulative packet losses shown in Figure 8 is different from that shown in Figure 6. Cumulative packet losses shown in figure 8 is derived based on cumulative rates and is not an exact or true representation as shown in Figure 6.

The case study that is presented above will be the basis for comparison with the controlled case. The open-loop case presented above will be compared with MPC controller using packet accumulation model. Since in both open-loop and closed loop cases as disruptions can be avoided by buffering at destination, effectiveness of the controller will be evaluated by comparing packet losses, and playback start time.

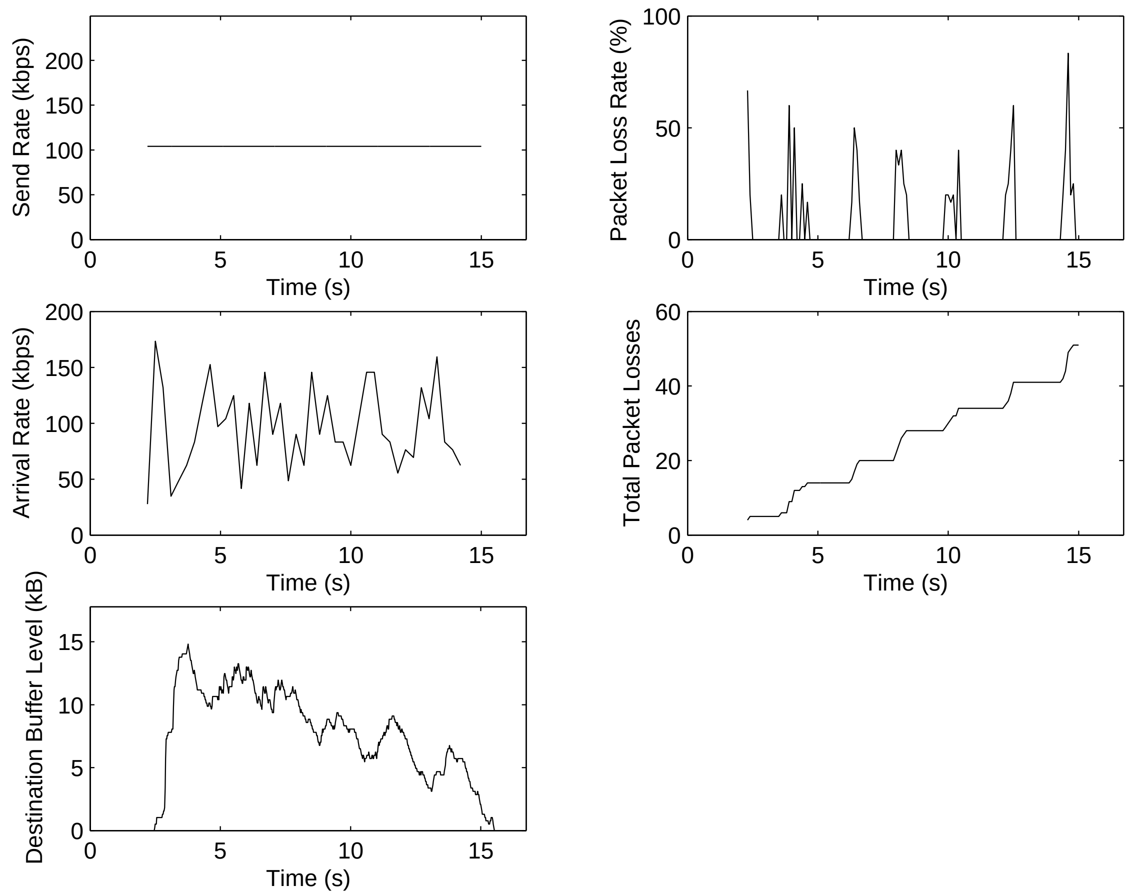

Fig. 6. Buffer Level, Send and Arrival Rates, and Loss Plots for Open-Loop Control with 100 kbps Source Send Rate Using Network 1 Traffic Conditions.

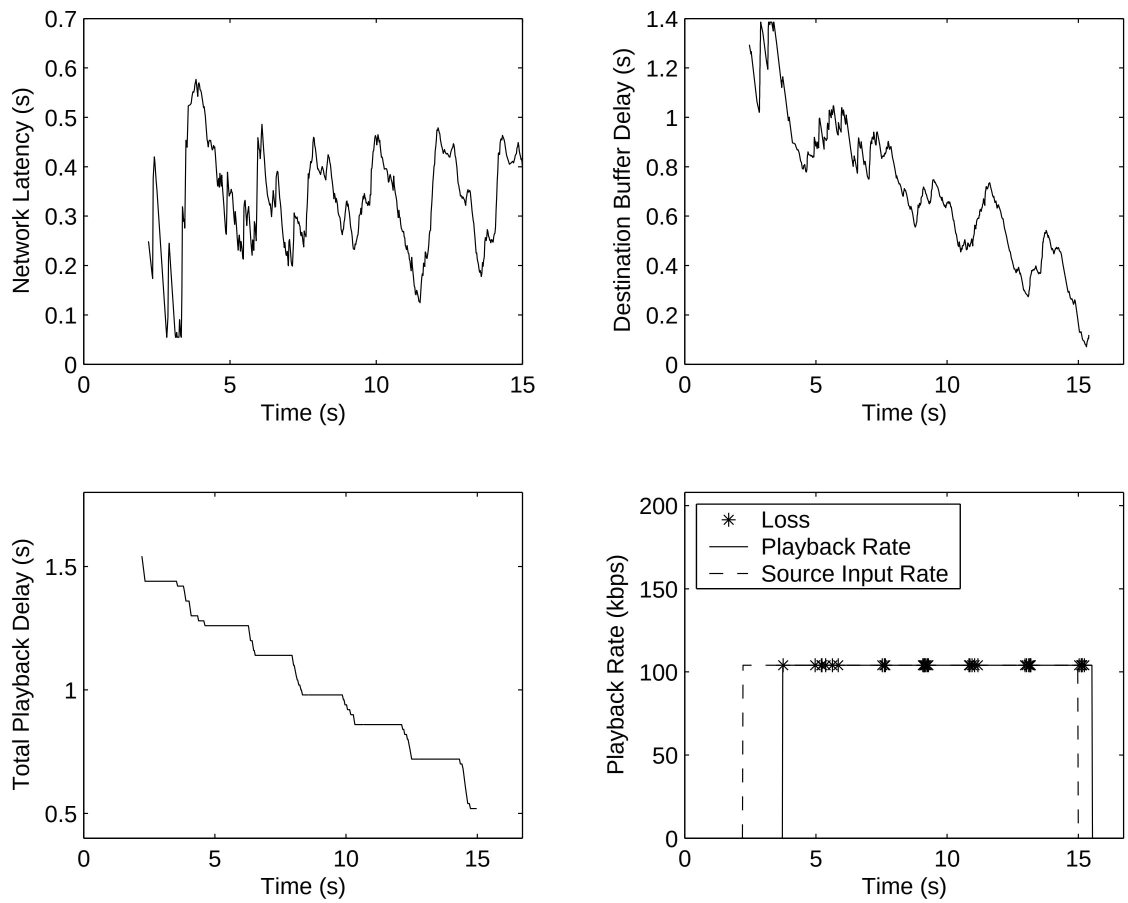

Fig. 7. Delays and Playback Rate Plots for Open-Loop Control of 100 kbps Source Send Rate Using Network 1 Traffic Conditions.

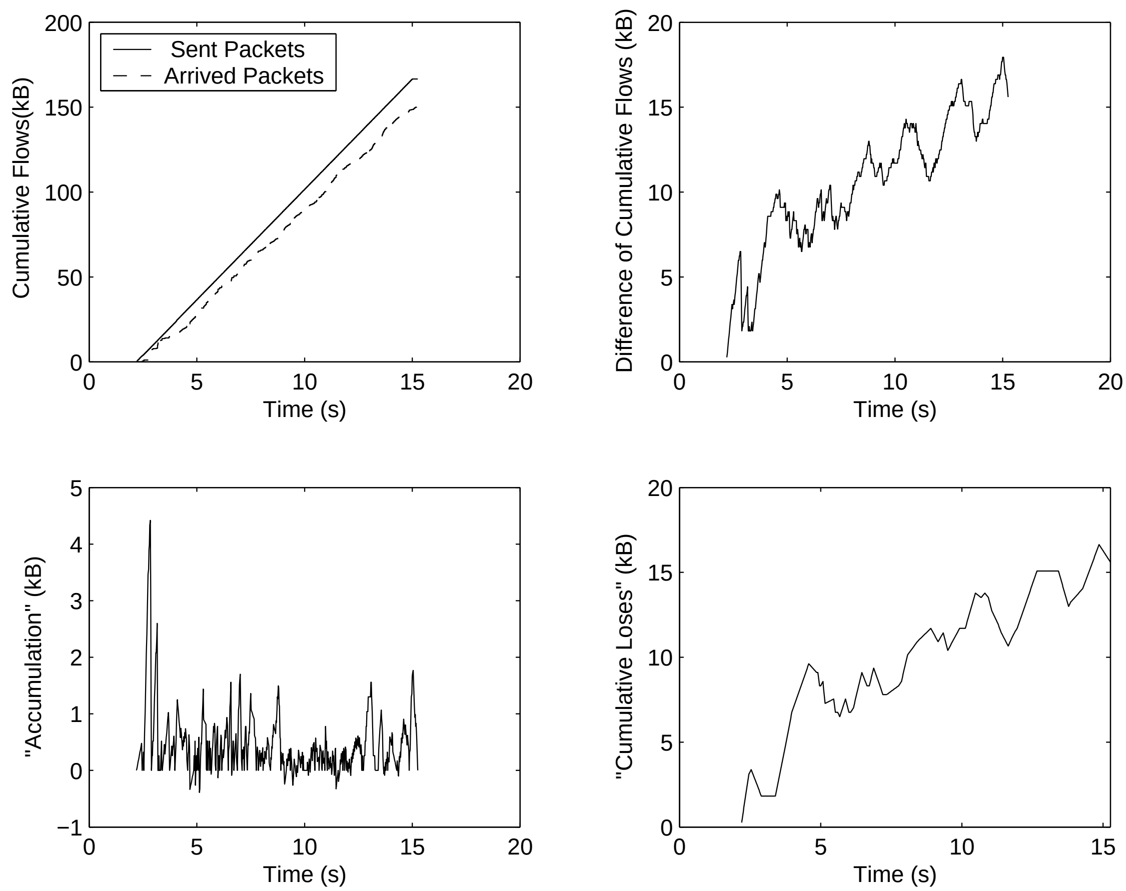

Fig. 8. Cumulative Flows and Accumulation for Open-Loop Control of 100 kbps Source Send Rate Using Network 1 Traffic Conditions.

2. Unconstrained MPC Using Packet Accumulation Model for Network 1

MPC is developed using packet accumulation model to investigate its impact on losses and playback start time. Various degrees of freedom as discussed in previous chapter are varied systematically and their impacts are studied. The MPC is implemented by using the available predictor that was developed off line. Input-output measurements that are available from the open-loop cases are used to compute the future set of controls for the specified time horizon. The prediction horizon is chosen to be about 3 seconds with 20 milliseconds as sampling rate. Receding horizon for implementation of control sequence is 1 second. These horizons are retained when the controller is tested on different networks. Figure 9 shows 150-step-ahead prediction of packet accumulation.

Mean Square Error (MSE), is used as a performance metric for the developed predictor. MSE is defined is as

$$MSE = \frac{\sum_{k=1}^{N}[y(k) - \hat{y}(k+p/k)]^2}{\sum_{k=1}^{N} y(k)^2} \times 100. \tag{4.1}$$

where N is the total number of samples, $y(k)$ the output measurement, $\hat{y}(k+p/k)$ the predicted output and P is the prediction horizon. MSE of 150-step-ahead prediction of packet accumulation using network 1 for application send rate of 100 kbps is about $52.99\%. Penalizing factor \lambda$ is set to 0.01. Objective of the MPC is set to track packet accumulation varying between 900 to 1200 bytes. Once the above parameters are fixed, initial source buffer is varied to see the impact of the controller. This method is followed for all cases. Figure 10 shows how the variation of the source buffer impacts the packet losses and the playback start time.

Figure 10 shows that the MPC controller impacts the losses and well as playback start time. Losses are reduced by 29.41% and playback start time is reduced

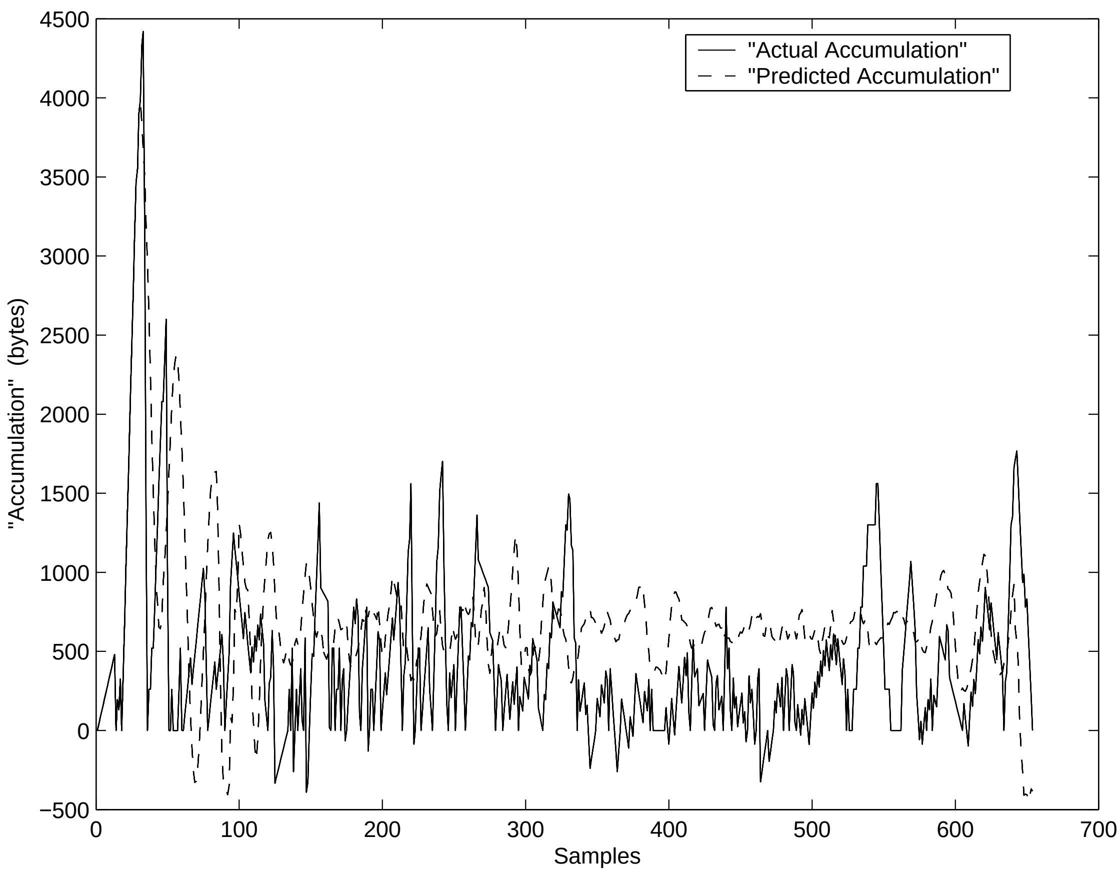

Fig. 9. 150-Step-Ahead Prediction of Packet Accumulation Using Network 1 for Application Send Rate of 100 kbps.

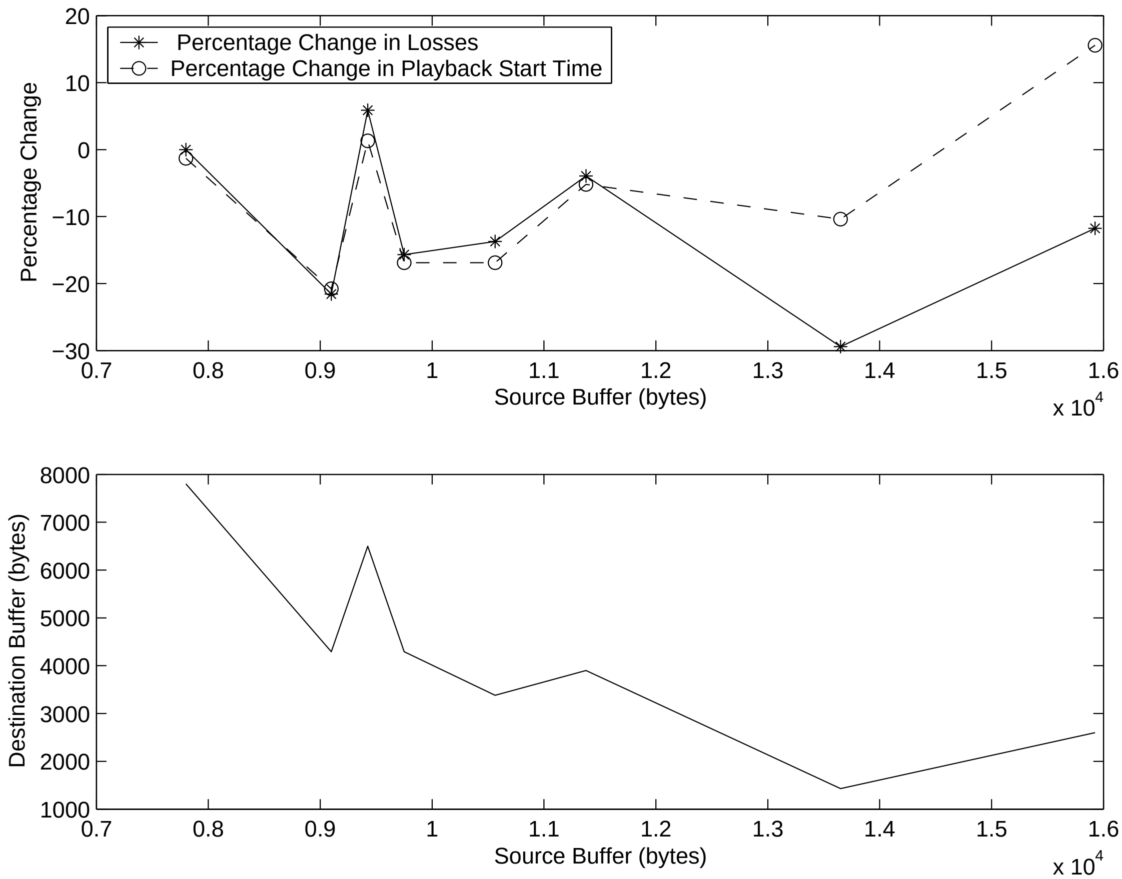

Fig. 10. Percentage Change in Losses and Percentage Change in Playback Start Time for MPC Using Packet Accumulation Model for Network 1.

by 10.38%. Note that minimal buffering at destination of 1430 bytes is required to eliminate the disruption as against 14300 bytes in open-loop case. Decrease in playback start time is seen because packet losses that occurred during the initial buffering period at the destination in open-loop case are minimized. Figures 11, 12, and 13 demonstrate the results of MPC simulation.

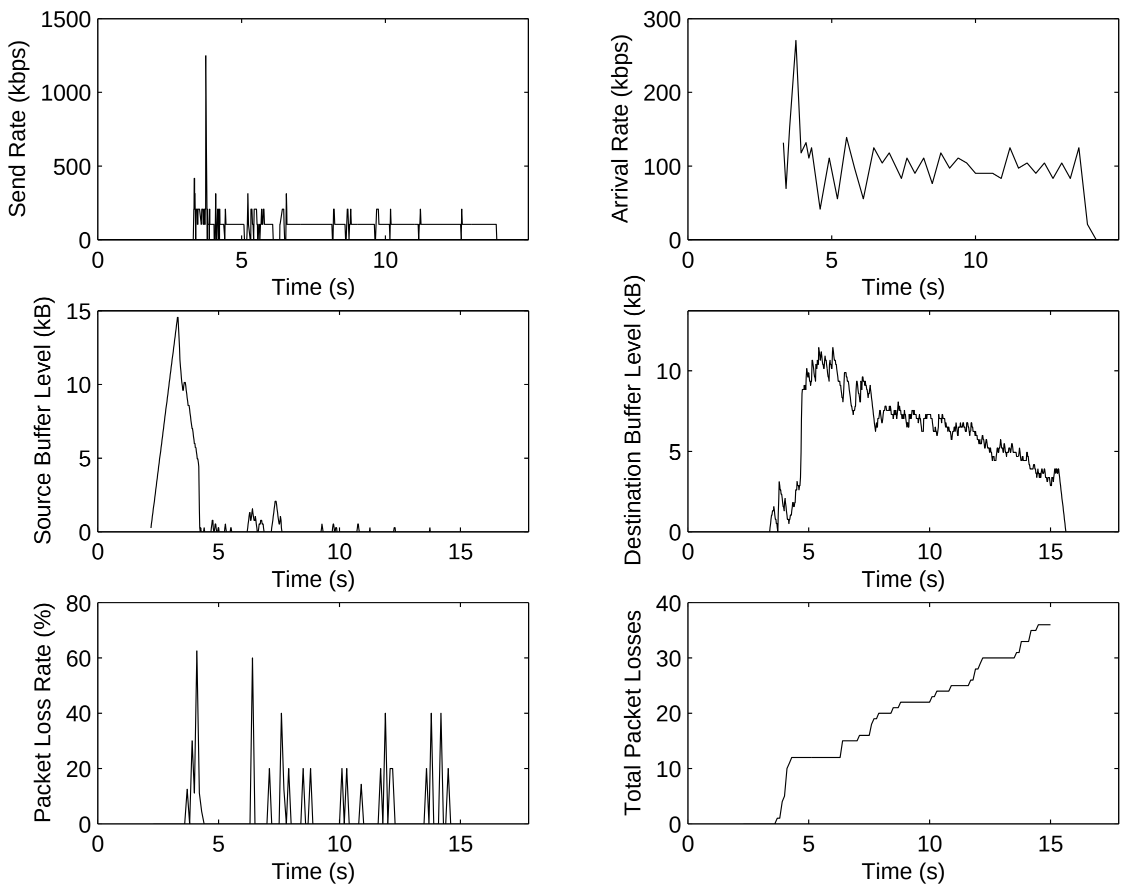

Fig. 11. Buffer Level, Send and Arrival Rates, and Loss Plots for MPC Simulation Using Network 1 Traffic Conditions.

Clearly, improvements in losses and playback start time are evident from Figures 11 and 12 respectively. The MPC controller is also effective in achieving its goal by maintaining the accumulation of packets between 900 to 1200 bytes as shown in

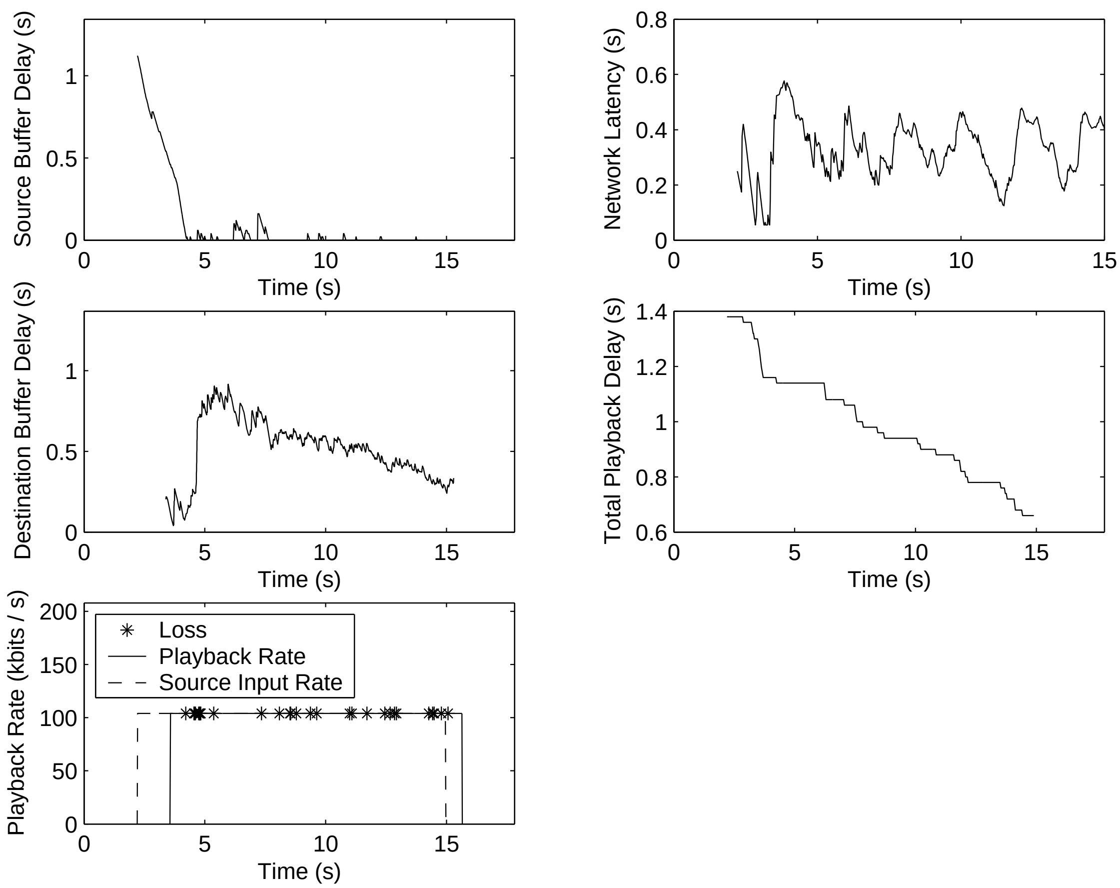

Fig. 12. Delays and Playback Rate Plots for MPC Simulations Using Network 1 Traffic Conditions.

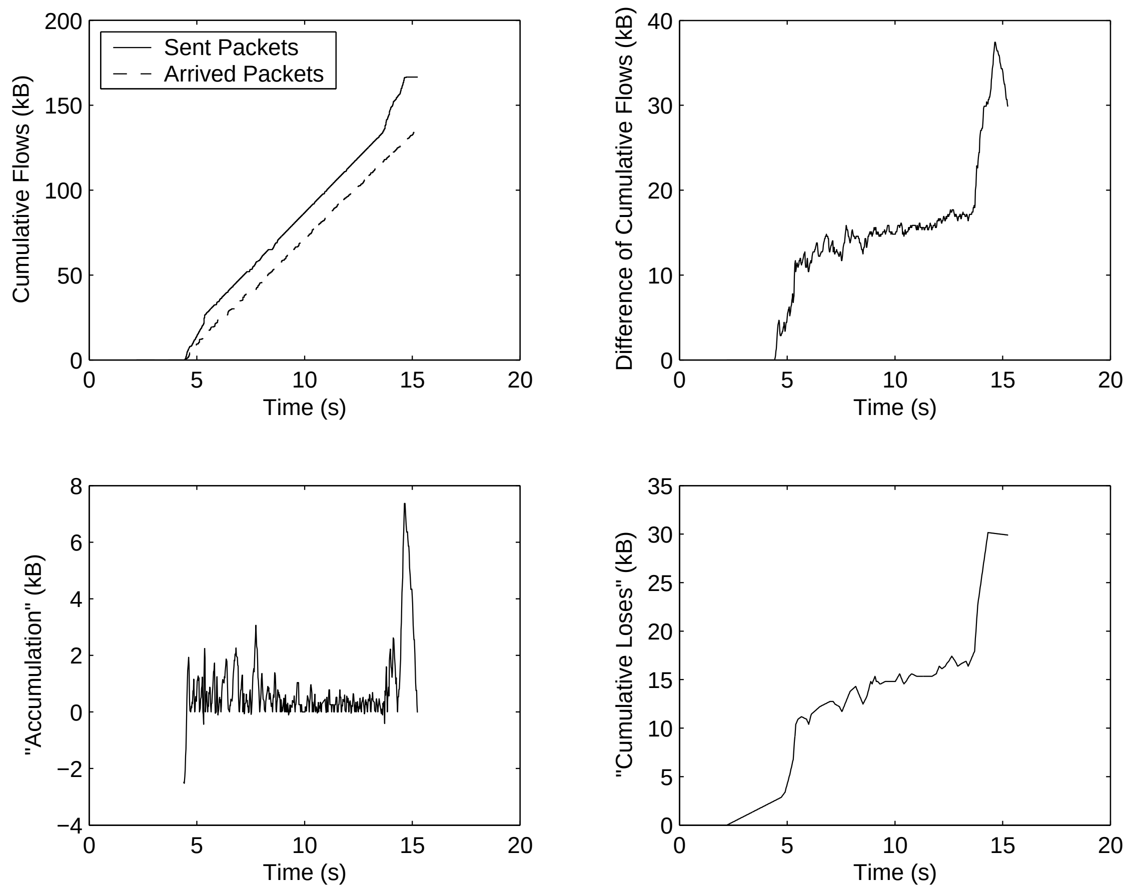

Fig. 13. Cumulative Flows and Accumulation for MPC Simulation Using Network 1 Traffic Conditions.

Figure 13.

Similarly, MPC is implemented on remaining networks as given in table I and its impact on losses and play back start time is summarized in table II.

Table II. Performance of MPC Controller for 100 kbps Send Rate under Different Network Conditions.

Network	**Percentage Change in Packet Losses Compared to Open-Loop**	**Percentage Change in Playback Start Time Compared to Open-Loop**
Network 1	-29.41%	-10.38%
Network 2	-27.77%	+42.10%
Network 3	-16%	+4%
Network 4	-38.46%	+97.29%

MPC works effectively in reducing the packet losses in all the different networks. Except in case of network 1, an increase in playback start time is observed in all the other networks. Note that in case of network 4, percentage change in playback start time is +97.29%. This may be acceptable if the increase is within the interactive range of real-time media application.

Its important to recollect that except for the variation in source buffer, and some minimal buffering at destination all the other tuning parameters are kept constant to study the impact of MPC controller on different networks. The results are significant because the tuning parameters for MPC is reduced to two. Results also indicate that the predictor is robust.

C. Effectiveness of the Unconstrained MPC Using Packet Accumulation Model for Send Rate of 50 kbps

Source input rate is varied to evaluate the effectiveness of the developed MPC. In this case the source input rate is reduced from 100 kbps to 50kbps. Table III summarizes the percentage packet losses for different networks with 50 kbps as input rate. Observe that losses increases marginally when the bit rate is decreased. Increase in losses are evident by comparing Tables I and III, where increase in throughput of cross flows is visible when the bit rate reduced. The smaller end-to-end flow when competing for network bandwidth gets overwhelmed by the larger cross traffic flows.

Table III. Percentage Packet Losses of Different Networks with 50 kbps as Source Input Rate.

Network	**Percentage Packet Losses**	**Throughput of Cross Flows (Mb)**
Network 1	8.26%	17.88
Network 2	5.46%	17.8
Network 3	5.30%	18.05
Network 4	2.96%	17.18

1. Open-Loop Simulations for Network 4

MPC is implemented on different networks as given in table III and its impact on losses and play back start time is summarized. Network 3 is ignored as it has losses similar to those of network 2. Figures 14, 15, and 16 present the open-loop simulations for bitrate of 50 kbps for Network 4. Figure 14 indicates that 4615 bytes of buffering at destination eliminates the disruption during playback in open-loop case.

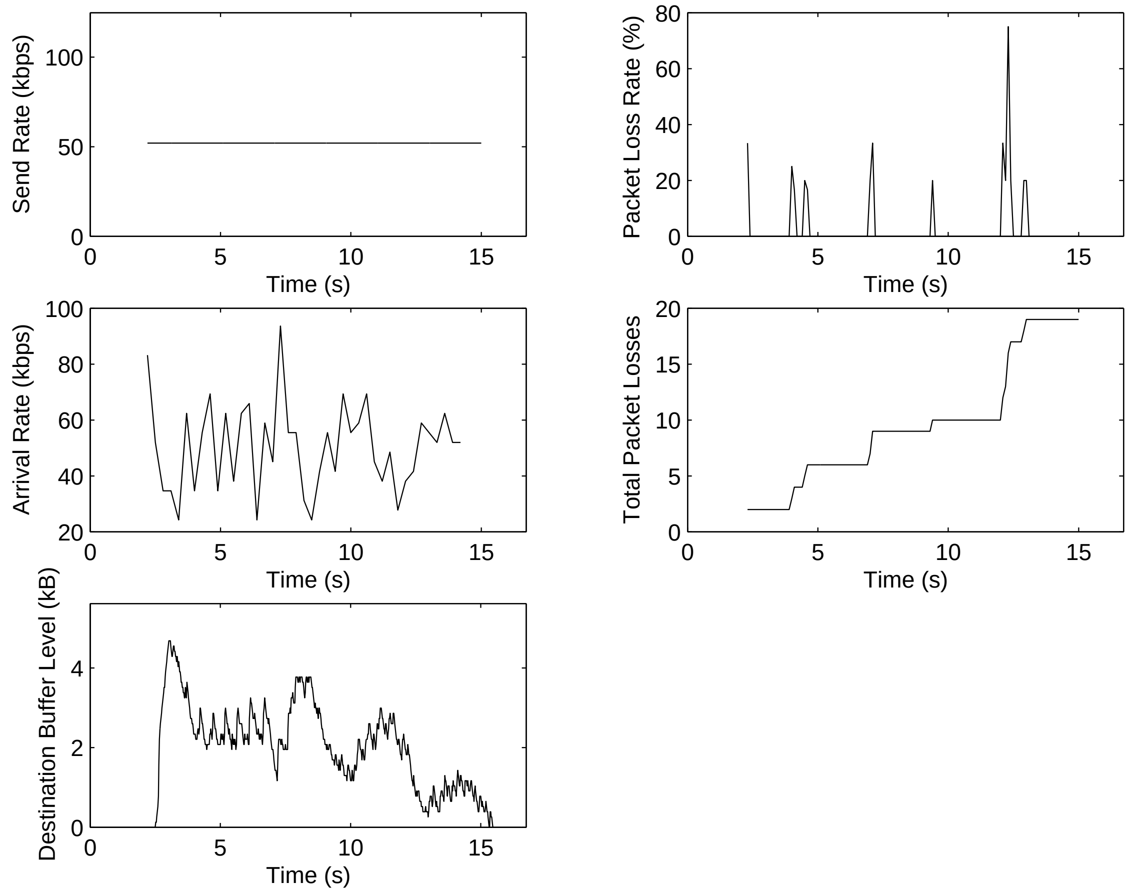

Fig. 14. Buffer Level, Send and Arrival Rates, and Loss Plots for Open-Loop Control with 50 kbps Source Send Rate Using Network 4 Traffic Conditions.

2. Unconstrained MPC Using Packet Accumulation Model for Network 4

Figure 17 shows 150-step-ahead prediction of packet accumulation. MSE of 150-step-ahead prediction of packet accumulation using network 4 for application send rate of 50 kbps is about 50.66%. Once again source buffer is varied to see the impact of the

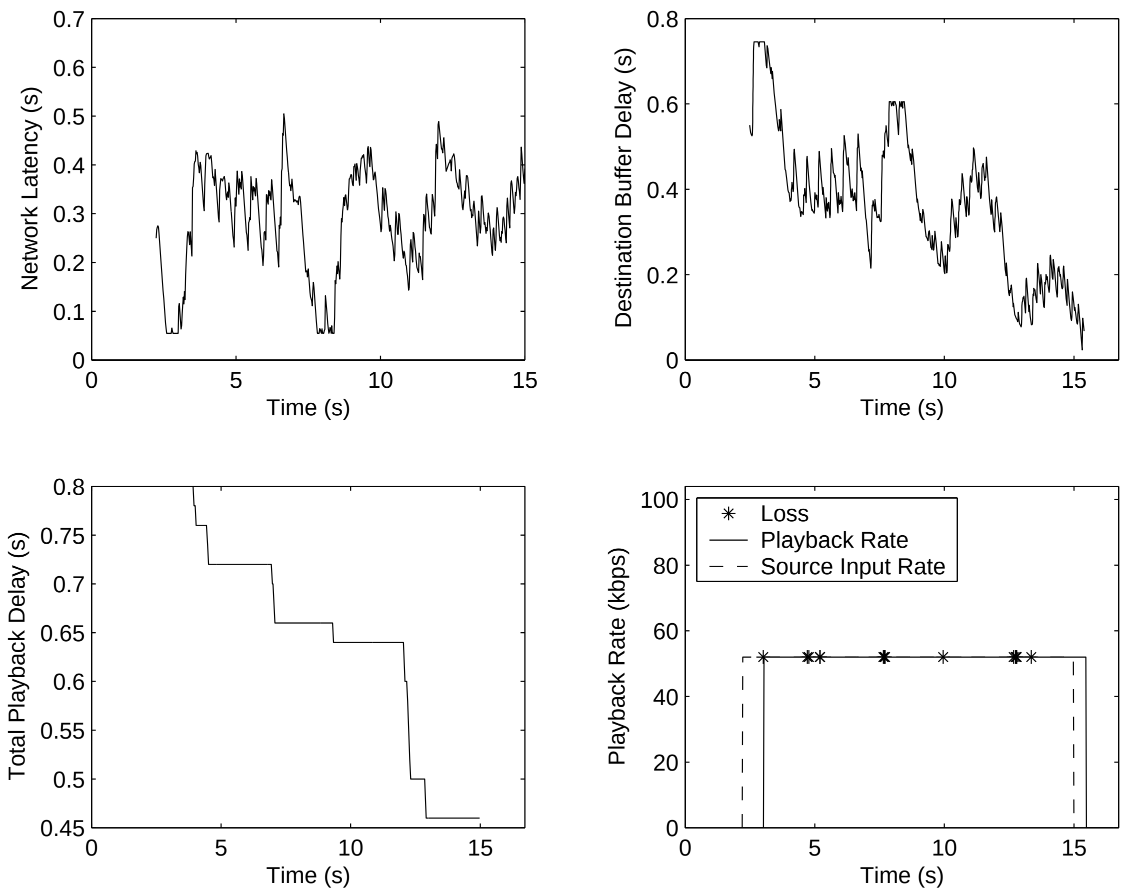

Fig. 15. Delays and Playback Rate Plots for Open-Loop Control with 50 kbps Source Send Rate Using Network 4 Traffic Conditions.

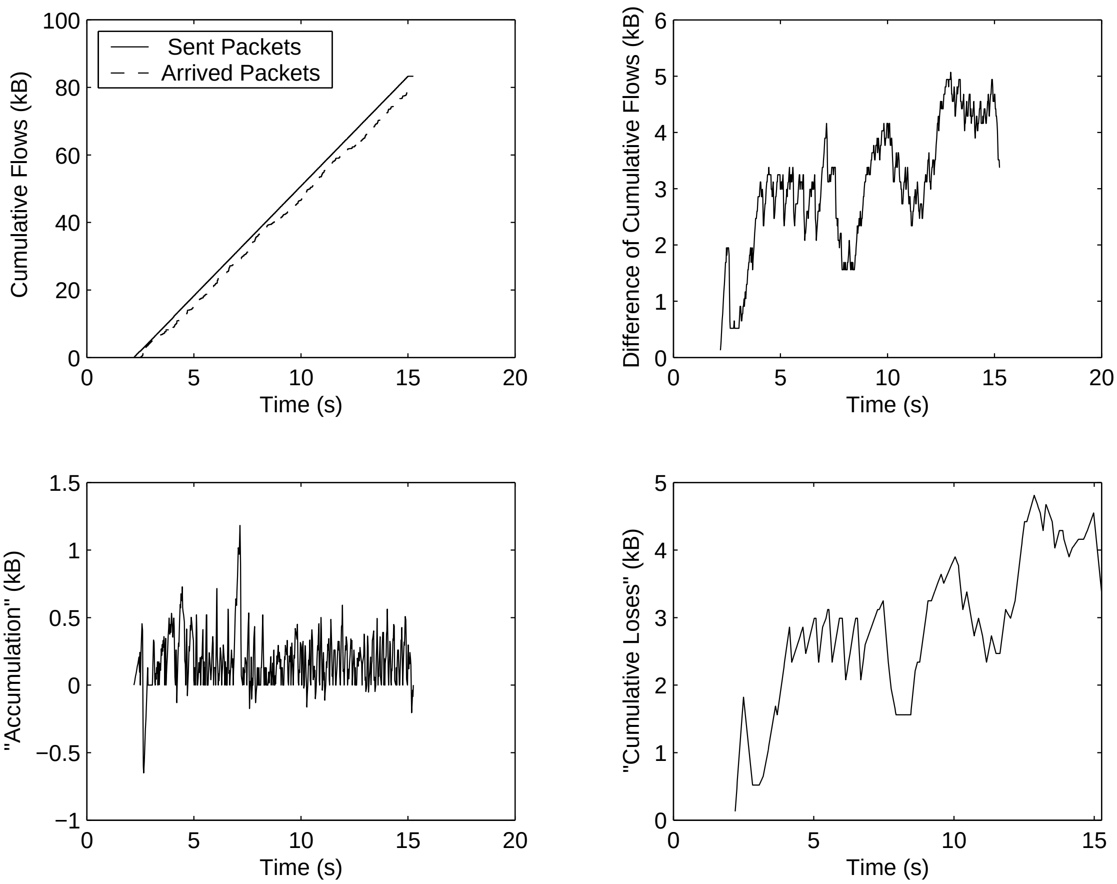

Fig. 16. Cumulative Flows and Accumulation for Open-Loop Control with 50 kbps Source Send Rate Using Network 4 Traffic Conditions.

controller. Figure 18 shows how the variation of the source buffer impacts the packet losses and the playback start time.

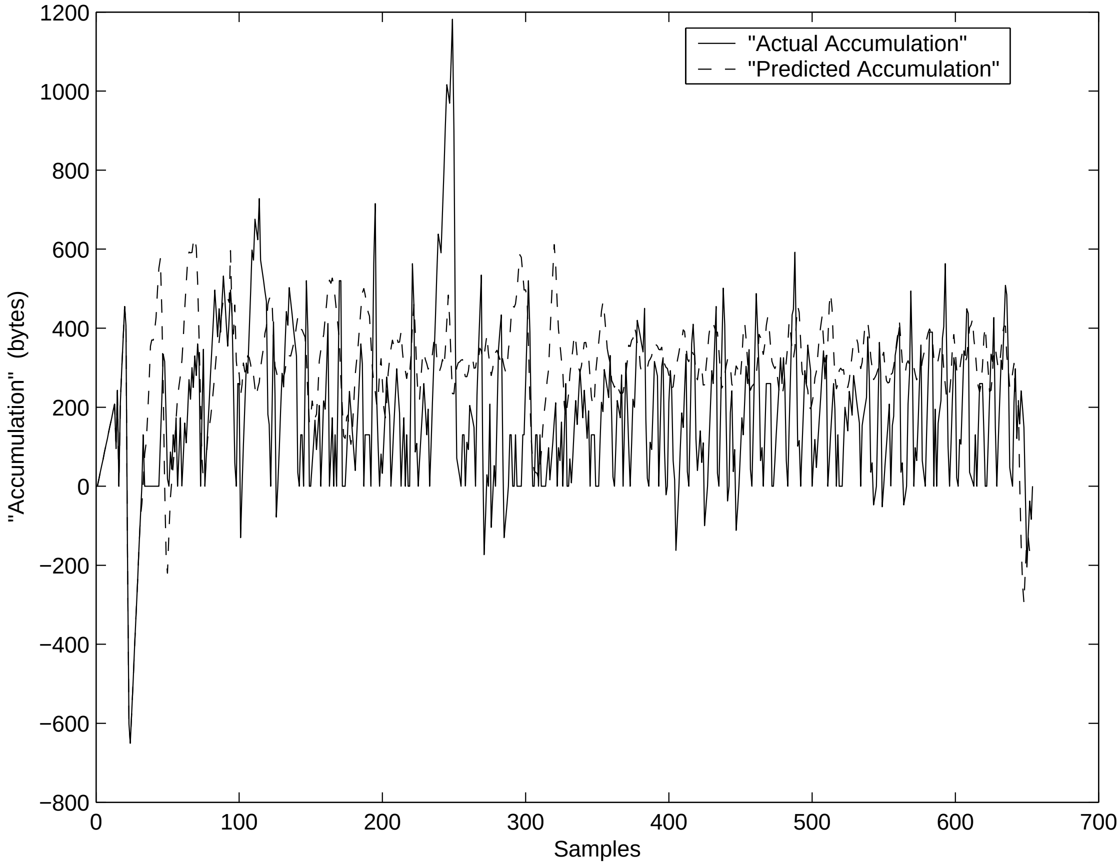

Fig. 17. 150-Step-Ahead Prediction of Packet Accumulation Using Network 4 for Application Send Rate of 50 kbps.

Figure 18 shows that the MPC controller reduces the losses with increase in playback start time. Losses are reduced by 31.57% and playback start time is increased by 25%. Note that minimal buffering at destination required to eliminate the disruption is 950 bytes. Figures 19, 20, and 21 demonstrate the results of MPC simulation.

Similarly, MPC is implemented on remaining networks as given in table III and its impact on losses and play back start time is summarized in table IV.

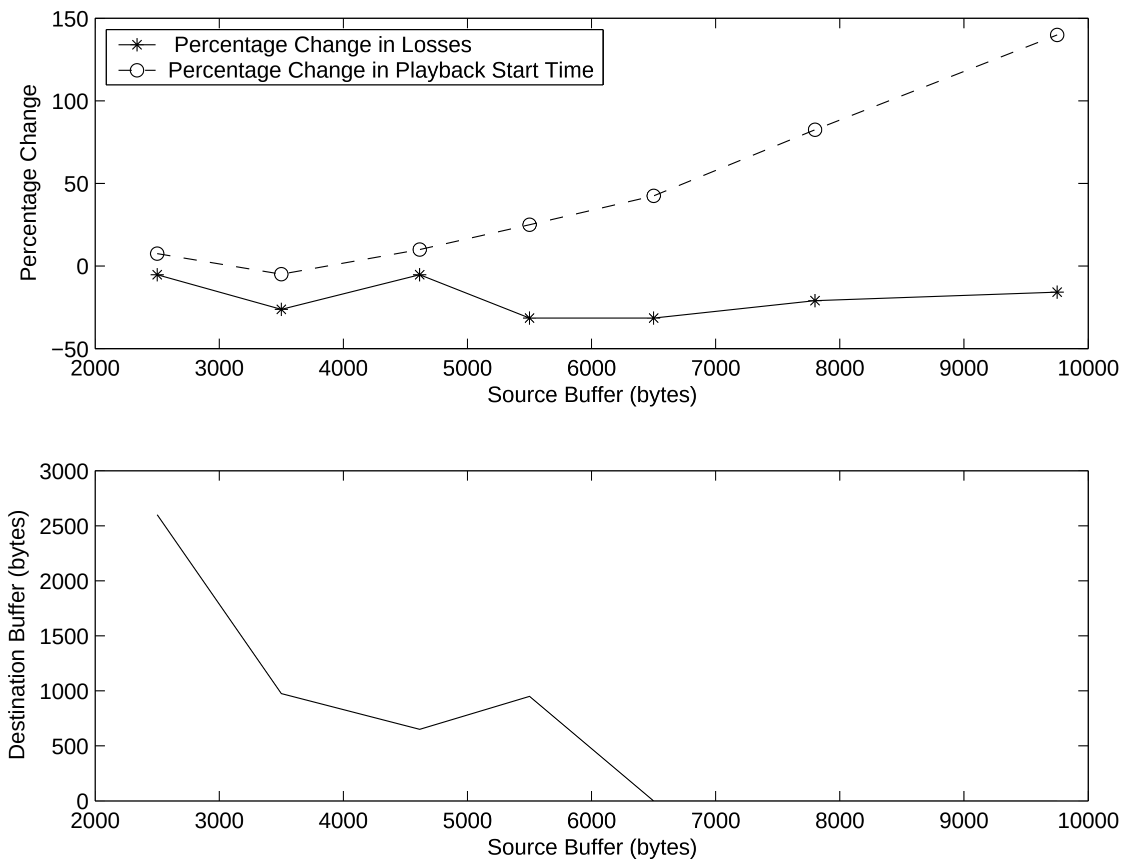

Fig. 18. Percentage Change in Losses and Percentage Change in Playback Start Time for MPC Using Packet Accumulation Model for Network 4.

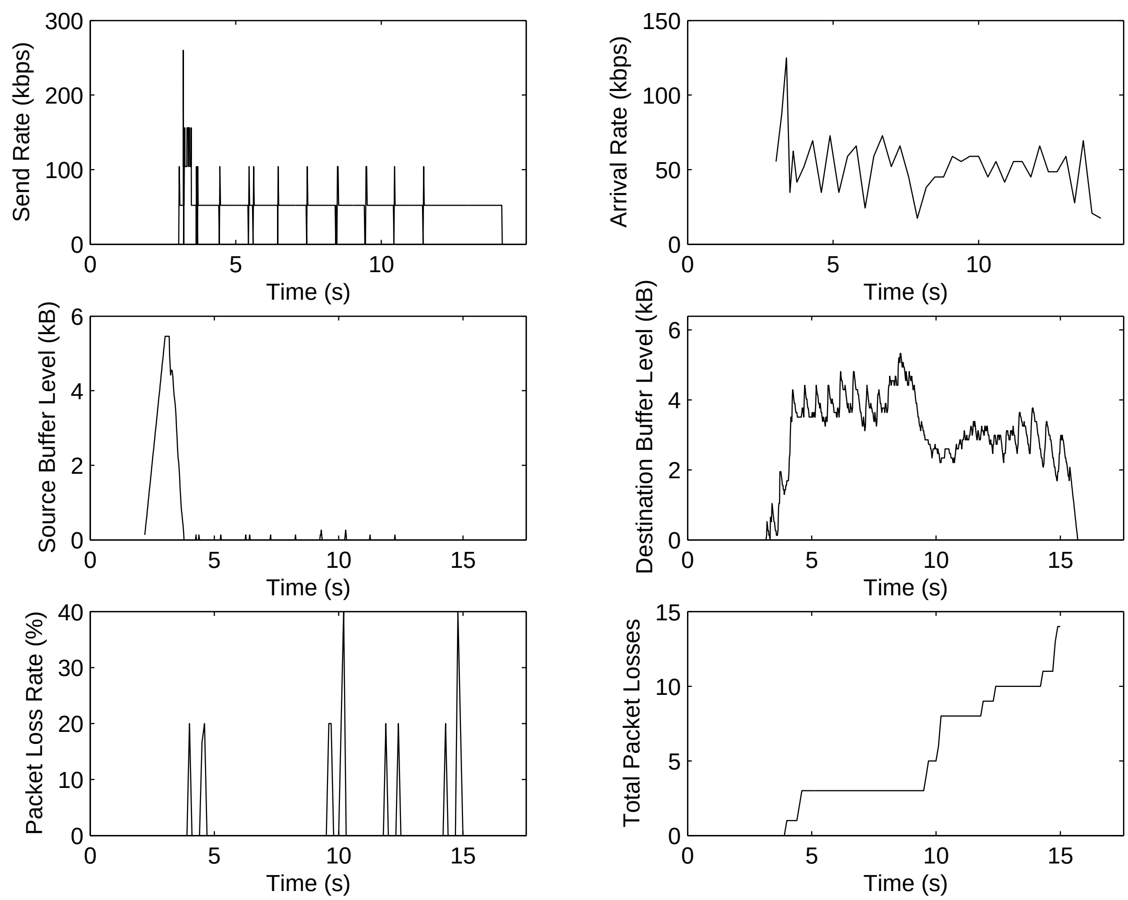

Fig. 19. Buffer Level, Send and Arrival Rates, and Loss Plots for MPC Simulation Using Network 4 Traffic Conditions.

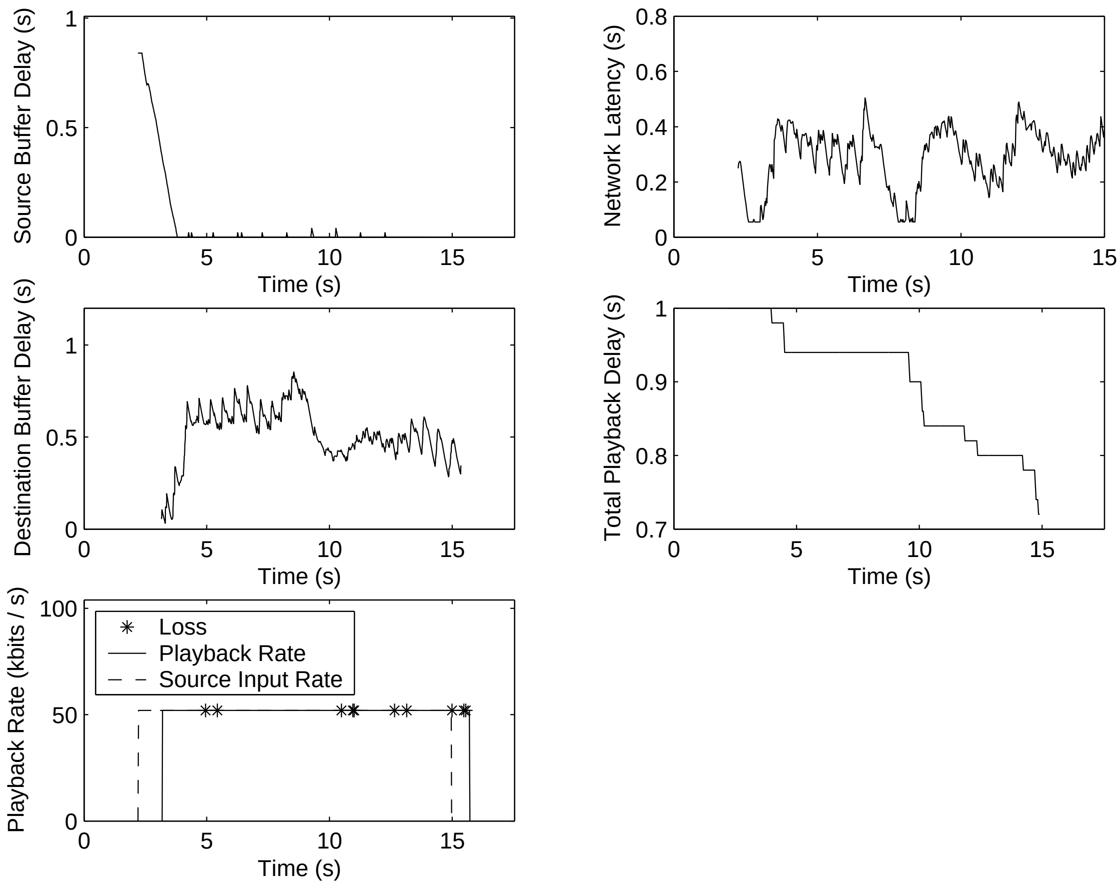

Fig. 20. Delays and Playback Rate Plots for MPC Simulations Using Network 4 Traffic Conditions.

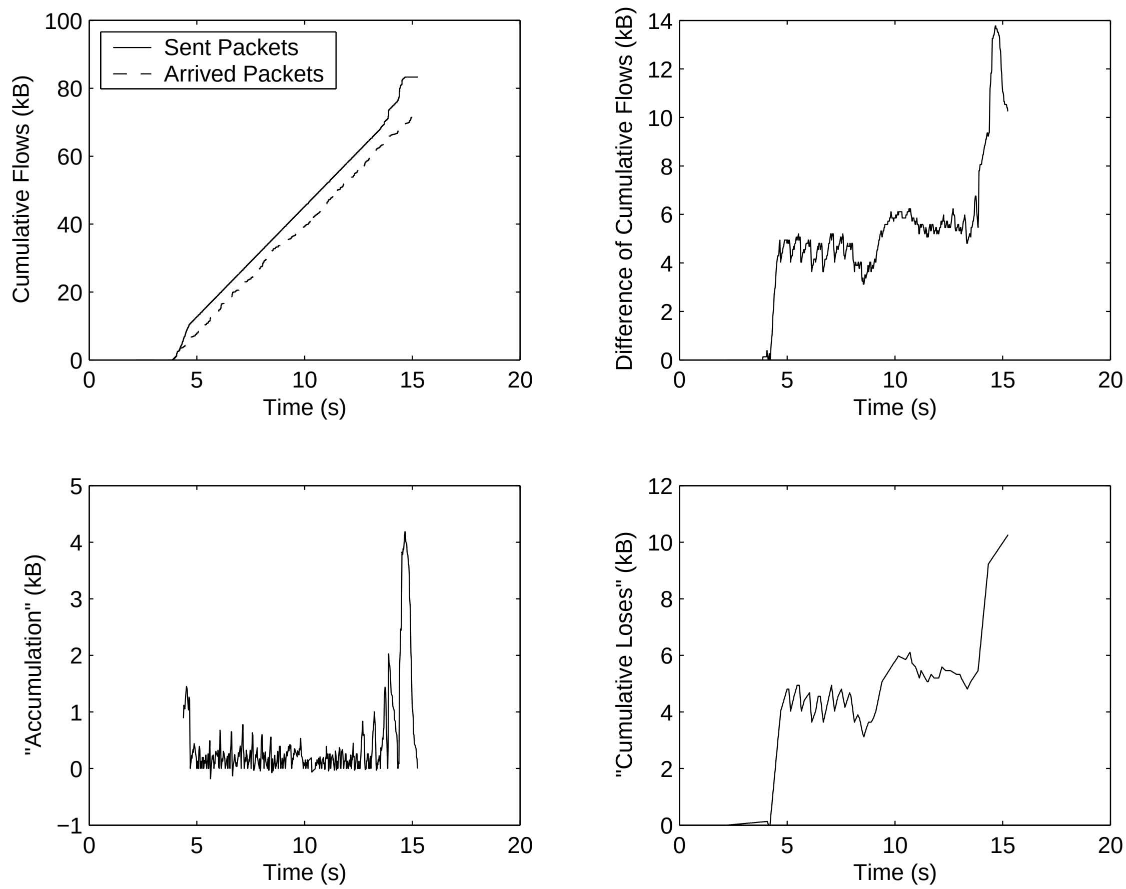

Fig. 21. Cumulative Flows and Accumulation for MPC Simulation Using Network 4 Traffic Conditions.

Table IV. Performance of MPC Controller for 50 kbps Send Rate under Different Network Conditions.

Network	**Percentage Change in Packet Losses Compared to Open-Loop**	**Percentage Change in Playback Start Time Compared to Open-Loop**
Network 1	-18.86%	-11.26%
Network 2	-5.71%	+8.38%
Network 4	-31.57%	+25%

D. Effectiveness of the Unconstrained MPC Using Packet Accumulation Model for Send Rate of 200 kbps

Effectiveness of controller is now studied by increasing the bit rate from 100 kbps to 200 kbps. Percentage packet losses of different networks with source input rate at 200 kbps are shown in table V. Marginal decrease in packet losses for different networks are observed. This occurs because throughput of cross traffic flows have decreased in case of 200 kbps when compared to the throughput of cross traffic flows of 100 kbps.

1. Open-Loop Simulations for Network 2

MPC is implemented on different networks as given in table V and its impact on losses and play back start time is studies. Figures 22, 23, and 24 present the open-loop simulations for bitrate of 200 kbps for Network 2. Figure 22 indicates that 17810 bytes of buffering at destination eliminates the disruption during playback in open-loop case.

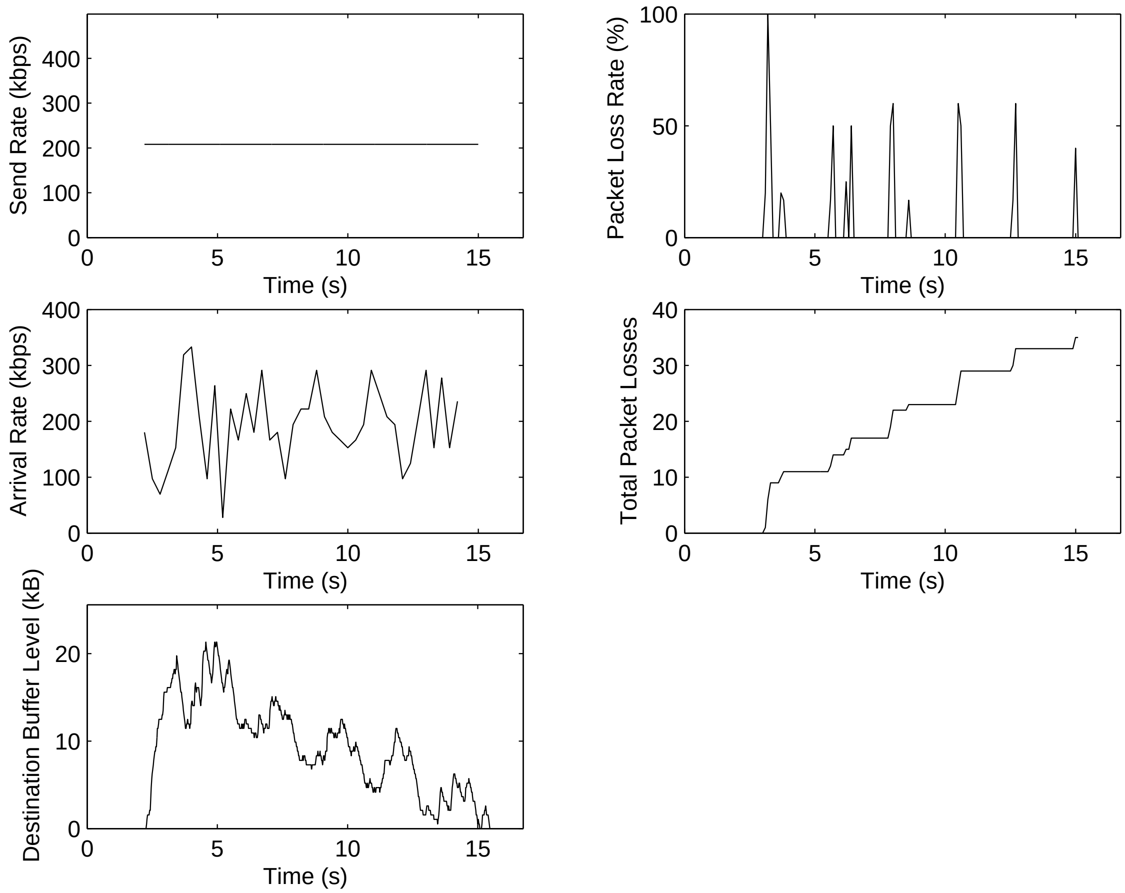

Fig. 22. Buffer Level, Send and Arrival Rates, and Loss Plots for Open-Loop Control with 200 kbps Source Send Rate Using Network 2 Traffic Conditions.

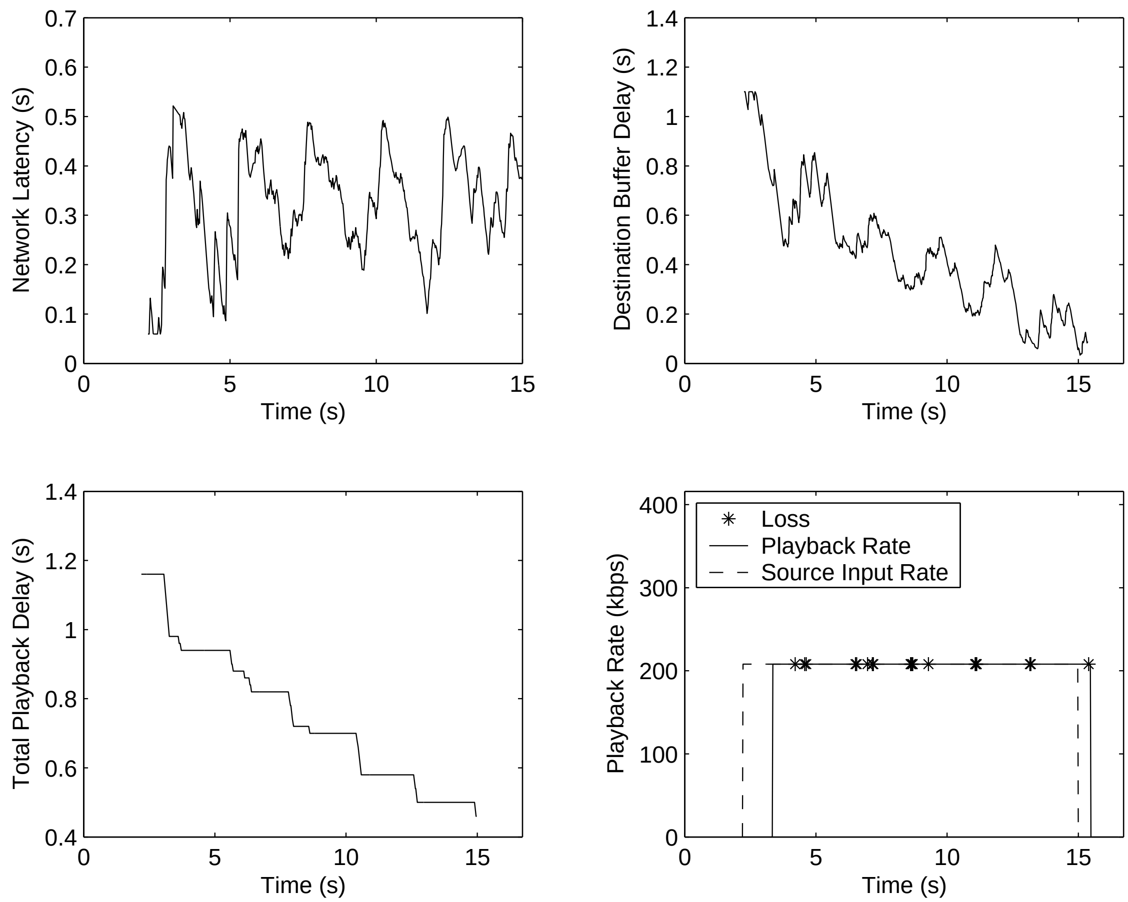

Fig. 23. Delays and Playback Rate Plots for Open-Loop Control with 200 kbps Source Send Rate Using Network 2 Traffic Conditions.

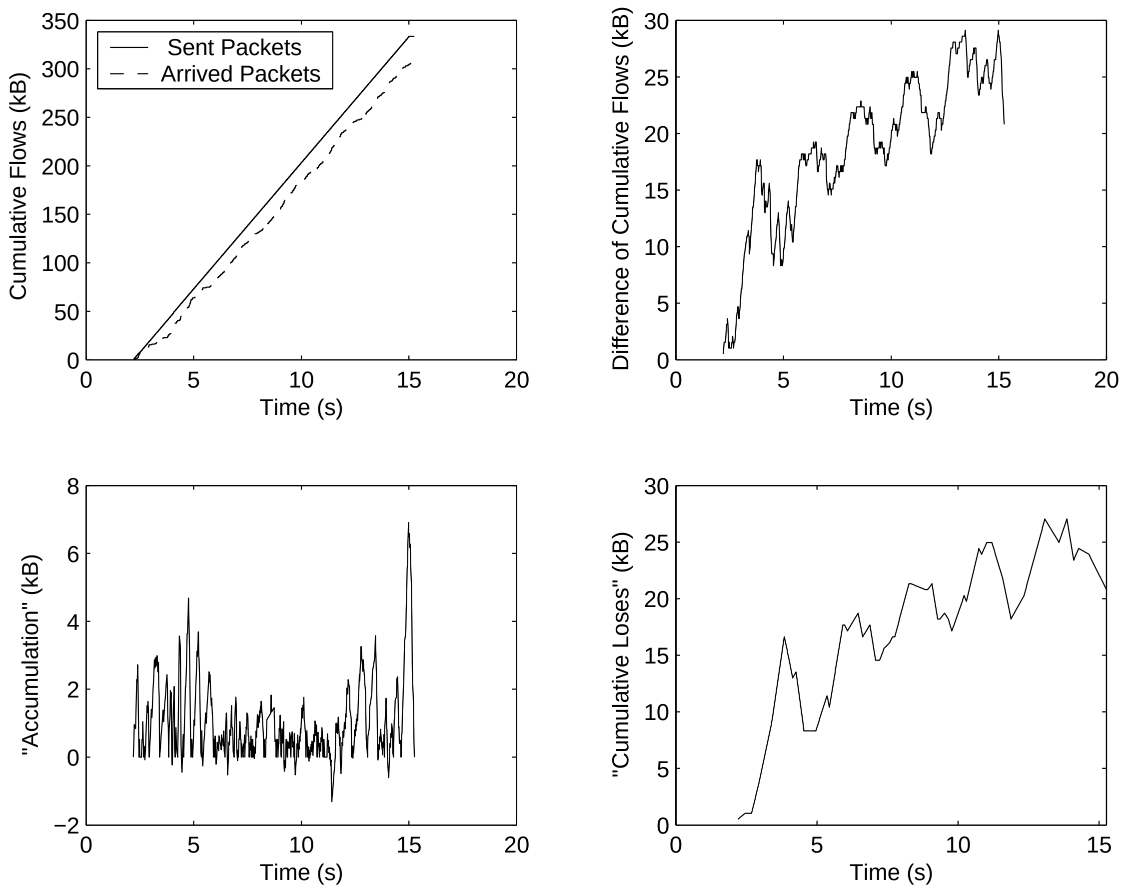

Fig. 24. Cumulative Flows and Accumulation for Open-Loop Control with 200 kbps Source Send Rate Using Network 2 Traffic Conditions.

Table V. Percentage Packet Losses of Different Networks with 200 kbps as Source Input Rate.

Network	**Percentage Packet Losses**	**Throughput of Cross Flows (Mb)**
Network 1	7.95%	17.83
Network 2	5.46%	17.57
Network 3	3.74%	17.52
Network 4	1.71%	17.07

2. Unconstrained MPC Using Packet Accumulation Model for Network 2

In this case, predictor that was developed off-line and used in cases of 100, and 50 kbps is retained. Penalizing factor λ is set to a very small value, close to zero, and the objective of controller is to set to track an accumulation varying between 500 and 1750 bytes. Controller is tuned in order to achieve feasible control outputs that can be implemented. Once again source buffer is varied to study the effectiveness of the controller in minimizing packet losses and playback start time. Figure 25 shows 150-step-ahead prediction of packet accumulation. MSE of 150-step-ahead prediction of packet accumulation using network 2 for application send rate of 200 kbps is about 60.63.66%.

Figure 26 shows that the MPC controller reduces the losses with increase in playback start time. Losses are reduced by -8.57% but playback start time is increased by 129.1%. The increase in playback start time is very large and may not be acceptable. Figures 27, 28, and 29 demonstrate the results of MPC simulation. Figure 29 shows that the controller is unable to achieve its objective. The accuracy of

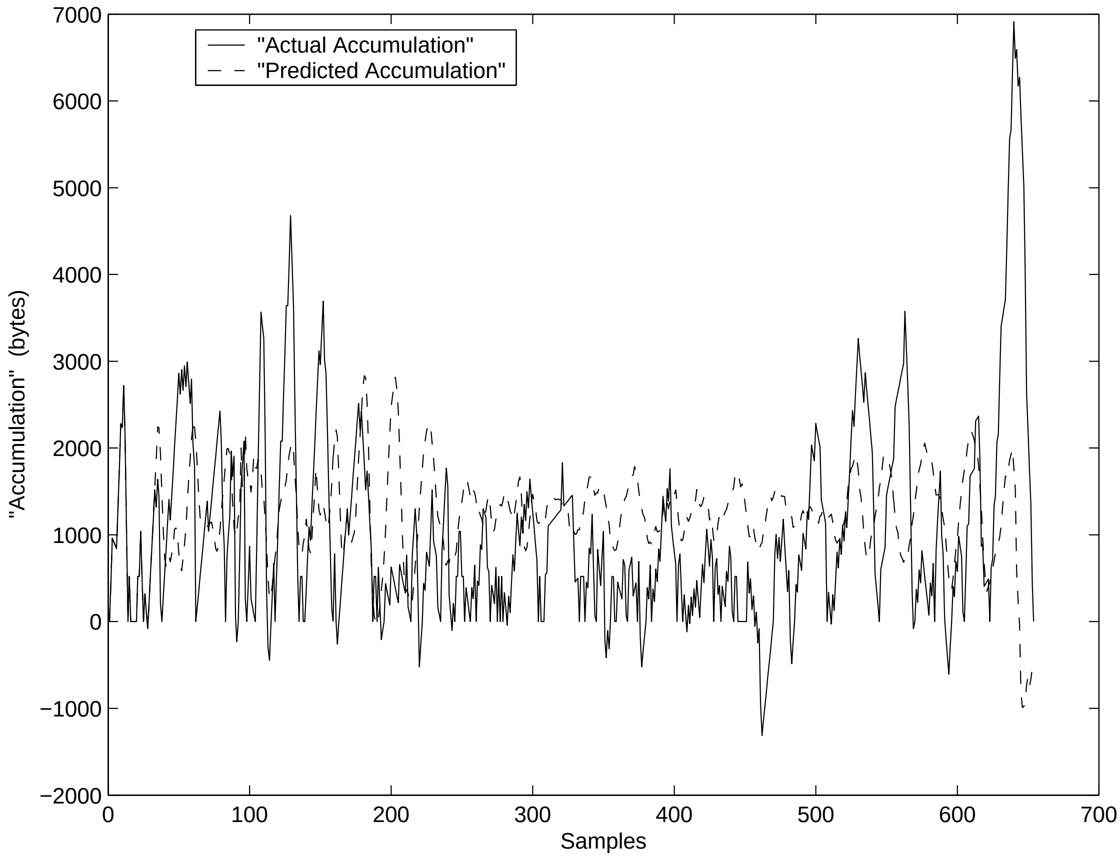

Fig. 25. 150-Step-Ahead Prediction of Packet Accumulation Using Network 2 for Application Send Rate of 200 kbps.

implementing small control actions is difficult because of the large packet size. This adversely effects the performance of the controller.

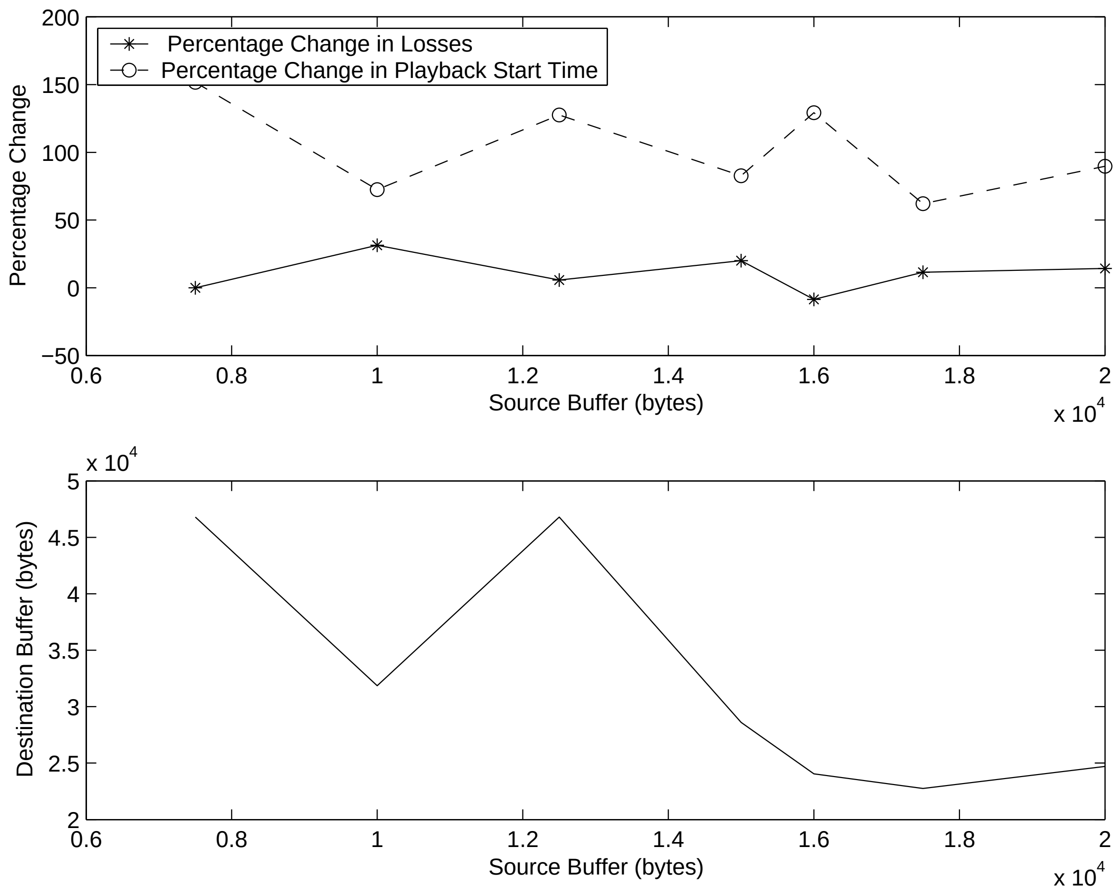

Fig. 26. Percentage Change in Losses and Percentage Change in Playback Start Time for MPC Using Packet Accumulation Model for Network 2.

Table VI shows the performance of MPC on different networks.

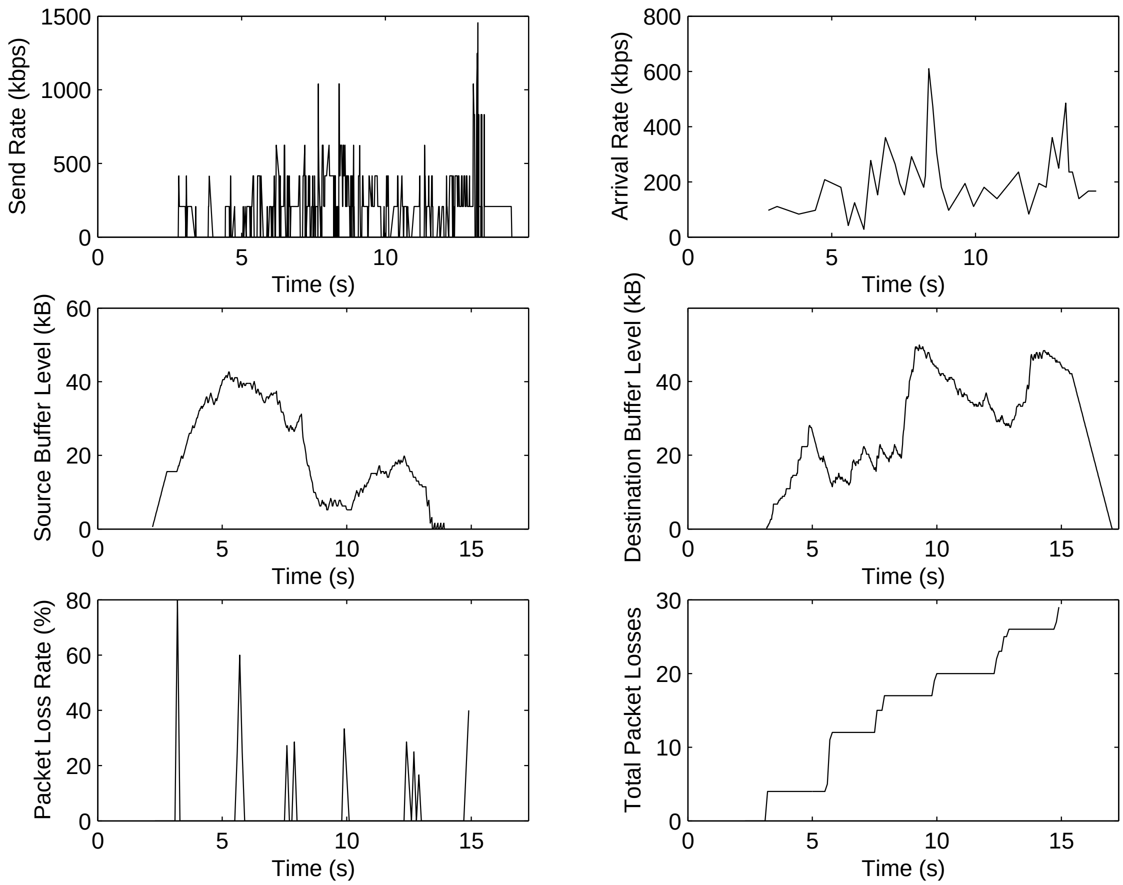

Fig. 27. Buffer Level, Send and Arrival Rates, and Loss Plots for MPC Simulation Using Network 2 Traffic Conditions.

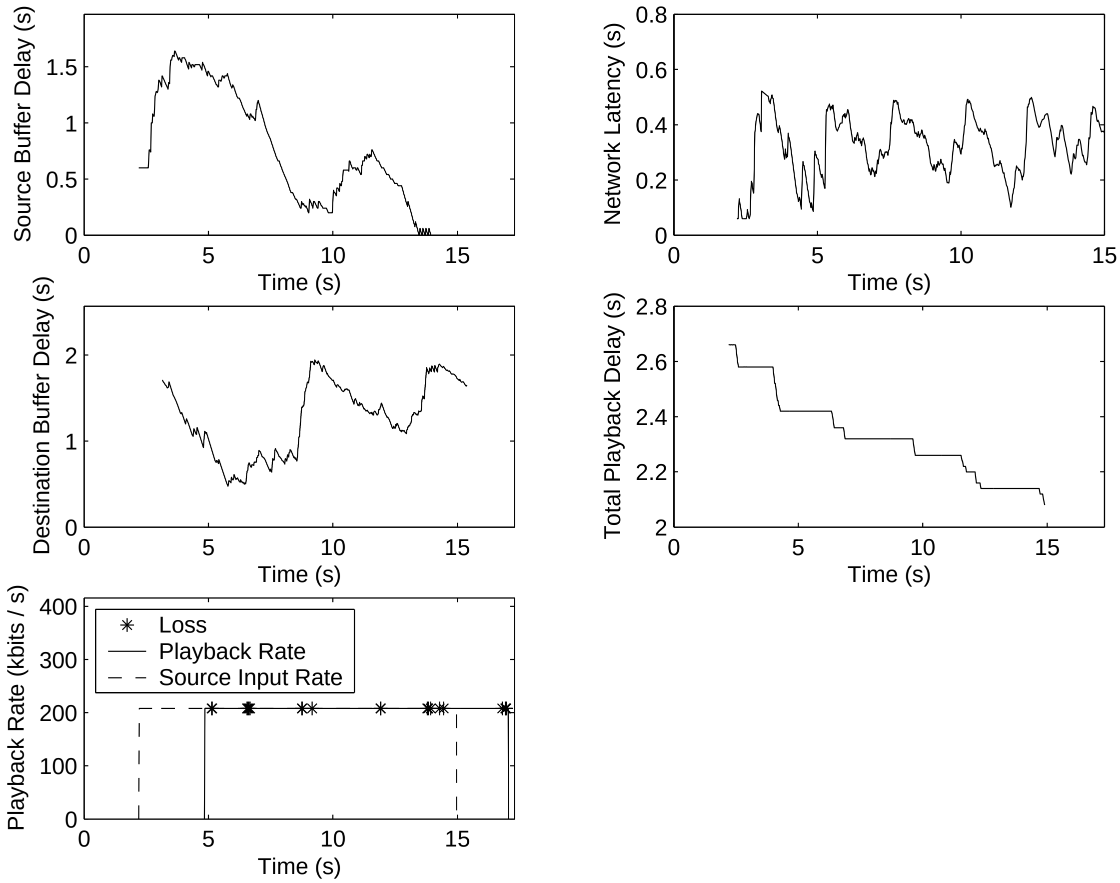

Fig. 28. Delays and Playback Rate Plots for MPC Simulations Using Network 2 Traffic Conditions.

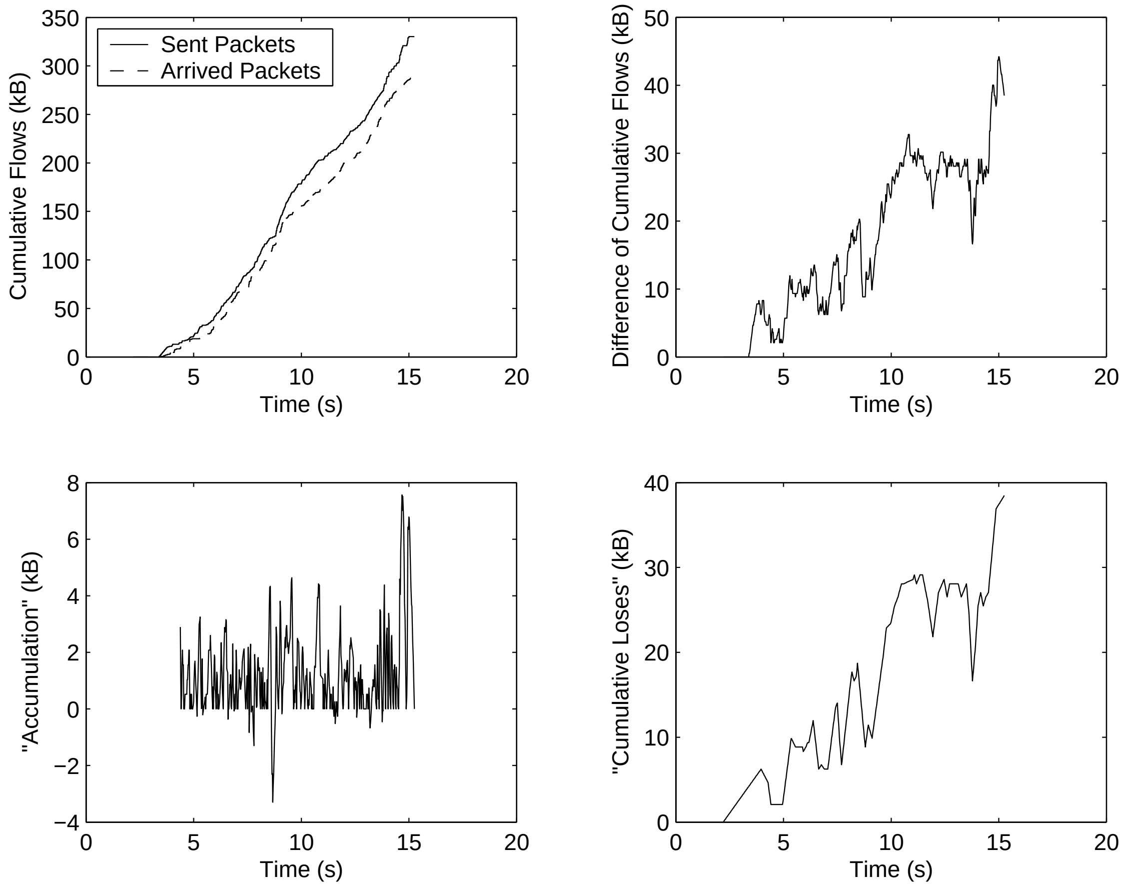

Fig. 29. Cumulative Flows and Accumulation for MPC Simulation Using Network 2 Traffic Conditions.

Table VI. Performance of MPC Controller with 200 kbps Source Send Rate under Different Network Conditions.

Network	Percentage Change in Packet Losses Compared to Open-Loop	Percentage Change in Playback Start Time Compared to Open-Loop
Network 1	-33%	-26.02%
Network 2	-8.75%	+129%
Network 3	-12.5%	+19.14%
Network 4	-18.18%	+91.17%

E. Effectiveness of the Unconstrained MPC on Varying Network Traffic Conditions Using Packet Accumulation Model for Send Rate of 100 kbps

A case is presented wherein all the four networks are combined in one simulation to test the effectiveness of the developed MPC controller. Table VII gives the percentage packet losses for the open-loop case. Tuning parameters are same as those used by the controller in case of individual networks. Destination buffer is chosen to be 50% of the uncontrolled case and source buffer is chosen to be 9750 bytes. These are optimum values at which all the networks perform reasonably well in terms of minimizing losses and eliminating disruptions.

Figures 30, 31, and 32 depict the open loop plots. In open-loop case no control strategy is implemented. Send rate, arrival rate, destination buffer level, packet loss rate, and cumulative packet losses are shown in Figure 30. Network latency,

Table VII. Percentage Packet Losses for the Varying Network Traffic Conditions with 100 kbps as Source Input Rate.

Network	Percentage Packet Losses
Combined Network	7.07%

destination buffer delay, total playback delay and playback rate are shown in Figure 31. Total playback delay is the sum of network delay and the destination buffer delay. In open-loop case no source buffering is performed but in control led cases buffering at source will be shown. Playback rate is compared to the source input rate as shown in Figure 31.

Figures 33, 34, and 35 depict the closed loop plots. Clearly, from figures 33, and 34 and table VIII, its evident that controller performs well by decreasing packet losses by 39.68% and increasing the playback start time by 2.2%.

Table VIII. Performance of MPC Controller with 100 kbps Source Send Rate under Varying Network Taffic Conditions.

Network	Percentage Change in Packet Losses Compared to Open-Loop	Percentage Change in Playback Start Time Compared to Open-Loop
Varying Network Condition	-39.68%	+2.2%

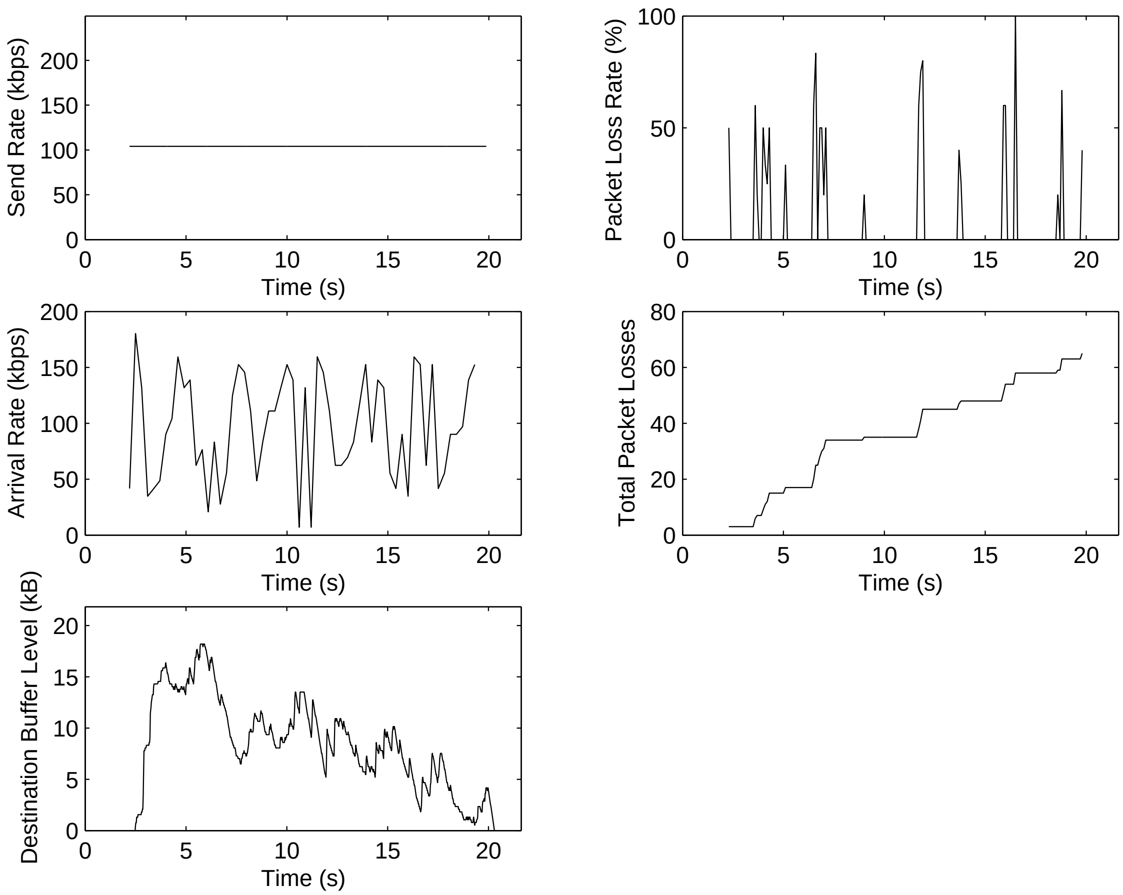

Fig. 30. Buffer Level, Send and Arrival Rates, and Loss Plots for Open-Loop Control with 100 kbps Source Send Rate Using Varying Network Traffic Conditions.

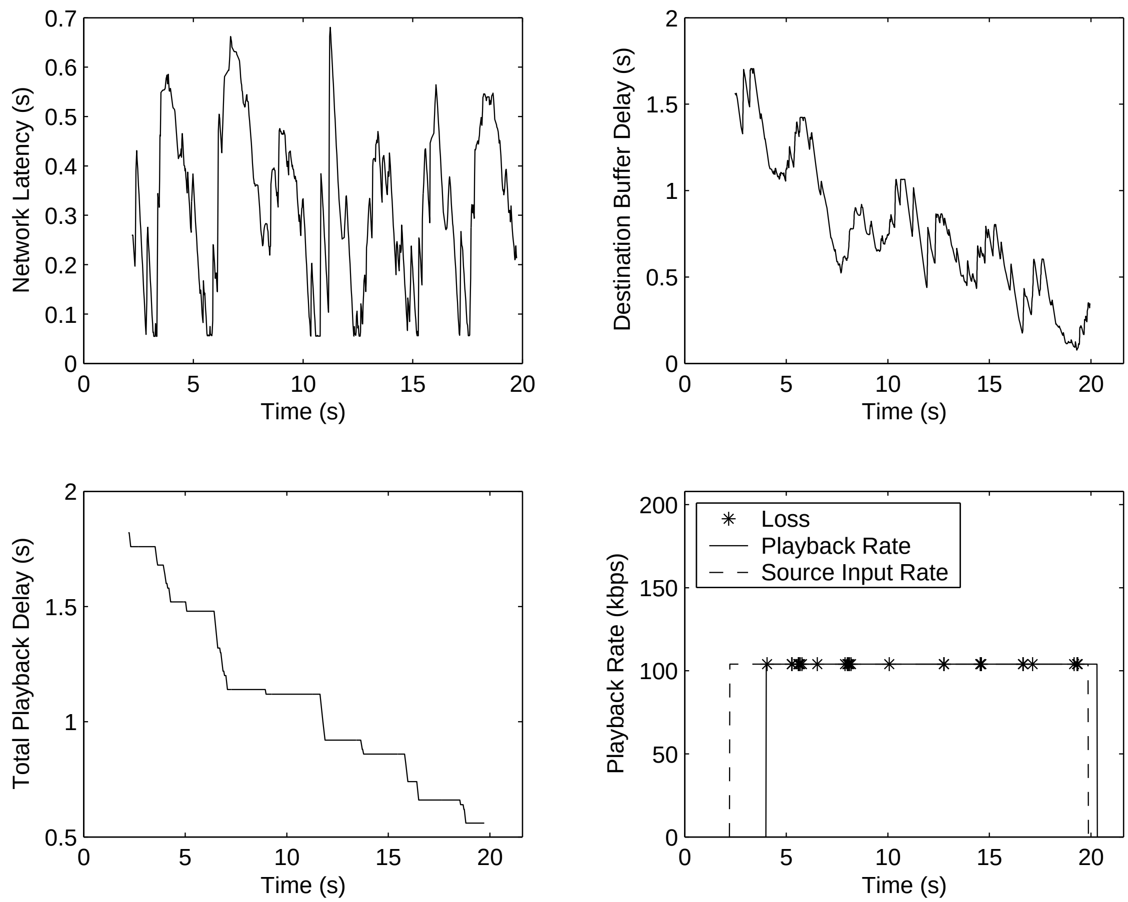

Fig. 31. Delays and Playback Rate Plots for Open-Loop Control with 100 kbps Source Send Rate Using Varying Network Traffic Conditions.

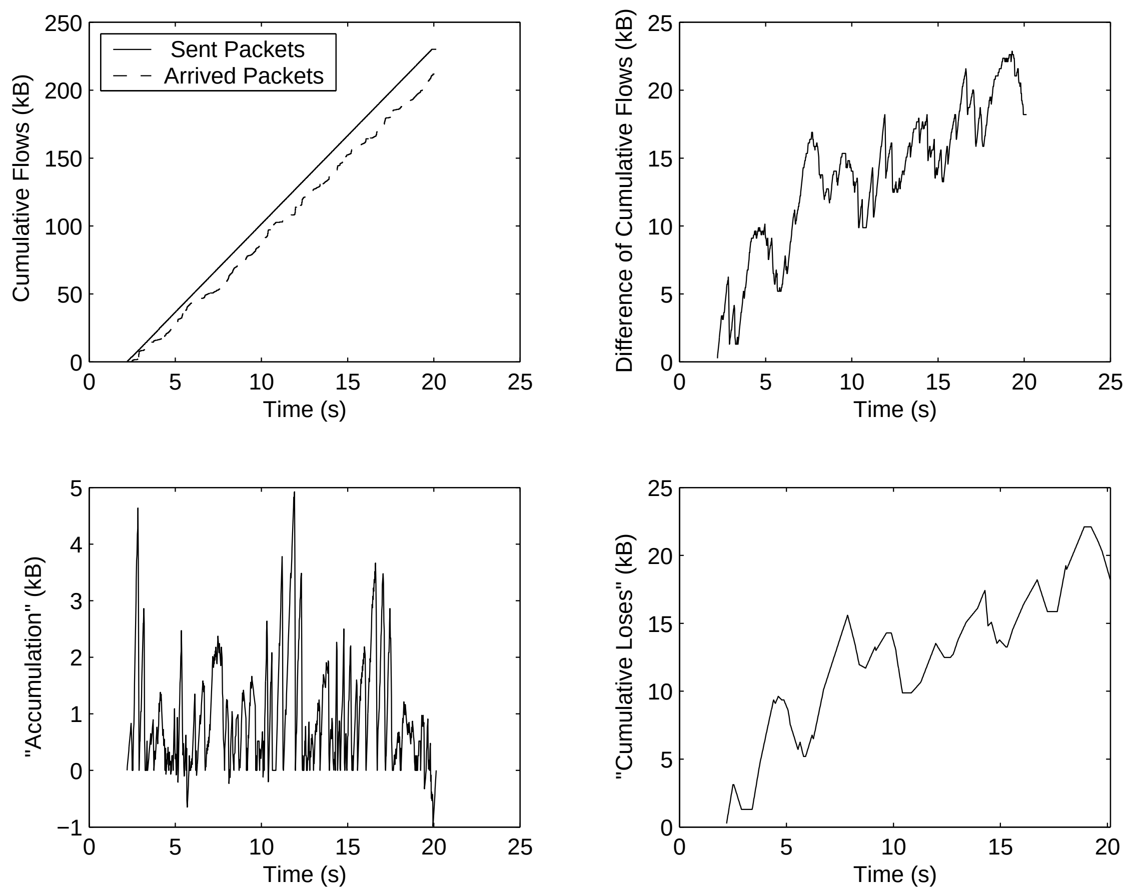

Fig. 32. Cumulative Flows and Accumulation for Open-Loop Control with 100 kbps Source Send Rate Using Varying Network Traffic Conditions.

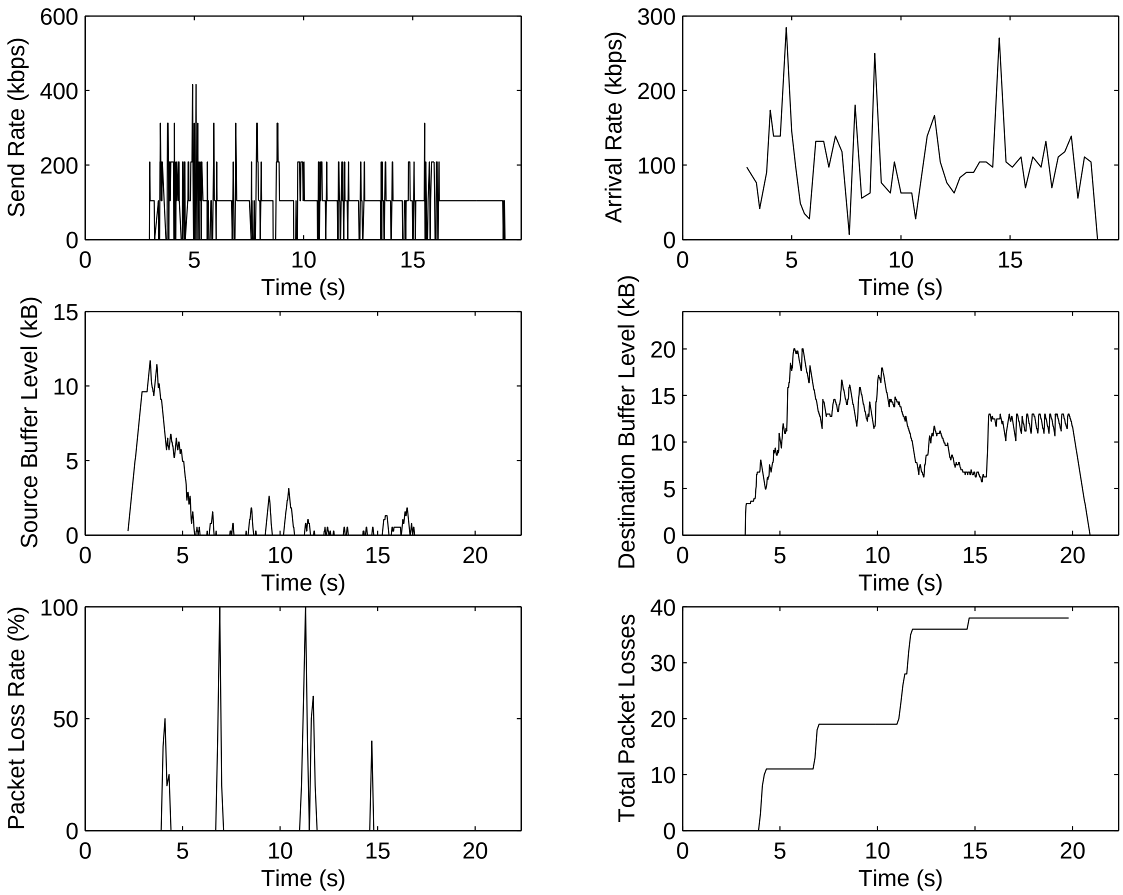

Fig. 33. Buffer Level, Arrival and Send Rates, and Loss Plots for MPC Simulation Using Varying Network Traffic Conditions.

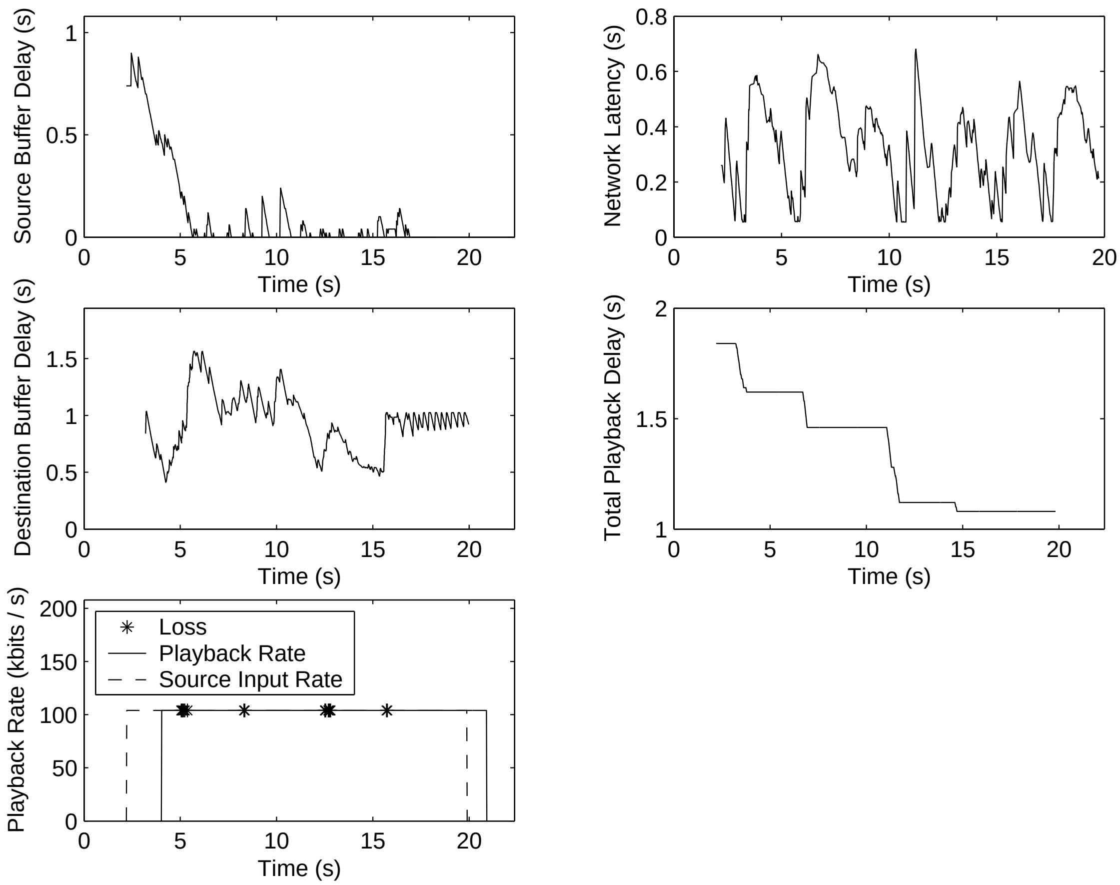

Fig. 34. Delays and Playback Rate Plots for MPC Simulation Using Varying Network Traffic Conditions.

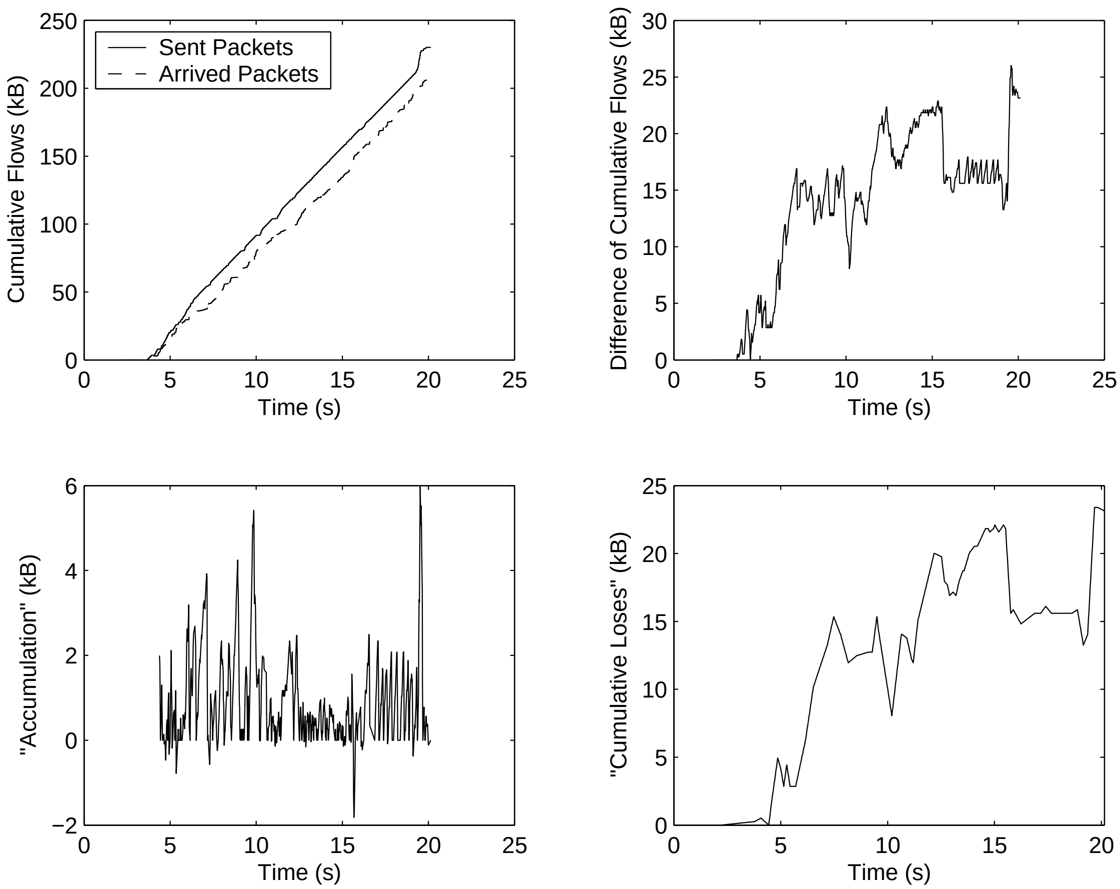

Fig. 35. Cumulative Flows and Accumulation for MPC Simulation Using Varying Network Traffic Conditions.

F. Effectiveness of the Unconstrained MPC on Network 1 Using Packet Accumulation Model with Combined Source Send Rate of 100 and 50 kbps

The effectiveness of the developed controller is evaluated by sending two different bit rates in one simulation using Network 1. The total simulation lasts for about 20 seconds. In the first 8 second of the simulation, source input rate of 100 kbps is send. In the later part of simulation source input rate is reduced to 50 kbps. Table IX gives the percentage packet losses for the open-loop case. Tuning parameters are same as those used by the controller in case of 100 kbps and 50 kbps. Initial destination buffer is chosen to be 50% of the uncontrolled case and initial source buffer is chosen to be 9750 bytes.

Table IX. Percentage Packet Losses for Network 1 with Combined Send Rates of 100 and 50 kbps.

Network	**Percentage Packet Losses**
Network 1	8.84%

Figures 36, 37, and 38 depict the open loop plots. Figures 39, 40, and 41 depict the closed loop plots.

From figures 39, 40, and table X, it is shown that controller performs well by decreasing packet losses by 21.42% and decreasing playback start time by 16.48%

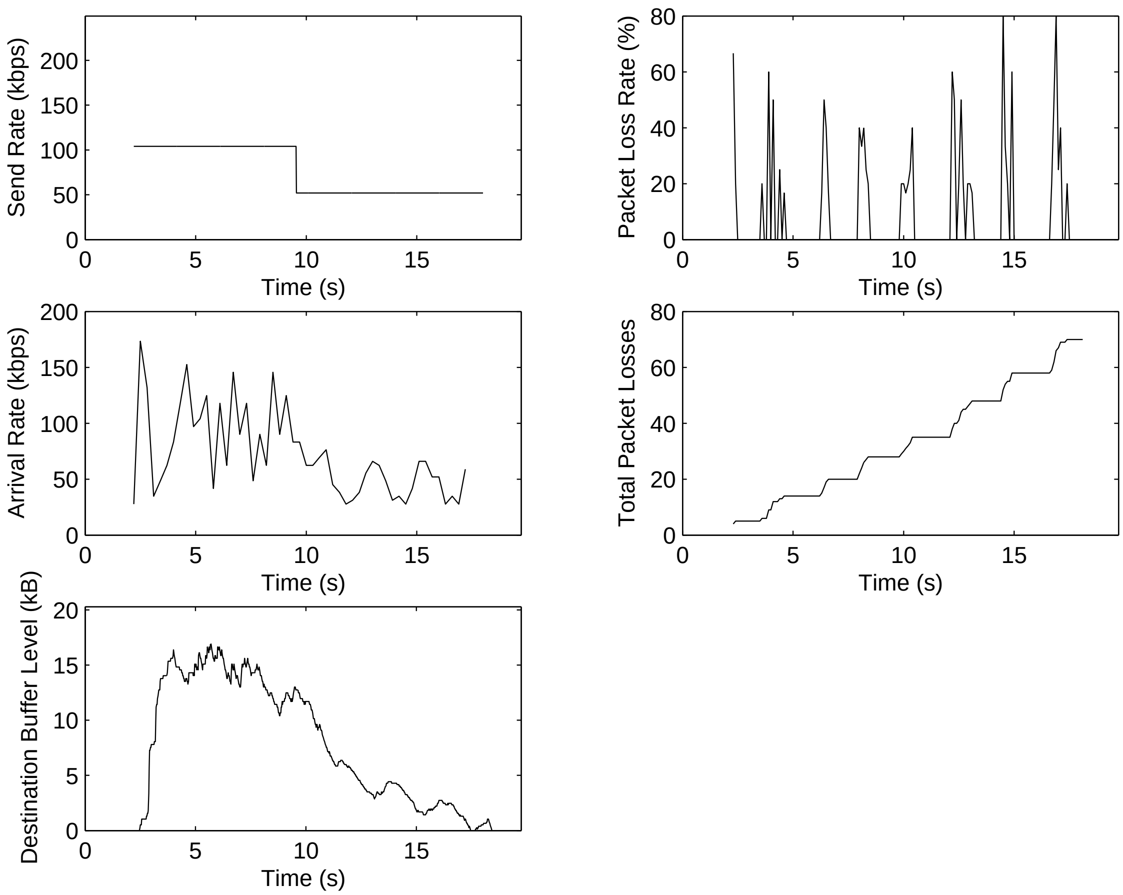

Fig. 36. Buffer Level, Send and Arrival Rates, and Loss Plots for Open-Loop Control with Combined Send Rates Using Network 1 Traffic Conditions.

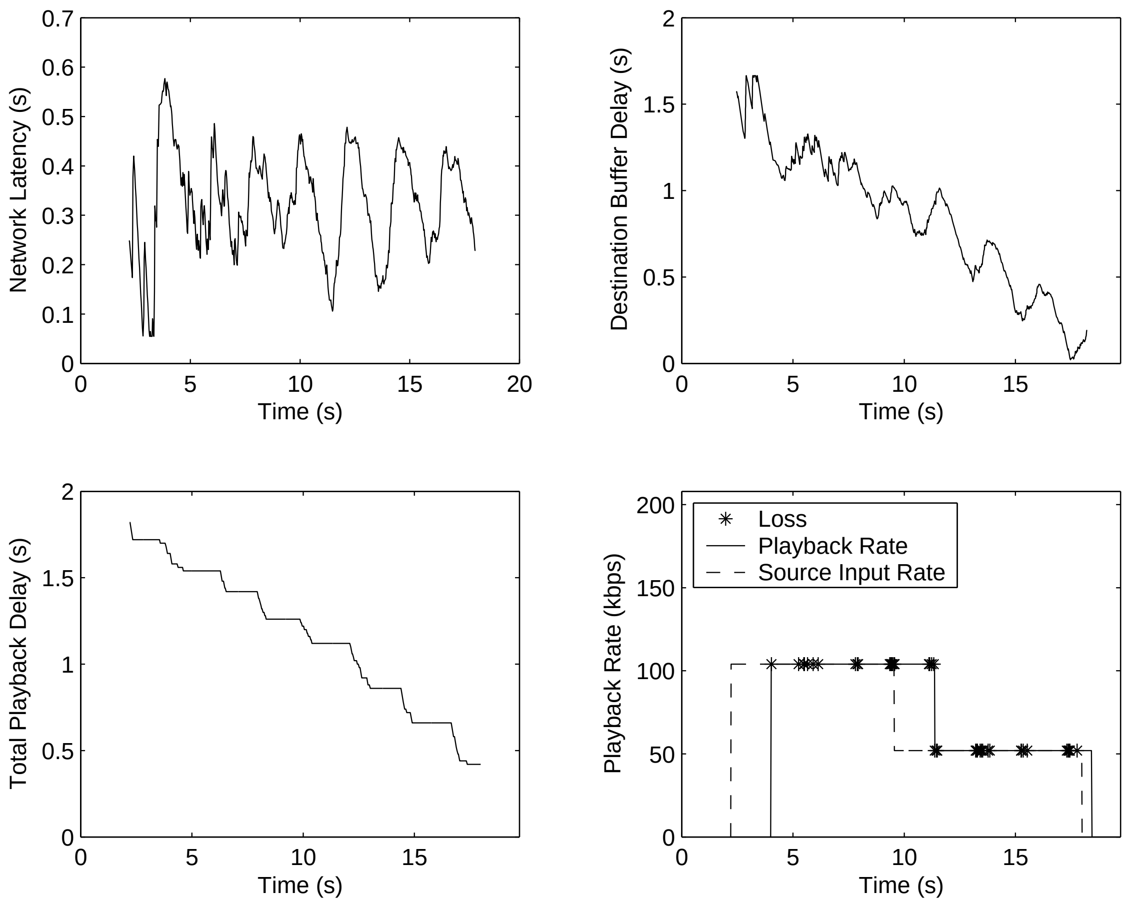

Fig. 37. Delays and Playback Rate Plots for Open-Loop Control with Combined Send Rates Using Network 1 Traffic Conditions.

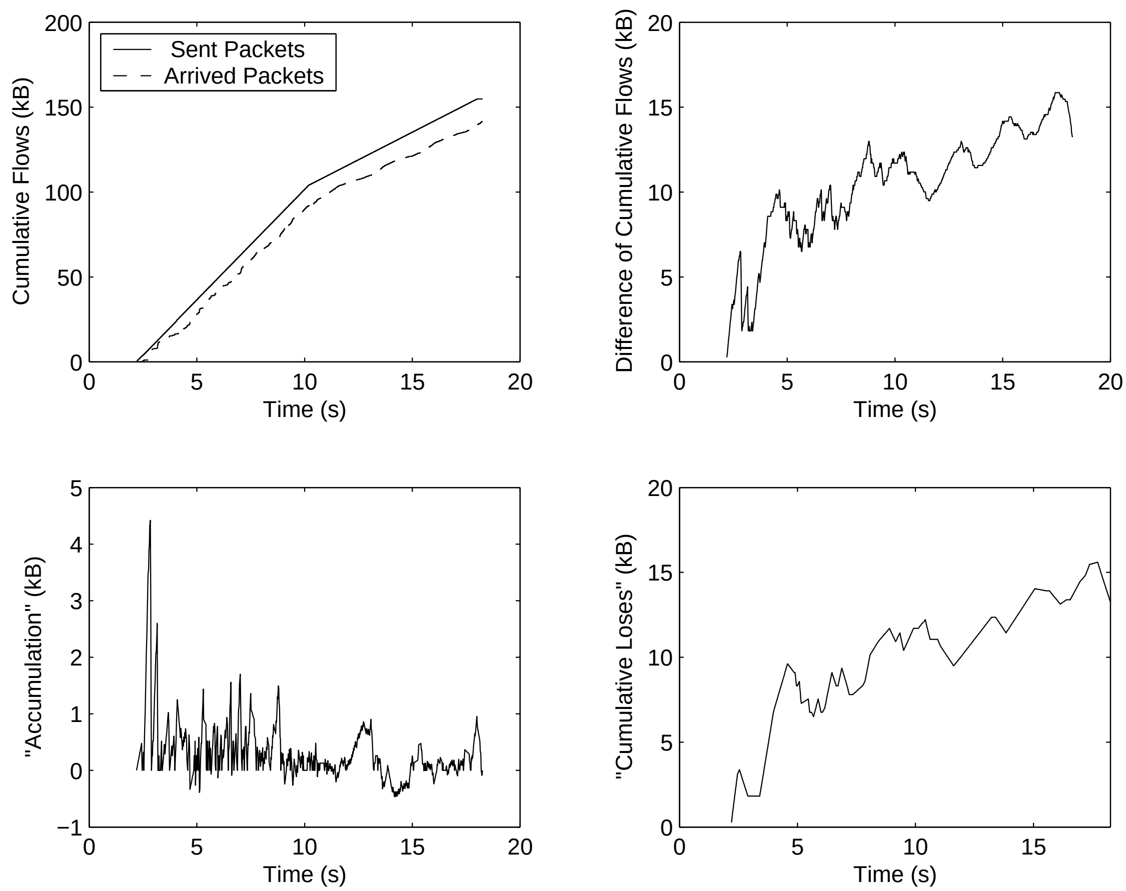

Fig. 38. Cumulative Flows and Accumulation for Open-Loop Control with Combined Send Rates Using Network 1 Traffic Conditions.

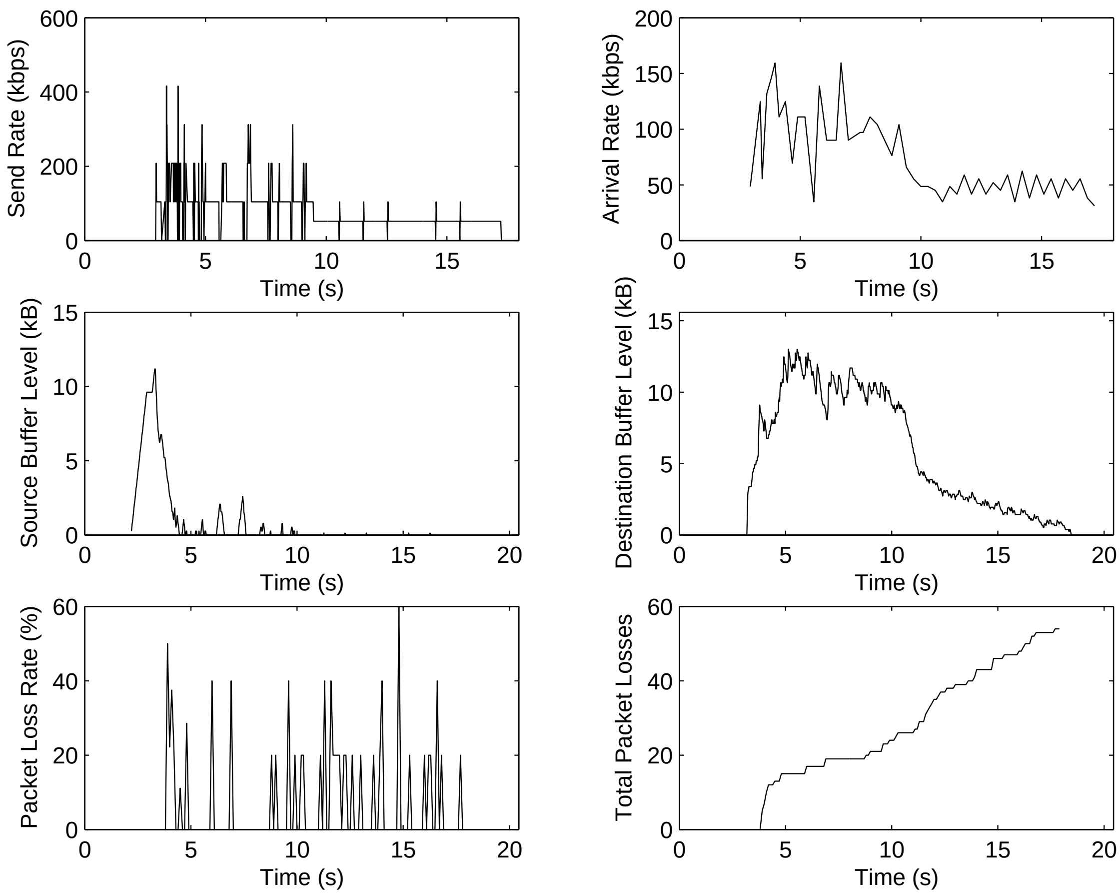

Fig. 39. Buffer Level, Rate, and Loss Plots for MPC Simulation with Combined Send Rates Using Network 1 Traffic Conditions.

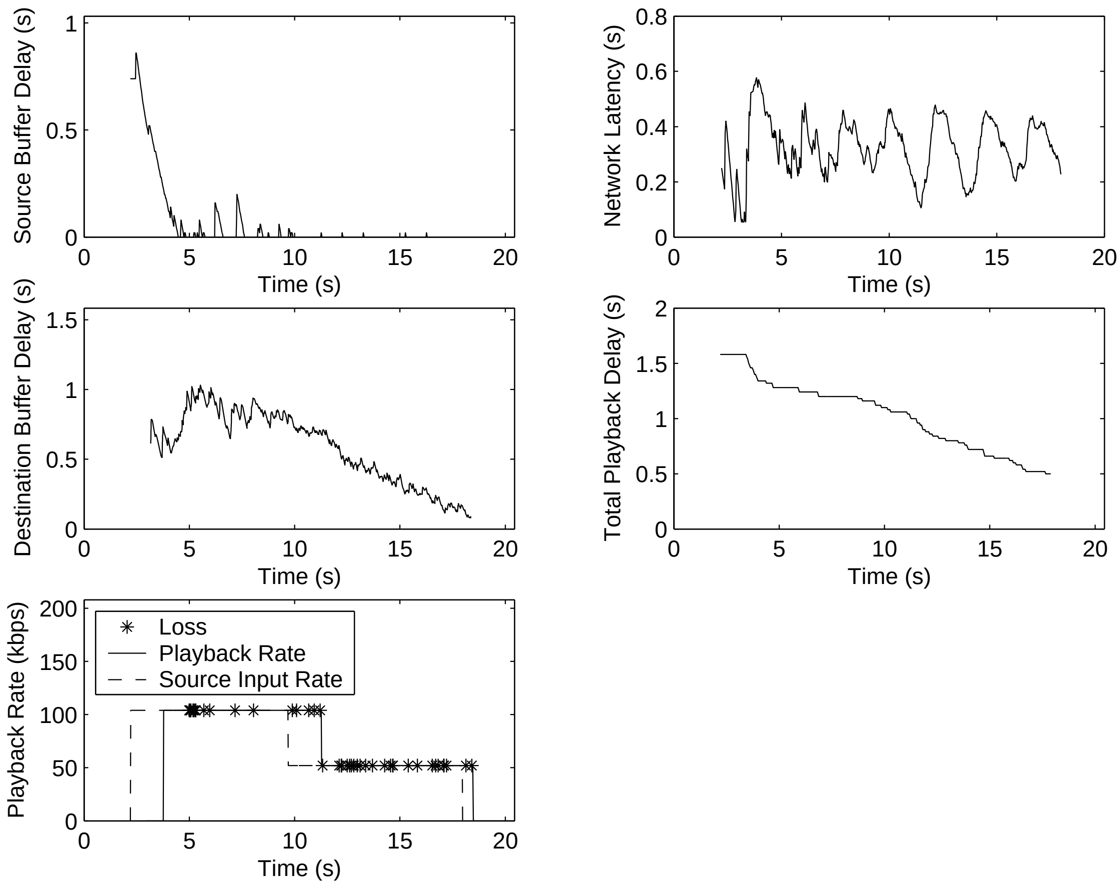

Fig. 40. Delays and Playback Rate Plots for MPC Simulation with Combined Send Rates Using Network 1 Traffic Conditions.

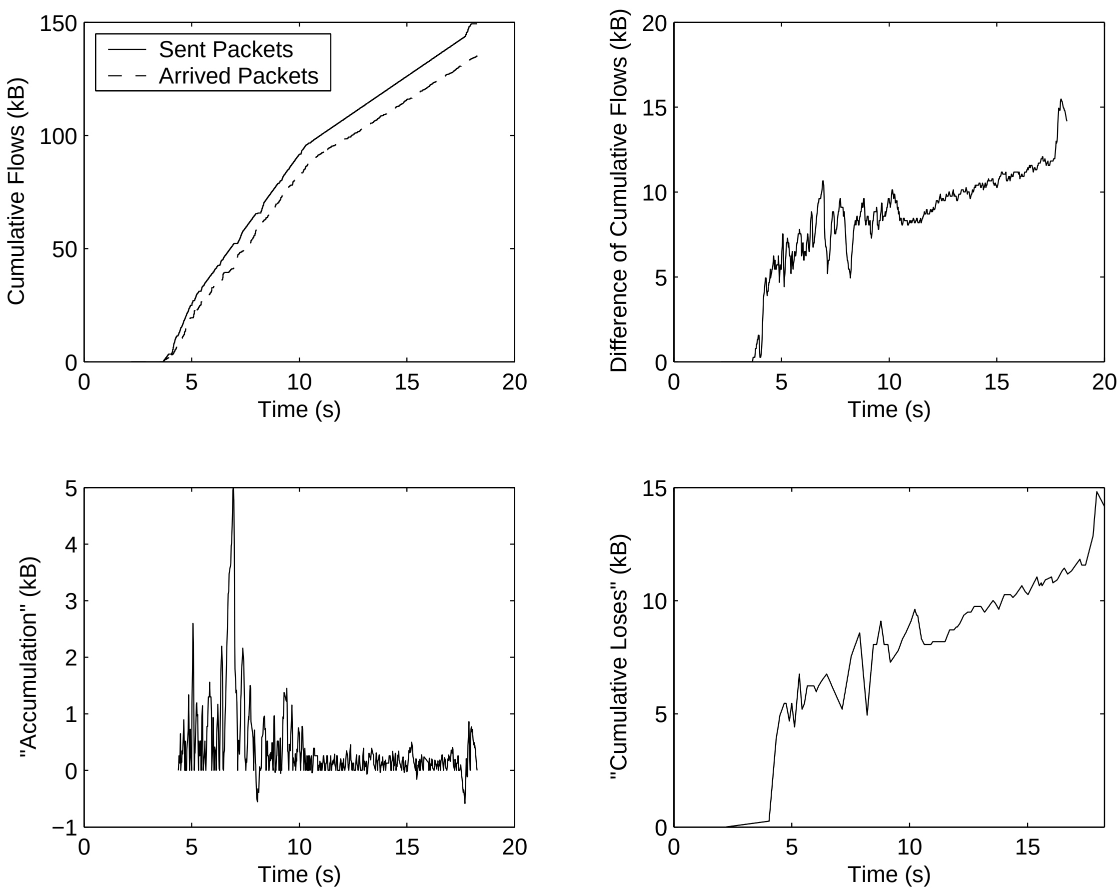

Fig. 41. Cumulative Rates and Accumulation for MPC Simulation with Combined Send Rates Using Network 1 Traffic Conditions.

Table X. Performance of MPC Controller with Combined Bit Rates of 100 and 50 kbps under Network 1 Conditions.

Network	**Percentage Change in Packet Losses Compared to Open-Loop**	**Percentage Change in Playback Start Time Compared to Open-Loop**
Network 1	-21.42%	-16.48%

CHAPTER V

SUMMARY AND CONCLUSIONS

The objective of this research is to improve the QoS for real-time media applications over best-effort networks. This is achieved by considering controller performance criteria and implementing Unconstrained MPC control schemes at the application level.

A. Summary

The problem definition and objectives are stated in Chapter I. Relevant literature related to computer networks and application of control theory as applied to computer science are reviewed as in introduction to this area of research.

Various end-to-end network measurements that could be used as feedback signals are discussed in Chapter II. QoS metrics along with major assumptions of this thesis are stated in this chapter. Several control methods along with desired controller performance criteria are mentioned in Chapter III. Unconstrained MPC is developed. The reasons for choosing predictive methods over reactive methods for development of the controller are as follows:

- Best effort networks have time-varying time delays. Its well known fact that controller developed based on reactive schemes fails to perform well in case systems having time varying time delays.

- Best effort networks cannot be modelled based on first principles. Hence, empirical modeling based on input-output measurements is a natural choice

- Constraints placed by the decoder on the source buffer can be handled in case of MPC schemes.

Developed control method is evaluated in Chapter IV using MATLAB and ns-2. QoS guarantees are met by reducing packet losses, eliminating disruptions and with either increase or decrease in playback start time. Special cases are also considered in Chapter IV, including combination of different network congestion levels, and effect of different bit rates in same simulation. The main premise of the predictive controller is that when accumulation and congestion levels are high, the send rate is reduced to prevent losses and alleviate network congestion.

B. Conclusions

Some conclusions that can be made based on the results of this work are:

- Unconstrained Model Predictive Control strategies can improve application level QoS of real-time media applications over best-effort networks compared to the open-loop case.

- QoS can be improved by minimizing packet losses. A trade off exists, as decrease in packet losses is accomplished by increase in playback start time. This is due to the buffering at the the source. Decrease in playback start time is achieved in cases where the congestion levels are high.

- Controller works well when losses are varied from 2% to 8% In cases where no packet losses or extreme packet losses occurs, feedback control has little flexibility in minimizing losses.

- A major conclusion of this work is that once a robust predictor is developed, packet losses can be minimized based on tuning of source buffer. This reduces the degree of freedom of the controller to one. Adding destination buffer eliminates disruptions and increases the tuning parameters to two.

- Packet accumulation captures the congestion level of the network. Packet accumulation when used as feedback signal in MPC, works well in meeting controllers desired objectives.

C. Recommendations for Future Work

Some recommendations for future research include:

- Effectiveness of modeling and implementation of MPC with actual network measurements from the Internet should be considered.
- Implementaion of MPC with constraints should be considered. Incorporation of constraints for send rate and buffers should be considered.
- Implementation of MPC with multiple output signals should be considered. Feasibility of modeling different loss signals, such as loss rate and cumulative packet losses should be considered.
- Based of preliminary results, lookup table for tuning of the controller should be considered. Based on different send rates and network loss levels, look up table for source buffer can be considered.
- Ns-2 support for end-to-end analysis and control should be considered. Better interface to call ns-2 as a function from Matlab can be developed. Use of OPNET for simulations and controls should also be considered.

REFERENCES

[1] Y. Xia, D. Harrison, S. Kalyanaraman, K. Ramachandran, and A. Venkatesan, "An accumulation-based congestion control model," *IEEE International Conference on Communications*, vol. 1, pp. 657–663, May 2003.

[2] J.B Postel, "Transmission control protocol. DARPA Internet program protocol specification," [Online]. Available: http://www.freesoft.org/CIE/RFC/793/, Accessed on December 2001.

[3] S. Floyd and K. Fall, "Promoting the use of end-to-end congestion control in the Internet," *IEEE/ACM Transactions on Networking*, vol. 7, no. 4, pp. 458–472, August 1999.

[4] S. Floyd, K. Fall, J. Padhye, and J. Widmer, "Equation-based congestion control for unicast applications," in *ACM SIGCOMM '00*, Stockholm, Sweden, August 2000, pp. 43–56.

[5] D. Bansal and H. Balakrishnan, "TCP-friendly congestion control for real-time streaming applications," [Online]. Available: http://nms.lcs.mit.edu/papers/cm-binomial.html, Accessed on December 2001.

[6] D. Dwyer, S. Ha, J-R. Li, and V. Bhargavan, "An adaptive transport protocol for multimedia communication," in *Proceedings of IEEE Multimedia Computing and Systems*, June 1998, pp. 23–32.

[7] T. Turletti. and C. Huitema, "Videoconferencing on the Internet," *IEEE Transactions on Networking*, vol. 4, no. 3, pp. 340–350, June 1996.

[8] J. Bolot and T. Turletti, "A rate control mechanism for packet video in the Internet," in *Proceedings of the Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, Boston, Massachusetts, April 1994, vol. 3, pp. 1216–1223.

[9] B. Smith, "Cyclic-udp: A priority-driven best-effort protocol," [Online]. Available: http://www.cs.cornell.edu/Info/Faculty/Brian_Smith.html/Publications, Accessed on December 2001.

[10] S. C. Liew and D. C. Tse, "A control-theoretic approach to adapting VBR compressed video for transport over a CBR communications channel," *IEEE Transactions on Networking*, vol. 6, no. 1, pp. 42–55, February 1998.

[11] A. Kolarov and G. Ramamurthy, "A control-theoretic approach to the design of an explicit rate controller for ABR service," *IEEE/ACM Transactions on Networking*, vol. 7, no. 5, pp. 741–753, October 1999.

[12] S. Mascolo, D. Cavendish, and M. Garla, "ATM rate based congestion control using a Smith predictor: An EPRCA implementation," in *Proceedings of the Fifteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, San Francisco, California, 1996, vol. 2, pp. 569–576.

[13] L. BenMohamed and Y.T. Wang, "A control-theoretic ABR explicit rate algorithm for ATM switches with Per-VC queuing," in *Proceedings of the Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies*, San Francisco, California, 1998, vol. 1, pp. 183–191.

[14] Y.S. Sun, F. Tsou, M. Chang Chen, and Z. Tsai, "A TCP-friendly congestion control scheme for real-time packet video using prediction," in *Proceedings of the*

1999 Global Telecommunications Conference, Rio de Janeiro, Brazil, December 1999, vol. 3, pp. 1818–1822.

[15] D. Wu, Y. T. Hou, W. Zhu, Y. Zhang, and J. M. Peha, "Streaming video over the Internet: Approaches and directions," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 282–300, March 2001.

[16] D. Wu, Y. T. Hou, W. Zhu, H. Lee, T. Chiang, Y. Zhang, and H. J. Chao, "On end-to-end architecture for transporting MPEG-4 video over the Internet," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 6, pp. 923–941, September 2000.

[17] D. Wu, Y. T. Hou, and Y. Zhang, "Transporting real-time video over the Internet: Challenges and approaches," in *Proceedings of the IEEE*, December 2000, vol. 88, pp. 1855–1875.

[18] H. Ohsaki, M. Morita, and M. Murata, "On modeling round-trip dynamics of the Internet using system identification," *IEICE Transactions on Communication*, vol. E85-B, no. 1, pp. 1–9, January 2002.

[19] S. Doddi, "Emperical modeling of end-to-end delay dynamics in best-effort networks," M.S. thesis, Texas A&M University, College Station, Texas, 2004.

[20] I. Yeom, "ENDE: An end-to-end network delay emulator," M.S. thesis, Texas A&M University, College Station, Texas, 1998.

[21] S. Parekh, N. Gandhi, J. Hellerstein, D. Tilbury, T. Jayram, and J. Bigus, "Using control theory to achieve service level objectives in performance management," in *Proceedings of the 2001 IEEE/IFIP International Symposium on Integrated Network Management Proceedings*, May 2001, pp. 841–854.

[22] N. Gandhi, D. M. Tilbury, S. Parekh, and J. Hellerstein, "Feedback control of a lotus notes server: Modeling and control design," in *Proceedings of the 2001 American Control Conference*, Arlington, Virginia, June 2001, vol. 5, pp. 3000–3005.

[23] T. F. Abdelzaher and C. Lu, "Modeling and performance control of Internet servers," in *Proceedings of the 39th IEEE Conference on Decision and Control*, Sydney, Australia, December 2000, vol. 3, pp. 2234–2239.

[24] J. Aweya, D. Y. Montuno, Q. Zhang, and L. Orozco-Barbosa, "Congestion control using a multi-step neural predictive technique," in *Proceedings of the 1998 IEEE Global Telecommunications Conference*, Sydney, Australia, November 1998, vol. 3, pp. 1705–1714.

[25] Y. Gu, H. O. Wang, Y. Hong, and L. G. Bushnell, "A predictive congestion control algorithm for high speed communications networks," in *Proceedings of the 2001 American Control Conference*, Arlington, Virginia, June 2001, vol. 5, pp. 3779–3780.

[26] D. McNamee, C. Krasic, K. Li, A. Goel, E. Walthinsen, D. Steere, and J. Walpole, "Control challenges in multi-level adaptive video streaming," in *Proceedings of the 39th IEEE Conference on Decision and Control*, Sydney, Australia, December 2000, vol. 3, pp. 2228–2233.

[27] B. Li, D. Xu, and K. Nahrstedt, "Optimal state prediction for feedback-based QoS adaptations," in *Seventh International Workshop on IWQoS*, London, England, June 1999, pp. 37–46.

[28] M. Morita, H. Ohsaki, and M. Murata, "Designing a delay-based adaptive congestion control mechanism using control theory and system identification for

tcp/ip networks," *in Proceedings of SPIE's International Symposium on the Convergence of Information Technologies and Communications (ITCom 2002)*, pp. 132–43, 2002.

[29] A. P. Black, J. Huang, R. Koster, J. Walpole, and C. Pu, "Infopipes: An abstraction for multimedia streaming," Technical Report Number CSE 02-001, Department of Computer Science and Engineering, OGI School of Science and Engineering, OR, USA, January 2002.

[30] H. Gorecki, S. Fuksa, P. Grabowski, and A. Korytowski, *Analysis and Synthesis of Time Delay Systems*, John Wiley & Sons, New York, 1989.

[31] M. Malek-Zavarei and M. Jamshidi, *Time-Delay Systems: Analysis, Optimization, and Applications*, Elsevier Science Pub. Co., New York, 1987.

[32] J. E. Marshall, *Time-Delay Systems: Stability and Performance Criteria with Applications*, Ellis Horwood, New York, 1992.

[33] S. Niculescu, M. Fu, and H. Li, "Delay-dependent closed-loop stability of linear systems with input delay: An LMI approach," in *Proceedings of the 36th Conference on Decision and Control*, San Diego, California, December 1997, vol. 2, pp. 1623–1628.

[34] M. T. Nihtilä, "Finite pole assignment for systems with time-varying input delays," in *Proceedings of the 30th Conference on Decision and Control*, Brighton, England, December 1991, vol. 1, pp. 927–928.

[35] E. Poulin and A. Pomerleau, "PI settings for integrating processes based on ultimate cycle information," *IEEE Transactions on Control Systems Technology*, vol. 7, no. 4, pp. 509–511, July 1999.

[36] L. Wang and W. R. Cluett, "Tuning PID controllers for integrating processes," *IEE Process Control Theory Applications*, vol. 144, no. 5, pp. 385–392, September 1997.

[37] E. P. and A. Pomerleau, "PID tuning for integrating and unstable processes," *IEE Process Control Theory Applications*, vol. 143, no. 5, pp. 429–435, September 1996.

[38] S. Majhi and D. P. Atherton, "Modified Smith predictor and controller for processes with time delay," *IEE Process Control Theory Applications*, vol. 146, no. 5, pp. 359–366, September 1999.

[39] P. Quet, B. Ataslar, A. Iftar, H. Ozbay, S. Kalyanaraman, and T. Kang, "Rate-based flow controllers for communication networks in the presence of uncertain time-varying multiple time-delays," *Automatica*, vol. 38, no. 6, pp. 917–928, March 2002.

[40] G. A. Dumont, A. Elnaggar, and A. Elshafei, "Adaptive predictive control of systems with time-varying time delay," *International Journal of Adaptive Control and Signal Processing*, vol. 7, pp. 91–101, 1993.

[41] J. W. Mangan III, "Adaptive control of media applications in best effort networks using reliable transport protocols," M.S. thesis, Texas A&M University, College Station, Texas, 2002.

[42] P. Vettori and S. Zampieri, "Controllability of delay-differential systems in the behavioral approach," in *Proceedings of the 38th Conference on Decision and Control*, Phoenix, Arizona, December 1999, vol. 1, pp. 233–238.

[43] J. Klamka, "Controllability of semilinear systems with delays in control," in

Proceedings of the 1999 IEEE Conference on Systems, Man, and Cybernetics, Tokyo, Japan, October 1999, vol. 1, pp. 89–94.

[44] M.M. Ekanayake, K. Premaratne, C. Douligeris, and P.H. Bauer, "Stability of discrete-time systems with time-varying delays," in *Proceedings of the 2001 American Control Conference*, Arlington, Virginia, June 2001, vol. 5, pp. 3914–3919.

[45] J. Wei, S. Wenzhong, F. Shumin, and S. Shiji, "On the controllability and the stabilizability of control delay systems," in *Proceedings of the Third World Congress on Intelligent Control and Automation*, Hefei, People's Republic of China, June 2000, vol. 4, pp. 2846–2849.

[46] "Network Simulator 2 (ns-2)," Information Science Institute, University of Southern California, Los Angeles, 2002.

[47] James F. Kurose and Keith W. Ross, *Computer Networking A Top-Down Approach Featuring the Internet*, Addison-Wesley, New York, 2001.

VITA

Vivek Khariwal was born in Bangalore, Karnataka, India on the 6th of August, 1978. He received his B.E. degree in mechanical engineering from Bangalore University, Bangalore, India, in 2000. In August 2001, he joined the master's degree program in mechanical engineering at Texas A&M University, and received his M.S. degree in August of 2004.

Permanent Address:

248, III Main Road,

Chamrajpet,

Bangalore, Karnataka, India 560018.

Contact phone: 91-80-26602506

Contact email address: khariwal@tamu.edu

The typist for this thesis was Vivek Khariwal.