Resource Allocation and Congestion Control Strategies for Networked Unmanned Systems

Kamal Bouyoucef

A Thesis
in
The Department
of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy at
Concordia University
Montréal, Québec, Canada

October 2008

Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-63453-0
Our file Notre référence
ISBN: 978-0-494-63453-0

ABSTRACT

Resource Allocation and Congestion Control Strategies for Networked Unmanned Systems

Kamal Bouyoucef, Ph.D.
Concordia Unviersity, 2008

It is generally agreed that communication is a critical technological factor in designing networked unmanned systems (NUS) that consist of a large number of heterogeneous assets/nodes that may be configured in ad-hoc fashion and that incorporate intricate architectures. In order to successfully carry out the NUS missions, communication among assets need to be accomplished efficiently.

In contrast with conventional networks, NUSs have specific features that may render communication more complex. The main distinct characteristics of NUS are as follows: (a) heterogeneity of assets in terms of resources, (b) multiple topologies that can be fully-connected, (c) real-time requirements imposed by delivery timeliness of messages under evolving and uncertain environments, (d) unknown and random time-delays that may degrade the closed-loop dynamics performance, (e) bandwidth constraints reflecting differences in assets behavior and dynamics, and (f) protocol limitations for complying with the wireless features of these networks.

The NUS system consists of clusters each having three nodes, namely, a sensor, a decision-maker, and an actuator. Inspired by networked control systems (NCS), we introduced a generic framework for NUSs. Using the fluid flow model (FFM), the overall dynamical model of our network cluster is derived as a time-delay dependent system.

The following three main issues are investigated in this thesis, bandwidth allocation, an integrated bandwidth allocation and flow rate control, and congestion control.

To demonstrate the difficulty of addressing the bandwidth allocation control problem, a standard PID is implemented for our network cluster. It is shown that in presence of feedback loops and time-delays in the network, this controller induces flow oscillations and consequently, in the worst-case scenario, network instability. To address this problem, nonlinear control strategies are proposed instead. These strategies are evaluated subject to presence of unknown delays and measurable/estimated input traffic. For different network configurations, the error dynamics of the entire controlled cluster is derived and sufficient stability conditions are obtained. In addition, our proposed bandwidth allocation control strategy is evaluated when the NUS assets are assumed to be mobile. The bandwidth allocation problem is often studied in an integrated fashion with the flow rate control and the connection admission control (CAC). In fact, due to importance of interaction of various components, design of the entire control system is often more promising than optimization of individual components. In this thesis, several robust integrated bandwidth allocation and flow rate control strategies are proposed.

The third issue that is investigated in this thesis is the congestion control for differentiated-services (DiffServ) networks. In our proposed congestion control strategies, the buffer queue length is used as a feedback information to control locally the queue length of each buffer by acting on the server bandwidth and simultaneously a feedback signaling notifies the ordinary sources regarding the allowed maximum rate. Using sliding mode generalized variable structure control techniques (SM-GVSC), two congestion control approaches are proposed, namely, the non degenerate and degenerate GVS control approaches. By adopting decentralized end-to-end, semi-decentralized end-to-end, and distributed hop-by-hop control approaches, our proposed congestion control strategies are investigated for a DiffServ loopless

mesh network (Internet) and a DiffServ fully-connected NUS. Contrary to the semi-decentralized end-to-end congestion control strategy, in the distributed hop-by-hop congestion control strategy, each output port controller communicates the maximum allowed flow rate only to its immediate upstream node(s) and/or source(s). This approach reduces the required amount of information in the flow control when compared to other approaches in which the allowed flow rate is sent to all the upstream sources communicating through an output port.

DEDICATION

This thesis is dedicated to my parents, Hocine and Tassadit, who waited so long for this.

My deepest dedication goes to my wife Houria, who did more than her share around the house as I was always with my books.

I dedicate this thesis to my kids Ghiles and Mazigh, who needed more than what a dad student could give.

My thanks also go to my brothers and sisters as well as all my family members.

ACKNOWLEDGEMENTS

I would be delighted to thank the many people who have made this thesis possible. I owe my most sincere gratitude to my supervisor Professor K. Khorasani, Department of Electrical and Computer Engineering at Concordia University, for his untiring guidance during my Ph.D. program, his patience, his motivation to tackling several issues, his consistence and his rigor in analyzing problems and presenting solutions, as well as his support.

I deeply wish to thank Professor S. Alkass, Department of Mechanical Engineering at Concordia University, Professor Ahmed K. Elhakeem, Department of Electrical and Computer Engineering, at Concordia University, and Professor S. Hashtrudi Zad, Department of Electrical and Computer Engineering at Concordia University, for serving as internal committee members, for their constructive criticism and unselfish comments. I would also like to acknowledge Professor Brandon W. Gordon, Department of Mechanical Engineering, at Concordia University and Professor A. Aghdam, Department of Electrical and Computer Engineering, at Concordia University, both presently in sabbatical leave for their advises in my proposal and seminar presentations.

I am indebted to Professor Luc Baron of Departement de Genie Mecanique at Ecole Polytechnique de Montreal, for serving as an external examiner.

My thanks go to all my colleagues and friends particularly those from System and Control Laboratory for their enjoyable and fruitful discussions.

Last but not least, my thanks are also extended to all the other faculty members and Staff in the Department of Electrical and Computer Engineering for assistance in fulfillment of this degree requirement.

TABLE OF CONTENTS

List of Figures

List of Tables

List of Symbols and Abbreviations

N-CUS	Network-Centric Unmanned Systems.
NCS	Network Control Systems.
FFM	Fluid Flow Model.
SM-GVSC	Sliding Mode-Generalized Variable Structure Control.
GOCF	Generalized Observable Canonical Forms.
IDCC	Integrated Dynamical Congestion Controller.
E2E	End-To-End.
HBH	Hop-By-Hop.
ATM	Asynchronous Transfer Mode.
TCP/IP	Transport Control Protocol/Internet Protocol.
CAC	Connection Admission Control.
ABR	Available Bit Rate.
QoS	Quality of Service.
AQM	Active Queue Management.
$x_{l_h f}$	Queue length in the buffer f of the output port l_h.
$e_{l_h f}$	Error between the actual queue state and its reference.
$\mathbb{F}_f^c, E_f^c$	State and error buffer occupancies, respectively.
C_{l_h}, C_{serv}	Capacity of the queue server $l_h f$ and its maximum, respectively.
λ_{il_h}, λ_{ol_h}	The arriving and the outgoing customers rate at the buffer $l_h f$, respectively.
$\hat{\lambda}_{il_h}$	Estimate of the arriving customers rate and the estimation error, respectively.
η	State variable incorporating the state and the estimation error of the traffic.
$M_{l_j f}$	Number of immediate upstream senders.
$F_{lk,f}^{hj}$	Process function.

$S^{hj}_{lk,f}$	Switch.
$\tau^{hj}_{lk,f}$	Block delay.
φ	Upper bound of the derivative of the time delay $\tau^{hj}_{lk,f}$.
μ, γ	The proposed congestion controller design parameters.
$sgn(e)$	Sign function.
Γ, σ	The proposed estimator design parameters.
ϕ^c_f	The matrix incorporating switching and process function quantities in the overall cluster model.
ν	Positive scalar utilized in the performance index.
Ξ	Vector of the state and the input variables utilized in the Hamiltonian function.
$f = p, r$	Indices corresponding to the Premium and the Ordinary services, respectively.
$\bar{\lambda}^U_{i,f}$	Source traffic input utilized in the functional differential equation.
$A, \mathbb{A}, B, \mathbb{B}$	Matrices utilized in the cluster error dynamics.
$\mathbb{C}, P, Q, R_0$	Matrices utilized in the Lyapunov Krasovskii functional candidates.
$\mathbb{L}_i$	Linear matrix inequalities.

Chapter 1

Introduction

In order to reduce the loss of human life in hazardous and hostile environments, scientists have always thought and predicted new generations of machines that are able to accomplish such perilous expeditions. Besides, after sixty years, it appears that Clarence "Kelly" Johnson's foresight is coming to fruition. Indeed, in 1944, the legendary founder of Lockheed's Skunk Works and designer of the SR-71 and U-2 aircraft, predicted that the future of military aviation would belong to Uninhabited or Unmanned Aerial Vehicles (UAVs). Even though the aerial vehicles have more dynamic degrees of freedom in comparison to ground vehicles, these vehicles do belong to a general class of Unmanned Systems (USs). The Unmanned Systems (USs) might be defined as a mechatronic system with no human operator aboard and which using increased autonomy operate in an intelligent manner using active or passive seekers or guidance and intelligent agents to identify and accomplish complex tasks. Unmanned vehicles have been fielded in several domains in the recent past, ranging from battlefields to Mars. Most major efforts have been funded by various U. S. Government agencies. While military applications are the predominant unmanned systems opportunity, civilian and commercial applications will become an increasingly important aspect of this business. Such applications will likely include telecommunications, weather reconnaissance, border patrol, civil emergency support and Earth science to name just a few. For example, in the Earth science missions, we might name applications that would allow measurement of the geophysical processes associated with natural hazards such as earthquakes, landslide, and volcanoes as they are manifested by deformations in the Earth's crust. Measurements of the crustal deformation would be made by an interferometric synthetic aperture radar (SAR) carried by the UAV platform. An other mission is the one that supports measurements of the dynamics of the breakup of polar glacier and polar ice sheets. The measurements enable direct observation of the evolution in time of ice and land topography, iceberg volume, glacier profiles, and glacier channel profiles and provide data for validating simulations of these dynamics and their interaction with the ocean environment.

Without loss of generality, mobile or stationary, the USs include different categories such as unmanned aerial vehicles (UAV), unmanned ground vehicles (UGV), unmanned underwater vehicles (UUV), unmanned surface vehicles (USV), unattended munitions (UM), and unattended ground sensors (UGS). Missiles, rockets, and their sub munitions, and artillery are not considered unmanned systems. In fact, the introduction of jungle or urban environments necessitate the consideration of vertical aspects for Unmanned Air Vehicles (UAV) and introduction of Unmanned Ground Vehicles (UGV) while riverine or littoral operations suggest the use of Unmanned Surface Vehicles (USV).

The unmanned systems domain is not anymore a fiction, but rather has already entered a practical era and besides unmanned systems industry is very flourishing. Several military operational UAVs, such as Predator, Hunter, and Shadow; and developmental UAVs, such as Global Hawk, have already demonstrated their significant capabilities during the recent military operations, and a multitude of unmanned vehicles have been already realized and put on the market by different vendors. Among them, we may name a few such as the Tactical Aerospace Group (TAG), the Auxilia and C2hub, the Joint Architecture for Unmanned Systems JAUS Working Group, the Northrop Grumman Corporation and Boeing.

It is widely accepted that communication and navigation constitute a central issue for unmanned systems. The challenge is to provide energy-efficient, low-latency, sufficient bandwidth communications and GPS-based localization and navigation. The radio transmissions should have low probabilities of detection and intercept as well as GPS transmissions that should be properly encrypted to prevent interception or alteration.

1.1 Problem Statement

The future envisaged network of unmanned systems consists of a number of heterogeneous nodes or assets configured in ad-hoc fashions and with intricate architectures. Depending on the application, these assets are network configured to concurrently perform Command, Control, Communication, Intelligence, Surveillance, and Reconnaissance (C3ISR) operations. Depending on the application, different tasks can be dedicated to unmanned systems, and generally these tasks may be classified as sensing, processing or decision making and actuating. The network can be thus organized as an ensemble of clusters consisting of a certain number of interconnected nodes performing the above mentioned tasks. Each vehicle might participate in two networks: An external network that enables the coordination and communication of the vehicles with other entities, such as peer vehicles or ground stations, and an internal network that connects on-board sensors, actuators and other devices. The current communication that is implemented with proprietary, application-specific, and point-to-point and fixed-bandwidth channels-based inflexible protocols cannot be used for such complicated networks. Therefore, the sheer number of the unmanned assets or nodes used and their corresponding dynamics, coupling interactions, network bandwidth constraints, data latency (delays in the time of arrival of information and data), scheduling and real-time requirements, data acquisition, control, processing and routing requirements, etc. would introduce a new set of challenging problems which call for the development of innovative and novel scalable, evolvable, intelligent and distributed algorithms and protocols. However, in order to define the problem statement for unmanned system communication, it would be judicious to start surveying the existing network metrics that may have in common certain fundamental features such as wireless communication, mobility, and real-time requirements. Among the existing networks, wireless sensor network (WSN) and Networked Control Systems (NCS) are useful for this purpose. The major features of networked unmanned systems (NUS) are as follow:

1.1.1 Heterogeneity

Assets in Unmanned Systems are very heterogeneous in their hardware and software. Depending on their destination and application, assets may be Unmanned Ground Vehicles (UGVs), Unmanned Aerial Vehicles (UAVs), Unmanned Marine Vehicles, and Unattended Ground Sensors (UGSs). Therefore, their size may differ and consequently their resources like energy, memory, computation and communication would be limited. Hence, due to the heterogeneity feature of assets [69], several problems of compatibility and adaptability may arise. In terms of communication problems due to these devices being diverse, trade offs may be necessary to organize the ensemble of nodes by finding the optimal topology.

1.1.2 Network topology

According to their decision level with respect to an application, assets can be organized either in hierarchical or flat architecture. The communication topology may be star, ring, tree, mesh, or fully connected and end-to-end versus hop-by-hop. It is dynamic due to the time varying link condition and asset mobility or addition or removal of assets. In addition, topology is subject to an interaction problem and coupling effects that could lead to unstable or unmanageable dynamics. By considering all the interactions and loops, a given topology could be fully-connected which is well-known in communication community as the most complicated topology. This issue and its related challenges is still an open area for further research.

1.1.3 Real-time requirements

As in NCS that may be defined as a control system communicating with sensors and actuators over a communication network, assets in NUS also can receive information from any entity of the network, process it and send the decision made to any entity of the same or other networks. In addition, both NCS and NUS interact with the physical environment. Therefore, NUS may be classified in the same research area as the NCS that is called control over network. In fact, control that is a central element in all applications related to the next generation of intelligent machines may intervene differently with respect to communication networks, however two main areas have been generally defined, control over networks and control of networks.

In control over networks, and specifically in NCS the crucial problem is how to ensure the real-time requirements and the convergence of all considered disciplines namely control, communication and computing. In fact, the operation correctness within NCSs with maximal level of performance, delivery timelines of messages must fulfill the involved physical environment real-time constraints, whereas the non-deterministic communication network could not guarantee any reliability and timelines of messages. These networks are characterized by a lack of QoS such as packet loss, delays, delay jitter, and bandwidth limitations. The problem of lack of message reliability or "dead-information" is mainly due to latency or delay. We believe that the real-time requirements in NUS are almost similar to those of NCSs or slightly more complex due to the bi-directionality of information.

On the other hand, and in control of networks area the basic problems include controlling congestion across network links, routing the flow of packets through the network, caching and updating data at multiple locations, and managing power levels for wireless networks. Furthermore, there may be more challenges in presence of some features such as the extremely large scale system architecture, the decentralized nature of the control system, the stability in presence of varying time lags, uncertainty and variations in the network and also the other issues of diverse traffic characteristics in terms of arrival statistics at both the packet and flow time scales, and different requirements on quality of service, in terms of delay, bandwidth, and loss probability, that the network must support.

1.1.4 Time-delay

It is widely known that in closed-loop control systems, time-delay may degrade the dynamic performance, and even worst may lead to instability of the system [55], [79]. In communication networks, time-delay is usually considered as a summation of different delays that can be mainly caused by the propagation of messages (cell/packet) through the communication medium, the queueing in different buffers, and processing time, congestion, absence of synchronization between clocks and so on. For instance, to quantify the time-delay in NCS, basically we approximate it by the propagation time of messages from sensors to actuator and controller to actuator since the process time in the controller is much smaller. Therefore, time delay can be crucial for NUS since their assets may be far from each other (links to base station or satellite) while their topology that is characterized by several loops, interactions, and coupling effects may be mesh or even fully connected.

1.1.5 Bandwidth constraints

The dynamics of heterogeneous assets are different in speed. Consequently, the frequencies of messages exchange are different, and their bandwidths are allocated accordingly. On the other hand, the entire NUS dynamics is subject to continuous changes due to asset mobility and adding/deleting of assets. Therefore, and for a maximal utilization, the bandwidth allocation should be dynamic in order to adapt to these changes. It is widely accepted that resource allocation constitutes one of the most important issues in sensor networks. Several efforts are based on deploying fairness of bandwidth allocation which are linked to congestion problems. Usually, transport protocols are used to remedy these problems.

1.1.6 Protocol limitations

In order to reach good performance for networks, protocols are designed to provide fairness in bandwidth allocation, congestion control, reduce packet loss, and ensure end-to-end reliability. Recent works have shown that [59] conventional transport protocols designed for internet such as User Data Protocol (UDP) and Transport Control Protocol (TCP), cannot be directly applied to wireless sensor networks. In internet, UDP is unreliable even though it is simple and connectionless, whereas TCP is reliable but slower and complex (connection-oriented). One of the main TCP drawbacks in wireless sensor networks is that it assumes that the packet loss is due to congestion, which is not totally true, since packet loss may also be due to the high error bit rate in the link. Consequently, TCP may have a degraded throughput (TCP triggers rate reduction since it assumes that packet loss are due to congestion), tardy response resulting on longer time to mitigate congestion and more packet loss (end-to-end congestion control in TCP versus hop-by-hop control), a lack of fairness of bandwidth allocation and data collections (depending on the topology utilized, flow and congestion control may discriminate some nodes), and high bandwidth and energy consumption (because of the end-to-end based TCP).

1.2 Motivations/Applications

We believe that we are not very far from the world with ubiquitous autonomous machines, that would share our space. It is widely accepted that the area of unmanned systems is new and flourishing so that the number of military and civilian applications is rapidly increasing. On Earth or in space, for mapping, patrolling, searching and rescuing, the application range is very wide thanks to the technological progress and the fact that the devices are inexpensive. Recently, governments of several countries are investing more in research on unmanned systems while as mentioned previously, many companies are investigating and producing such systems. In United States of America, the government and the Department of Defence (DoD) are emphasizing this area and injecting prodigious budgets. However, the consideration of such systems as networks opens undeniably several challenging and promising research opportunities.

1.3 Objectives

It is worth noting that communication problems in NUS have not been extensively investigated and covered in the literature. We are interested in investigating this issue by first defining the main metrics. We then study the resource allocation and congestion control problems. Before studying these issues, it is important to evaluate the complexity of the system dynamics with intricate topologies such as fully-connected structures (in presence of interactions and coupling effects). Stability analysis of these networks in presence of time-delays constitutes also one of our goals. We are also interested in modeling mobility within our proposed framework.

We propose a control strategy that is based on feedback linearization approach but it is empowered with robustness features and capabilities.

1.4 Literature Review

1.4.1 Communication issues in NUS

In unmanned systems, and particularly in cooperative UAV control, the research topics cover computer vision for real-time navigation and networked computing and communication strategies for distributed control, as well as traditional aircraft-related topics such as collision avoidance and formation flight.

Specifically, communication is important in unmanned autonomous vehicles for reporting observations to human experts or to data repositories, for obtaining high-level directions concerning their tasks and objectives, and for coordinating among multiple assets.

The challenge with the future generation of unmanned networks that consist of a number of heterogeneous assets configured in ad-hoc fashions and with intricate architectures is to enable communication that supports scalable, evolvable, and intelligent protocols.

The communication limitations such as noise, delays, bandwidth, range and network topology, introduce multitude of problems that might be engendered and eventually several research issues may arise for investigations such as communication architecture (information exchange strategy) and topology (rings, trees etc.), autonomy level, and QoS improvement techniques (scheduling, traffic shaping, resource reservation, and admission control).

1.4.2 Importance of communication in NUS

Communication strategies consist of establishment of task ensemble according to the available resources in order to attain certain objectives. In the literature [52]

[67], [96], [99], and [112], several efforts have been made to develop strategies for unmanned systems that are subject to several constraints such as the way the assets are organized according to their topology and behavior.

Information exchange strategies should be designed to improve formation stability and performance and should also be robust to changes in the communication topology. However, for example in [42] the authors show how the topology of the information flow affects the stability and performance of the system as it performs a coordinated task. Using graph theory to model the communication topology and control theory tools they studied the interplay between the communication network and the vehicle dynamics. Their approach for information exchange methodology exhibits a separation principle which decouples the stability of the formation communication which they term information flow, and the local control of individual vehicles.

Furthermore, as far as the topologies of communication networks are concerned, it should be noted that some of these are very complex such as the mesh and the fully-connected topologies. For instance, in [120] the authors have argued that multiring topology is a solution that may ensure survivable, secure, and scalable group communications in military environments, especially for battlespace environments. The authors have shown that multiring architectures may decrease network diameter, hence improving end-to-end delays for applications.

1.4.3 Protocols and their limitations

Communication strategies are implemented using a set of formal and well-ordered rules called protocols describing how to transmit data across a network. In the backbone of such networks that is the Internet the hardware layer is the Ethernet. To really exploit the benefits of Ethernet, communication strategies must use the most common protocols. At the lowest level is the Internet Protocol (IP), and above it are the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP). TCP/IP is a stream-based protocol that keeps track of the data sent between two hosts to guarantee delivery in the correct order, whilst UDP/IP just sends packets without tracking. UDP is faster since it doesn't require much checking, but TCP is better when sending large amounts of data.

Currently, such protocols are ineluctably used in NUS. As an application of using IP, Liberatore's team from Case western University and Bashin from NASA Glenn Research Center have argued in [78] that the challenge enabling UAV communication that supports scalable, evolvable and intelligent protocols might be accomplished through the use of IP coupled with an appropriate design of local controller, real and non real-time middleware and agent-based software. In order to address the real-time problem due to the limitation of the communication between the human operator and the robot (vehicle), they proposed the control architecture that is composed of software agents organized as "virtual robots" that may vary the degrees of autonomy. In this way, the robot assumes higher level of control responsibility.

Regarding the TCP protocol, it is also addressed in unmanned system communication. For instance, in [88] Palazzi *et al* have studied a TCP-based communication architecture for unmanned systems. They studied the benefit of extension of this communication architecture to satellite services in case of emergency. Indeed, when fixed infrastructures collapse, access to satellite resources might represent the only means of communication, even though and especially in an urban environment, shadowing can strongly reduce the visibility time. In such cases they conclude that to improve performance they may act on both architectural and protocol (TCP) levels.

A large body of work has been developed to improve such protocols and to make them scalable, evolvable and intelligent. Furthermore, some studies conducted at Berkley have shown that current TCP protocols present important limitations.

For example, in [27], the authors concluded that in wireless networks packet loss which is assumed as a sign of congestion by TCP (either Reno or Friendly schemes), can also be caused by physical channel errors. Consequently, the congestion assumption breaks down and TCP seriously underutilizes the wireless bandwidth. Through experiments the authors found that TCP Friendly Rate Control (TFRC) achieves only 56% of the available wireless bandwidth. Knowing this TCP limitation, the same research team has proposed several enhanced variants of TCP in [28], [29], [30], and [100].

These TCP limitations have been recently surveyed in [119], in which also the transport protocols have been classified.

1.4.4 Differentiated-services (DiffServ) architecture

The evolution of Internet Protocol (IP) networks has been witnessed from providing a single best-effort service to providing multiple types of services with Quality of Service (QoS) guarantee, namely bounded packet delay and loss. Although the most current network traffic is still best-effort, users are demanding better QoS guarantees for their traffic, or QoS differentiation for their virtual private networks. The Integrated-Service (InterServ) framework is an attempt to achieve end-to-end per-flow QoS but was unsuccessful in large-scale networks mainly due to scalability and heterogeneity concerns. An alternate attempt to solve the challenges of providing QoS is the Differentiated-Services (DiffServ) framework.

However, two broad aggregate behavior groups have been adopted by the DiffServ working group [13]: the Expedited Forwarding (EF) Per-Hop Behavior (PHB) and the Assured Forwarding (AF) PHB. First, the EF-PHB can be used to build a low loss, low latency, low jitter, assured bandwidth end-to-end service, thus indirectly providing some minimum aggregated quality of service, whereas the AF-PHB group provides delivery of IP packets in four independently forwarded AF classes.

Within each AF class, an IP packet can be assigned three different levels of drop probabilities and each class can be provided with some minimum bandwidth and buffer guarantees.

Furthermore, in [89] the authors have adopted the same spirit as the IETF DiffServ working group and divide traffic into three basic types of service: Premium Traffic Service, Ordinary Traffic Service, and Best Effort Traffic Service. The premium traffic belongs to the first class of EF PBH whereas the best effort traffic belongs to its last class. On the other hand, the ordinary traffic belongs to the first class of the AF PBH.

The Premium Service requires strict guarantees of delivery, within given delay and loss bounds. It does not allow regulation of its rate. On the other hand, Ordinary Traffic allows the network to regulate its flow into the network. It cannot tolerate loss of packets but it can however tolerate queueing delays. Finally, Best Effort Service offers no guarantees on either loss or delay. It makes use of any instantaneous leftover capacity from Premium and Ordinary.

In the DiffServ network domain, several related issues have been investigated in the literature such as: (a) adaptive Connection Admission Control (CAC) that has been shown to be efficient in terms of bottleneck link utilization [73], (b) a methodology for providing the absolute DiffServ for real-time applications [121], (c) mechanisms and options proposed for Transport Control Protocol (TCP) performance enhancements which are evaluated within a Bandwidth on Demand (BoD)-aware satellite simulation environment [63], and (d) for scheduling control [81]. For comprehensive surveys the reader is referred to [19] for bandwidth and fairness issues or to [108] for absolute and relative DiffServ discussions.

1.4.5 Congestion control and bandwidth allocation

The above protocols, and specifically the transport protocols, provide the following functions: orderly transmission, flow and congestion control, loss recovery and possibly Quality of Service (QoS) guarantee such as timing requirement and fairness. Hence, transport protocols are used to mitigate congestion and reduce packet loss, to provide fairness in bandwidth allocation, and to guarantee end-to-end reliability.

Congestion is the problem which occurs when demand for a resource outstrips the capacity [7], [36], [45], [75], [111], and [123]. For instance, in Wireless Sensor Networks (WSN) congestion may be caused by many new factors such as the convergent nature of the upstream toward the sink and the bandwidth limitations. However, the two main congestion causes are the packet arrival rate exceeding packet service rate that occurs at sensor nodes close to the sink, and the second cause is due to the link level performance such as contention, interference and bit error that occurs on the link. Not only the congestion in WSNs has an impact on the reliability and application's QoS but also on its energy-efficiency. In effect, congestion leads to longer queueing time (packet loss), degradation of the link utilization and transmission collision if contention-based link protocols (CSMA: Carrier Sense Multiple Access) are used to share radio resources. All these consequences cause a waste of additional energy [2].

A common congestion method uses the queue length [56], packet service time [38] or ratio between packet service time and packet inter-arrival time at an intermediate node [118]. To overcome congestion one may either reduce the load of the network or increase the resources. Congestion control consists of preventing congestion before it happens or removing it after it happens. Congestion control approaches can be divided into two classes, namely the open-loop and the closed-loop [107]. In open-loop, the congestion control approaches are: retransmission policy, window policy (selective repeat better than Go-back-N), ACK policy (piggybacking), discarding policy (discard less sensitive packets) and admission policy (before admitting flow, check resource requirement, QoS). On the other hand, the main closed-loop control approaches are: chock packet (packet sent by routers to the sender to inform the congestion), hop-by-hop chock packet (if a router is congested it informs the previous upstream router to reduce the rate of outgoing packets), explicit signaling (the router experiencing the congestion can send an explicit signal, by setting one bit in a packet to inform the sender or the receiver of the congestion) and implicit signaling (the sender can detect an implicit signal of congestion and slow its sending rate.)

It is generally accepted that the problem of network congestion control remains a critical issue and a high priority, especially given the growing size and speed (bandwidth) of the increasingly integrated services demanded from fixed and mobile networks. Moreover, congestion may become unmanageable unless effective, robust, and efficient methods for congestion control are developed. It is worth noting that a number of popular congestion control designs were developed using intuition, mostly resulting in simple nonlinear control schemes. From the congestion control solution deployed in TCP [60] to the popular congestion control schemes proposed to address the Available Bit Rate (ABR) problem in Asynchronous Transfer Mode (ATM) [84] several solutions have been studied [25], [61]. As the link speed increases to satisfy demand, and as the demand on the network for better quality of service increases, empirical and analytical evidence have demonstrated the poor performance and cyclic behavior of the controlled TCP/IP Internet under certain conditions [104]. Furthermore, the interaction of additional nonlinear feedback loops can produce unexpected and erratic behavior [97], and hence new effective congestion control schemes are needed. Despite the successful application of control theory to other complex systems the development of network congestion control based on control theoretic concepts is still at the research stage.

Few control theory-based attempts can be found in the literature such as optimal [39], [101], linear [11], [97], predictive adaptive [92], fuzzy and neural [32], [37], [80], and nonlinear control [89], [91]. For instance in optimal-based congestion control approach, Srikant [39] has described and analyzed a joint scheduling, routing and congestion control mechanism for Wireless Sensor Networks (WSN). Their main contribution is to prove the asymptotic optimality of a primal-dual congestion controller that models different versions of TCP. The queue lengths serve as common information to different layers of the network protocol stack.

The same author has surveyed in [110] recent research on the topic of Internet congestion control. He has emphasized the role of optimization, control theory and stochastic models in developing algorithms that perform well even in the presence of delays and stochastic disturbances such as arrivals and departures of flows. Note that no application and no results have been provided in these two studies.

On the other hand, using linear control approach Rohrs *et al* [97] have proposed and simulated a congestion control algorithm that can eliminate the large steady-state oscillations from the results of nonlinear algorithms. Linear control techniques employed here apply to the case where multivalued feedback is provided directly. Note that in [97] the proposed congestion control algorithm has been simulated and evaluated on one node, whereas it is well-known that performance of such linearized control systems are valid only when the system operates around a linear region. In [74], Rhors *et al* have proposed an effective control strategy for the explicit rate congestion controller. The main contributions of this paper are: congestion control algorithm enhancements including convergence rate improvements, queue depth management, and a method to reduce coefficient bias without compromising convergence or significantly increasing computational complexity. Similar to [74], the authors in [97] consider also a single node.

Using predictive adaptive (feedback and feedforward) control techniques, the authors of [92] propose a congestion control strategy which integrates Connection Admission Control (CAC) and flow control. It provides robust, effective and efficient congestion control with guaranteed QoS and high utilization of link capacity. It also tolerates (fairly) long propagation delays because it uses traffic prediction that, due to the (high-level) correlation present in some Variable Bit Rate (VBR) traffic, can forecast beyond the propagation delay. The sensitivity to traffic modeling is also addressed by using adaptive predictive control. This study has been conducted on one node in which the authors assume no interaction between any of the outgoing links.

Using fuzzy logic techniques, in [32], [33] the authors have proposed a new Active Queue Management (AQM) scheme implemented in TCP/IP networks within the differentiated-services framework to provide effective congestion control by achieving high utilization, low losses and delays. The design of the fuzzy knowledge base is kept simple, using the linguistic interpretation of the system behavior. This proposed controller constitutes an alternative to the existing Random Early Detection (RED) approach. Robustness and fast response as well as fewer queue fluctuations have been obtained with the proposed fuzzy system when compared to the classical RED in a differentiated-services framework. Even though the authors consider a network topology with multiple bottleneck links, no analytical results have been provided and no consideration regarding the interaction between outgoing links is mentioned.

In the same context in the domain of intelligent control approaches, a congestion control method for ATM networks is studied in [37] using neural networks. In the proposed method, the coding rate for multimedia stream and the corresponding user percentage are taken as the controller output to adjust the cells arrival rate to the multiplexer buffer. This control method not only minimizes the cell loss rate, but also guarantees the quality of information such as voice sources, which utilizes the ATM networks as high as possible.

In [91], the authors have introduced a combined control of connection admission, flow rate and bandwidth for ATM based networks. A fluid flow model in state variable form describes the time-varying mean behavior of the queueing system. The integrated control strategy is derived using nonlinear control theory.

For a differentiated-services framework [13], Pitsillides *et al* [89] have also derived an adaptive nonlinear congestion controller using information on the status of each queue in the network. Known as the Integrated Dynamic Congestion Control (IDCC), this congestion control scheme that is general and independent of technology, (such as TCP/IP or ATM) is designed also on the basis of the same concepts of nonlinear control theory and fluid flow model [1,26,35,65].

Validated in [9], [43], [47], [48], [98], [103], [107], and [113], the utilized fluid flow model (FFM) is of first-order and simpler than a detailed Markovian queueing probabilistic models. In fact, modeling queueing systems is a crucial issue and many work might be found in the literature. Using the conservation principle the FFM proposed by Agnew [1] might be used for modeling either queueing and contention systems as well as loss systems [12].

However, the FFM which is relevant to our research is also studied in [113] in order to model the time-varying behavior of ad-hoc mobile networks. Indeed, network queues are described using fluid-flow based differential equation models which are solved using numerical methods, while node mobility is modeled using an adjacency matrix topology representation whose values are determined via discrete event simulation techniques. Note that even though the network topology utilized is interesting in the sense that the closed-loops are considered, the congestion control problem is <u>not</u> studied.

To aggregate traffic in packet-switched network management and control, bandwidth allocation should be made in an efficient way to provide quantitative packet-level QoS. But due to the unpredictable, unknown statistical characteristics of the aggregate traffic this issue is still not fully solved.

In the literature several efforts have been deployed to address the bandwidth allocation problem and to propose adaptive algorithms that guarantee aggregate traffic packet-level QoS metrics, such as the average queue length, packet loss, and packet delay [31], [41], [66], [90], and [106]. Based on online measurement of the system state and measured traffic information and possibly traffic prediction, the bandwidth controller adjusts the allocated bandwidth on a time scale between a packet level and a connection level such that the QoS requirement is achieved while maintaining a high bandwidth utilization.

According to the underlying control techniques used, or the QoS metrics guaranteed, the existing algorithms can be classified as being either open-loop or closed-loop. The open-loop control approaches involve predicting the input traffic rate using the past history whereas the closed-loop control approaches can be further categorized based on the guaranteed QoS metrics, including the average queue length [90], loss [106], and delay [31], [66].

To summarize the various classes, one starts with the control category that uses the queue length. Based on fluid flow model (FFM), lot of work on Adaptive Bandwidth Control (ABC) attempt to maintain the average queue length at some desired target value so that the packet loss and the average queueing delay are kept low. In [93], to improve the global network performance and robustness a multilevel optimal control approach is developed to coordinate the decentralized nodal bandwidth control.

To handle a finite buffer and a finite capacity situation, the authors of [91] have proposed a proportional control in the context of the ATM Available Bit Rate (ABR) service. In order to maintain a low average queue length the authors of [90] have derived a bandwidth control using feedback linearization and robust adaptive concepts.

The second class of non-zero loss guarantee is divided into direct and indirect approaches. First, a non-zero loss might be guaranteed when ABC is designed to ensure that the cumulative loss converges to the desired (non-zero) target packet loss rate and stays at that level. Using integral control in [53], the allocated bandwidth is adjusted in proportion to the difference between the measured loss and the target loss.

On the other hand, the packet loss might be guaranteed indirectly after translating the loss requirement into some other performance measure [106]. In [106], using fuzzy control the average queue length is maintained between a queue threshold pair that depends on the target loss rate. Note that at low target loss rate, the target queue length can be so small that the control approach becomes ineffective.

The third class guarantees a zero loss by allocating the bandwidth that matches the arrival rates which is best implemented in open-loop control. Indeed, zero loss can be expected whenever the service rate is sufficiently greater than the arrival rate [41]. Guaranteeing a delay requirement, the fourth class is studied in [66], in which bandwidth adjustment is used to guarantee a probabilistic delay bound. For a single server the probabilistic delay bound requirement is that the probability that the packet queueing delay will be greater than some value is less than some other small value.

The last class guarantees a multiple QoS metrics. In [31], a class-level ABC is developed for a qualitative Assured Forwarding (AF) DiffServ class with absolute and relative differentiation of delay and rate guarantees over the duration of a busy period.

Furthermore, in NCS [5] the sampling periods of control systems (thus their bandwidth consumption) is linked to the congestion level fed back from the network.

In other words, NCSs adapt their bandwidth usage by varying their sampling periods so as to avoid congestion in the network.

1.5 Thesis Contributions

The research conducted in this thesis attempts to investigate the following issues.

■ **Communication metrics for NUS's**

To successfully carry out NUS missions, communication among assets need to be conducted efficiently. In contrast with conventional communication networks, NUSs have specific features that may render the communication problem more complex. In this research, the main NUS characteristics are stated as follows: (a) heterogeneity of assets in terms of resources, (b) topology that may be fully-connected [42], (c) real-time requirements that are imposed by delivery timelines of messages for the involved physical assets in uncertain environments, (d) time-delays that may degrade performance of the closed-loop dynamics, (e) bandwidth constraints reflecting differences in assets characteristics, and (f) protocol limitations in mainly complying with wireless features of these networks [27], [59]. In order to model and simulate the above NUS characteristics, a differentiated-services fully-connected network is clustered. Each NUS cluster is defined as a group of three nodes, namely, a sensor, a decision-maker, and an actuator. By using the fluid flow model (FFM), the overall dynamics of our cluster is derived as a time-delay dependent system.

■ **Bandwidth allocation control for NUS**

In this thesis, and corresponding to our defined time-delay dependent NUS cluster, the resource (bandwidth) allocation problem is investigated for different configurations in terms of possible interconnections of various assets. We have demonstrated the difficulty of allocating the bandwidth to different nodes of a NUS by implementing a standard PID controller. In effect, through simulations we have shown that depending on interconnections between the network nodes, the behavior of the closed-loop system vary considerably such that the network might become even unstable. This difficulty can be however addressed by using elaborated nonlinear control approaches.

The first nonlinear control approach that is investigated is the input-output linearization-based congestion controller. By assuming that the input traffic and the delays are measurable, the controller has demonstrated a good performance and can achieve the imposed steady state error and the transient design specifications.

The second nonlinear congestion controller is designed using the sliding mode machinery. By assuming that the input traffic and the delays are measurable, the time-delay dependent control system dynamics is derived and shown to be asymptotically stable for a fully-connected configuration. For unknown delay and measurable input traffic, we have derived the time-delay dependent control system dynamics which is shown to be asymptotically stable for a cascade configuration. Under the same assumptions on the delay and the input traffic and for configuration having a feedback traffic (loop), the stability of the derived control dynamics is guaranteed provided that the time-delay is upper bounded.

For unknown delay and by estimating the input traffic, the error dynamics of the overall NUS cluster is derived as a generic time-delay system for different network configurations. Moreover, by considering the node mobility, our proposed robust congestion controller is evaluated for a wide range of operating points with demonstrated good performance and stability.

- **Combined bandwidth allocation and flow rate control**

In this research, three robust integrated bandwidth allocation and flow rate control strategies are proposed. The proposed controllers maintain the average queue length of each buffer of the network by dynamically allocating the bandwidth to the server, and control the flow rate.

The first robust control strategy is an improved version of the strategy proposed in [91]. This strategy is empowered by robustness capabilities and constitutes a benchmark solution to the third strategy. The first and the second strategies do not need any knowledge about the incoming traffic in order to compute their commands while in the third strategy, the input traffic is assumed to be measured or estimated. A comparative study between the first and the third proposed strategies is conducted and the results can be stated as follows: a) for any network configuration, proposed strategies guarantee convergence and stability of the time-delayed network, b) the buffer queue lengths responses obtained by using the first strategy are quite oscillatory while those obtained by using the third strategy are oscillation free, c) the results obtained by using the third strategy for a measurable input traffic are much better than those obtained for estimated input traffic, and therefore, an improved estimator would compensate for the overshoots (e.g. $\sigma - modification$).

■ **Robust congestion control strategies for differentiated-services**

Our research investigations have led to several congestion control strategies. In these strategies, the buffer queue length is used as a feedback information to control locally the queue length of each buffer by acting on the server bandwidth and simultaneously sending back to the ordinary sources the allowed maximum rate. Our proposed congestion strategies can be categorized into three classes. Using the sliding mode generalized variable structure techniques (SM-GVSC), the first and the second classes of congestion control strategies are called as *non degenerate* and *degenerate GVS control approaches*, respectively. In fact, the GVS control technique is recently introduced in [44] to alleviate the undesirable chattering phenomenon by designing a control law that switches on the derivatives of the control input (non degenerate case) rather than on the control input itself as in the classical variable structure (CVS). In cases where the GVS design leads to a control law that does not exhibit the derivatives of the control input, it is then denoted as the *degenerate* *case.*

In the first class, three different decentralized end-to-end control strategies are proposed. The end-to-end control approach refers to the fact that the node controller notifies all the upstream ordinary sources in order to reduce the traffic in case of congestion. The first control strategy consists of solving the sliding mode condition which is inspired from the equivalent control-based sliding mode control method. The second control strategy consists of adding a discontinuous term to the classical input-output linearization-based control law for robustness purposes. Finally, the last control strategy utilizes the hyperplane convergence equation as a nonlinear feedback control.

The second class of congestion control strategies (degenerate case) uses the hyperplane convergence equation as a nonlinear feedback control. Three strategies are proposed, namely, the decentralized end-to-end control, the semi-decentralized end-to-end control, and the distributed hop-by-hop control.

For a measured input traffic, the decentralized end-to-end control strategy is first implemented on a Diffserv single node, whose time-delayed dependent congestion control dynamics is derived analytically in order to guarantee stability of the closed-loop system. The proposed congestion controller has demonstrated good performance and robustness to a burst of input traffic. The decentralized end-to-end control strategy is then implemented on a Diffserv cascade network and the resulting time-delay control dynamics is shown to be stable. Simulations results have shown that for the Premium service, the buffer queues converge to their respective reference sets and for the Ordinary service, the convergence of the buffer queues can be reached but with oscillatory responses inducing a steady state error.

By estimating the input traffic and adding the bandwidth control of the Ordinary service, a semi-decentralized end-to-end control strategy is proposed and implemented first on a three nodes cascade network and then on a five nodes cascade network. For the three nodes (cascaded) configuration, the controller has demonstrated good performance and convergence. For the five nodes configuration, the simulation results have shown that the buffer queue length responses corresponding to the ordinary service are oscillatory. One can argue that the semi-decentralized end-to-end control approach is limited due to a lack of scalability.

By estimating the input traffic and adding the bandwidth control of the Ordinary service, a distributed hop-by-hop congestion controller is investigated first on a general mesh network and then on a NUS cluster. Note that contrary to the mesh network, the NUS cluster is mainly characterized by traffic feedbacks. For each network, the resulting closed-loop network cluster error dynamics is studied as a time-delay system and stability conditions are derived that are only dependent on the configuration of the network. The analytical results are confirmed through a number of simulation case studies. For a five nodes network, our control has demonstrated good performance to dynamic input traffic resulting in a good scalability property. For our network cluster, a comparative study is conducted to demonstrate and illustrate the advantages and superiority of our proposed congestion controller when compared to a conventional method (which in general results in an unstable closed-loop system).

The third class of congestion control strategies are designed using the classical input-output linearization-based nonlinear control theory. In fact, this controller is derived for further comparison with the above GVS congestion controllers. The comparison study will be conducted to show the performance and robustness capabilities of the GVS congestion controllers.

1.6 Thesis Structure

As illustrated by Figure 1.1, this thesis is structured as follows. After reviewing the

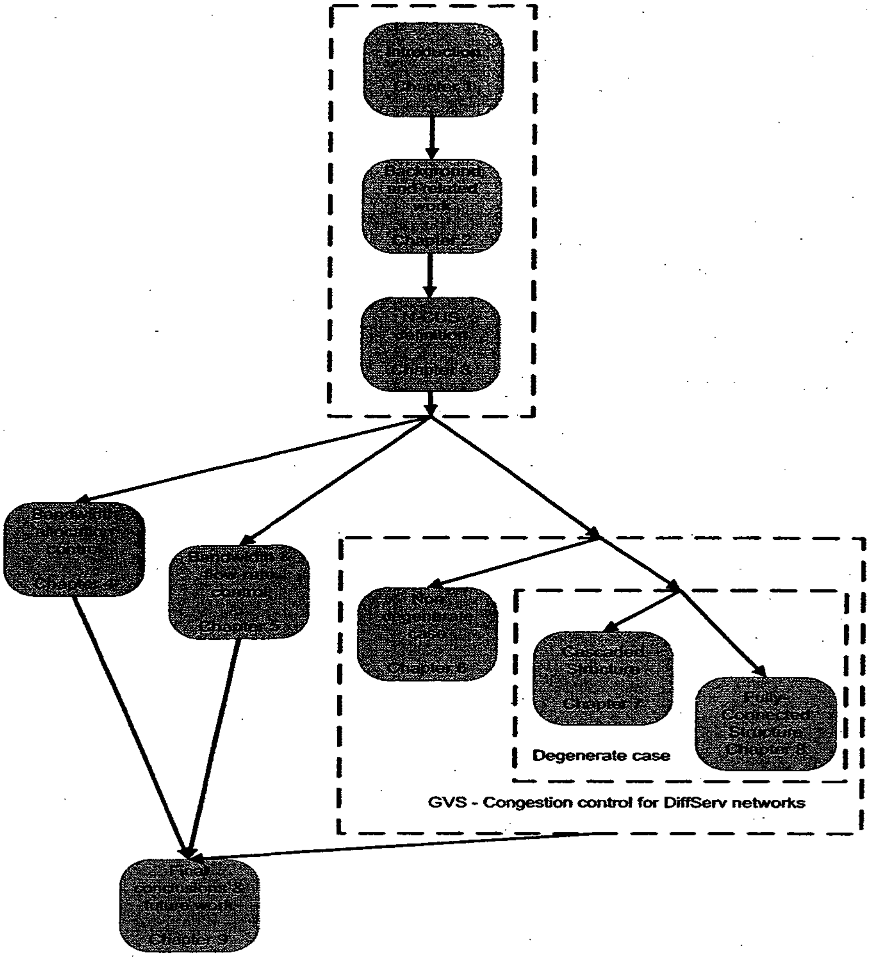

Introduction
Chapter 1
Background and related work
Chapter 2
NUS definition
Chapter 3
Bandwidth allocation control
Chapter 4
Bandwidth & flow rate control
Chapter 5
Non-degenerate case
Chapter 6
Cascaded Structure
Chapter 7
Fully-Connected Structure
Chapter 8
Degenerate case
GVS - Congestion control for DiffServ networks
Final conclusions & future work
Chapter 9

Figure 1.1: Thesis Structure.

necessary tools for designing our proposed control strategies and the related work in Chapter 2, an integrated bandwidth allocation and flow rate control and the congestion control problems are introduced. A NUS cluster is defined and modeled in Chapter 3. Chapters 4-8 constitute the core of our research. Chapter 4 is devoted to the bandwidth allocation control and Chapter 5 treats the integrated bandwidth allocation and flow rate control. Chapters 6-8 are devoted to the congestion control for DiffServ networks. The non-degenerate GVS control strategies are investigated in Chapter 6 and the degenerate GVS control strategies are studied in Chapters 7 and 8. In Chapter 7, we proposed our first method of the degenerate GVS control strategies and our second method is proposed in Chapter 8. Final conclusions and future work are stated in Chapter 9.

Chapter 2

Backgound

2.1 Introduction

This research addresses bandwidth allocation and congestion control problems. The sliding mode machinery is utilized in order to design the proposed robust model-based algorithms. In virtue of the robustness capabilities of the sliding mode control, an inaccurate/uncertain queueing model can be used. The fluid flow model that is one of the most intuitive and popular techniques that model a wide range of networked systems [1], [26], [65], [125], [126], [128] is introduced in the first section.

In communication networks, bandwidth allocation, flow control, and connection admission control can be studied either separately or preferably in a combined way. In the second section of this chapter, the integrated control of bandwidth, flow rate, and connection admission proposed in [91] is discussed. As far as the congestion control problems are concerned, our proposed solutions constitute alternatives to the integrated dynamic congestion controller (IDCC) in [89].

The aforementioned sliding mode-variable structure control (SM-VSC) machinery is reviewed in the last section of this chapter. The SM-VSC approach that we use is also known as sliding mode Generalized Variable Structure Control

(SM-GVSC) [44]. Contrary to the sliding mode classical variable structure control (SM-CVSC) that is studied in the context of differential geometry [95], [116], the SM-GVSC control uses the differential algebra tools.

2.2 Overview of the Fluid Flow Model (FFM)

Among a number of formal models that may be used in describing the queueing systems, the fluid flow model is one of the most intuitive and popular techniques that could model a wide range of networked systems [1]. The fact that the FFM can be used both at the edge and inside the network as well as for wired and wireless communication networks indicates its merit as a unifying framework for evaluation of the performance of these networks. Indeed, based on the flow conservation principle, the following equation of FFM represents the rate of change of information $\dot{x}(t)$ queued in the buffer as a difference between the arriving input and the outgoing output information rates $f_{in}(t)$ and $f_{out}(t)$, respectively, assuming that no losses exist, that is

$$\dot{x}(t) = f_{in}(t) - f_{out}(t)$$

where $f_{in}(t) = \lambda_i(t)$ is the rate of arriving customers at the buffer queue and $f_{out}(t) = C(t)G(x,t)$, denotes the function of the state variable $x(t)$ representing the queueing length, with $G(x,t)$ representing the ensemble average utilization of the queue at time t, whereas $C(t)$ denotes the capacity of the queue server. In fact, the commonly utilized approach to determine $G(x(t))$, is to match the steady state of the above differential equation with that of a queueing theory model based on M/M/1. Note that in the Kandall's representation, M corresponds to the interarrival and service time distributions, respectively, while 1 designates the number of servers.

Validated by several research work [1], [9], [43], [47], [48], [98], [103], [107], and [113], the FFM may be represented according to different forms and the variant

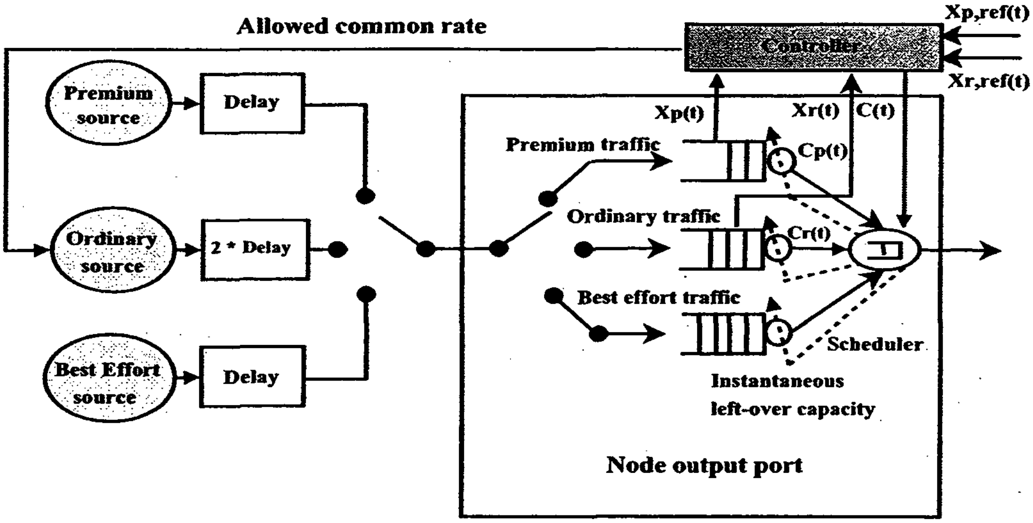


Figure 2.1: Control strategy implemented at each switch output port [17]

of the FFM we are interested in is much simpler than the probabilistic models [114], and also of a low-order complexity.

In the congestion control problem, the buffer queue length is maintained close to some desired value by acting on the buffer server capacity. However, for control purposes, the fluid flow model can be represented as a system of state and output equations as given in (2.1). Note that $y_f(t)$ is the output variable (this is to follow the standard structure of dynamical systems in the state space representation form) that is taken for simplicity to be the same as the state $x_f(t)$. One way to control the information flow is to divide the traffic into three main classes (refer to Figure 2.1), according to their priority and the degree of their importance [89]. The most important traffic class, i.e. the first class, is the Premium traffic service which requires strict guarantees of delivery within a given delay and loss bounds. The second class of traffic, the Ordinary traffic service allows the network to regulate the input flow, while it tolerates the queueing delay but doesn't tolerate the loss of information. It mainly uses the left over capacity from the premium traffic service.

The Best Effort traffic constitutes the third class which is less important than the above two classes, such that it utilizes the left over capacity from the premium and the ordinary traffics. Since the third class does not intervene with the control design, only the Premium and Ordinary traffic services are represented by the nonlinear equation (2.1) where $f = (p, r)$ with p and r indicating the premium and the ordinary buffer dynamics, respectively,

$$\begin{cases} \dot{x}_f(t) = -C_f(t)\dfrac{x_f(t)}{x_f(t)+1} + \lambda_{if} \\ y_f(t) = x_f(t) \end{cases} \tag{2.1}$$

2.3 Integrated Control of Bandwidth, Flow Rate and Connection Admission

Traffic control and resource management are crucial in order to guarantee the desired grade of service in multimedia and multiservice broadband networks as for example in Asynchronous Transfer Mode (ATM) based networks. Bandwidth allocation, flow control and Connection Admission Control (CAC) might be studied in a separate way or preferably in an integrated way. In [91], the authors state that the design of the entire system and the interaction of the various components is often more important than the optimization of individual components. However, they propose a nonlinear control solution in the form of a combined dynamic feedback controller for bandwidth, flow control and connection admission. In ATM architectures, the ABR traffic that can be seen as a premium traffic may compete the network resources with a Guaranteed traffic. To avoid cell losses of the ABR traffic, the latter needs to be controlled. As illustrated in Figure 2.2, the controller ensures an appropriate CAC policy (bounds guaranteed traffic), exerts influence on ABR traffic flow into the server (to avoid congestion) and using feedback information from the network state (queue length) it dynamically allocates bandwidth (cell-server-rate, capacity)

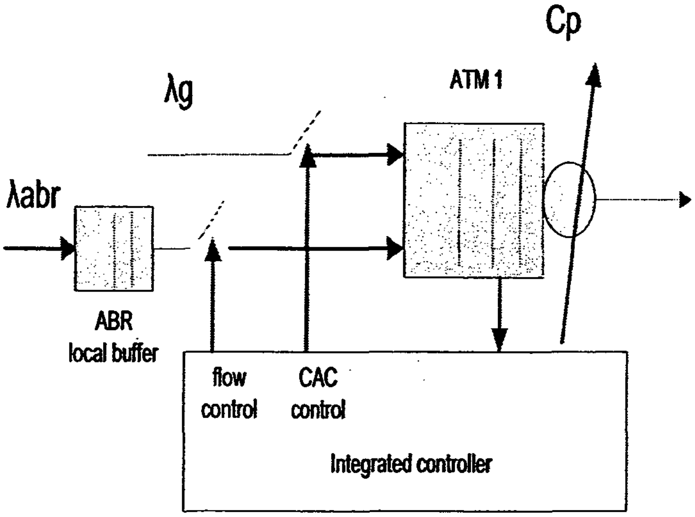


Figure 2.2: Integrated control concept [91]

to ABR traffic (to improve network resource utilization).

Flow control is exercised on the output of the ABR source which is considered as a local buffer. Any unmet ABR traffic demand can be thought as stored in the local buffer and whenever the resources become available it is paced in the network. In [91], the authors first consider a single node with ABR traffic and derive their proposed nonlinear control strategy and then an extension to guaranteed traffic and multiple nodes are considered. Using the FFM of equation (2.1), the model (2.2) describes the time varying mean behavior of Available Bit Rate (ABR) traffic, which competes with Guaranteed traffic

$$\dot{x}(t) = -C(t)\frac{x(t)}{x(t)+1} + \lambda_i^{ABR}(t) + \lambda_i^{guar}(t);\; x(t_0) = x_0 \tag{2.2}$$

where $0 \leq x(t) \leq x_{bufsizemax}$ with $x_{bufsizemax}$ is the maximum buffer size and $0 \leq C(t) \leq C_{serv}$ with C_{serv} is the maximum server capacity. For simplicity, we consider directly the case of non-zero guaranteed traffic. Defining the state error as $e(t) = x(t) - x_{ref}$, where x_{ref} is the desired reference queue length, equation (2.2) can be written as

$$\dot{e}(t) = -C(t)\frac{e(t) + x_{ref}}{e(t) + x_{ref} + 1} + \lambda_i^{ABR}(t) + \lambda_i^{guar}(t) \tag{2.3}$$

In this case, by choosing $C(t) = \alpha_p(1 + e(t) + x_{ref}) = \alpha_p(1 + x(t))$ one obtains [91] $\dot{e}(t) = -\alpha_p e(t) - \alpha_p x_{ref} + \lambda_i^{ABR}(t) + \lambda_i^{guar}(t)$ where α_p is the design parameter. Note that the approach considered by the authors is well known as an input/output linearization-based feedback control. The approach permits to choose a closed-loop dynamics having a stable behavior with predefined specifications. Two cases can be considered, namely, measurable or unmeasurable guaranteed traffic.

Case 1: Measurable Guaranteed traffic

By choosing $\lambda_i^{ABR}(t) = \alpha_p x_{ref} - \lambda_i^{guar}(t)$, and the above $C(t)$, equation (2.3) becomes $\dot{e} = -\alpha_p e(t)$, whose solution is given for all $t \geq t_0$ by

$$e(t) = E_{xp}^{-\alpha_p(t-t_0)} e(t_0); \;\; and$$

$$x(t) = x_{ref} + E_{xp}^{-\alpha_p(t-t_0)}(x(t_0) - x_{ref})$$

since $0 \leq x(t_0) \leq x_{ref}$, then $0 \leq x(t) \leq x_{ref}$ for all $t \geq t_0$. Note that E_{xp} denotes the exponential function symbol.

Case 2: Unmeasurable Guaranteed Traffic

By choosing $\lambda_i^{ABR}(t) = \alpha_p x_{ref}$ one obtains

$$\dot{e} = -\alpha_p e(t) + \lambda_i^{guar}(t)$$

whose solution is

$$e(t) = E_{xp}^{-\alpha_p(t-t_0)} e(t_0) + \int_{t_0}^{t} E_{xp}^{-\alpha_p(t-\tau)} \lambda_i^{guar}(\tau) d\tau$$

If λ_{imax}^{guar} is an upper bound for λ_i^{guar} that is known (can be provided by an appropriate CAC policy) then

$$e(t) \leq E_{xp}^{-\alpha_p(t-t_0)} e(t_0) + \frac{\lambda_{imax}^{guar}}{\alpha_p}(1 - E_{xp}^{-\alpha_p(t-t_0)})$$

or

$$x(t) \leq x_{ref} + \frac{\lambda_{imax}^{guar}}{\alpha_p} + E_{xp}^{-\alpha_p(t-t_0)}(x(t_0) - \frac{\lambda_{imax}^{guar}}{\alpha_p})$$

Depending on the values of x_{ref}, λ_{imax}^{guar} and according to the dependence of λ_i^{ABR} with respect to α_p when

$$\frac{\lambda_{imax}^{guar}}{x_{bufcap} - x_{ref}} \ll \frac{C_{max}}{1 + x_{bufcap}}$$

where C_{max} is such that $C_{max} \leq C_{serv}$, λ_i^{ABR} has to exceed a certain minimum value λ_{imin}^{ABR} for control strategy to be effective in presence of guaranteed traffic [91].

Now in order to extend the application of controlling the bandwidth allocation and flow rate to n ATM switches, let us write the allocated capacity for each switch as follows

$$C^j(t) = \alpha_p^j(1 + x^j)$$

and the ABR traffic is now limited to

$$\lambda_i^{ABR}(t) \leq \bar{\alpha} e_{max}$$

where $\bar{\alpha} = min(\alpha_p^1, \ldots, \alpha_p^n)$, $e_{max} = min(x_{max}^1, \ldots, x_{max}^n)$ and subscript j indicates an ATM switch j. Note that index j designates the j^{th} switch whereas index i utilized in λ_i indicates input traffic. The control strategy is clearly fully decentralized and the independence of the propagation delay between the switches is present. Regarding the CAC policy, for the ABR traffic its admission is only limited by the local buffer traffic source, whereas for the guaranteed traffic rate its admission must remain less than a portion of C_{max}, as given below [91]

$$\lambda_{imax}^{guar} < \frac{C_{max}(x_{bufcap} - x_{ref})}{1 + x_{bufcap}}$$

Remark 2.1. *The above results are interesting in the sense that the decentralized controller combines the three strategies to control the bandwidth, the flow and the connection admission. However, in [91], the sensitivity is evaluated through simulations* *and the results are not shown analytically, second the stability and the sensitivity to the time-delays are not studied in presence of feedback loops. This is of particular importance since they use a feedback linearization technique that cannot ensure robustness. These issues are all addressed and considered in our investigations.*

2.4 Integrated Dynamic Congestion Controller (IDCC) Strategy

General and independent of the technology, as for example in Transport Control Protocol/Internet Protocol (TCP/IP) or Asynchronous Transfer Mode (ATM), Pitsillides, *et al* [89] proposed the Integrated Dynamical Congestion Controller (IDCC) to solve the congestion problem in networks with multiple classes of traffics. In their paper, packet term is used for both IP packets and ATM cells whereas switch is used for both ATM switch and IP routers. By differentiating each class of traffic, the control objective for each class is "decoupled" from the rest, thus simplifying the overall control design. The control strategy is model-based dynamic feedback linearization, with proportional plus integral action and adaptation.

Each Origin-Destination (OD) flow may be classified as Premium service, Ordinary service and Best Effort service, as discussed previously. Assuming their reference model is a generic output buffered switch with K input and K output ports, each output port has one physical or logical queue for each traffic class. Because of the limited link capacity at each output port there is a rate mismatch between the flow into and out of the queue resulting in congestion problem.

To solve this problem, the IDCC is implemented at each output port. By tightly controlling each output port, the overall network is tightly controlled. For the premium traffic service, by dynamically allocating the node capacity the queue length in the node's buffer is controlled. The desired reference setpoint for the queue is chosen by the network operator to indirectly guarantee acceptable bounds for the maximum delay and loss. Indeed, as mentioned previously the IDCC is based on the FFM given by equation (2.1), in which $\lambda_{ip}(t) \leq \hat{k}_p^{id} < C_{serv}$, where $\hat{k}_p^{id}$ is a constant indicating the maximum rate that could be allocated to the incoming premium traffic (i.e. through the connection admission policy) and C_{serv} is the physical capacity of the server. By substituting for the derivative of the state in equation (2.1), instead its error derivative $\dot{e}_p(t) = \dot{x}_p(t)$ and nothing that $e_p(t) = x_p(t) - x_p^{ref}$, where x_p^{ref} is the desired average state of the premium traffic buffer, and by using feedback linearization and adaptive control ideas, the premium control input that is the capacity $C_p^{id}(t)$ which is bounded by $C_p^{id}(t) \leq C_{serv}$ is chosen as

$$C_p^{id}(t) = max[0, min\{C_{serv}, \mu^{id}(t)\}] \tag{2.4}$$

where

$$\mu^{id}(t) = \rho_p^{id}(t)\frac{1 + e_p(t) + x_p^{ref}(t)}{e_p(t) + x_p^{ref}(t)}[\alpha_p^{id}e_p(t) + k_p^{id}(t)] \tag{2.5}$$

with

$$\rho_p^{id}(t) = \begin{cases} 0 \ , & if \ \ x_p(t) \leq 0.01 \\ 1.01x_p(t) - 0.01 \ , & if \ \ 0.01 < x_p(t) \leq 1 \\ 1 \ , & if \ \ x_p(t) > 1 \end{cases} \tag{2.6}$$

and

$$\dot{k}_p^{id}(t) = P_r[\delta_p^{id}e_p(t)], \ \ 0 \leq k_p^{id}(0) \leq \hat{k}_p^{id}$$

where $P_r[.]$ is the projection operator defined as

$$P_r[\delta_p^{id}e_p(t)] = \begin{cases} \delta_p^{id}e_p(t) \ , & if \ \ 0 \leq k_p^{id}(t) \leq \hat{k}_p^{id} \\ \delta_p^{id}e_p(t) \ , & if \ \ k_p^{id}(t) = \hat{k}_p^{id}, e_p(t) \leq 0 \\ \delta_p^{id}e_p(t) \ , & if \ \ k_p^{id}(t) = 0, e_p(t) \geq 0 \\ 0 \ , & otherwise \end{cases} \tag{2.7}$$

where $\alpha_p^{id} > 0$, and $\delta_p^{id} > 0$, are design constants that affect the convergence rate and performance. Whenever, the premium service has excess capacity beyond that is required to maintain its QoS at the prescribed levels (as set by the queue length reference value), it offers it to the ordinary traffic service.

On the other hand, for the ordinary traffic service, using the leftover capacity after serving the premium class, and dynamically acting on the flow of the ordinary traffic into the network, the queue length of the ordinary traffic buffer is controlled. The ordinary traffic buffer queue length reference is also chosen by the network operator whereas the ordinary rate that is the control variable is calculated by the IDCC scheme. This rate is subsequently sent to the implicated sources to inform them about the maximum allowed rate they can transmit over the next control interval. In effect, attempting that the average buffer size $x_r(t)$ remains close to the desired value x_r^{ref}, the FFM-based ordinary traffic control strategy consists of first choosing $C_r^{id}(t)$ such that

$$C_r^{id}(t) = max[0, C_{serv} - C_p^{id}(t)] \tag{2.8}$$

and then by using the feedback linearization the ordinary controlled traffic rate $\lambda_{ir}^{id}(t)$ is chosen as

$$\lambda_{ir}^{id}(t) = max[0, min\{C_r^{id}(t), g^{id}(t)\}] \tag{2.9}$$

where

$$g^{id}(t) = C_r^{id}(t)(\frac{x_r(t)}{1+x_r(t)}) - \alpha_r^{id} e_r(t) \tag{2.10}$$

with $\alpha_r^{id} > 0$ is a design constant.

According to the best effort traffic, the corresponding control strategy utilizes any instantaneous leftover capacity to transmit a packet from the best effort buffer. This increases the network utilization during periods of insufficient supply of packets from both the premium and the ordinary traffic services.

Remark 2.2. *Reference [89] has introduced a significant contribution to the control of networks area specifically to the differentiated-services architecture. The proposed* *congestion control approach uses nonlinear control tools and adaptive concepts. Nevertheless, we believe that the fact that [89] does not consider the coupling effects may constitute a serious limitation of the proposed decentralized IDCC. In addition, whilst the control system may reach insensitivity to time-delays in the premium channels, we believe that the ordinary channels would be sensitive to time-delays especially given that the considered feedback linearization approach is not known to be robust. Finally, [89] does not consider feedback loops in a network environment, a situation that will change the dynamic interactions of such control systems. In this thesis, all these aspects will be investigated and addressed formally.*

2.5 Sliding Mode Variable Structure Control (SM-VSC)

2.5.1 Principles of SM-VSC

The existence of the sliding mode in a manifold is due to the discontinuous nature of the variable structure control which is a switching between two distinctively different system structures. Such a system is characterized by a potentially excellent performance that includes insensitivity to parameter variations and a complete disturbance rejection. It should be noted that the SM-VSC control design does not require an accurate model of the plant. Furthermore, as far as the MIMO (multi input/multi output) systems are concerned, a decomposition into a SISO (single input/single output) switching system permits one to simplify the design procedure. However, since the switching could not practically be implemented with an infinite frequency (as required theoretically for the ideal sliding mode case), the discontinuity generates a chattering in the control, which may excite high-frequency dynamics that are neglected in the model, and therefore can cause damage to the system.

Consider the nonlinear dynamic system in which the time variable is not explicitly shown, that is

$$\dot{x} = f_s(x) + g_s(x)U + \delta \tag{2.11}$$

where $x \in X$ is an open set of R^n, $f_s(x) = [f_{1s}, f_{2s}, \ldots, f_{ns}]^T$, $g_s(x) = [g_{1s}, g_{2s}, \ldots, g_{ns}]^T$ are vector fields defined on R^n with $g_s(x) \neq 0$, $\forall x \in X$, δ is the external disturbance, and the control is defined such that $U : R^n \to R$.

Assume a hypersurface $S = \{x \in \Re^n : S(x) = 0\}$ denoted as the "sliding surface" on which discontinuous control functions of the type

$$U = \begin{cases} U^+(x) \;\; if \;\; S(x) > 0 \\ U^-(x) \;\; if \;\; S(x) < 0 \end{cases} \tag{2.12}$$

makes the surface attractive to the representative point of the system such that it slides until the equilibrium point is reached. This behavior occurs whenever the well-known sliding condition $S\dot{S} < 0$ [116] is satisfied. Using the directional derivative $L_{f_s}S$, this condition may be represented as $\lim_{s \to 0^+} (L_{f_s + g_s U^+} S) < 0$ and $\lim_{s \to 0^-} (L_{f_s + g_s U^-} S) > 0$, or by using the gradient $\nabla$of S and the scalar product$< .,. >$ as $\lim_{s \to 0^+} (< \nabla S, f_s + g_s U^+ >) < 0$ and $\lim_{s \to 0^-} (< \nabla S, f_s + g_s U^- >) > 0$.

A geometric illustration of this behavior is shown in Figure 2.3, in which the switching of the vector fields occurs on the hypersurface S. In effect, depending on the system state with respect to the surface, the control is selected such that the vector fields converge to the surface. One of the most popular SM-VSC approaches is the well-known *equivalent control method*, which corresponds to the ideal sliding mode. To this equivalent control U_{eq}, one may add a discontinuous control component ΔU as follows,

$$U = U_{eq} + \Delta U \tag{2.13}$$

where the equivalent control component U_{eq} is derived such that the previously defined hypersurface is a local invariant manifold. In effect, when $S(x) = 0$,

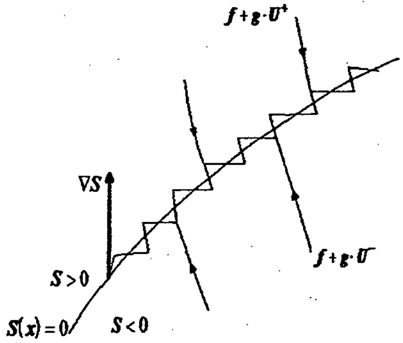


Figure 2.3: The geometric illustration of the sliding surface and the switching of the vector fields on the hypersurface S.

$L_{f_s+g_sU_{eq}}S =< \nabla S, f_s + g_sU_{eq} >= 0$, then

$$U_{eq} = -\frac{< \nabla S, f_s >}{< \nabla S, g_s >} = -\frac{L_{f_s}S}{L_{g_s}S} \tag{2.14}$$

where the second component corresponds to the discontinuous control so that $\Delta U = -bsgn(S)$, with the gain b chosen to be greater than the amplitude of the perturbation signal δ.

2.5.2 Sliding Mode Generalized Variable Structure Control (SM-GVSC) [44]

Recently, using powerful techniques based on linear algebra outstanding contributions to the control theory of dynamical systems have been made by M. Fliess [44]. Contrary to a classical variable structure (CVS) in which the equivalent control method may be utilized to any form of the model (2.1), the generalized variable structure (GVS) considers any system that is represented by both linear and nonlinear canonical forms known as *Generalized Canonical Forms* (GCF), which may explicitly exhibit time derivatives of the control input. The case in which the canonical form does not contain these derivatives is called the *degenerate* case. The presence of derivatives in the canonical forms is suitable for chattering alleviation purpose. These canonical forms might be obtained subject to the existence conditions of the differential primitive element for nonlinear systems, or cyclic element for linear systems in the context of differential algebra, by eliminating the state in the original Kalman state space representation. The elimination of the state in the model can be considered by using the Conte's theorem [34], [95].

Consider the nonlinear dynamical system

$$\begin{cases} \dot{x} = f_s(x) + g_s(x)u \\ y = h_s(x) \end{cases} \tag{2.15}$$

Assume that a strictly positive integer α_i exists such that $\alpha_i = d - r_d$, where r_d is the relative degree of the system with respect to the output y and d is such an integer that satisfies the following rank condition

$$rank \frac{\delta(h_s, \dot{h}_s, \ldots, h_s^{(d-1)})}{\delta x} = rank \frac{\delta(h_s, \dot{h}_s, \ldots, h_s^{(d-1)}, h_s^{(d)})}{\delta x} \tag{2.16}$$

The elimination of the state x in both state and output equations (2.15) will lead to the differential equation below

$$\zeta(y, \dot{y}, \ldots, y^{(d-1)}, y^{(d)}, u, \dot{u}, \ldots, u^{(\alpha_i)}) = 0 \tag{2.17}$$

By defining a new variable $z_i = y^{i-1}$ where $i = 1, \ldots, d$, and under the condition that the Jacobian $\frac{\delta(\zeta)}{\delta(y)^{(d)}}$ should be locally nonsingular, the transition from the implicit input-output representation to the locally explicit generalized observable canonical form (GOCF) given by (2.18) may be accomplished as follows

$$(GOCF) \begin{cases} \dot{z}_i = z_{i-1}; \; i = 1, 2, \ldots, n-1 \\ \dot{z}_n = \zeta(z, u, \dot{u}, \ldots, u^{(\alpha_i)}) \\ y = z_1 \end{cases} \tag{2.18}$$

Consider the reference tracking trajectory $W_R(t) = [y_R(t), \dot{y}_R(t), \ldots, y_R^{(n-1)}(t)]^T$, and define the tracking error vector $e(t) = [e_1, e_2, \ldots, e_n]^T = z(t) - W_R(t)$ such that

$e_1 = y - y_R$ and $e_i = e_1^{(i-1)} = z_i - y_R^{(i-1)}$ with $i = 2, \ldots, n$. The system (2.18) is now rewritten in the error state space as follows

$$\begin{cases} \dot{e}_i = e_{i+1}, \ (i = 1, \ldots, n-1) \\ \dot{e}_n = \zeta(W_R(t) + e(t), u, \dot{u}, \ldots, u^{(\alpha_i)}) - y_R^{(n)}(t) \\ e_1 = y - y_R \end{cases} \tag{2.19}$$

Finally, in the same error state space, assume a hypersurface $S(t) = e_n + \sum_{i=1}^{n-1} s_i e_i = 0$ on which the representative operating point of the control system slides until the origin whenever the sliding condition $S\dot{S} < 0$ holds [116]. The GVS control law may be derived either by solving this sliding condition or by applying some feedback linearization to the system (2.19). One type of feedback linearization may be stated as [44]

$$\zeta(z, u, \dot{u}, \ldots, u^{(\alpha_i)}) = \sum_{j=1}^{n} a_j z_j + \sum_{j=1}^{\alpha_i} b_j v_j \tag{2.20}$$

where $z = [z_1, \ldots, z_n]^T$, $v = [v_1, \ldots, v_{\alpha_i}]^T$ is the new input, and the coefficients a_i, b_i are chosen according to the stability conditions of the resulting linearized system. In our application we use another kind of feedback control that is based on the convergence of the hypersurface (ref to Chapter 6, Section 6.2.4).

2.6 Conclusion

The necessary tools for designing the proposed robust congestion control strategies are reviewed in this chapter. One starts by choosing an uncertain fluid flow model to design the model-based robust congestion control strategies. The combined control of bandwidth, flow rate, and connection admission concepts are discussed as studied in [91]. The concept of an integrated dynamic congestion controller (IDCC) as proposed in [89] is also reviewed and presented. Finally, the sliding mode variable structure control is debated in the context of the differential geometry for the classical variable structure and in the context of differential algebra for the generalized variable structure systems.

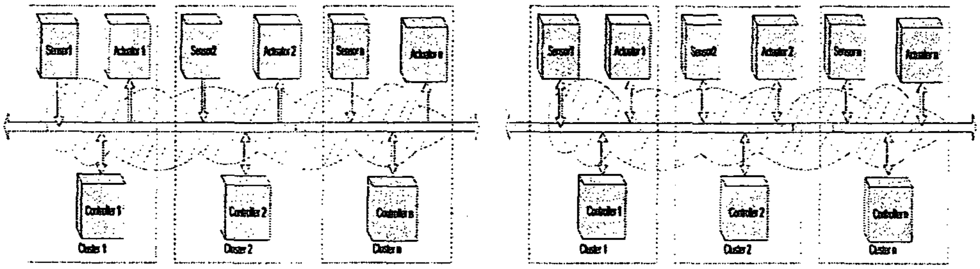
Chapter 3

Proposed Networked Unmanned Systems

3.1 Introduction

The aim of this research is to address the bandwidth allocation and congestion control problems for differentiated-services (DiffServ) networked unmanned systems (NUS). In NUS systems, asset tasks may be classified into three main classes, namely sensing, decision-making and actuating. Depending on the strategy that is adopted, NUS nodes are often organized in clusters [72] and in cooperation with other clusters they would autonomously or semi-autonomously achieve and contribute to accomplishing desired performance specifications and behaviors. By analogy to network control systems (NCS), we define in this chapter our NUS cluster as being a group of three nodes, namely, a sensor, a decision-maker, and an actuator.

To successfully carry out NUS missions, communication among assets need to be accomplished efficiently. In contrast with conventional communication networks, NUSs have specific features that may render the communication problems more complex. NUS are mainly characterized by (a) heterogeneity of their assets



(a) Generic setup of a NCS [122], [127]

(b) Generic setup of a NUS [18]

Figure 3.1: Generic setups of NCS and NUS, respectively.

in terms of size and resources, (b) topology that may be fully-connected [42], (c) real-time requirements that are imposed by delivery timelines of messages for the involved physical assets in uncertain environments, (d) time delays that may degrade performance of the closed-loop dynamics, (e) bandwidth constraints reflecting differences in speed of assets dynamics, and (f) protocol limitations with mainly compliance to wireless features of these networks [27], [59]. In this chapter, the clustering of a general NUS is considered and a generic setup is proposed as the NCS counterpart. Using the fluid flow model (FFM) discussed in Chapter 2, the overall differentiated-services (DiffServ) cluster model is derived.

3.2 Clustering of Networked Unmanned Systems

As mentioned in Chapter 1, the tasks dedicated to unmanned systems in order to perform Command, Control, Communication, Intelligence, Surveillance, and Reconnaissance (C3ISR) operations, can be classified as sensing, processing or decision making and actuating.

By analogy to networked control systems (NCS) [5] that are defined as control systems where sensors and actuators exchange data over a communication network (refer to Figure 3.1(a)), assets or nodes in NUS may: a) also receive information from other nodes, process them and send the decisions made to other nodes, b) interact with the physical environment, and c) be represented as a set of n clusters (see Figure 3.1(b)). Nevertheless, their difference lies in the fact that NUS is an interconnected network with bidirectional data and traffic flows. Indeed, data may be fed back to the upstream nodes (i.e. the senders) after certain processing by nodes and assets. This inherent positive feedback characteristic also known in the NCS literature as "closing the loop over the network" constitutes thus as one of the main challenges of NUS since it may lead, in the worst-case scenario, to instability of the network. Moreover, the nodes and the graphs are also not fixed, in the sense that they may be exchanged and grouped with other cluster nodes.

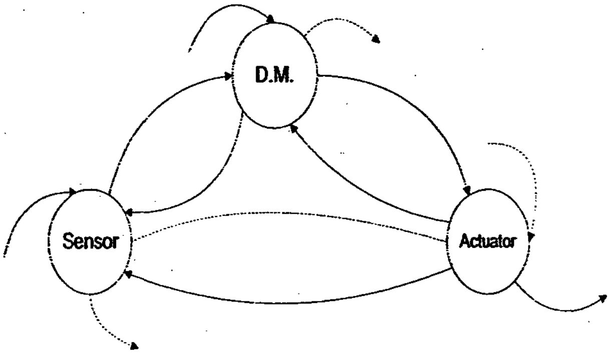


Figure 3.2: A single cluster module for a NUS obtained after clustering a large scale NUS that is shown in 3.1(b) [18].

3.3 General Networked Unmanned Systems with a Fully-Connected Topology

From the discussion in the above section, and after representing a NUS as an ensemble of clusters consisting of a sensor, a decision maker and an actuator, it is worth mentioning that these clusters can be interconnected differently leading to various topologies.

In communication networks, several topologies can be found such as star, ring,

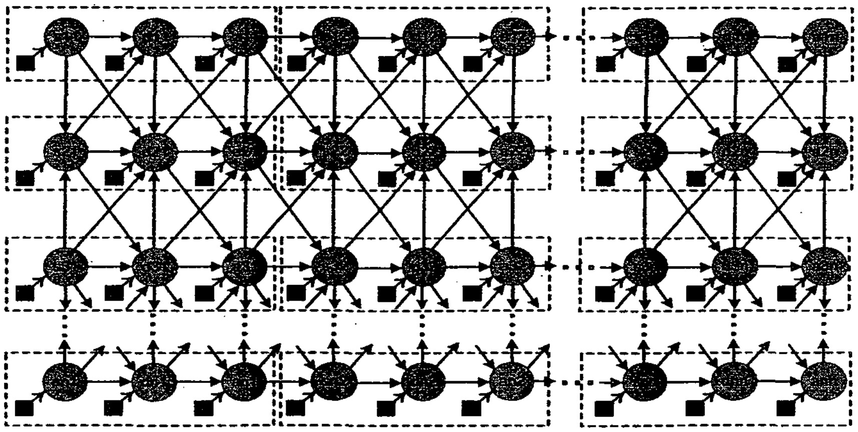


Figure 3.3: Proposed *nxn* nodes network.

mesh, and fully-connected. In the communication community, it is well-known that the mesh and the fully-connected topologies are the most complicated. In addition, when compared to the star and the ring topologies, the mesh topology is appropriate for distributed hop-by-hop control strategies. For example, a wireless mesh network topology that is called an *ad hoc multihop network* was successfully developed for industrial control and sensing at Massachusetts Institute of Technology (MIT) Media Lab. Even in harsh industrial environments, the mesh topology is able to ensure the reliability of the entire network [3], [4].

In agreement with the general network that is depicted by Figure 3.3, two topologies are considered in this thesis. The first topology as utilized in Chapter 8 to simulate an Internet network is basically a well-known mesh topology in which the traffic flow directions are set as follows: (a) the nodes are assumed to communicate in diagonal directions, and (b) the nodes located at the core of the network are assumed to be differently fed along the vertical directions, the nodes of the even rows are assumed to receive information flows while those of the odd rows are sending information flows. For example, by considering that *n* is an odd number, the nodes of the last row are hence sending vertically information to the nodes of the top row $n-1$. To each node of our network, one connects a source/user (denoted as U_h, $h = j_{11}, j_{12}, \ldots, j_{1n}, \ldots, j_{n1}, j_{n2}, \ldots, j_{nn}$ with $j = [s, d, a]$ where s, d, and a denote the sensor, the decision-maker, and the actuator, respectively.).

The second topology simulating a NUS network in Chapter 8 contains the interconnections characterizing the first topology as well as the interconnections (feedback traffic) between the cluster nodes, namely, the sensor (s_{ij}), the decision-maker (d_{ij}), and the actuator (a_{ij}). Note that in the network illustrated as a general NUS of nxn clusters by Figure 3.3, the connections characterizing the feedback traffic are not represented.

3.4 Differentiated-Services Cluster Definition

In the same spirit as the NUS shown in Figure 3.1(b), our network as represented in Figure 3.2 may be defined as one cluster with three interconnected nodes [14], namely a sensor, a decision-maker, and an actuator. The three nodes in a given cluster are primarily communicating with one another and ultimately communicate with other clusters. In Figure 3.2, the dashed lines represent the other connections that are not considered in this study.

The information types communicated among the cluster nodes could be considered as single traffic (Best Effort service) [18] or as multi services, namely a differentiated-services traffic [17, 19]. As illustrated in Figure 2.1 and according to the DiffServ Code Point (DSCP) [13], [85] and [89], our information traffic is generated at the edge of the network as three aggregates and then buffered in the node output port of the network core. In fact, each node output port consists of three physical or logical queues and consequently, the traffic is divided into three services, namely: the Premium service, the Ordinary service, and the Best Effort service [13], [85], [89]. The first two services are denoted in this thesis by p and

r, respectively. Between each source-buffer, a delay block is introduced to simulate and reflect any system inducing time-delay due to either propagation, transmission, processing delays, etc. Note that due to the round trip path of the flow control, the time-delay corresponding to the ordinary channel is two times the single trip time-delay of the Premium and the Best Effort channels as shown in Figure 2.1.

3.5 Cluster Model in the Open-Loop Control Configuration

It should be noted that since our goal is to propose and develop robust congestion control strategies, an uncertain model of the node, cluster, or network is sufficient for the purpose of our design. Our traffic and network are modeled using the Fluid Flow Model (FFM) approach [89], [98], [103] as discussed in Section 2. The model is much simpler than probabilistic models [115] and suitable for both decentralized and distributed control approaches (defined formally subsequently). It is well-known that among a number of formal models that may be used for describing the queueing systems, the FFM is one of the most intuitive and simple techniques that accurately models a wide range of networked systems [1]. The considered FFM has already been validated by several previous work [98], [103] and is of a low-order.

As illustrated in Figure 3.4 below, the fluid flow model of the l^{th} output port of the h^{th} node having dimension kxl can be described by the following state space representation

$$\begin{cases} \dot{x}_{l_h f}(t) = -C_{l_h f}(t)\dfrac{x_{l_h f}(t)}{x_{l_h f}(t)+1} + \lambda_{il_h f} \\ y_{l_h f}(t) = x_{l_h f}(t) \end{cases} \tag{3.1}$$

where $x_{l_h f}(t)$ denotes the length of the information queued in the buffer f of output port l_h, $C_{l_h f}(t)$ is the capacity of the queue server $l_h f$, and $\lambda_{il_h f}(t)$ is the rate of arriving customers at the buffer $l_h f$. The index $f = p, r$ indicates the Premium and the Ordinary services, respectively.

Since the FFM of equation (3.1) has $C_{l_h f}$ as the input (capacity), $x_{l_h f}$ as the state (queue length), y_f as the output, and $\lambda_{il_h f}$ as the stimuli (input traffic), then we may analyze its dynamics from the point of view of stability analysis after linearization. Indeed, around an equilibrium point $x_{l_h f_0}$, we proposed to linearize the FFM model (3.1) as given below

$$\dot{x}_{l_h f} = \frac{\lambda_{il_h f} - C_{l_h f}}{C_{l_h f}} x_{l_h f} + \lambda_{il_h f} \tag{3.2}$$

By now analyzing the ratio $\frac{\lambda_{il_h f} - C_{il_h f}}{C_{il_h f}}$, one may conclude that this system is stable whenever the condition $\lambda_{il_h f} < C_{l_h f}$ is satisfied, which is in fact obvious since the capacity should be larger than the input traffic to ensure the stabilization of the queue length. In fact, this system behaves as a tank with two valves for the incoming and the outgoing traffics. Simulation results have shown that if the capacity is not sufficiently large, the FFM model dynamics behaves as an integrator.

With reference to Figure 3.4, the installed switches on node channels permit any input to be channeled to any buffer. In other words, switches allow one to enable or disable any loop in the network. The switches S^s_k, S^d_{lk} and S^a_{lk} are introduced in the sensor, the decision maker and the actuator, respectively, to simulate any possible network topology and combination. It is assumed that the same switches are used for all services, namely the Premium, the Ordinary and the Best Effort traffics. Also, the indices s, d and a refer to the sensor, the decision maker, and the actuator, respectively. Three users/sources designated as U^s, U^d, U^a are connected to the respective cluster nodes.

The overall model for the cluster is described by introducing in each link a delay block $\tau^{hj}_{lk,f}$ to simulate the time-delays due to propagation, transmission, processing delays, and other factors. Moreover, to represent the actual processing of the traffic (data) that is taking place at each node (channel) a process function block $F^{hj}_{lk,f}$ is added in series with the block delay $\tau^{hj}_{lk,f}$. The process function $F^{hj}_{lk,f}$

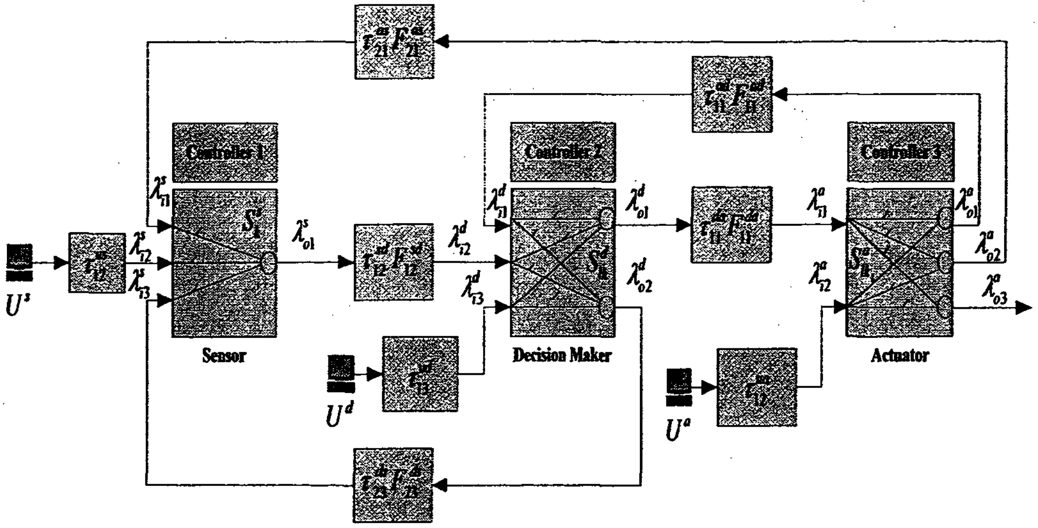


Figure 3.4: Detailed time delay dependent network cluster [14]

should be considered as modeling all various types of operations (such as image and voice compressions, dropping of packets, etc.).

Note that the traffic flow that is entering node j at the input k, f corresponds to the delayed traffic flow that is leaving node h from the output l, f with delay $\tau^{hj}_{lk,f}$, so that the relationship between these variables is written as

$$\lambda^{j}_{ik,f}(t) = F^{hj}_{lk,f}\lambda^{h}_{ol,f}(t - \tau^{hj}_{lk,f}) \tag{3.3}$$

where the indices i and o designate the node input and output, respectively, $j \in N_1$ with $N_1 = \{s, d, a\}$, $h \in N$ with $N = \{N_1, u^s, u^d, u^a\}$, $k = 1, 2, \ldots, m^j$, $l = 1, 2, \ldots, n^h$ with m^j denoting the number of inputs of node j and n^h denoting the number of outputs of node/user h, and $f = p, r$ indicates the Premium and the Ordinary services, respectively. Specifically, in the cluster considered in Figure 3.4, $m^s = m^d = n^a = 3$, $n^s = n^{u^s} = n^{u^d} = n^{u^a} = 1$, and $m^a = n^d = 2$.

Using the FFM model (3.1), the cluster that is shown by Figure 3.4 can be described as an open-loop sixth order nonlinear system governed by

$$\dot{x}^c_f(t) = -diag[C_{sf}\ C_{df}\ C_{af}]X^c_f + \phi^c_f\lambda^c_{o,f}(t) \tag{3.4}$$

where $X_f^c = [x_f^s \ (x_f^d)^T \ (x_f^a)^T]^T$ is the cluster state vector with indices c, s, d and a designating cluster, sensor, decision maker, and actuator, respectively, $x_f^d = [x_{1,f}^d \ x_{2,f}^d]^T$, $x_f^a = [x_{1,f}^a \ x_{2,f}^a \ x_{3,f}^a]^T$, C_{sf} a scalar, $C_{df} = diag[C_{1_d,f} \ C_{2_d,f}]$, $C_{af} = diag[C_{1_a,f} \ C_{2_a,f} \ C_{3_a,f}]$, and $\mathbb{F}_f^c = [\mathbb{F}_f^s \ (\mathbb{F}_f^d)^T \ (\mathbb{F}_f^a)^T]^T$ is the cluster occupancy vector with $\mathbb{F}_f^s = \frac{x_f^s(t)}{x_f^s(t)+1}$, $\mathbb{F}_f^d = [\frac{x_{1,f}^d(t)}{x_{1,f}^d(t)+1} \ \frac{x_{1,f}^d(t)}{x_{2,f}^d(t)+1}]^T$, and $\mathbb{F}_f^a = [\frac{x_{1,f}^a(t)}{x_{1,f}^a(t)+1} \ \frac{x_{1,f}^a(t)}{x_{2,f}^a(t)+1} \ \frac{x_{3,f}^a(t)}{x_{3,f}^a(t)+1}]^T$, $\phi_f^c = diag[S_{k_s}^s F_{k_s,f}^s \ \ S_{l_d k_d}^d F_{l_d k_d,f}^d \ S_{l_a k_a}^a F_{l_a k_a,f}^a] = diag[S_{k_s}^s \ \ S_{l_d k_d}^d \ \ S_{l_a k_a}^a] * diag[F_{k_s}^s \ F_{l_d k_d}^d \ F_{l_a k_a}^a]$ is the switching and the process matrices corresponding to the sensor, the decision-maker, and the actuator, respectively. Specifically, $S_{k_s}^s = [S_1^s \ S_2^s \ S_3^s]$, $S_{l_d k_d}^d = [(S_1^d)^T \ \ (S_2^d)^T]^T$, with $S_j^d = [S_{j1}^d \ \ S_{j2}^d \ \ S_{j3}^d] \ (j = 1, 2)$, $S_{l_a k_a}^a = [(S_1^a)^T \ \ (S_2^a)^T \ \ (S_3^a)^T]^T$ with $S_j^a = [S_{j1}^a \ \ S_{j2}^a] \ (j = 1, \ldots 3)$, $F_{k_s,f}^s = diag[F_{21,f}^{as} \ F_{12,f}^{us} \ F_{23,f}^{ds}]$, $F_{l_d k_d,f}^d = diag[F_{11,f}^{ad} \ F_{12,f}^{sd} \ \ F_{13,f}^{ud}]$, $F_{l_a k_a,f}^a = diag[F_{11,f}^{da} \ \ F_{12,f}^{ua}]$, $\lambda_{o,f}^c = [(\lambda_{o,f}^{2s})^T \ \ (\lambda_{o,f}^{2d})^T \ \ (\lambda_{o,f}^{2a})^T]^T$ is the output traffic vector with $\lambda_{o,f}^{2s} = [\lambda_{o2,f}^a(t - \tau_{21,f}^{as}) \ \lambda_{o1,f}^{u^s}(t - \tau_{12,f}^{u^s s}) \ \lambda_{o2,f}^d(t - \tau_{23,f}^{ds})]^T$, $\lambda_{o,f}^{2d} = [\lambda_{o1,f}^a(t - \tau_{11,f}^{ad}) \ \lambda_{o1,f}^s(t - \tau_{12,f}^{sd}) \ \lambda_{o1,f}^{u^d}(t - \tau_{13,f}^{u^d d})]^T$ and $\lambda_{o,f}^{2a} = [\lambda_{o1,f}^d(t - \tau_{11,f}^{da}) \ \lambda_{o1,f}^{u^a}(t - \tau_{12,f}^{u^a a})]^T$.

To model and simulate the network given by Figure 3.4 as realistic as possible, the following set of conditions and assumptions are taken into account and incorporated in our representations.

Assumption 3.1. *Each buffer of the network has a finite storage capacity. The queue length is bounded by* $0 \leq x_{bufsize,f} \leq x_{bufsizemax,f}$ *with* $f = p, r$. *In compliance with the priority of the stored aggregates and by denoting the Best Effort service by index b, the three buffers obey the following law,* $x_{bufsize,p} < x_{bufsize,r} << x_{bufsize,b}$.

Assumption 3.2. *In the network, all the nodes and users (sources) are assumed to be equidistant from one another so that the delay which is considered to be mainly dominated by the propagation factors can be taken as identical but time-varying and upper bounded by* $\tau_{lk,f}^{hj} = \tau(t) \leq \tau_{max}$ *for* $f = p, r$.

Assumption 3.3. *The server capacity* C_f *for each output port is assumed to be bounded so that* $0 \leq C_f \leq C_{serv}$.

3.6 Conclusion

In this chapter, a general networked unmanned system that can be seen as a mesh or fully-connected differentiated-services (DiffServ) network is introduced and characterized. Each NUS cluster is defined as a group of three nodes, namely, a sensor, a decision-maker, and an actuator. Using the fluid flow model, the overall dynamics of our cluster is derived as a time-delay dependent system. To simulate any possible network topology and configuration, a series of switches are introduced in each node. To model our network as realistic as possible, a set of conditions and assumptions are also enunciated.

Chapter 4

Bandwidth Allocation Control Strategies

4.1 Introduction

Bandwidth allocation constitutes one of the most crucial research issues in NUS systems. In fact, the dynamics of heterogeneous assets are mostly different in speed. Consequently, frequencies of message exchanges vary, and their bandwidths are to be allocated accordingly. On the other hand, the entire NUS dynamics is subject to continuous changes due to asset mobility and adding/deleting of assets. Therefore, for maximal utilization bandwidth allocation should be dynamic in order to adapt to the above changes. Furthermore, due to unpredictable and unknown statistical features of aggregate traffic, bandwidth allocation problems need to be addressed more reliably in order to provide quantitative packet-level QoS. Specifically, in NUS, by taking into account features such as closing loops over the network, dynamic changes in topologies, real time requirements, etc., the bandwidth allocation problem becomes more complex. In addition, NUS systems that will incorporate potentially adaptable and evolving structures, will make the closed-loop analysis extremely difficult.

In the literature several efforts have been deployed to address the bandwidth allocation problem and to propose adaptive algorithms that guarantee aggregate traffic packet-level QoS metrics, such as average queue length [90], loss [106], and delay [31,66].

As far as our NUS system that is depicted in Figure 3.4 is concerned, each network node buffer constitutes a potential bottleneck. To address the bandwidth allocation problem, the buffer queue length of each node output port can be controlled using the following methodologies, namely: a) a fully *centralized* control scheme in the sense of Definition 4.1 (given below) that is however not suitable for evolvable networks and cannot be scaled for large networks, b) a fully *decentralized* control scheme in the sense of Definition 4.2 that does not take into account the effects of other input traffics, and c) a *semi-decentralized* control scheme in the sense of Definition 4.3 which is more suitable for adaptive approaches since the estimated incoming traffic corresponds to the sum of all input traffics. Note that for the purpose of analysis and development of design methodologies the three nodes are represented on the basis of FFM which could represent a network cluster that consists of a sensor (e.g. a UAV), a decision maker (e.g. a C3) and an actuator (e.g. a UGV).

In this chapter and as illustrated in Figure 4.1, we investigate the bandwidth allocation problem by (a) showing that the dynamics of our time-delayed controlled NUS system could easily become unstable when a PID controller is implemented for controlling the buffer queue length of each output port buffer [14]; (b) implementing a nonlinear input-output linearization controller that has a complete knowledge about the input traffic and the delay, a scenario in which the buffer queue length control of the entire network is satisfactorily addressed [18]; (c) proposing a robust nonlinear feedback control approach that attempts to guarantee an average queue

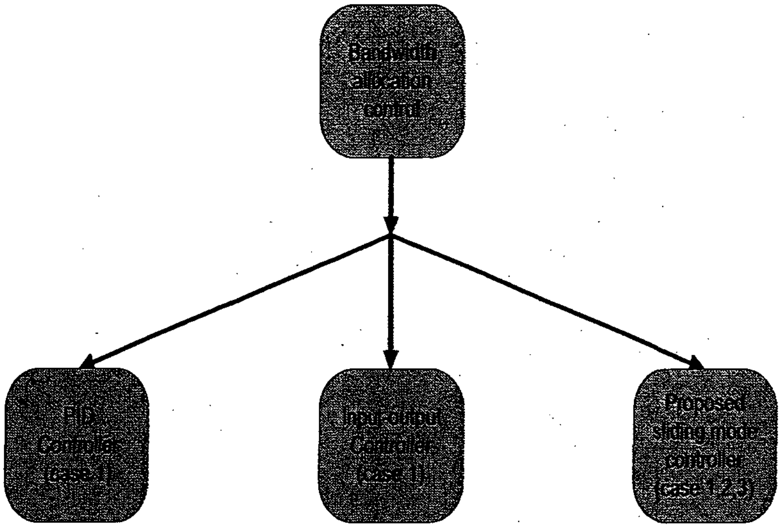

Bandwidth allocation control
PID Controller (case 1)
Input-output Controller (case 1)
Proposed sliding mode controller (case 1,2,3)

Figure 4.1: Chapter structure.

length in each bottleneck of our proposed three nodes cluster [18]; and (d) studying analytically and by simulation studies the effects of interaction of additional delayed feedback loops on performance and stability of the controlled system.

We now state our definitions needed in our subsequent discusssions.

Definition 4.1. *The control approach is called* **centralized** *if the overall* ***cluster*** *is controlled by the same controller that is designed on the basis of the entire cluster model.*

Definition 4.2. *The control approach is called* **decentralized** *if (a) each* ***output port buffer*** *of the cluster is controlled independently of other buffers controls, (b) each buffer controller is designed on the basis of the standard FFM given by equation* (3.1)*, and (c) any auxiliary input traffic is considered as a disturbance.*

Definition 4.3. *The control approach is called* **semi-centralized** *if (a) each* ***node*** *of the cluster is controlled independent of other nodes controls, and (b) each node controller is designed on the basis of the entire node model.*

Our proposed bandwidth allocation control approaches are model-based which implies that we will use buffer queue length and input traffic information. As far as the **incoming traffic** is concerned, it can be either **measured** or **estimated** according to the three cases considered below.

Case 1: The input traffic is assumed to be measured at the input port of the node and the channel between the input port and the controlled output port buffer is free of delay and attenuation.

Case 2: The input traffic is assumed to be measured at the input port of the node and an unknown time-delay and attenuation exist between the input port and the controlled output port buffer.

Case 3: The input traffic is estimated on the basis of the state of the controlled output port buffer.

4.2 Bandwidth Allocation Using PID Control

To demonstrate the difficulty of allocating the bandwidth to different assets of a NUS system, we start with the implementation of a standard PID controller. Our main objective is to tightly control the buffer queue length by implementing separately for each output port of a node an appropriate controller to satisfy design requirements. However, for sake of establishing a benchmark for comparative evaluation and analysis we start with the standard PID controller whose parameters are obtained for one buffer and then duplicated for all the other buffers. In order to achieve the desired control specifications for one buffer, that is, zero steady state error and no overshoot, the controller structure selected is a proportional and derivative control. Effectively, the above specifications are achievable only when no feedback is present in the network, that is the architecture considered is a cascade of three nodes in series. However, as shown through simulations the control objectives are not achievable as soon as feedbacks are introduced into the network.

In fact, the feedback can be seen as a positive reaction that can effectively change the system dynamics. Consequently, the PID parameters have to be re-adjusted in order to obtain the PID controller results that are presented below. Introducing the integral action leads in our case to an overshoot. The formal definition of the PID control structure may be expressed as

$$C(t) = k^p e(t) + k^i \int e(t)dt + k^d \frac{de(t)}{dt} \tag{4.1}$$

where $e(t) = x(t) - x_{ref}(t)$ refers to the tracking error and k^p, k^i, and k^d denote the proportional, the integral and the derivative design parameters. Our main aim in the next section is to evaluate the closed-loop control system performance and its sensitivity with respect to the time-delay in the unmanned network architecture as shown in Figure 3.4.

4.2.1 Simulation results of bandwidth allocation using PID control

In this section, simulation results for the above PID controller for each output port of the three-node unmanned system network constituting a cluster that is shown in Figure 3.4, are presented. In these results, we consider constraints on the buffer input flow as represented by $0 < \lambda^j_{ik}(t) \leq 4000$ packet/s, buffer queue length $0 < x^j_l(t) \leq 128$ packet and capacity of the queue server is limited to $0 < C^j_l(t) \leq 40000$ packet/s. In fact, these constraints follow assumptions that are stated in Chapter 3. Furthermore, as represented in each node of Figure 3.4, switches are installed on node channels to enable or disable any input to be forwarded to any other buffer by considering binary values. In other words, by properly selecting switches, one would be able to enable or disable any loop in the NUS.

Sensor switch is designated by the vector $S^s_k = [S^s_1\ S^s_2\ S^s_3]$, $S^d_{lk} = [S^d_{11}\ S^d_{12}\ S^d_{13};$

$S^d_{21}\ S^d_{22}\ S^d_{23}]$ is a 2x3 matrix that represents the decision maker switch, and the actuator switch is a 3x2 matrix denoted by $S^a_{lk} = [S^a_{11}\ S^a_{12};\ S^a_{21}\ S^a_{22};\ S^a_{31}\ S^a_{32}]$. In this section, we have chosen to present simulation results corresponding to three scenarios or configurations. We start with the first configuration as represented by $S^s_k = [0,1,0], S^d_{lk} = [0,1,0;0,0,0]$, and $S^a_{lk} = [0,0;0,0;1,0]$ which corresponds to a three nodes cascade (with no feedback traffic in the network). The second configuration introduces one feedback that connects the output port 1 of the actuator to the input 1 of the decision maker, which corresponds to $S^s_k = [0,\ 1,\ 0], S^d_{lk} = [1,\ 1,\ 0;\ 0,\ 0,\ 0]$, and $S^a_{lk} = [1,\ 0;\ 0,\ 0;\ 1,\ 0]$. The third configuration is denoted by $S^s_{lk} = [1,\ 1,\ 1]$, $S^d_{lk} = [1,\ 1,\ 0;\ 1,\ 1,\ 0]$, and $S^a_{lk} = [1,\ 0;\ 1,\ 0;\ 1,\ 0]$ which corresponds to the fully-connected configuration in which all the feedback loops are considered. As far as the time-delay τ^{hj}_{lk} is concerned, simulations are conducted with $\tau^{hj}_{lk} = 0.01$ s and $\tau^{hj}_{lk} = 0.1$ s. Furthermore, note that in the network illustrated in Figure 3.4, the process function F^{hj}_{lk} is set equal to 1. The discussion regarding the performance of the proposed PID controllers are provided below.

Let us start with the first configuration corresponding to the switching combinations $S^s_k = [0,1,0], S^d_{lk} = [0,1,0;0,0,0], S^a_{lk} = [0,0;0,0;1,0]$. The graphs shown in Figure 4.2 correspond to the case of the time-delay of 0.01 s. The graphs on the first row represent the actual buffer queue sizes for (a) the sensor, (b) the decision maker (port 1), and (c) the actuator (port 3), respectively, while the second row illustrates the graphs for their respective capacities. It can readily be seen that the three node states converge to their desired set point references with a good steady state but with excessive overshoots. Note that all the other ports are not of concern as they are not selected for this scenario. When the time-delay is increased for this configuration to 0.1 s, there is no major change other than the introduction of a very small delay in the response of the system.

The simulation results for the second configuration of switches, namely $S^s_k =$

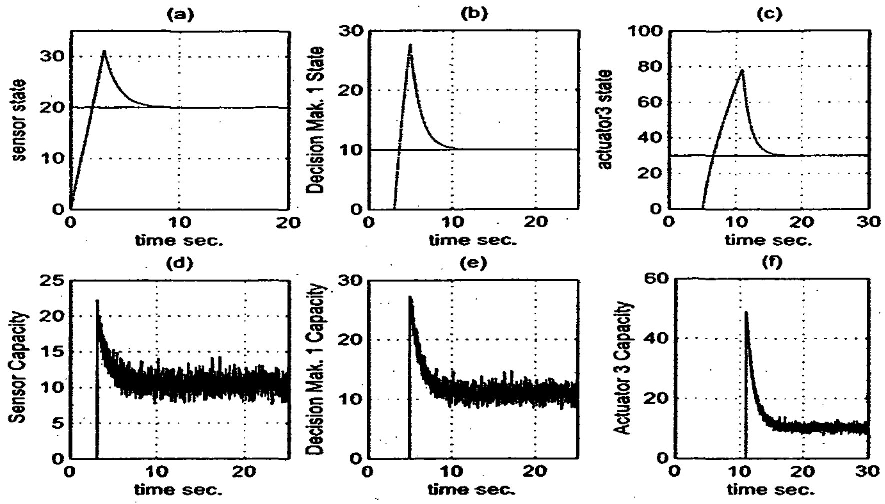


Figure 4.2: The PID-based bandwidth allocation control for configuration $S_k^s = [0,1,0], S_{lk}^d = [0,1,0;0,0,0]$, and $S_{lk}^a = [0,0;0,0;1,0]$ and the time-delay of 0.01s (buffer states: packet, capacity: packet/s).

$[0,1,0], S_{lk}^d = [1,1,0;0,0,0]$, and $S_{lk}^a = [1,0;0,0;1,0]$ introduces a single loop so that the port 1 of the actuator feeds back to the port 1 of the decision maker. Setting the time-delay initially to 0.01 s, the plots in Figure 4.3 illustrate the actual queue states of (a) the sensor, (b) the decision maker port 1, and (c) the actuator port 1, respectively. Since the sensor is not associated with the network loop, the dynamics of the sensor node is the same as that of the previous configuration, whereas the other two nodes produce an unacceptable transient performance (excessively large overshoots), although result in zero state errors. The second row of plots in Figure 4.3 depicts the respective control variables (capacities) of the nodes considered in the first row. Similar results are shown for the output port 3 of the actuator as illustrated in the third row.

In this scenario by increasing the delay to 0.1 s, we may observe according to the simulation results that are shown in Figure 4.4, that all the queues are relatively controlled up to the simulation time of 28 s where the control of the decision maker (Figure 4.4-b) and the actuator (Figure 4.4-c-d) fails. This phenomena can

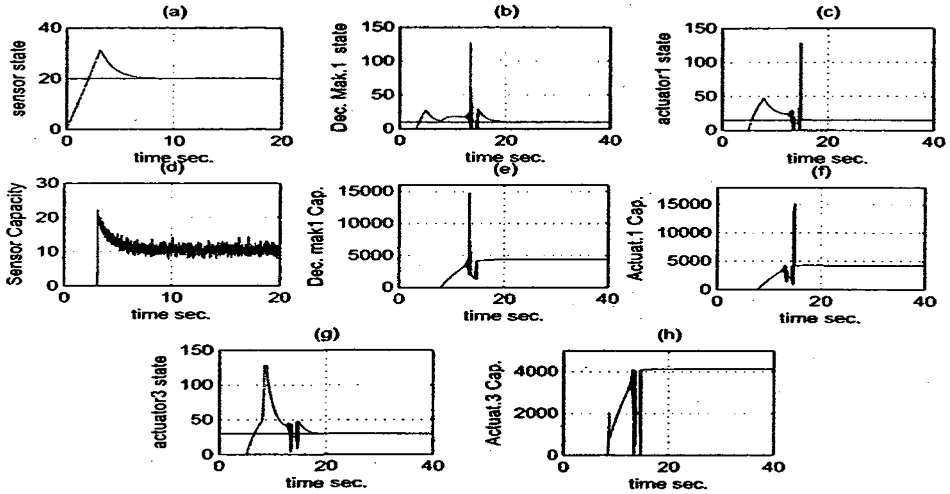


Figure 4.3: The PID-based bandwidth allocation control for configuration $S^s_k = [0,1,0], S^d_{lk} = [1,1,0;0,0,0]$, and $S^a_{lk} = [1,0;0,0;1,0]$ and the time-delay of 0.01s (buffer states: packet, capacity: packet/s).

be explained by noting that the positive feedback introduced by the loop integrates indefinitely the input flows and that implies that the capacities (graphs e, f, h) increase until a bifurcation occurs in the control system for which the system dynamic becomes unstable.

The third configuration that is considered for the PID controller is $S^s_{lk} = [1,1,1], S^d_{lk} = [1,1,0;1,1,0]$, and $S^a_{lk} = [1,0;1,0;1,0]$ which corresponds to invoking all the loops (fully-connected network). Setting the delay to 0.01 s, one may observe that for all the queue states illustrated in Figure 4.5, and the rows 1 and 3, the resulting overshoots in the transient behavior is excessively large, although good steady state errors are achieved. On the other hand, rows 2 and 4 show each capacity of the respective queueing state as presented in the previous rows. Furthermore, for the this configuration, increasing the time-delay to 0.1 s does not have a substantial effect on the control behavior. Nevertheless, a small delay occurs (the delay is of the order of 0.22 s magnitude for the port 3 of the actuator). The corresponding results are not included here.

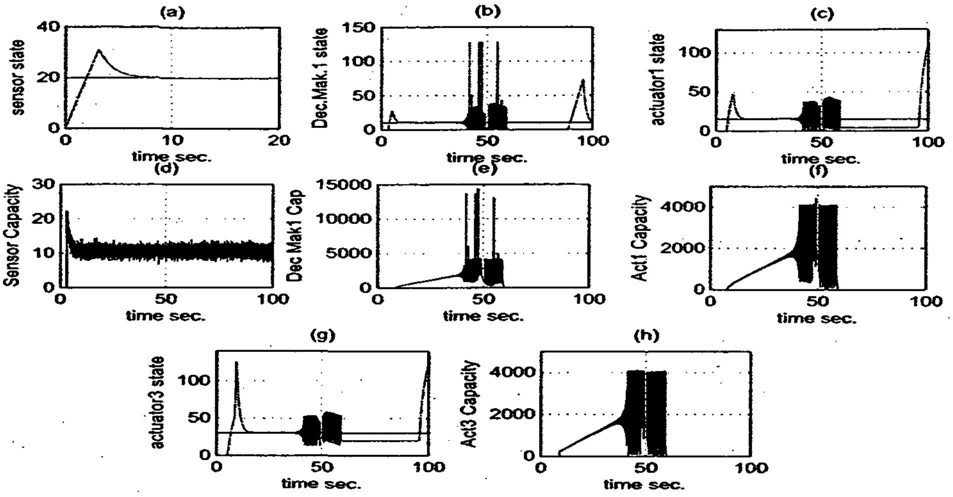


Figure 4.4: The PID-based bandwidth allocation control for configuration $S_k^s = [0,1,0], S_{lk}^d = [1,1,0;0,0,0]$, and $S_{lk}^a = [1,0;0,0;1,0]$ and the time-delay of 0.1s (buffer states: packet, capacity: packet/s).

4.3 Bandwidth Allocation Control using Input-Output Linearization Approach

In the previous section we have shown that the bandwidth allocation problem could not be satisfactorily addressed by a standard PID controller. It would be interesting to investigate instead a nonlinear control approach such as the input-output linearization control even though it is well-known that this method does not enjoy very good robustness properties.

At the output port of each node, an input-output feedback linearization based controller is now designed and implemented. The controller is derived on the basis of the first-order nonlinear model of equation (3.1). Consider that the capacity $C(t)$ is the control variable, and $x(t)$ is the output to be controlled. Input-output linearization of the simple first-order nonlinear model implies that the corresponding relative degree is one and there is no internal dynamics. To stabilize the resulting integrator, a simple output feedback may be used. As the controller design procedure

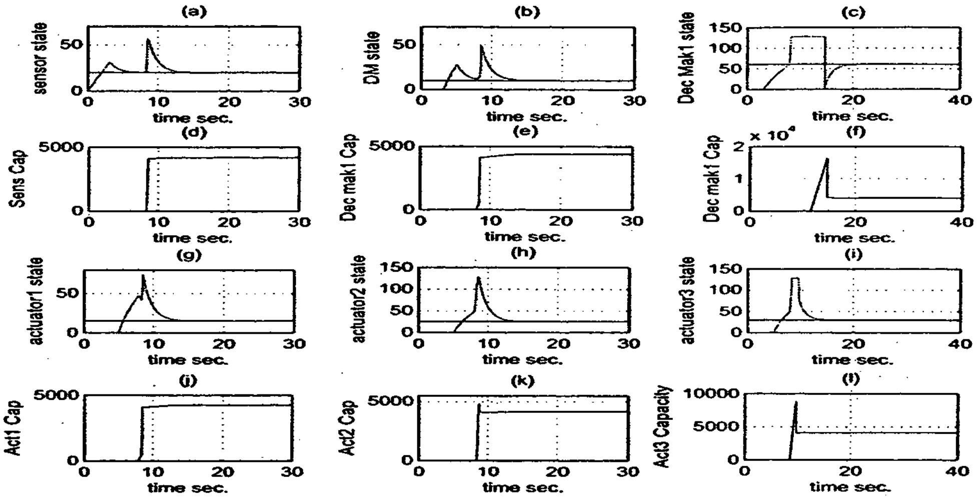


Figure 4.5: The PID based bandwidth allocation control for configuration $S^s_{lk} = [1,1,1], S^d_{lk} = [1,1,0;1,1,0]$, and $S^a_{lk} = [1,0;1,0;1,0]$ and the time-delay of 0.01s (buffer states: packet, capacity: packet/s).

is the same for all buffers (no coupling is taken into account in the control design), let us without loss of any generality, determine the controller corresponding to the sensor buffer. Namely,

$$\dot{x}^s(t) = -C^s(t)\frac{x^s(t)}{x^s(t)+1} + F^s_k S^s_k \lambda^s_{ik} \tag{4.2}$$

where $\lambda^s_{ik} = [\lambda^s_{i1}\lambda^s_{i2}\lambda^s_{i3}]^T$ so that the input-output feedback linearization controller is given by

$$C^s(t) = [F^s_k S^s_k \lambda^s_{ik}(t) + K_{io}(x^s(t) - x^s_{ref}(t))]\frac{x^s(t)+1}{x^s(t)} \tag{4.3}$$

where K_{io} is the controller gain to be specified. In the next section, simulation results that utilize the above nonlinear controller are provided.

4.3.1 Simulation results of bandwidth allocation control using input-output linearization approach

In order to compare the PID-based simulation results with those that are obtained based on the input-output feedback linearization, we need to maintain the same simulation conditions, mainly in terms of the noted constraints, switching combinations, propagation delays, and the random input flow characteristics. However, the gain of the input-output feedback linearization controller is set to 80 and the process function (attenuation) is set equal to $F_{lk}^{hj} = 1$. Following the same procedure as in the PID case, we start with the first configuration $S_k^s = [0,1,0], S_{lk}^d = [0,1,0;0,0,0], S_{lk}^a = [0,0;0,0;1,0]$. Setting the time-delay to 0.01 s, the plots in Figure 4.6-a, b, and c (same graph order as in the PID case) show clearly that the desired specifications in terms of the steady state errors and the overshoots are satisfied. Indeed, as illustrated by graphs (a, b, and c), the buffer queue lengths converge to their corresponding set points. This conclusion is also valid even when the time-delay is set to 0.1 s (not shown here).

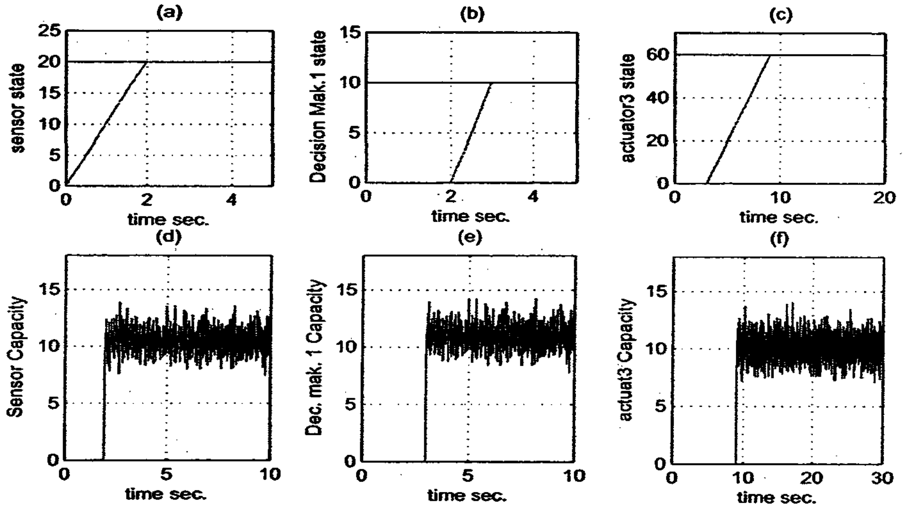


Figure 4.6: The I/O linearization-based bandwidth allocation control for the configuration $S_k^s = [0,1,0]$, $S_{lk}^d = [0,1,0;0,0,0]$, and $S_l^a = [0,0;0,0;1,0]$ and the time-delay of 0.01s (buffer states: packet, capacity: packet/s).

Consider now the second switching configurations $S_k^s = [0,\ 1,\ 0]$, $S_{lk}^d = [1,\ 1,\ 0;\ 0,\ 0,\ 0]$, $S_l^a = [1,\ 0;\ 0,\ 0;\ 1,\ 0]$. Contrary to the PID scenario, according to simulation results shown in Figure 4.7-a, b, c, and g, we may observe that the nonlinear controller satisfies and maintains the imposed design specifications for both the steady state error and the overshoot. Note also that even for a time-delay of 0.1 s, the effects on the control behavior is less significant (Figure 4.8-a, b, c, g) in comparison to the effects that they have on the PID control approach that clearly fails. Hence, we may conclude that the nonlinear controller is capable of coping with the network dynamics for which a PID controller cannot when the propagation delay is increased.

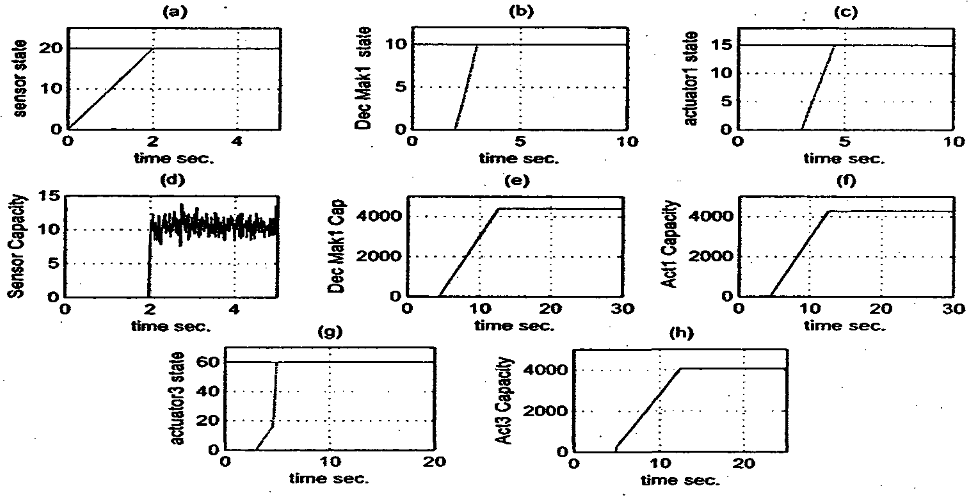


Figure 4.7: The I/O linearization-based bandwidth allocation control for configuration $S^s_k = [0,1,0]$, $S^d_{lk} = [1,1,0;0,0,0]$, and $S^a_l = [1,0;0,0;1,0]$ and the time-delay of 0.01s (buffer states: packet, capacity: packet/s).

The last configuration to be considered and utilized for comparison with the PID strategy is $S^s_{lk} = [1,1,1], S^d_{lk} = [1,1,0;1,1,0], S^a_{lk} = [1,0;1,0;1,0]$. According to the simulation results shown in Figure 4.9 (same order of the graphs as in the PID case) we may conclude that when the feedback linearization controller is utilized one is capable of achieving the desired performance specifications in both the transient as well as the steady state values. Eventhought the results for the time-delay of 0.1 s are not shown here, we may also state that the feedback linearization technique is still less sensitive to the time-delay when compared to the PID control strategy.

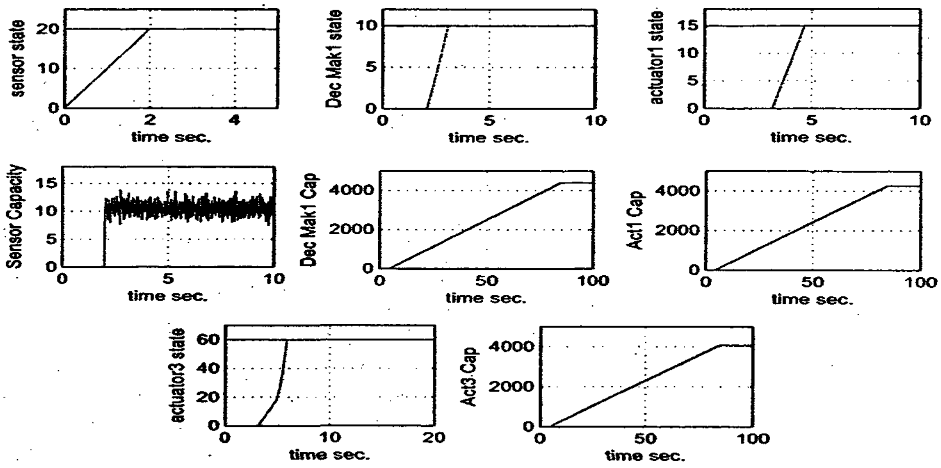


Figure 4.8: The I/O linearization-based bandwidth allocation control for configuration $S^s_k = [0,1,0]$, $S^d_{lk} = [1,1,0;0,0,0]$, and $S^a_l = [1,0;0,0;1,0]$ and the time-delay of 0.1s (buffer states: packet, capacity: packet/s).

4.4 Proposed Robust Bandwidth Allocation Control Strategy

In this section, we propose a robust bandwidth allocation control strategy by using sliding mode machinery. As shown in Figure 4.10, our proposed control strategy is investigated according to the three cases stated in Section 1. The investigations are also subject to the switching configurations of the network as well as to the node mobilty.

4.4.1 Proposed controller design

In each node of our network, the output ports (buffers) constitute potential bottlenecks. However, as shown in Figure 3.4, for each node a controller is assigned to regulate the queue length of each buffer by allocating appropriate bandwidth to the output port server. To design our proposed robust controller, let us rewrite the fluid flow model (3.1) corresponding to the l^{th} buffer in the error state space after defining

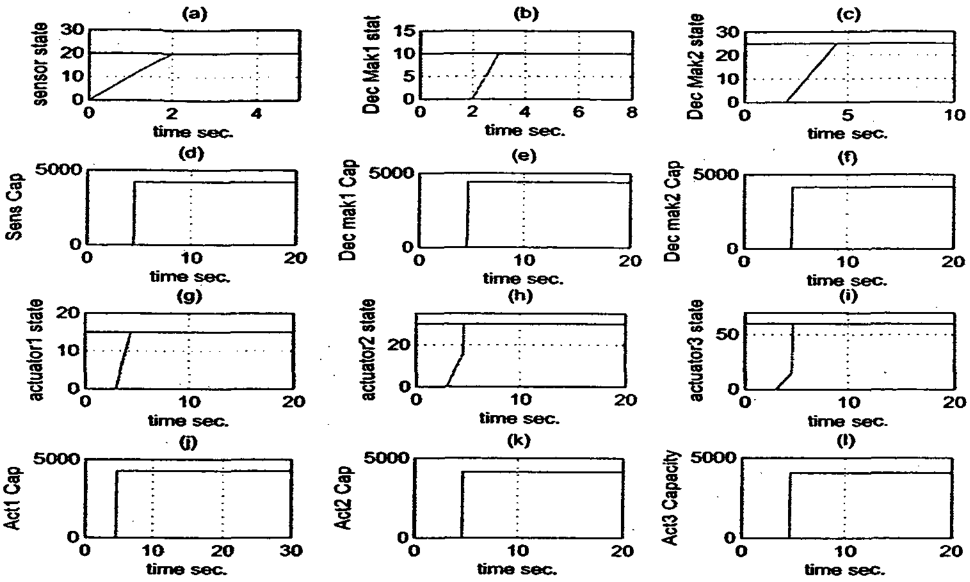


Figure 4.9: The I/O linearization-based bandwidth allocation control for configuration $S_k^s = [1,1,1]$, $S_{lk}^d = [1,1,0;1,1,0]$, and $S_l^a = [1,0;1,0;1,0]$ and the time-delay of 0.01s (buffer states: packet, capacity: packet/s).

the state error variables $e_l(t) = x_l(t) - x_{l_{ref}}(t)$, $\dot{e}_l(t) = \dot{x}_l(t) - \dot{x}_{l_{ref}}(t)$ where $x_{l_{ref}}$ denotes the desired reference queue length. By substituting these in model (3.1), the following error dynamics given by equation (4.4) is obtained

$$\begin{cases} \dot{e}_l(t) = -C_l(t)\dfrac{e_l(t) + x_{l_{ref}}(t)}{e_l(t) + x_{l_{ref}}(t) + 1} + \lambda_{il}(t) - \dot{x}_{l_{ref}}(t) \\ y_l(t) = e_l(t) + x_{l_{ref}}(t) \end{cases} \tag{4.4}$$

Considering that the resulting FFM in the error state space representation is given by equation (4.4), the objective of our proposed bandwidth allocation control is to ensure performance and robustness of the closed-loop system to uncertainties and unmodeled dynamics [16,116]. This will be accomplished provided that for the above system a sliding mode exists in some neighborhood of the linear switching surface that is defined as $S_l(t) = e_l(t)$. Using the sliding mode feedback control (refer to Chapter 2), the bandwidth allocation control law is now derived and expressed as

$$C_l(t) = E_l^{-1}(t)[\mu_l e_l(t) + \gamma_l sgn(e_l(t)) + \lambda_{il}(t) - \dot{x}_{l_{ref}}(t)] \tag{4.5}$$

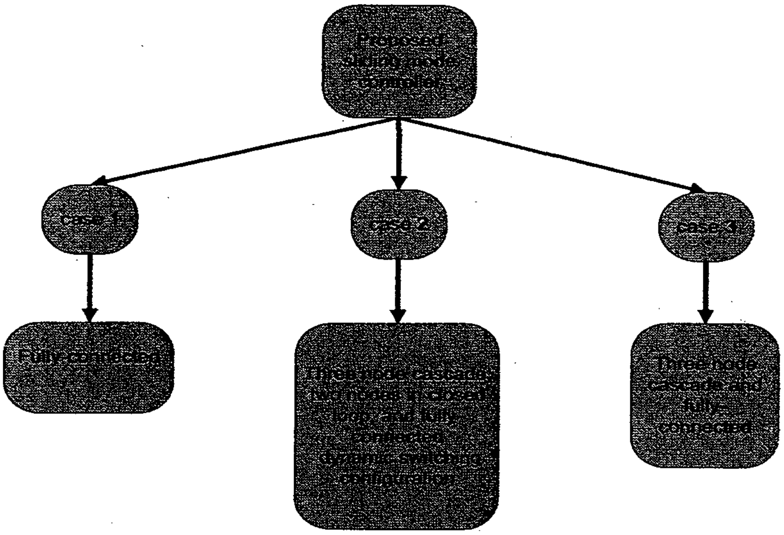

Proposed sliding mode controller
case 1
case 2
case 3
Fully connected
Three node cascade: two nodes in closed loop, and fully connected dynamic switching configuration
Three node cascade and fully-connected

Figure 4.10: The recapitulative investigations on the proposed sliding mode bandwidth allocation control.

where $E_l(t) = \frac{e_l(t)+x_{l_{ref}}(t)}{e_l(t)+x_{l_{ref}}(t)+1}$ is the l^{th} buffer occupancy, μ_l, γ_l are the design parameters assigned to buffer l and

$$sgn(e_l) = \begin{cases} 1 \ if \ e_l > 0 \\ 0 \ if \ e_l = 0 \\ -1 \ if \ e_l < 0 \end{cases} \tag{4.6}$$

The resulting error dynamics of the l^{th} closed-loop controlled buffer is now given by

$$\dot{e}_l(t) = -\mu_l e_l(t) - \gamma_l sgn(e_l(t)) \tag{4.7}$$

Theorem 4.1. *Given the l^{th} buffer of the cluster as shown in Figure 3.4 and described by the FFM model* (4.4), *the feedback control law governed by equation* (4.5) *guarantees the asymptotic stability of the error dynamics* (4.7) *provided that μ_l and γ_l are selected as positive design parameters.*

Proof: Choose a candidate Lyapunov function as $V(e_l, t) = \frac{e_l^2(t)}{2}$ such that for all $e_l(t) \neq 0 \Rightarrow V(e_l) > 0$. The time derivative of V along the trajectories of (4.7) can be obtained as follows

$$\dot{V}(e_l,t) = -\mu_l e_l^2(t) - \gamma_l e_l(t) sgn(e_l(t)) \tag{4.8}$$

It is clear that since $sgn(e_l(t)) = \frac{e_l(t)}{|e_l(t)|}$, $\dot{V}(e_l)$ is strictly negative definite whenever the design parameters μ_l and γ_l are selected to be positive, and hence $e_l(t) \rightarrow 0$ as $t \rightarrow \infty$. This completes the proof. ▲

Remark 4.1. *The control law given in Theorem 4.1 holds as (a) a decentralized control approach in the sense of Definition 4.2 when applied to our network in which no input coupling is considered , and (b) a semi-decentralized control approach in the sense of Definition 4.3 when applied to our network with input coupling considerations.*

4.5 Time-Delay Dependent Control System Dynamics

It is widely known that in closed-loop control systems, time-delay and latency may degrade dynamic performance and in the worst case may lead to instability of the system. In communication networks, time-delays are usually considered as a combination of different delays that can be caused by propagation of messages (cell/packet) through communication medium, transmission time, processing time, and so on. Therefore, time-delays effects can be crucial for NUS systems since their assets may be far from each other (links to base station or satellite) while their topology that is characterized by several loops, interactions, and coupling effects may be mesh or even fully-connected.

In the first section of this chapter as well as in [14], we have evaluated by simulations the effects of constant time-delay on the control dynamics of the network (see Figure 3.4) when buffer queue lengths are controlled with a standard PID and Input-Output linearization nonlinear control approaches and have found that the nonlinear approach can handle the buffer queue length control of the entire cluster. First, the incoming traffic to the buffer and subsequently the time-delays are considered as measurable as defined in Case 1 of Section 4.1, and second time-delay dependent control dynamics is not derived analytically and the stability conditions are not presented. In this section, we implement our proposed robust bandwidth allocation control strategy given by equation (4.5) on the delay dependent NUS system of Figure 3.4 for different types of input traffic and topologies. The input traffic is considered in compliance with Cases 1, 2 and 3 that are defined at the beginning of this chapter.

4.5.1 Measurable input traffic according to Case 1

4.5.1.1 Fully-connected switching configuration

Assume that the attenuation F_{lk}^{hj} in the network is fixed and is known *a priori* by the controller and the input traffic $\lambda_{il}(t)$ is measurable as defined in Case 1 so that the dynamics of the overall fully-connected cluster may be described as,

$$\dot{e}(t) = -Ae(t) - Bsgn(e(t)), where \quad \begin{cases} e(t) = & [e^s(t)\ e_1^d(t)\ e_2^d(t)\ e_1^a(t)\ e_2^a(t)\ e_3^a(t)]^T \\ A = & diag[\mu^s\ \mu_1^d\ \mu_2^d\ \mu_1^a\ \mu_2^a\ \mu_3^a] \\ B = & diag[\gamma^s\ \gamma_1^d\ \gamma_2^d\ \gamma_1^a\ \gamma_2^a\ \gamma_3^a] \end{cases} \tag{4.9}$$

Lemma 4.1. *Given that the input traffic is measurable as defined in Case 1, using the control law given by Theorem 4.1 as a semi-decentralized control approach in the sense of Definition 4.3, then the closed-loop dynamics described by* (4.9) *for our time-delayed fully-connected cluster is asymptotically stable provided that the design parameters* μ_l^j *and* γ_l^j *are selected to be positive.*

Proof: Since matrices A and B in the error dynamics of equation (4.9) are diagonal and whose elements are the positive design parameters μ_l^j and γ_l^j, then according to Theorem 4.1, the dynamics of the fully-connected cluster is asymptotically stable. ▲

4.5.1.2 Simulation results for a fully-connected configuration using our proposed bandwidth allocation controller

To confirm the above analytical results, simulations are conducted on our network and the results are depicted in Figure 4.11. For all the simulation results presented below the sampling time is set to $T_s = 1$ ms and the proposed robust controller parameters are set to $\mu_l^j = 100$ and $\gamma_l^j = 0.1$. The external input traffic to the overall cluster corresponds to $\lambda_{i2}^s(t)$ which is a random signal of mean 1000 packet/s (this mean is set to 10 packet/s in one simulation corresponding to two nodes switching configuration) and variance of 5 packet/s, and the cluster output is $\lambda_{o3}^a(t)$. Any node input traffic is assumed to be measurable according to Case 1 and is bounded such that $0 \leq \lambda_{ik}^j(t) \leq 4000$ packet/s. The buffer capacities are finite and their queues are bounded such that $0 \leq x_l^j(t) \leq 128$. Finally, the constraint on the maximum available service capacity is $0 \leq C_l^j(t) \leq 40000$ packet/s, whereas the process function (attenuation) is considered to be equal to 1. Furthermore, as represented in each node of Figure 3.4, switches are installed on node channels to enable or disable any input to be forwarded to any buffer by considering binary values.

In the first simulation, we consider a fully-connected cluster that corresponds to the switching configuration $S_{lk}^s = [1, 1, 1], S_{lk}^d = [1, 1, 0; 1, 1, 0], S_{lk}^a = [1, 0; 1, 0; 1, 0]$. By setting buffer queue lengths references to $x_{ref}^s = 20$ packet, $x_{1_{ref}}^d = 10$ packet, $x_{2_{ref}}^d = 25$ packet, $x_{1_{ref}}^a = 15$ packet, $x_{2_{ref}}^a = 30$ packet and $x_{1_{ref}}^d = 60$ packet, Figure 4.11 shows buffer queues step responses and their respective capacities for the time-delay τ_{lk}^{hj} equal to both 1 ms and 60 ms. The graphs in the first and the third rows of Figure 4.11 represent the actual buffer queue sizes and the desired references for

(a) sensor, (b), (c) decision maker (ports 1, 2), and (g), (h), (i) actuator (ports 1, 2, 3), respectively. In the second and fourth rows, the graphs illustrate capacities for (d) sensor, (e), (f) decision-maker (ports 1, 2), and (j), (k), (l) actuator (ports 1, 2, 3), respectively. It can readily be seen that for the time-delay of 1 ms the three node states converge to their desired set point references with a good settling time and steady state error performance. This convergence takes place even if we increase the time-delay to 60 ms but results in the settling time as expected to be greater. One may thus conclude that the time-delay dependent cluster control dynamics remains stable for a measurable delay and latency.

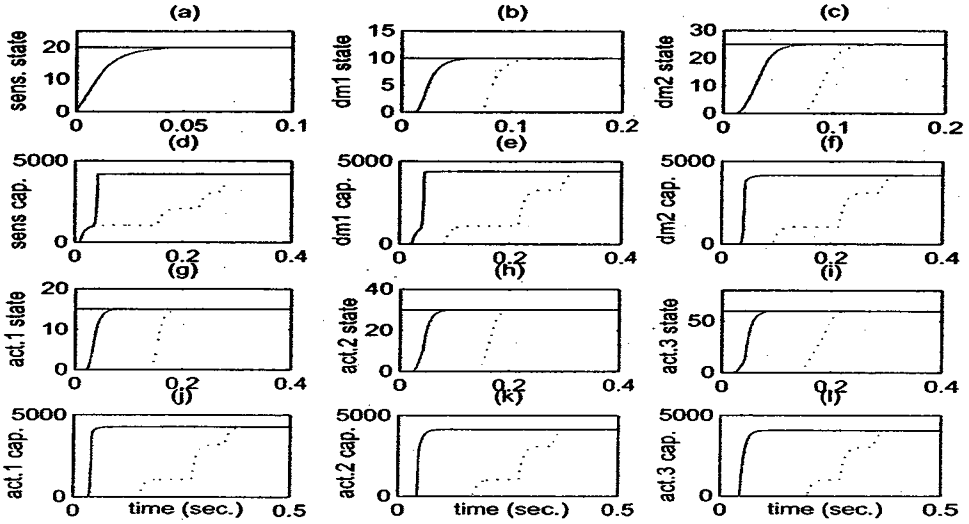


Figure 4.11: The queue length (packet) and control (capacity: packet/s) behavior for a fully-connected network in presence of measurable delay (solid line: 1 ms, dashed line: 60 ms) .

4.5.2 Measurable input traffic according to Case 2

For a more realistic situation, we now assume that the buffer input traffic is measurable according to Case 2 which also means that the controller has no *a priori* knowledge about the time-delay except its upperbound and the attenuation in the network. Let us consider different switching configurations.

4.5.2.1 Three nodes cascade switching configuration

The simplest configuration set for the cluster is a three nodes cascade (without any loop).

Assumption 4.1. *Given that the delay τ^{hj}_{lk} and the attenuation F^{hj}_{lk} corresponding to any link between two nodes of our proposed cluster as shown in Figure 3.4 are time-invariant, the time-delayed signals within the cluster may be approximated by their first-order linear functions as shown below*

$$\lambda^{j}_{ik}(t) = F^{hj}_{lk}\lambda^{h}_{ol}(t-\tau^{hj}_{lk}) = F^{hj}_{lk}\lambda^{h}_{ol}(t) - F^{hj}_{lk}\tau^{hj}_{lk}\dot{\lambda}^{h}_{ol}(t)$$

4.5.2.2 Derivation of the three nodes cascade error dynamics based on Assumption 4.1

To derive the error dynamics of our network, let us consider that the delayed signal can be approximated by their first-order linear function as stated in Assumption 4.1 and the signum function $sgn(e)$ is replaced by the well-known saturation function $sat(e)$ given by equation 4.10 in order to avoid derivatives of $sgn(e)$.

$$sat(e) = \begin{cases} 1 & e > 0 \\ \dfrac{e}{\varepsilon} & -\varepsilon < e < \varepsilon \\ -1 & e < 0 \end{cases} \tag{4.10}$$

Therefore, the time-delay dependent control network dynamics is governed by,

$$\dot{e}(t) = -Ae(t) - Bsat(e(t)) - C\frac{d(sat(e))}{dt} + D, \quad where \tag{4.11}$$

$$A = \begin{pmatrix} \mu^s & 0 & 0 \\ a_{21} & \mu^d_1 & 0 \\ a_{31} & a_{32} & \mu^a_3 \end{pmatrix}, \; B = \begin{pmatrix} \gamma^s & 0 & 0 \\ b_{21} & \gamma^d_1 & 0 \\ b_{31} & b_{32} & \gamma^a_3 \end{pmatrix}, \; D = \begin{pmatrix} 0 \\ d_2 \\ d_3 \end{pmatrix} \tag{4.12}$$

$$
with \begin{cases}
a_{21} = F_{12}^{sd}\tau_{12}^{sd}(\mu^s)^2 + (F_{12}^{sd} - 1)\mu^s \\
a_{31} = -F_{12}^{sd}F_{11}^{da}(\mu^s)^2\mu_1^d\tau_{12}^{sd}\tau_{11}^{da} + \tau_{11}^{da}F_{11}^{da}\mu^s(\mu^s - \mu_1^d(F_{12}^{sd} - 1)) + (F_{11}^{da} - 1)\mu^s \\
a_{32} = F_{11}^{da}\tau_{11}^{da}(\mu_1^d)^2 + (F_{11}^{da} - 1)\mu_1^d \\
b_{21} = -F_{12}^{sd}\tau_{12}^{sd}\mu^s + F_{12}^{sd} - 1 \\
b_{31} = \mu^s F_{11}^{da}\tau_{11}^{da}(F_{12}^{sd}\tau_{12}^{sd}\mu_2^d - 1) - F_{12}^{sd}\tau_{12}^{sd}\mu_2^d(F_{12}^{sd} - 1) + F_{11}^{da} - 1 \\
b_{32} = -F_{12}^{sd}\tau_{12}^{sd}\mu_2^d + F_{11}^{da} - 1 \\
d_2 = (F_{12}^{sd} - 1)\lambda_{i2}^s - F_{12}^{sd}\tau_{12}^{sd}\dot{\lambda}_{i2}^s \\
d_3 = -F_{12}^{sd}\tau_{12}^{sd}\mu_2^d[(F_{12}^{sd} - 1)\lambda_{i2}^s - F_{12}^{sd}\tau_{12}^{sd}\dot{\lambda}_{i2}^s] + (F_{11}^{da} - 1)\lambda_{i2}^s - F_{11}^{da}\tau_{11}^{da}\dot{\lambda}_{i2}^s
\end{cases}
\tag{4.13}
$$

Lemma 4.2. *Provided that Assumption 4.1 holds and the input traffic is measurable according to Case 2, using the control law given by Theorem 4.1 as a decentralized control approach in the sense of Definition 4.2, then the closed-loop dynamics that is described by equation* (4.11) *for a time-delayed cluster configured as a three nodes cascade is stable if the design parameters* μ_l^j *and* γ_l^j *are selected to be positive.*

Proof: From the error dynamics governed by equation (4.11), one may observe that (a) matrices A and B given by equation (4.12) are triangular and whose eigenvalues are the diagonal terms which are the positive design parameters μ_l^j and γ_l^j, and (b) the term with $\frac{d(sat(e))}{dt}$ given as

$$
\dot{sat}(e) = \begin{cases} \dfrac{1}{\varepsilon} & -\varepsilon < e < \varepsilon \\ 0 & elsewhere \end{cases} \tag{4.14}
$$

is always bounded, therefore the closed-loop error dynamics (4.11) is stable, provided that D is bounded. ▲

4.5.2.3 Simulation results for a cascade configuration using our proposed bandwidth allocation controller

The above stability result may be illustrated through simulation results that are obtained for the switching configuration $S_k^s = [0,1,0], S_{lk}^d = [0,1,0;0,0,0], S_{lk}^a = [0,0;0,0;1,0]$. This is a cluster of three nodes cascaded without any feedback loop. This also corresponds to a configuration in which buffers of the sensor, the decision-maker (port1), and the actuator (port3) are involved. Keeping the same simulation conditions as in the previous simulation studies, and assuming that the input traffic is measurable according to Case 2, the first row of Figure 4.12 shows the graphs of the buffer queue length step responses for the sensor (a), the decision maker (b), and the actuator (c), respectively, and their respective capacities are depicted in the second row for the sensor (d), the decision-maker (e), and the actuator (f), respectively. One can conclude that according to graphs of Figure 4.12 when the time-delay is set to 1 ms (solid line), the buffer queue lengths converge to their respective references with a good transient and steady state error performance and even for the time-delay of 60 ms (dashed line) except a delay in the transmission, other properties such as convergence and stability are maintained which confirms our analytical synthesis results.

4.5.2.4 Derivation of the three nodes cascade error dynamics by relaxing Assumption 4.1

When relaxing Assumption 4.1 and by considering the delays is identical so that $\tau_{hj}^{lk} = \tau$ and the process function $F_{hj}^{lk} = F$, the time-delay dependent network dynamics is now governed by,

$$\dot{e} = -A_1 e + A_2 e(t-\tau) - A_3 sgn(e) + A_4 sgn(e(t-\tau)) - B_1 \lambda_{i2}^s + B_2 \lambda_{i2}^s(t-\tau) \quad (4.15)$$

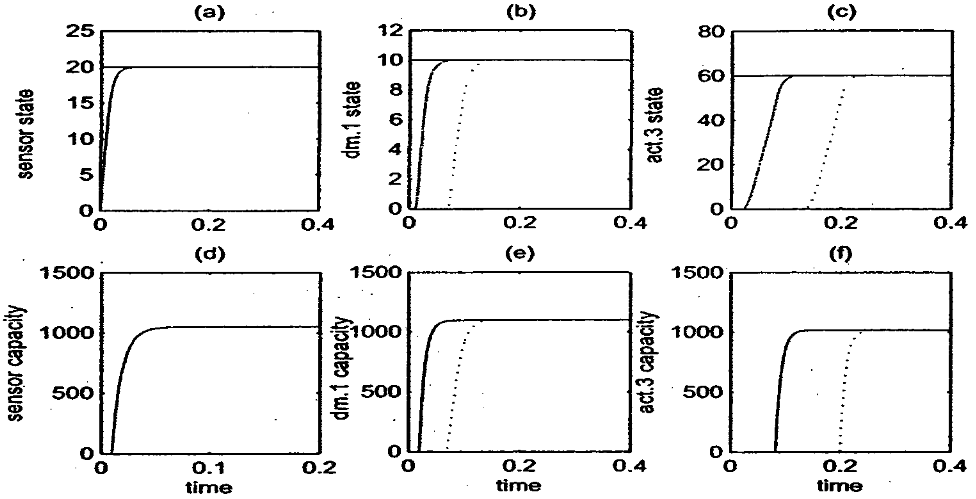


Figure 4.12: The queue length (packet) and control (capacity: packet/s) behavior for a three nodes cascade network in presence of non-measurable delay (solid line: 1 ms and dashed line: 60 ms).

where $e = [e^s \; e_1^d \; e_3^a]^T$, λ_{i2}^s is the external input traffic generated by the source/user,

$$A_1 = \begin{pmatrix} \mu^s & 0 & 0 \\ \mu^s & \mu_1^d & 0 \\ \mu^s & \mu_1^d & \mu_3^a \end{pmatrix}, \; A_2 = \begin{pmatrix} 0 & 0 & 0 \\ F\mu^s & 0 & 0 \\ F\mu^s & F\mu_1^d & 0 \end{pmatrix}, \; A_3 = \begin{pmatrix} \gamma^s & 0 & 0 \\ \gamma^s & \gamma_1^d & 0 \\ \gamma^s & \gamma_1^d & \gamma_3^a \end{pmatrix} \tag{4.16}$$

$$A_4 = \begin{pmatrix} 0 & 0 & 0 \\ F\gamma^s & 0 & 0 \\ F\gamma^s & F\gamma_1^d & 0 \end{pmatrix}, B_1 = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, B_2 = \begin{pmatrix} 0 \\ F \\ F \end{pmatrix} \tag{4.17}$$

Lemma 4.3. *Provided that the input traffic is measurable according to Case 2, using the control law given by Theorem 4.1 as a semi-decentralized control approach in the sense of Definition 4.3, then the closed-loop dynamics described by equation* (4.15) *for a time-delayed cluster configured as a three nodes cascade is asymptotically stable independent of the delay if the design parameters μ_s, μ_d and μ_a are selected to be positive and $\lambda_{i2}^s \equiv 0$, that is.*

Proof: Using the frequency domain approach analysis for systems with commensurate delays, the stability of our error dynamics (4.15) can be determined by its characteristic function if $\lambda_{i2}^s \equiv 0$, that is

$$P(s; e^{-\tau}) = det(sI - A_1 - A_2 e^{-\tau s})$$

$$= det \begin{pmatrix} s+\mu_s & 0 & 0 \\ \mu_s(1-Fe^{-\tau s}) & s+\mu_d & 0 \\ \mu^s(1-Fe^{-\tau s}) & \mu_d(1-Fe^{-\tau s}) & s+\mu_a \end{pmatrix} = (s+\mu_s)(s+\mu_d)(s+\mu_a) \quad (4.18)$$

Since, the characteristic function (4.18) has no zero or root in the right-half plane, the error dynamics (4.15) is stable independent of the delay if the design parameters μ_s, μ_d and μ_a are selected to be positive. This completes the proof. ▲

4.5.2.5 Two nodes switching configuration in closed-loop

Although we have shown that the cascade system with unknown time-delay is stable, it is well-known that the closed-loop system with unknown time-delays may become unstable. In fact, this is the case for time-delay dependent networked control systems known also as networked sense-and-respond systems [6]. To study this inherent feature, let us consider in our network a switching combination that configures a simple closed-loop time-delayed control system. Specifically, let us set the switching configuration as $S_k^s = [0,1,1]$, $S_{lk}^d = [0,0,0;0,1,0]$, $S_l^a = [0,0;0,0;0,0]$, which corresponds to nodes sensor and decision-maker with a lower loop. Combination of these switches introduces a single loop so that output port 2 of the decision-maker feeds back to the third input of the sensor node. By using the FFM model (4.4) and provided that Assumption 4.1 holds and the process function is set equal to 1, one obtains the error dynamics having the form (4.11), where now

$$A = \begin{pmatrix} \frac{2\mu^s}{2-\mu^s\tau_{23}^{ds}} & 0 \\ -\frac{(\mu^s)^2\tau_{12}^{sd}}{2-\mu^s\tau_{23}^{ds}} & \mu_2^d \end{pmatrix}, B = \begin{pmatrix} \frac{2\gamma^s}{2-\mu^s\tau_{23}^{ds}} & 0 \\ -\frac{\mu^s\tau_{12}^{sd}\gamma^s}{2-\mu^s\tau_{23}^{ds}} & \gamma_2^d \end{pmatrix}, C = \begin{pmatrix} -\frac{\tau_{23}^{ds}}{2-\mu^s\tau_{23}^{ds}} & 0 \\ \frac{\mu^s\tau_{23}^{ds}\tau_{12}^{sd}\gamma^s}{4-2\mu^s\tau_{23}^{ds}} + \frac{\tau_{12}^{sd}\gamma^s}{2} & 0 \end{pmatrix} \quad (4.19)$$

and $D = [d_1 \;\; d_2]^T$ where $d_1 = \frac{\tau_{23}^{ds}}{2-\mu^s\tau_{23}^{ds}}\dot{\lambda}_{i1}^s$ and $d_2 = -(\frac{\mu^s\tau_{23}^{ds}\tau_{12}^{sd}}{4-2\mu^s\tau_{23}^{ds}} + \frac{\tau_{12}^{sd}}{2})\dot{\lambda}_{i1}^s$

Lemma 4.4. *Provided that Assumption 4.1 holds and the input traffic is measurable according to Case 2, using the control law given by Theorem 4.1 as a semi-decentralized control approach in the sense of Definition 4.3, then there exists an upper bound* $\tau_{12}^{sd} < \frac{2}{\mu^s}$ *such that our time-delayed cluster configured as a two nodes (sensor and decision-maker) with a lower loop is stable if the design parameters* μ *and* γ *are selected to be positive and* D *is a bounded signal.*

Proof: From the error dynamics that is described by (4.11) with matrices A, B, C and D given by equation (4.19), one may observe that (a) matrices A and B are triangular and whose eigenvalues are the diagonal elements which are positive if μ^s, γ^s, μ_2^d and γ_2^d are positive and $\tau_{12}^{sd} < \frac{2}{\mu^s}$, (b) the term with $\frac{d(sat(e))}{dt}$ is always bounded according to equation (4.14), therefore the closed-loop error dynamics is stable. This completes the proof. ▲

4.5.2.6 Simulation results for a two nodes configuration using our proposed bandwidth allocation controller

Using the two nodes configuration, two external input traffics are considered in the simulation studies below. First, we consider the traffic with a mean of 1000 packet/s and second, we consider the traffic with a mean of 10 packet/s. Note also that the traffic is assumed to be measurable according to Case 2 and the process function $F_{lk}^{hj} = 1$. As shown in Figure 4.13, the two buffer queue states to be controlled and their respective capacities are represented in the first column for a time-delay of 1 ms and the external input traffic mean of 1000 packet/s, and in the second column when the time delay is increased to 60 ms keeping the input traffic mean to 1000, and finally in the third column when the input traffic mean is set to 10 packet/s and keeping the time-delay to 60 ms. One can observe that for both traffic means the control dynamics is sensitive to the time-delay. Indeed, by comparing for example the graphs (a) and (b) for the sensor and (g) and (h) for the decision-maker it is clear that the performance is degraded corresponding to the delay of 60 ms. This sensitivity may also be seen for input traffic mean of 10 packet/s through oscillatory graphs (c) and (i) corresponding to the sensor and the decision maker, respectively.

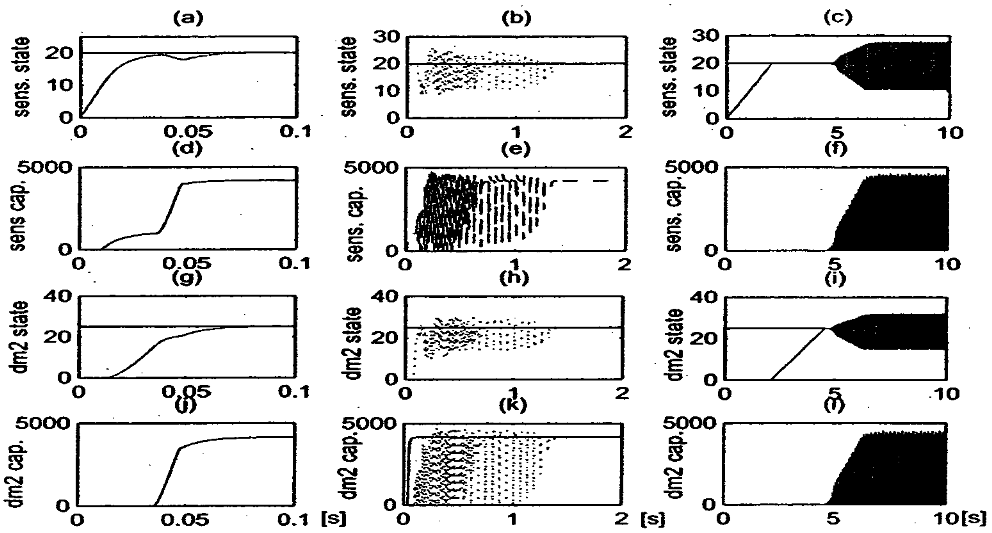


Figure 4.13: The queue length (packet) and control (capacity: packet/s) behavior for a two nodes switching configuration; columns 1 & 2: (random traffic mean of 1000 packet/s, solid line: 1 ms delay and dashed line: 60 ms delay); column3: (random traffic mean of 10 packet/s, 60 ms delay).

4.5.2.7 Fully-connected switching configuration

The last switching configuration that we consider here for measurable input traffic (as defined in Case 2) corresponds to $S^s_{lk} = [1,1,1], S^d_{lk} = [1,1,0;1,1,0], S^a_{lk} = [1,0;1,0;1,0]$, which represents a fully-connected network. Assuming that both τ^{hj}_{lk} and F^{hj}_{lk} are *unknown* by the controller but time-invariant and bounded, the overall closed-loop error dynamics of the fully-connected cluster has now the same form as in equation (4.11) with $B = A = -A_1^{-1}A_2$, where

$$A_1 = \begin{pmatrix} H_1 & H_2 & H_3 & H_4 & H_5 & 0 \\ H_6 & H_7 & H_8 & H_9 & H_{10} & 0 \\ H_{11} & H_{12} & H_{13} & H_{14} & H_{15} & 0 \\ D_{43}\tau_{11}^{da}\mu^s & D_{39}\tau_{11}^{da}\mu_1^d & D_{42}\tau_{11}^{da}\mu_2^d & (1 + D_{38}\tau_{11}^{da}\mu_1^a) & D_{40}\tau_{11}^{da}\mu_2^a & 0 \\ D_{49}\tau_{11}^{da}\mu^s & D_{45}\tau_{11}^{da}\mu_1^d & D_{48}\tau_{11}^{da}\mu_2^d & D_{44}\tau_{11}^{da}\mu_1^a & (1 + D_{46}\tau_{11}^{da}\mu_2^a) & 0 \\ D_5\tau_{11}^{da}\mu^s & D_{51}\tau_{11}^{da}\mu_1^d & D_{54}\tau_{11}^{da}\mu_2^d & D_{50}\tau_{11}^{da}\mu_1^a & D_{52}\tau_{11}^{da}\mu_2^a & 1 \end{pmatrix}, \tag{4.20}$$

with $H_1 = 1 + (D_{12}\tau_{21}^{as} + D_{13}\tau_{23}^{ds})\mu^s$, $H_2 = (D_4\tau_{21}^{as} + D_5\tau_{23}^{ds})\mu_1^d$, $H_3 = (D_{10}\tau_{21}^{as} + D_{11}\tau_{23}^{ds})\mu_2^d$, $H_4 = (D_1\tau_{21}^{as} + D_3\tau_{23}^{ds})\mu_1^a$, $H_5 = (D_6\tau_{21}^{as} + D_7\tau_{23}^{ds})\mu_2^a$, $H_6 = (D_{24}\tau_{11}^{ad} + D_{25}\tau_{12}^{sd})\mu^s$, $H_7 = 1+(D_{16}\tau_{11}^{ad}+D_{17}\tau_{12}^{sd})\mu_1^d$, $H_8 = (D_{22}\tau_{11}^{ad}+D_{23}\tau_{12}^{sd})\mu_2^d$, $H_9 = (D_{14}\tau_{11}^{ad}+D_{15}\tau_{12}^{sd})\mu_1^a$, $H_{10} = (D_{18}\tau_{11}^{ad}+D_{19}\tau_{12}^{sd})\mu_2^a$, $H_{11} = (D_{36}\tau_{11}^{ad}+D_{37}\tau_{12}^{sd})\mu^s$, $H_{12} = (D_{28}\tau_{11}^{ad}+D_{29}\tau_{12}^{sd})\mu_1^d$, $H_{13} = 1 + (D_{34}\tau_{11}^{ad} + D_{35}\tau_{12}^{sd})\mu_2^d$, $H_{14} = (D_{26}\tau_{11}^{ad} + D_{27}\tau_{12}^{sd})\mu_1^a$, $H_{15} = (D_{30}\tau_{11}^{ad} + D_{31}\tau_{12}^{sd})\mu_2^a$, A_2 is a (6x6) matrix that is delay independent and which is a function of the attenuation F_{lk}^{hj} and switch S_{lk}^{hj}, and D is a vector that depends on the time-delay τ_{lk}^{hj}, input traffic λ_{i1}^s, F_{lk}^{hj} and S_{lk}^{hj}.

Given that the vector D is assumed to be bounded, stability of the fully connected error dynamic may be considered by using the same approach as that used in the previous cases. Matrix A given by equation (4.20) should be Hurwitz to ensure stability of the fully-connected dynamics. The proposed approach for deriving stability conditions through the upper bound of the delay τ_{lk}^{hj} may result in conservative conditions.

4.5.2.8 Simulation results for a fully-connected configuration using our proposed bandwidth allocation controller

For a fully-connected switching configuration and assuming that the input traffic is measurable according to Case 2, we have conducted two simulation studies. First, we start with the same initialization as in the above simulations to show through transient responses (refer to Figures 4.14, and 4.15) the behavior of our fully-connected control dynamics and then we follow with another situation in which initializations are changed dynamically in order to show that our proposed control strategy can handle a wide range of operating conditions with different initial conditions (refer to Figures 4.16, and 4.17). Note also that Figures 4.14, and 4.17 illustrate in the same order the graphs of the buffer queue states and capacities as in Figure 4.11.

By setting the same reference points as in the simulations corresponding to Figure 4.11, Figure 4.14 shows the results for delays of 1 ms (solid line) and 8 ms (dashed line). One can clearly observe that for 8 ms delay even if the performance is degraded (specially graph (g)) the convergence and tracking is still achieved. Note that beyond delay of 8 ms, the stability of the closed-loop system fails at least for one buffer queue as shown in Figure 4.15. Indeed, setting delay to 60 ms, as shown in Figure 4.15 the queue lengths corresponding to decision maker port 1 (plots b and e) fail and other transient responses are oscillatory with plots a, c, and g showing overshoots.

The last simulations are conducted for different operating points with queue references set to $x^s_{ref} = \{20, 40\}$ packet, $x^d_{1_{ref}} = \{10, 25\}$ packet, $x^d_{2_{ref}} = \{25, 10\}$ packet, $x^a_{1_{ref}} = \{15, 30\}$ packet, $x^a_{2_{ref}} = \{30, 20\}$ packet, and $x^a_{3_{ref}} = \{60, 30\}$ packet. Corresponding to the delay of 1 ms, Figure 4.16 shows that the proposed robust bandwidth allocation controller converges with good performance for a wide range of operating points. By increasing the delay to 60 ms, one may observe in Figure 4.17 that except for the buffer control of the decision-maker (port1) which fails other buffer control dynamics converge to their desired references with degraded step responses. In addition, performance of all their second step responses are as good as the case with 1 ms delay. This behavior is due to the initial conditions that are non-zero for the second step inputs.

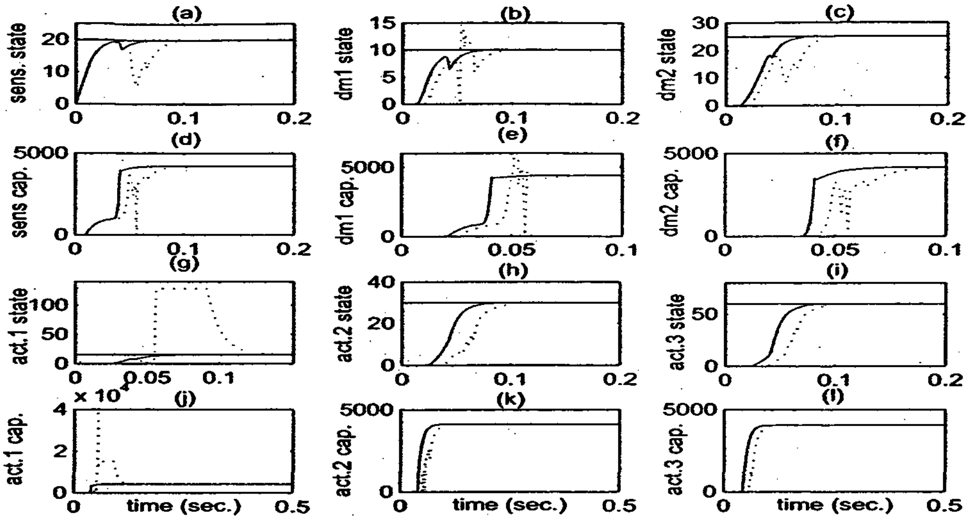


Figure 4.14: The queue length (packet) and control (capacity: packet/s) behavior for a fully connected configuration in presence of a non-measurable time-delay (solid line: 1 ms and dashed line: 8 ms).

4.5.3 Input traffic estimated according to Case 3

After considering Cases 1 and 2 corresponding to measurable input traffic, let us now consider Case 3 in which the input traffic is estimated. Note also that in the derivations below (a) Assumption 4.1 is now not required, (b) time-delays are considered to be identical so that $\tau_{hj}^{lk} = \tau$, and (c) the process function $F_{hj}^{lk} = F$.

4.5.3.1 Three nodes cascade switching configuration

The time-delay dependent network dynamics by incorporating the dynamics of the input traffic estimations now is governed by

$$\dot{\eta} = -\mathbb{A}_1\eta + \mathbb{A}_2\eta(t-\tau) - \mathbb{A}_3 sgn(e) + \mathbb{A}_4 sgn(e(t-\tau)) + \mathbb{B}_1 U^s(t-\tau) \qquad (4.21)$$

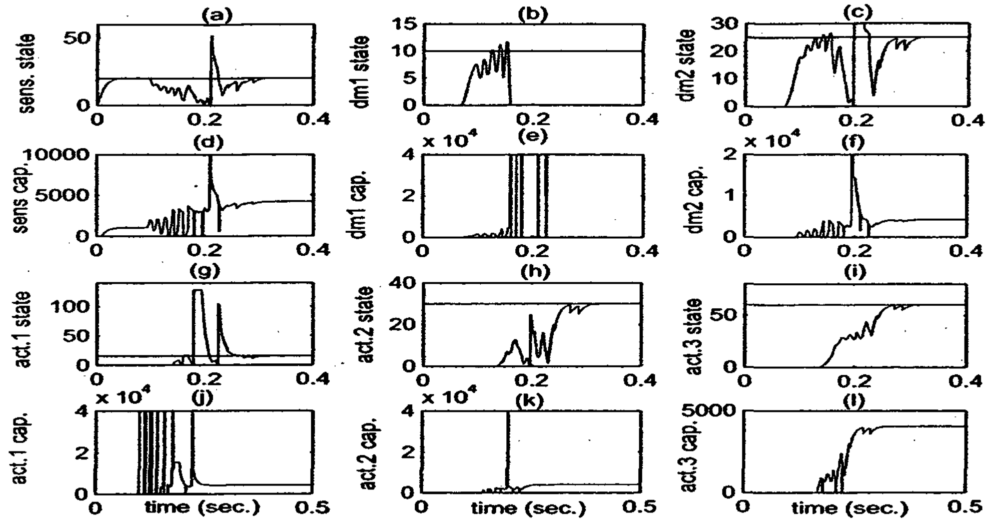


Figure 4.15: The queue length (packet) and control (capacity: packet/s) behavior for a fully-connected configuration in presence of a non-measurable time-delay of 60 ms.

where $\eta = [e^T\ \ \hat{\lambda}^T]^T$, $e = [e^s\ \ e_1^d\ \ e_3^a]^T$, $\hat{\lambda} = [\hat{\lambda}^s\ \ \hat{\lambda}_1^d\ \ \hat{\lambda}_3^a]^T$, $\mathbb{B}_1 = [1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$,

$$\mathbb{A}_1 = \begin{pmatrix} -\mu^s & 0 & 0 & -1 & 0 & 0 \\ 0 & -\mu^d & 0 & 0 & -1 & 0 \\ 0 & 0 & -\mu^a & 0 & 0 & -1 \\ \Gamma_s & 0 & 0 & 0 & 0 & 0 \\ 0 & \Gamma_d & 0 & 0 & 0 & 0 \\ 0 & 0 & \Gamma_a & 0 & 0 & 0 \end{pmatrix}, \ \mathbb{A}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ F\mu^s & 0 & 0 & F & 0 & 0 \\ 0 & F_1\mu_1^d & 0 & 0 & F & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{4.22}$$

$$\mathbb{A}_3 = \begin{pmatrix} -\gamma^s & 0 & 0 & 0 & 0 & 0 \\ 0 & -\gamma^d & 0 & 0 & 0 & 0 \\ 0 & 0 & -\gamma^a & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \ and\ \mathbb{A}_4 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ F\gamma^s & 0 & 0 & 0 & 0 & 0 \\ 0 & F\gamma_1^d & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{4.23}$$

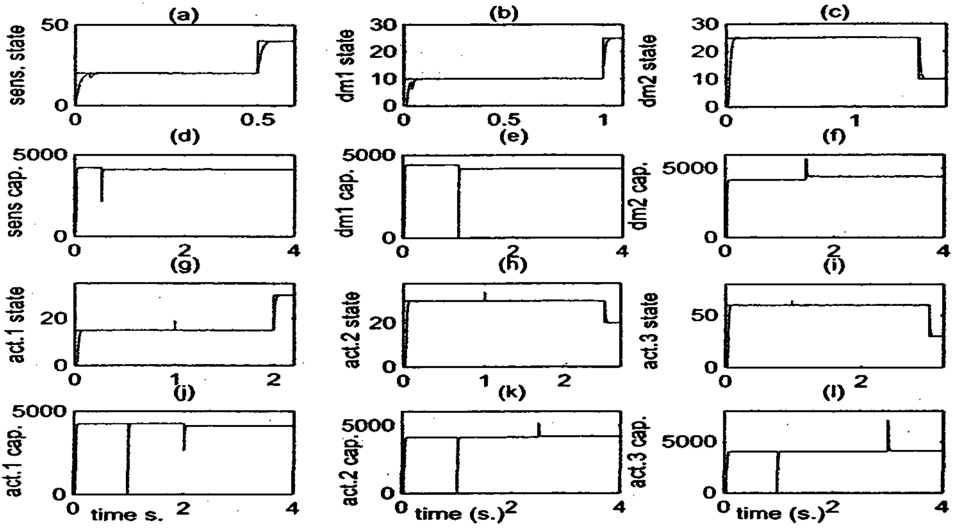


Figure 4.16: The queue length (packet) and control (capacity:packet/s) behavior for a fully-connected configuration with a wide range of operating points in presence of a non-measurable time-delay of 1 ms.

4.5.3.2 Fully-connected switching configuration

The time-delay dependent network dynamics is now governed by

$$\dot{\eta} = -\mathbb{A}_1\eta + \mathbb{A}_2\eta(t-\tau) - \mathbb{A}_3 sgn(e) + \mathbb{A}_4 sgn(e(t-\tau)) + \mathbb{B}_1\lambda^s_{i2}(t) \qquad (4.24)$$

where $\eta = [e^T\ \hat{\lambda}^T]^T$, $e = [e^s\ e^d_1\ e^d_2\ e^a_1\ e^a_2\ e^a_3]^T$, $\hat{\lambda} = [\hat{\lambda}^s\ \hat{\lambda}^d_1\ \hat{\lambda}^d_2\ \hat{\lambda}^a_1\ \hat{\lambda}^a_2\ \hat{\lambda}^a_3]^T$,

$$\mathbb{A}_1 = \begin{pmatrix} -A_1 & -I \\ \Gamma & 0 \end{pmatrix},\ \mathbb{A}_2 = \begin{pmatrix} A_2 & B_1 \\ 0 & 0 \end{pmatrix},\ \mathbb{A}_3 = \begin{pmatrix} A_3 \\ 0 \end{pmatrix},\ \mathbb{A}_4 = \begin{pmatrix} A_4 \\ 0 \end{pmatrix},\ \mathbb{B}_1 = \begin{pmatrix} B_1 \\ 0 \end{pmatrix} \qquad (4.25)$$

with $A_1 = diag[\mu_s\ \mu_{d1}\ \mu_{d2}\ \mu_{a1}\ \mu_{a2}\ \mu_{a3}]^T$, $A_3 = -diag[\gamma_s\ \gamma_{d1}\ \gamma_{d2}\ \gamma_{a1}\ \gamma_{a2}\ \gamma_{a3}]^T$, $B_2 = [S^s_2\ 0\ 0\ 0\ 0\ 0]^T$, $A_2 = B_1A_1$, $A_4 = B_1A_3$, and

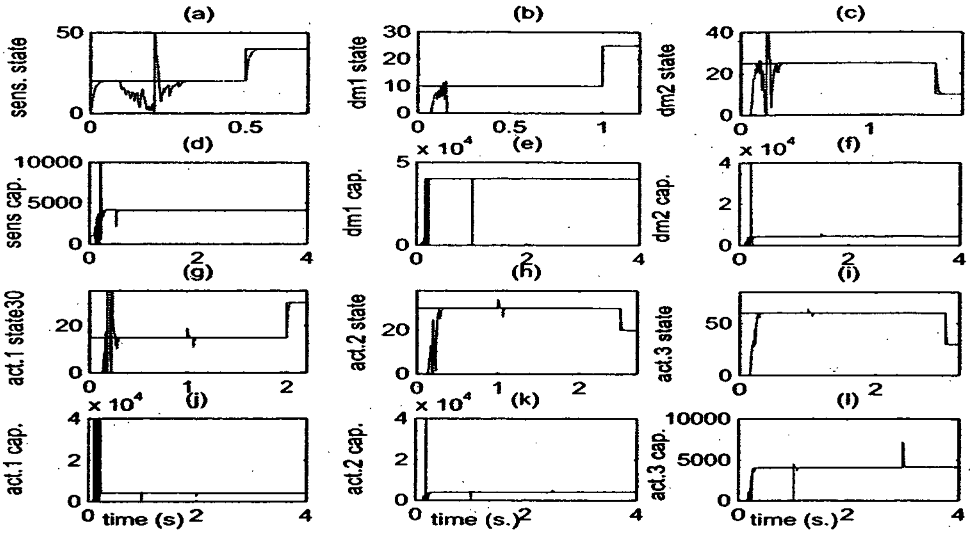


Figure 4.17: The queue length (packet) and control (capacity:packet/s) behavior for a fully-connected configuration with a wide range of operating points in presence of a non-measurable time-delay of 60 ms.

$$B_1 = \begin{pmatrix} 0 & 0 & S_3^sF & 0 & S_1^sF & 0 \\ S_{12}^dF & 0 & 0 & S_{11}^dF & 0 & 0 \\ S_{22}^dF & 0 & 0 & S_{21}^dF & 0 & 0 \\ 0 & S_1^aF & 0 & 0 & 0 & 0 \\ 0 & S_2^aF & 0 & 0 & 0 & 0 \\ 0 & S_3^aF & 0 & 0 & 0 & 0 \end{pmatrix} \qquad (4.26)$$

Remark 4.2. *The stability analysis of the obtained error dynamics* (4.21) *and* (4.24) *corresponding to the cascaded and fully-connected configurations, respectively can be studied by using a less conservative approach as it will be studied in Chapter 8. In effect, it can be observed that the error dynamics* (4.24) *is similar to that given by equation* (8.46), *therefore the stability analysis is omitted here.*

4.6 Bandwidth Allocation Control with Dynamic Switching Configurations

Given that assets in a NUS systems are mobile, it is important to verify that our proposed control strategies do hold in such situations. As shown in Figure 3.4, block delays and switches are considered in the cluster nodes to model the changes and the addition/removal of assets. It is worth noting that during these changes assets may be at certain locations in which their communication range could be reduced or even lost. The effects of node mobility on our proposed time delayed network control are evaluated through the following simulations by changing dynamically the switching configurations. These configurations change every control interval according to a lookup table containing all possible combinations. First, by setting the same reference points as in the simulations corresponding to Figure 4.14, Figure 4.18 shows that for a measurable input traffic according to Case 2 and in presence of unknown delay of 3 ms the convergence is achieved.

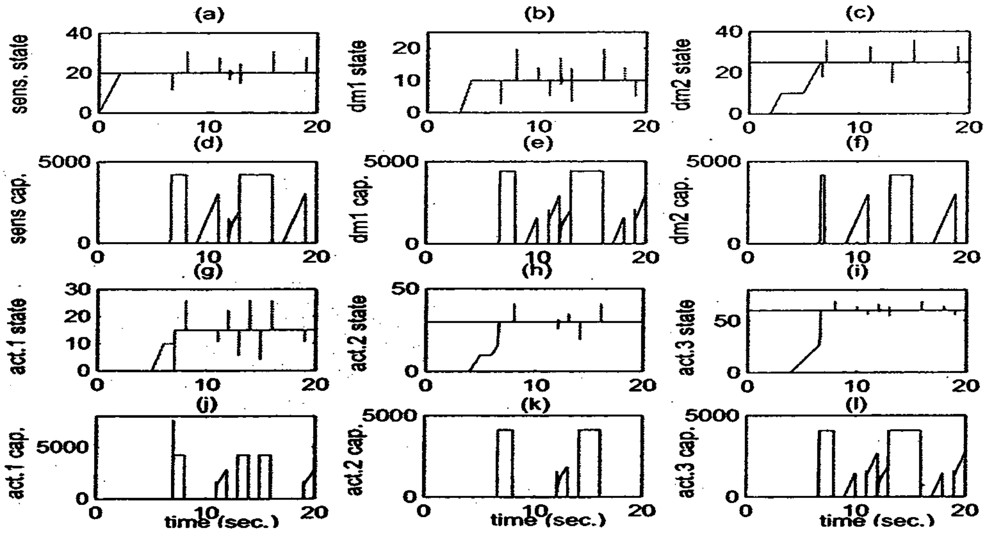


Figure 4.18: The queue length (packet) and control (capacity: packet/s) behavior in presence of dynamic changes of switching configurations and a non-measurable time-delay of 3 ms.

Note that beyond a delay of 4 ms, the convergence of the buffer queue lengths of the closed-loop system to their respective references can not be achieved. This is shown at least for one buffer queue as depicted in Figure 4.19 corresponding to a time-delay of 60 ms.

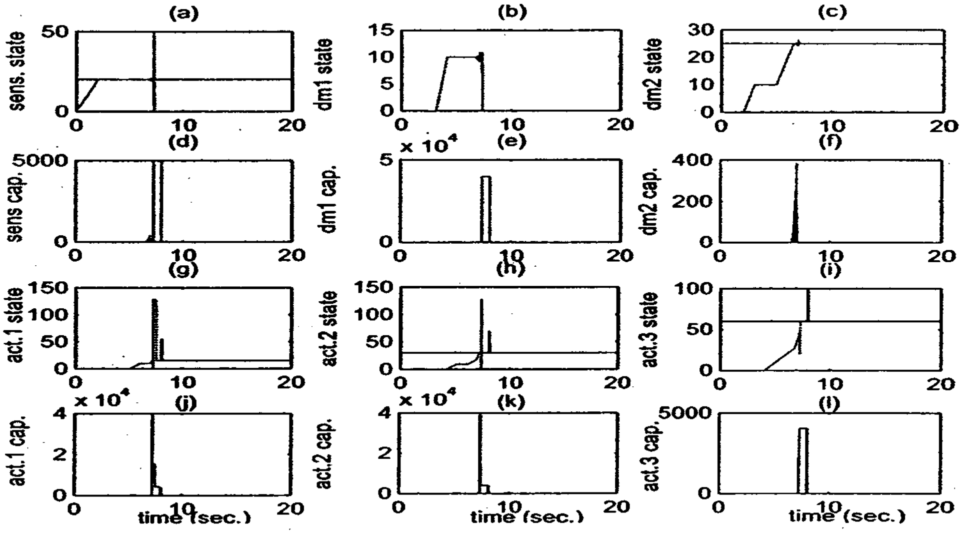


Figure 4.19: The queue length (packet) and control (capacity:packet/s) behavior in presence of dynamic changes of switching configurations and a non-measurable time-delay of 60 ms.

Next, simulations corresponding to different operating points with the queue references set to $x^s_{ref} = \{20, 40\}$ packet, $x^d_{1_{ref}} = \{10, 25\}$ packet, $x^d_{2_{ref}} = \{25, 10\}$ packet, $x^a_{1_{ref}} = \{15, 30\}$ packet, $x^a_{2_{ref}} = \{30, 20\}$ packet, and $x^a_{3_{ref}} = \{60, 30\}$ packet are conducted. In presence of an unknown time-delay of 3 ms and when the switching configuration dynamically changes, Figure 4.20 shows that the convergence is achieved for a wide range of operating points. Beyond an unknown time-delay of 5 ms, the buffer queue lengths of the closed-loop system do not converge to their respective references as shown in Figure 4.21 corresponding to a time-delay of 60 ms. In compliance with the good performance obtained in the presence of a time-delay of 3 ms and a dynamic change of the switching configurations, one may conclude that the resulting loss of performance for a time-delay of 60 ms is primarily due to time-delays rather than dynamic changes in the switching configurations.

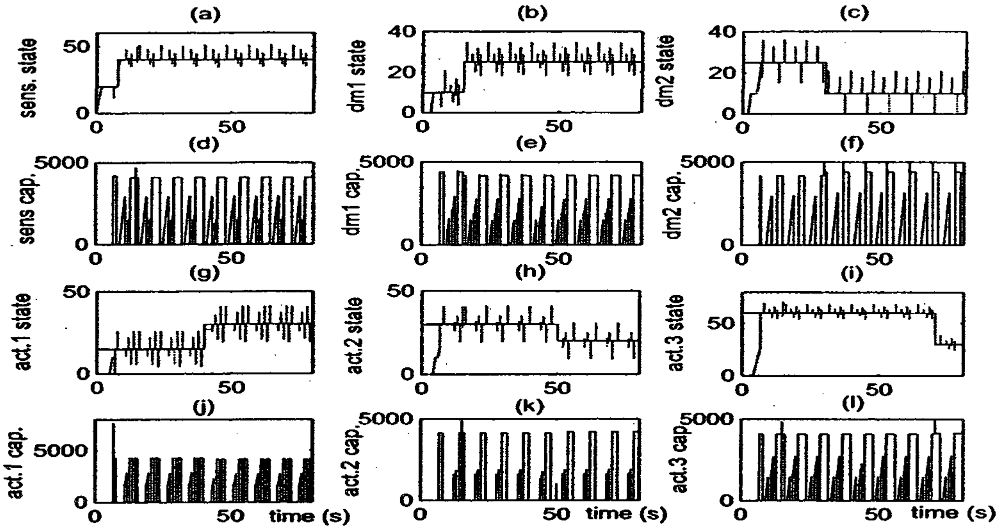


Figure 4.20: The queue length (packet) and control (capacity: packet/s) behavior with a wide range of operating points in presence of dynamic changes of switching configurations and a non-measurable time-delay of 3 ms.

4.7 Conclusion

In this chapter, a networked unmanned system (NUS) where each cluster has a common characteristic (real-time requirements, feedback loops, etc.) that consists of three nodes namely, a sensor, a decision-maker, and an actuator is considered. We have investigated the problem of resource (bandwidth) allocation within our proposed network for different configurations in terms of possible interconnections of various assets. We have considered both measurable and estimated input traffics.

We have demonstrated the challenges in allocating the bandwidth to different nodes of a NUS system by implementing a standard PID controller. Through simulations it is observed that by introducing feedback loops in our time-delayed NUS cluster configuration, control of the overall network can fail. In other words, depending on the interconnections between the network nodes, the behavior of the

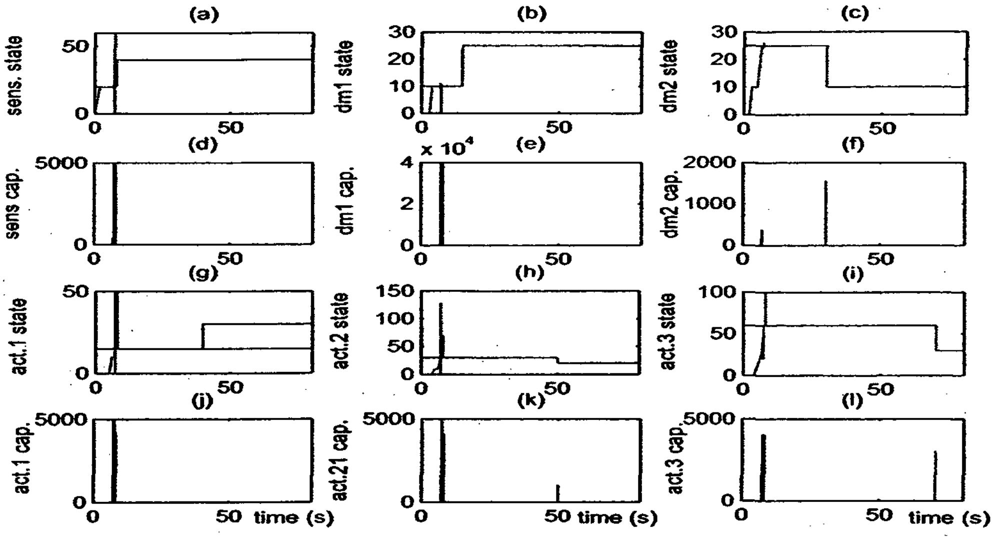


Figure 4.21: The queue length (packet) and control (capacity: packet/s) behavior with a wide range of operating points in presence of dynamic changes of switching configurations and a non-measurable time-delay of 60 ms.

PID-based closed-loop system vary considerably such that the network might become even unstable. This problem can be however addressed by using more elaborate nonlinear control approaches.

The first nonlinear control approach that is investigated is the input-output linearization controller. By assuming that the input traffic and time-delay are measurable, the controller has demonstrated good performance and achieves the imposed desired design specifications for both the steady state error and the transient (overshoot behaviors).

The second nonlinear controller is designed by using the sliding mode machinery. Assuming that the input traffic is measurable and time-delay is unknown, the time-delay dependent control system dynamics is derived and is shown to be asymptotically stable for a fully-connected configuration. The delayed control dynamics is also shown to be stable when a loop is added to our configuration, provided that the time-delay is upper bounded. Finally, by estimating the input traffic and corresponding to different network configurations, the overall NUS cluster error dynamics is derived as a generic time-delay system. In addition, our proposed robust controller is evaluated when node mobility is considered. We have demonstrated that for a wide range of operating points good performance and stability can be accomplished provided that the time-delay is upper bounded.

Chapter 5

Integrated Bandwidth Allocation and Flow Rate Control

5.1 Introduction

In communication networks, several mechanisms are studied to control traffic and avoid congestion, such as call admission control, input rate regulation, bandwidth allocation, routing, queue scheduling, and buffer management. It is widely accepted that in certain circumstances dynamic integrated control solutions may turn out to be more efficient than individual solutions.

In this chapter and illustrated in Figure 5.1, by using nonlinear feedback control theory, we propose three robust control strategies that dynamically integrate bandwidth allocation and flow rate control techniques. The first and third proposed strategies are evaluated on a network cluster that is defined in Chapter 3 under non-stationary conditions. These conditions occur when the statistics of the incoming traffic to the network queues are time-variant as well as topological changes take place in the network. Indeed, simulations are performed for dynamic traffics, under various network configurations and in presence of time-delays.

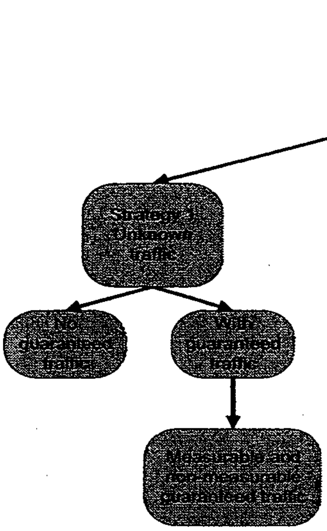

Proposed sliding mode controller
Strategy 1 Unknown traffic
Strategy 2 Unknown traffic
Strategy 3 Measured/ estimated traffic
No guaranteed traffic
With guaranteed traffic
Measurable and non-measurable guaranteed traffic
Three node cascade and fully connected

Figure 5.1: A recapitulative diagram of the investigations on the proposed sliding mode bandwidth allocation control.

5.2 First Proposed Robust Control Strategy

The first robust control strategy that we propose is an improved version of the strategy proposed in [91] (refer to Chapter 2). The improvement objective is to robustify the approach in [91] that uses a simple input-output linearization feedback control by adding a discontinuous control component to the control law. The design procedure for our proposed robust control strategy relies on nonlinear feedback control that uses sliding mode techniques. As illustrated in Figure 5.2, the proposed robust control strategy attempts on one hand to maintain the average queue length of each network buffer by dynamically allocating the bandwidth to the server while on the other hand to control the flow rate.

For the l^{th} output port of node h, our controller is designed in the error state space representation by first defining the sate error variables as $e_{l_h} = x_{l_h} - x_{l_h ref}$ and $\dot{e}_{l_h} = \dot{x}_{l_h} - \dot{x}_{l_h ref}$ where $x_{l_h ref}$ denotes the reference trajectory of the buffer queue length. By substituting the state error in the model governed by equation (3.1), the

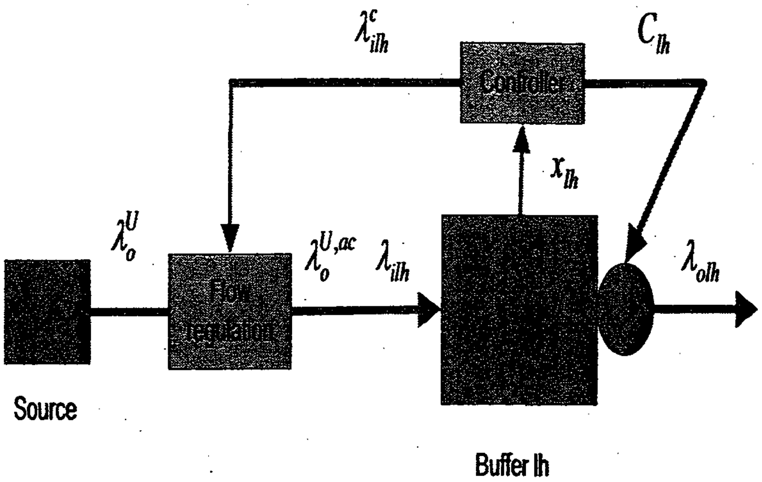


Figure 5.2: Integrated bandwidth allocation and flow rate control principle.

state space representation can be rewritten in the error space as follows

$$\begin{cases} \dot{e}_{l_h}(t) = -C_{l_h}(t)\dfrac{e_{l_h}(t) + x_{l_h ref}(t)}{e_{l_h}(t) + x_{l_h ref}(t) + 1} + \lambda_{il_h}(t) - \dot{x}_{l_h ref}(t) \\ y_{l_h}(t) = e_{l_h}(t) + x_{l_h ref}(t) \end{cases} \tag{5.1}$$

The robust control law is now derived and expressed according to

$$\begin{cases} C_{l_h}(t) = \mu_{l_h}(e_{l_h} + x_{l_h ref} + 1) + E^{-1}_{l_h}(t)\gamma_{l_h} sgn(e_{l_h}(t)) \\ \lambda^{c}_{l_h}(t) = \mu_{l_h} x_{l_h ref} + \dot{x}_{l_h ref}(t) \end{cases} \tag{5.2}$$

where $E_{l_h}(t) = \frac{e_{l_h}(t)+x_{l_h ref}(t)}{e_{l_h}(t)+x_{l_h ref}(t)+1}$ is the l^{th} buffer occupancy with μ_{l_h} and γ_{l_h} are design parameters assigned to buffer l that ensures closed-loop asymptotic convergence of the error dynamics (5.1). Note that index c in $\lambda^{c}_{l_h}(t)$ denotes *control*. The control law (5.2) is derived first by selecting the flow rate to be proportional (according to the design gain) to the desired queue length and then by applying a feedback control to obtain the bandwidth allocation control law. The resulting error dynamics of the l^{th} closed-loop buffer is now given by

$$\dot{e}_{l_h}(t) = -\mu_{l_h} e_{l_h}(t) - \gamma_{l_h} sgn(e_{l_h}(t)) \tag{5.3}$$

Remark 5.1. *As mentioned earlier, the proposed control given by equation* (5.2) *is an improvement of the controller proposed in [91]. Furthermore, note that in case of congestion the buffer input traffic* λ_{ilh} *corresponds to the maximum allowed traffic* λ^c_{ilh} *computed by the controller.*

Theorem 5.1. *For the* l^{th} *buffer of node h described by the FFM model* (5.1)*, by adopting the end-to-end semi-decentralized control approach, the nonlinear feedback flow rate and bandwidth control laws* (5.2) *guarantee asymptotic stability of the error dynamics* (5.3) *if* μ_{l_h} *and* γ_{l_h} *are selected as positive parameters.*

Proof: Since the error dynamics of equation (5.3) is the same as the one given by equation (4.7), therefore the proof of Theorem 5.1 follows along the same lines as as that for Theorem 4.1. ▲

Remark 5.2. *It is worth noting that the main advantage of the above strategy is that there is no need to measure or estimate the input traffic since the control law does not depend on it.*

5.2.1 Case study: The ABR traffic competing with the guaranteed traffic

Assume that we are given two types of traffics, namely guaranteed traffic $\lambda^g_{i_{l_h}}$ and the ABR traffic $\lambda_{i_{l_h}ABR}(t)$ are competing for network resources. Note that the FFM model given by equation (3.1) is now written in the error state space as follows

$$\begin{cases} \dot{e}_{l_h}(t) = -C_{l_h}(t)\dfrac{e_{l_h}(t) + x_{l_href}(t)}{e_{l_h}(t) + x_{l_href}(t) + 1} + \lambda_{il_hABR}(t) + \lambda^g_{il_h}(t) - \dot{x}_{l_href}(t) \\ y_{l_h}(t) = e_{l_h}(t) + x_{l_href}(t) \end{cases} \tag{5.4}$$

Let us start with our first strategy so that Theorem 5.1 is applied to model (5.4). Two cases are considered depending on $\lambda^g_{il_h}$ that may be measurable or unmeasurable while $\lambda_{i_{l_h}ABR}(t)$ is considered to be unknown as in the above case.

$\lambda^g_{il_h}$ measurable

By assuming that the guaranteed traffic is measurable while the ABR traffic is unknown, the flow rate and bandwidth control laws are obtained and given as follows

$$\begin{cases} C_{l_h}(t) = \mu_{l_h}(e_{l_h} + x_{l_href} + 1) + E^{-1}_{l_h}(t)\gamma_{l_h} sgn(e_{l_h}(t)) \\ \lambda^c_{il_hABR}(t) = \mu_{l_h} x_{l_href}(t) - \lambda^g_{il_h}(t) + \dot{x}_{l_href}(t) \end{cases} \tag{5.5}$$

where $E_{l_h}(t) = \frac{e_{l_h}(t)+x_{l_href}(t)}{e_{l_h}(t)+x_{l_href}(t)+1}$ is the l^{th} buffer occupancy with μ_{l_h} and γ_{l_h} are the design parameters assigned to the buffer l that ensures asymptotic convergence of the error dynamics (5.4) that is governed by

$$\dot{e}_{l_h}(t) = -\mu_{l_h} e_{l_h}(t) - \gamma_{l_h} sgn(e_{l_h}(t)) \tag{5.6}$$

Lemma 5.1. *For the l^{th} buffer of node h described by the FFM model* (5.4), *by adopting the end-to-end semi-decentralized control approach, the nonlinear feedback flow rate and bandwidth control laws* (5.5) *guarantee asymptotic stability of the error dynamics* (5.6) *if μ_{l_h} and γ_{l_h} are selected as positive parameters.*

Proof: Since the error dynamics of equation (5.6) is the same as the one given by equation (4.7), therefore the proof of Lemma 5.1 is the same as that for Theorem 4.1. ▲

$\lambda^g_{l_h}$ unmeasurable

Considering that both traffics are unknown, let us now set the flow rate and bandwidth control laws as described below

$$\begin{cases} C_{l_h}(t) = \mu_{l_h}(e_{l_h} + x_{l_href} + 1) + E^{-1}_{l_h}(t)\gamma_{l_h} sgn(e_{l_h}(t)) \\ \lambda_{l_hABR}(t) = \mu_{l_h} x_{l_href}(t) + \dot{x}_{l_href}(t) \end{cases} \tag{5.7}$$

where $E_{l_h}(t) = \frac{e_{l_h}(t)+x_{l_href}(t)}{e_{l_h}(t)+x_{l_href}(t)+1}$ is the l^{th} buffer occupancy with μ_{l_h} and γ_{l_h} are the design parameters assigned to buffer l that ensures boundedness of the error dynamics (5.4) in the closed-loop that is governed

$$\dot{e}_{l_h}(t) = -\mu_{l_h} e_{l_h}(t) - \gamma_{l_h} sgn(e_{l_h}(t)) + \lambda^g_{il_h}(t) \tag{5.8}$$

Lemma 5.2. *For the l^{th} buffer of node h described by the FFM model* (5.4), *by adopting the end-to-end semi-decentralized control approach, the nonlinear feedback flow rate and bandwidth control laws* (5.7) *guarantee stability of the error dynamics* (5.8) *if μ_{l_h} and γ_{l_h} are selected as positive parameters provided that $\lambda^g_{il_h}$ is bounded.*

Proof: Since the error dynamics of equation (5.8) corresponds to the error dynamics given by (4.7) with an additional term $\lambda^g_{il_h}$ that may be considered as an external input, then the error dynamics (5.8) is stable whenever $\lambda^g_{il_h}$ is bounded and the design parameters μ_{l_h} and γ_{l_h} are selected as positive. ▲

The convergence of these error dynamics can be also seen through their corresponding solutions. The general solution for the error dynamics (5.8) is described by equation (5.9), namely

$$e_{l_h}(t) \leq \begin{cases} E_{xp}^{-\mu_{l_h}(t-t_0)}[e_{l_h}(t_0) - \frac{\lambda^g_{il_h max}}{\mu_{l_h}} - \frac{\gamma_{l_h}}{\mu_{l_h}}] + \frac{\lambda^g_{il_h max}}{\mu_{l_h}} + \frac{\gamma_{l_h}}{\mu_{l_h}} & if \quad e_{l_h} < 0 \\ E_{xp}^{-\mu_{l_h}(t-t_0)}[e_{l_h}(t_0) - \frac{\lambda^g_{il_h max}}{\mu_{l_h}}] + \frac{\lambda^{ig}_{l_h max}}{\mu_{l_h}} & if \quad e_{l_h} = 0 \\ E_{xp}^{-\mu_{l_h}(t-t_0)}[e_{l_h}(t_0) - \frac{\lambda^g_{il_h max}}{\mu_{l_h}} + \frac{\gamma_{l_h}}{\mu_{l_h}}] + \frac{\lambda^g_{il_h max}}{\mu_{l_h}} - \frac{\gamma_{l_h}}{\mu_{l_h}} & if \quad e_{l_h} > 0 \end{cases} \tag{5.9}$$

where E_{xp} denotes the exponential function.

5.2.2 Simulation results using the first proposed robust control strategy

The first robust control strategy proposed above is now implemented as an end-to-end semi-decentralized control approach on the delay dependent network that is depicted by Figure 3.4. In the end-to-end control approach, the controller maintains locally the averaged l^{th} buffer queue size of node h close to the desired value and regulates the incoming traffic rate to node h by informing all upstream sources that are using node h regarding the maximum allowed rate.

Two sets of simulations are given in this section. In the first set, our network cluster is configured as a three nodes cascade by disabling all the loops. In the second set our network cluster is configured as fully-connected by incorporating all the loops (with feedback traffics). Given that delays is a crucial problem [76], the proposed control strategy is evaluated subject to the presence of time-delays of 1 ms and 60 ms.

For all the simulations presented below the sampling time is set to $T_s = 1$ ms and the proposed control parameters are set to $\mu_{l_h} = 15$ and $\gamma_{l_h} = 7500$. As far as the traffic is concerned, dynamic sources are considered in order to excite a broad range of the frequency modes of the system. Indeed, the mean of the random input traffic varies sinusoidally from 0 to $6 * 10^3$ packets/s with a frequency of 0.5 rad/s and variance of 10^3 packets/s. Note that the input traffic is bounded such that $0 \leq \lambda_{il_h} \leq 4000$ packet/s.

In compliance with Assumption 3.1, the buffer queue sizes are constrained according to the bounds $0 \leq x_{l_h} \leq 128$ packets while the maximum available service capacity is constrained according to Assumption 3.3 so that $0 \leq C_{l_h} \leq 4 * 10^4$ packets. Furthermore, the process function gains F_{lk}^{hj} are set equal to unity in all the links.

The first set of simulations is shown in Figure 5.3 for the time-delays of 1 ms (solid line) and 70 ms (dashed line). Note that with this cascade configuration, one is concerned with only the sensor port, the first output port of the decision-maker labeled $DM21$ corresponding to λ_{o1}^{d} in Figure 3.4, and the third output port of the actuator node labeled $Act33$ corresponding to λ_{o3}^{a}.

From Figure 5.3, we may observe that (a) the sensor, the decision-maker DM21, and the actuator Act33 buffer queue lengths all converge to their respective references which are set to [50, 60, 100] packets even in presence of time delay of 60 ms but with a steady state error, (b) the speed of transient responses is good, and (c) the behavior of the queue state responses are quiet oscillatory.

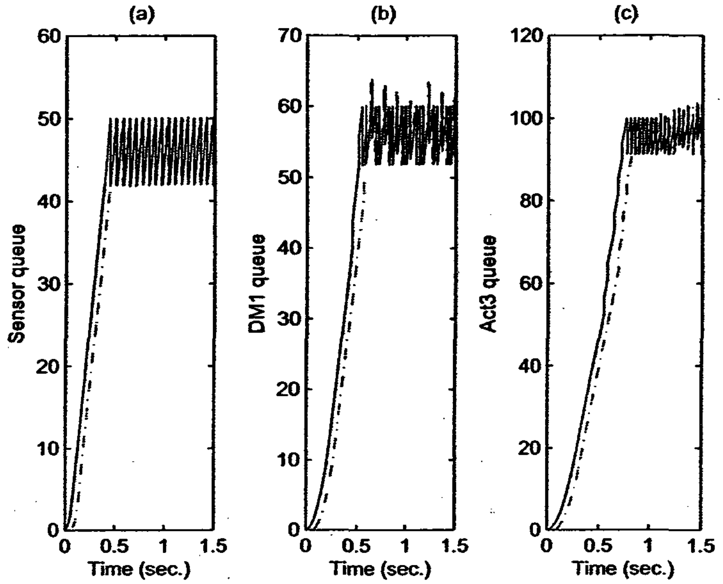


Figure 5.3: Simulation results for the cascade configuration in presence of time-delays (solid line: 1 ms, dashed line: 60 ms) and dynamic source traffics by using the first proposed strategy.

The second set of simulations are shown in Figure 5.4 for the time delays of 1 ms (solid line) and 70 ms (dashed line) and the operating points are set to 50 packet for the sensor, to [60 70] packet for the first and second outports of the decision-maker, respectively, and to [25 30 100] packet for the first, second, and third output ports of the actuator, respectively. The first row graphs of Figure 5.4 depict the queue state responses corresponding to (a) the sensor, (b) the decision-maker $DM21$, and (c) the second output port of the decision-maker labeled $DM22$. The second row graphs of Figure 5.4 are devoted to the three output ports of the actuator, namely

(d) *Act*31, (e) *Act*32, and (f) *Act*33, respectively. Note that the same remarks can be stated as for the results that are obtained for a cascade configuration. We may therefore conclude that the first strategy has an advantage that is the input traffic does not need to be measured and estimated but has a drawback that the responses of the queues are oscillatory.

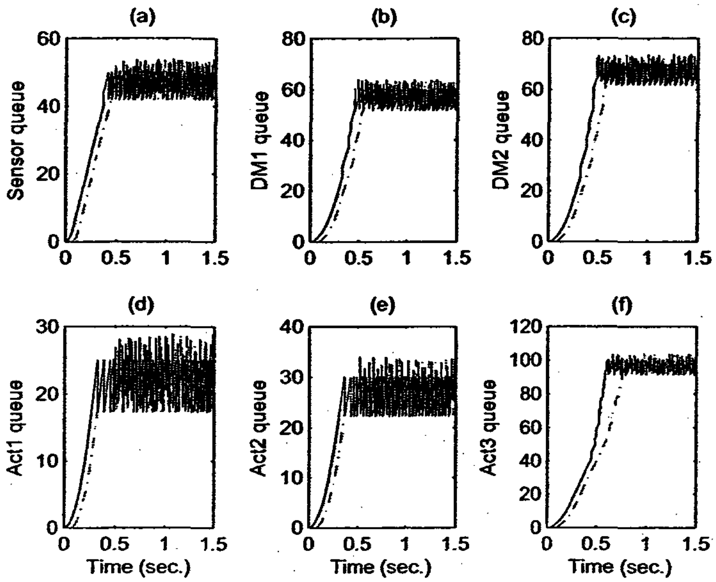


Figure 5.4: Simulation results obtained for the fully-connected configuration in presence of time-delays (solid line: 1 ms, dashed line: 60 ms) and dynamic source traffics by using the first proposed strategy.

5.3 Second Proposed Robust Control Strategy

The second control strategy that we propose is also robust since it is designed based on the sliding mode control tools. For the l^{th} output port of node h, our controller is designed in the error state space representation by using the same state error variables as in the first strategy. The robust controller is expressed as follows

$$\begin{cases} C_{l_h}(t) = \mu_{l_h}(e_{l_h} + x_{l_h ref} + 1) \\ \lambda^c_{il_h}(t) = \mu_{l_h} x_{l_h ref} - \gamma_{l_h} sgn(e_{l_h}(t)) + \dot{x}_{l_h ref}(t) \end{cases} \tag{5.10}$$

where μ_{l_h} and γ_{l_h} are the design parameters that are assigned to the buffer l that ensures asymptotic convergence of the closed-loop error dynamics (5.1).

The control law (5.10) is obtained first by selecting the bandwidth allocation control law as $C_{l_h}(t) = \mu_{l_h}(e_{l_h} + x_{l_h ref} + 1)$ and then applying a feedback control to obtain the flow rate. The resulting error dynamics of the l^{th} closed-loop controlled buffer is now given by

$$\dot{e}_{l_h}(t) = -\mu_{l_h} e_{l_h}(t) - \gamma_{l_h} sgn(e_{l_h}(t)) \tag{5.11}$$

Theorem 5.2. *For the l^{th} buffer of node h described by the FFM model* (5.1), *by adopting the end-to-end semi-decentralized control approach, the nonlinear feedback flow rate and bandwidth control laws given by equation* (5.10) *guarantee asymptotic stability of the error dynamics* (5.11) *if μ_{l_h} and γ_{l_h} are selected as positive parameters.*

Proof: Since the error dynamics of equation (5.11) is the same as the one given by (4.7), therefore the proof of Theorem 5.2 is the same as that for Theorem 4.1. ▲

Remark 5.3. *As for the strategy proposed above, note that there is no need to measure or estimate the input traffic for this control strategy too.*

5.4 Third Proposed Robust Control Strategy

Designed in the error state space representation as in the previous two strategies, the third robust control law is given by

$$\left\{ C^{ac}_{l_h}(t) = max\{0, min[C^{max}_{l_h}(t), C_{l_h}(t)]\} \right. \tag{5.12a}$$

$$\left\{ \lambda_{il_h}^{max,ac}(t) = max\{0, min[C_{l_h}^{max}(t), \lambda_{il_h}^{max}(t)]\} \right. \tag{5.12b}$$

where index *ac* designates the actual value. By applying the SM-VSC machinery to equation (5.12a), the buffer capacity is set to

$$C_{l_h}(t) = E_{l_h}^{-1}(\mu_{l_h} e_{l_h} + \gamma_{l_h} sgn(e_{l_h}) + \hat{\lambda}_{il_h} - \dot{x}_{l_{href}}) \tag{5.13}$$

where $E_{l_h} = \frac{x_{l_h}}{x_{l_h}+1} = \frac{e_{l_h}+x_{l_{href}}}{e_{l_h}+x_{l_{href}}+1}$ is the buffer occupancy, $\hat{\lambda}_{il_h}$ is the estimate of the input traffic that is obtained by using the following update law

$$\left\{ \dot{\hat{\lambda}}_{il_h} = \Gamma_{l_h} e_{l_h}(t) \right. \tag{5.14}$$

where Γ_{l_h}, μ_{l_h}, and γ_{l_h} are the design parameters. Note that the buffer capacity $C_{l_h}^{ac}$ is bounded by its maximum value $C_{l_h}^{max}$ which is equal to C_{serv}.

In equation (5.12b), the term

$$\lambda_{il_h}^{max} = C_{l_h}^{max} E_{l_h r} - \mu_{l_h r} e_{l_h r} - \gamma_{l_h r} sgn(e_{l_h r}) + \dot{x}_{l_h ref} \tag{5.15}$$

is the maximum ordinary traffic flow rate that is allowed by the controller and whose actual value $\lambda_{il_h r}^{max,ac}(t)$ is bounded by $C_{l_h}^{max}$. The resulting error dynamics of the l^{th} closed-loop controlled output port of node *h* is now given by

$$\dot{e}_{l_h} = -\mu_{l_h} e_{l_h} - \gamma_{l_h} sgn(e_{l_h}) + \tilde{\lambda}_{il_h} \tag{5.16}$$

where $\tilde{\lambda}_{il_h} = \lambda_{il_h} - \hat{\lambda}_{il_h}$. The behavior of the closed-loop system (5.16) is now described by the following theorem.

Theorem 5.3. *For the l^{th} output port of node h described by the Fluid Flow Model* (5.1), *by adopting the end-to-end semi-decentralized control approach, the nonlinear feedback bandwidth allocation and flow rate control law governed by equations* (5.12a), (5.12b), *and* (5.14) *guarantees asymptotic stability of the error dynamics* (5.16) *if $\mu_{l_h f}$, $\gamma_{l_h f}$, $\Gamma_{l_h f}$ $(f = p, r)$ are selected as positive design parameters that are assigned to the output port l_h.*

Proof: Let us choose a candidate Lyapunov function $V(e_{l_h}; \tilde{\lambda}_{il_h}) = \frac{e_{l_h}^2}{2} + \frac{\Gamma_{l_h}^{-1}\tilde{\lambda}_{il_h}^2}{2}$, where $\tilde{\lambda}_{il_h} = \lambda_{il_h} - \hat{\lambda}_{il_h}$ such that $V(e_{l_h}, \tilde{\lambda}_{il_h})$ is positive definite in $\mathbb{D} = \{(e_{l_h}; \tilde{\lambda}_{il_h}) \in \Re^2\}$. The time derivative of the Lyapunov function along the trajectories of (5.16) and (5.14) is given by

$$\dot{V}(e_{l_h}; \tilde{\lambda}_{il_h}) = -\mu_{l_h f} e_{l_h}^2 - \gamma_{l_h} e_{l_h} sgn(e_{l_h}) + \Delta_{l_h}(t) \tag{5.17}$$

where $\Delta_{l_h}(t) = e_{l_h}(t)\tilde{\lambda}_{il_h}(t) + \Gamma_{l_h}^{-1}\tilde{\lambda}_{il_h}(t)\dot{\tilde{\lambda}}_{il_h}(t)$. Given that $e_{l_h}(t)sgn(e_{l_h}(t))$ is always positive since

$$sgn(e_{l_h}(t)) = \begin{cases} 1 \ if \ e_{l_h}(t) > 0 \\ 0 \ if \ e_{l_h}(t) = 0 \\ -1 \ if \ e_{l_h}(t) < 0 \end{cases} \tag{5.18}$$

and if one now defines

$$\dot{\tilde{\lambda}}_{il_h} = -\Gamma_{l_h} e_{l_h}(t) \tag{5.19}$$

or equivalently for a slowly-time varying case,

$$\dot{\hat{\lambda}}_{il_h} = \Gamma_{l_h} e_{l_h}(t) \tag{5.20}$$

which defines the dynamics of the parameter estimate $\hat{\lambda}_{il_h}$, then equation (8.7) becomes

$$\dot{V}(e_{l_h}, \tilde{\lambda}_{il_h}) = -\mu_{l_h f} e_{l_h}^2 - \gamma_{l_h} e_{l_h} sgn(e_{l_h}) \tag{5.21}$$

which is negative semi-definite. Using the La Salle's invariance principle [68], let us define $\mathbb{S} = \{(e_{l_h}, \tilde{\lambda}_{il_h}) \in \mathbb{D} | \dot{V}(e_{l_h}, \tilde{\lambda}_{il_h}) = 0\}$. For $\dot{V}(e_{l_h}, \tilde{\lambda}_{il_h}) = 0$, $e_{l_h} = 0 \Rightarrow \mathbb{S} = \{(e_{l_h}, \tilde{\lambda}_{il_h}) \in \mathbb{D} | e_{l_h} = 0\}$. Assuming that $(e_{l_h}, \tilde{\lambda}_{il_h}) \in \mathbb{S} \ \forall \ t \Rightarrow e_{l_h}(t) = 0 \ \forall \ t$, and $\dot{e}_{l_h}(t) \equiv 0 \Rightarrow 0 = -A_{l_h} e_{l_h} - B_{l_h} sgn(e_{l_h}) + \tilde{\lambda}_{il_h}$ which implies that $e_{l_h}(t) = 0$ and $\tilde{\lambda}_{il_h}(t) = 0$. Therefore, the only invariant solution in $\mathbb{S}$ is the trivial solution $e_{l_h}(t) = 0$ and $\tilde{\lambda}_{il_h}(t) = 0$. Hence, the equilibrium point $e_{l_h} = 0$ and $\tilde{\lambda}_{il_h} = 0$ is asymptotically stable. ▲

5.4.1 Simulation results using the third proposed robust control strategy

The simulations given in this section are intended to demonstrate the effectiveness of the third proposed robust control solution as an end-to-end semi-decentralized approach for the delay dependent cluster that is depicted in Figure 3.4. As explained above, in the end-to-end control approach the controller maintains locally the averaged buffer queue size of node h close to the desired value and regulates the incoming ordinary traffic rate to node h by informing all upstream sources that are using node h regarding the maximum allowed rate.

Note that in order to compare the first and third proposed robust control strategies, the simulations presented are conducted under the same conditions. In this section, two sets of simulations are presented using the third proposed robust control strategy. In the first set we implement a non-adaptive controller that has a complete knowledge about the input traffic while in the second set of simulations we consider an adaptive controller in which the input traffic is estimated according to the update law (5.14). For both sets of simulations, two topologies are considered, namely the three nodes cascade as well as the fully-connected topologies.

In the simulations presented below (a) the sensitivities to the time delays of 1 ms and 60 ms are evaluated, (b) the sampling time is set to $T_s = 1$ ms, (c) the control parameters are selected to be $\mu_{l_h} = 500$ and $\gamma_{l_h} = 5 * 10^{-3}$, (d) the mean of the random input traffic varies sinusoidally from 0 to $6 * 10^3$ packets/s with a frequency of 0.5 rad/s and variance of 10^3 packets/s, it should be noted that this input traffic corresponds to a "slowly time-varying" signal, and (e) the input traffic is not considered as a slowly time-varying according to the following specifications that is a frequency of 5 rad/s. Note also that the traffic is generated in the Simulink environment as a random signal with a Gaussian distribution.

Moreover, the input traffic is bounded such that $0 \leq \lambda_{l_h} \leq 4000$ packets/s. In compliance with Assumption 3.1, the buffer queue sizes are constrained according to the bounds $0 \leq x_{l_h} \leq 128$ packets while the maximum available service capacity is constrained according to Assumption 3.3 so that $0 \leq C_{l_h} \leq 4 * 10^4$ packet/s. Furthermore, the process function gains F_{lk}^{hj} are set equal to unity in all the links.

The first set of simulations is devoted to the implementation of the non adaptive version of the third robust control strategy on our network cluster that is configured first as a three nodes cascade and then as a fully-connected. For time-delays of 1 ms (solid line) and 60 ms (dashed line), the results are shown in Figures 5.5 and 5.6, respectively.

From graphs of Figure 5.5, we may observe that (a) the sensor, the decision maker DM21, and the actuator Act33 buffer queue lengths all converge to their respective references which are set to $[50, 60, 100]$ packets even in the presence of time delay of 60 ms, (b) the speed of the transient responses is good, and (c) the behavior of the queue state responses are very smooth in comparison with the oscillatory responses that are obtained with the first strategy.

The simulation results presented in Figure 5.6 correspond to the fully-connected network where the first row graphs of Figure 5.6 shows the queue state responses corresponding to (a) the sensor, (b) the decision-maker $DM21$, and (c) the second output port of the decision-maker called $DM22$. The second row graphs of Figure 5.6 are devoted to the three output ports of the actuator, namely (d) $Act31$, (e) $Act32$, and (f) $Act33$. As for the results corresponding to the cascade configuration, we can readily observe through the buffer queue lengths responses that convergence and good performance are achieved if the traffic is assumed to be measurable.

The second set of simulations are carried out for our network cluster that is configured first as a three nodes cascade and then as a fully-connected network using the adaptive version of the third proposed robust control strategy. The results are shown in Figures 5.7 and 5.8 for the time-delays of 1 ms (solid line) and 60 ms

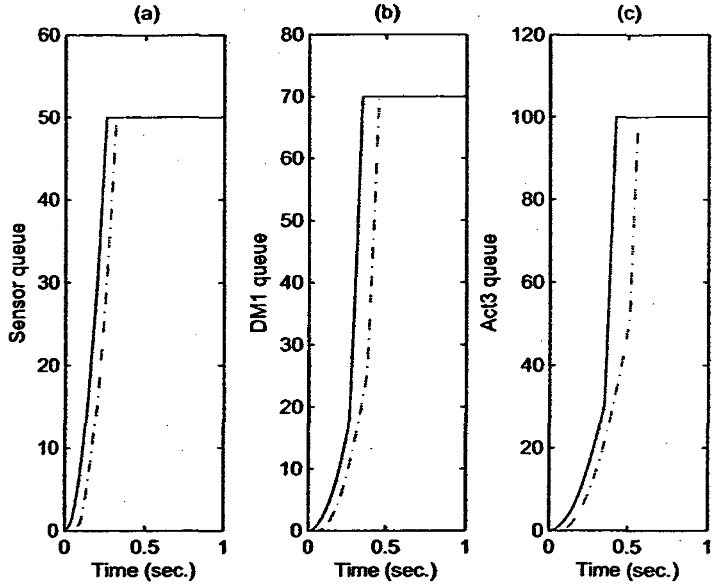


Figure 5.5: Simulation results for the cascade configuration in presence of time delays (solid line: 1 ms, dashed line: 60 ms) and dynamic source traffics by using the third proposed non-adaptive strategy (measured traffic).

(dashed line).

From the graphs of Figure 5.7 corresponding to the cascade configuration, we may assert that (a) the sensor, the decision maker DM21, and the actuator Act33 buffer queue lengths all converge to their respective references which are set to $[50, 60, 100]$ packets even in presence of time delay of 60 ms, (b) the transient responses show overshoots due to the uncertainties in the estimation of the input traffic, (c) the steady state is free of oscillations, and (d) the settling time is longer compared to the first strategy as expected.

For the simulation results presented in Figure 5.8 corresponding to the fully-connected network, one may make the same conclusions as for the above cascade configuration, namely that the convergence of all the buffer queue lengths to their respective references is achieved, and the transients show overshoots and longer

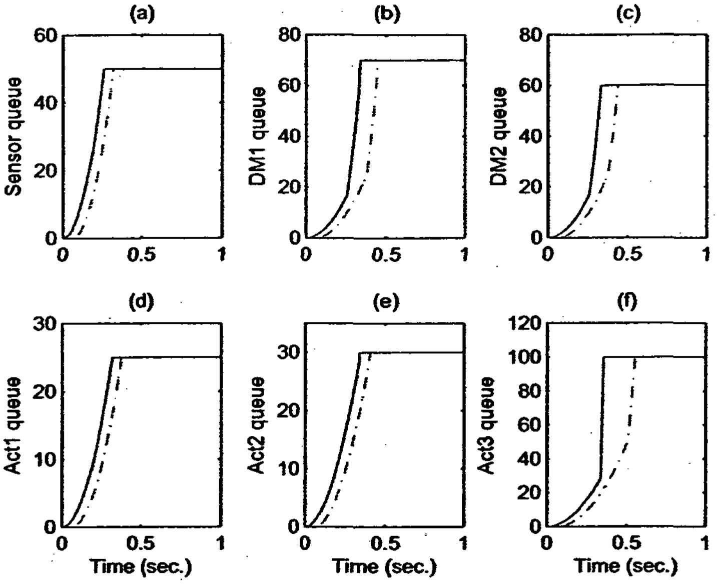


Figure 5.6: Simulation results for the fully-connected configuration in presence of time delays (solid line: 1 ms, dashed line: 60 ms) and dynamic source traffics by using the third proposed non adaptive strategy (measured traffic).

settling times as compared to the first strategy as expected.

The last simulation are conducted for a non slowly time-varying input traffic. In effect, by setting (a) the frequency of the sinusoidal input traffic to 5 rad/s, (b) the time-delay to 60 ms, and (c) the operating points to 50 packet for the sensor, to [60 70] packet for the first and second outports of the decision-maker, respectively, and to [25 30 100] packet for the first, second, and third output ports of the actuator, respectively, the simulation results obtained for the fully-connected configuration by using the third proposed adaptive strategy are presented in Figure 5.9. The results show that our third proposed strategy also holds for a non slowly time-varying input traffic even though overshoots due to the estimation can be observed.

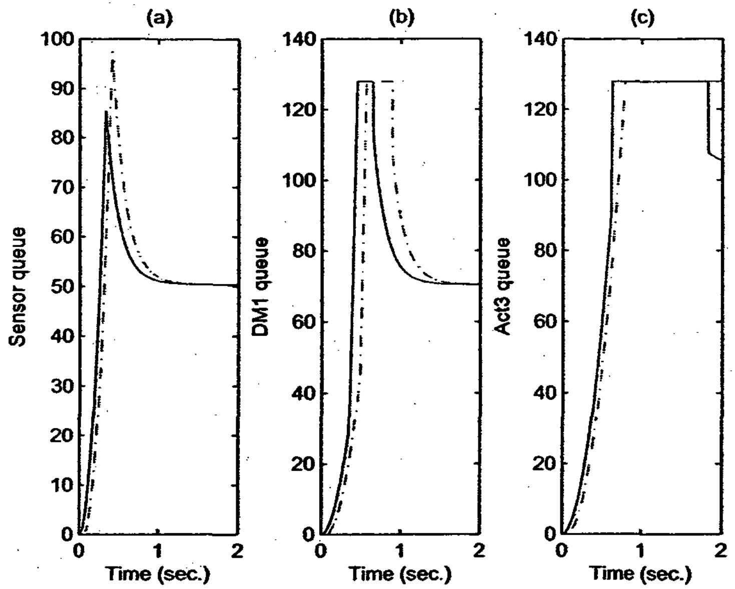


Figure 5.7: Simulation results for the cascade configuration in presence of time delays (solid line: 1 ms, dashed line: 60 ms) and dynamic source traffics by using the third proposed adaptive strategy (estimated traffic).

5.5 Conclusion

This chapter has focused on proposing three robust control strategies for integrating the bandwidth allocation and the flow rate control objectives. The proposed controllers attempts of on one hand to maintain the average queue length of each buffer of the network by dynamically allocating the bandwidth to the server while on the other hand to control the flow rate by notifying all upstream sources regarding the maximum allowed traffic rate.

The first robust control strategy is an improved version of the strategy that is proposed in [91]. This strategy is empowered by robustness capabilities and constitutes a benchmark solution for the third strategy. Indeed, the first and third control strategies are compared in simulations and the conclusions can be stated as follows, a) the first strategy as well as the second strategy do not need any

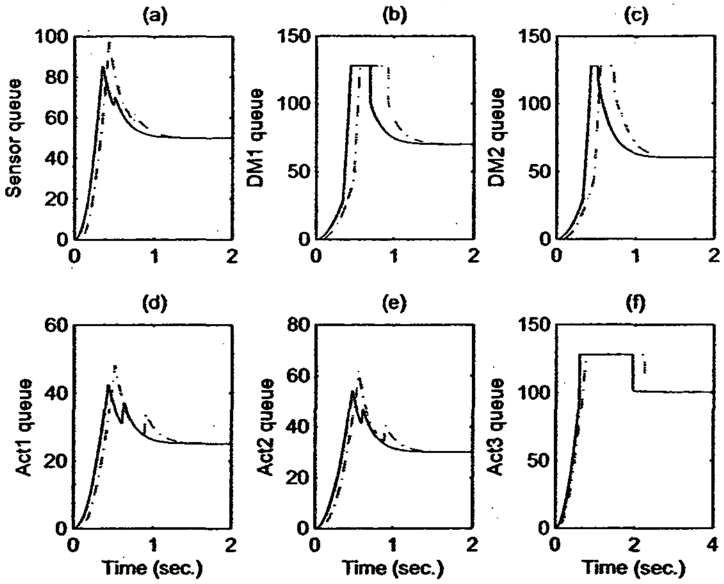


Figure 5.8: Simulation results for the fully-connected configuration in presence of time delays (solid line: 1 ms, dashed line: 60 ms) and dynamic source traffics by using the third proposed adaptive strategy (estimated traffic).

knowledge about the incoming traffic in order to compute their commands while the third strategy uses the estimated input traffic in its control law, b) using the first and third strategies for the network cluster configured as a cascade or a fully-connected network, all buffer queue lengths converge to their respective references even in presence of time delay of 60 ms, c) the buffer queue lengths responses that are obtained by using the first strategy are very oscillatory while those obtained by using the third strategy are oscillation-free, and d) by using the third strategy, the results that are obtained when the input traffic is assumed to be measurable are better than those when the input traffic is estimated (resulting in overshoots and long settling times). It should be noted that by adding the σ-modification [57, 64] to the adaptation law, the third strategy would constitute a significant step toward an efficient robust integrated algorithm as discussed in subsequent chapters.

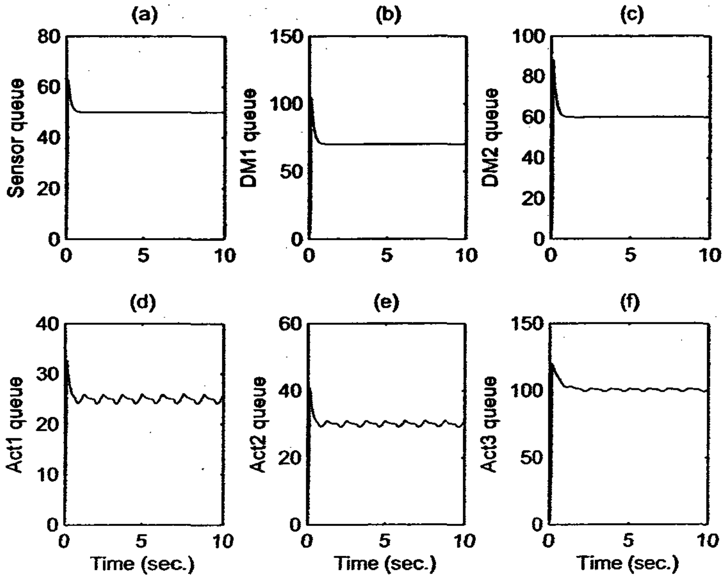


Figure 5.9: Simulation results for the fully-connected configuration in presence of a time-delay of 60 ms and dynamic source traffics (frequency=5 rad/s) by using the third proposed adaptive strategy (estimated traffic).

Chapter 6

SM-GVS Congestion Control for DiffServ Networks: Non-Degenerate Case

6.1 Introduction

Despite efforts that have been made by utilizing advanced control strategies such as optimal control, adaptive control, linear and nonlinear control [15], [80], [89], [94], [101], to name a few, to improve the network performance, the most commonly used congestion control methods in the literature are based on intuition and ad hoc techniques. Consequently, robustness of the congestion control problem remains unaddressed adequately. Our investigation in this chapter focuses on an interesting congestion control approach that has been studied by Pitsillides et. al. [89] (see Chapter 2). This methodology consists of development and analysis of a generic Integrated Dynamic Congestion Control (IDCC) scheme for controlling traffic using information on the status of each queue in the buffer. Pitsillides's control approach is based on a feedback linearization technique and can be classified as a network assisted congestion controller [71].

In contrast, the proposed control law introduced in this chapter is a kind of feedback linearization that is empowered by strong robustness features. In effect, in order to ensure the necessary robustness requirements of the congestion control, we use the sliding mode principle. The latter principle is basically to drive the nonlinear plant operating point along or nearby the vicinity of the specified and user-chosen hyperplane where it slides until it reaches the origin, by means of certain high-frequency switching control law. Once the system reaches the hyperplane, its order is reduced since it depends only on the hyperplane dynamics.

The existence of the sliding mode in a manifold is due to the discontinuous nature of the variable structure control which is switching between two distinctively different system structures. Such a system is characterized by an excellent performance, which includes insensitivity to parameter variations and a complete rejection of disturbances. However, since this switching could not be practically implemented with an infinite frequency as required for the ideal sliding mode, the discontinuity generates a chattering in the control, which may unfortunately excite high-frequency dynamics that are neglected in the model and thus might damage the actual physical system.

In view of the above, the sliding mode-variable structure control (SM-VSC) was restricted in practical applications until progresses in the electronics area and particularly in the switching devices in the nineteen seventies. Since then, the SM-VSC has reemerged with several advances for alleviating the undesirable chatter phenomenon. Among the main ideas is the approach based on the equivalent control component which is added to the discontinuous component [49], [116]. In fact, depending on the model parameters, the equivalent control corresponds to the SM existence condition. Second, the approach studied in [109] consists of the allocation of a boundary layer around the switching hyperplane in which the discontinuous control is replaced by a continuous one. In [10], [50] the gain of the discontinuous component is replaced by a linear function of errors. In [46], the authors propose a technique by replacing in the variable structure control the sliding mode by a sliding sector.

Most recent approaches consider that the discontinuity occurs at the highest derivatives of the control input rather than the control itself. These techniques can be classified as a higher-order sliding mode approaches in which the state equation is differentiated to produce a differential equation with the derivative of the control input [8] and [77]. Among them, a particular approach that is introduced in [44] and investigated in [16], [24], [95] uses differential algebraic mathematical tools. Indeed, by using the differential primitive element theorem in case of nonlinear systems and the differential cyclic element theorem in case of linear systems, this technique transforms the system dynamics into a new state space representation where the derivatives of the control inputs are involved in the generalization of the system representation. By invoking successive integrations to recover the actual control the chattering of the so-called Generalized Variable Structure (GVS) control is filtered out. In the following two sections, the sliding mode variable structure and input-output linearization nonlinear control concepts are utilized to design a congestion controller.

6.2 Proposed GVS Congestion Control Strategy

In this chapter, we utilize a sliding mode GVS machinery to design our proposed congestion control strategy. Basically, the GVS control approach consists of the transformation of the system dynamics into a new type of state representation exhibiting the derivatives of the control input and to which the output equation is associated to give what are called Generalized Observable and Controllable Canonical Forms (GOCF, GCCF). On the basis of these canonical forms, the control law might be derived using different approaches. Three control approaches are investigated in this chapter. The first approach consists of solving the well-known sliding condition inequality ($S\dot{S} < 0$). The second approach that is introduced in [82] uses the feedback control given by (2.20). The last approach considered to design our GVS congestion control law utilizes the hyperplane convergence equation as a nonlinear feedback control [95]. It is worth noting that the latter approach is one that have yielded the best results in terms of reducing chattering phenomena, robustness and performance attributes [24].

6.2.1 New fluid flow model in generalized observable canonical form

In order to obtain the generalized observable canonical form for the fluid flow model with derivatives of control variables, let us first rewrite for the h^{th} node of the DiffServ network, the FFM model that is given by equation (3.1) as follows

$$\begin{cases} \dot{x}_{hf}(t) = -C_{hf}(t)\dfrac{x_{hf}(t)}{x_{hf}(t)+1} + \lambda_{ihf} \\ y_{hf}(t) = x_{hf}(t) \end{cases} \tag{6.1}$$

Let us first transform the FFM given by equation (6.1) into a new FFM model (6.2) using the following variable change, $x_{1hf} = x_{hf}$ and $x_{2hf} = \frac{x_{hf}}{x_{hf}+1}$, with $f = p, r$ representing the Premium and the Ordinary services, respectively. Consequently, we now have the following new FFM model

$$\begin{cases} \dot{x}_{1hf} = -C_{hf}x_{2hf} + \lambda_{hf} \\ \dot{x}_{2hf} = \dfrac{-C_{hf}x_{2hf} + \lambda_{hf}}{(x_{1hf}+1)^2} \\ y_{hf} = x_{1hf} \end{cases} \tag{6.2}$$

To verify that the obtained second-order FFM model (6.2) corresponds to the original first-order FFM (2.1), simulations are conducted in which both models are subjected to the same sine wave input traffic and the results are shown in Figure 6.1. In the first row of Figure 6.1, the left and the right graphs depict the queue state for both models and their error, respectively. From these two graphs, one can argue that for a sine wave input traffic, the obtained second-order FFM model responds similarly to the original first-order FFM model.

Through the second row graphs, one may observe that both the output traffic $O1$ of the original first-order FFM (left graph) and the output traffic $O2$ of the obtained second-order FFM (right graph) models converge to the common input traffic $flow\ I$, verifying the flow conservation principle. This is also confirmed through the left graph of the third row, in which the output traffics $O1$ and $O2$ converge to each other. The last graph in the right depicts the second state x_{2hf}.

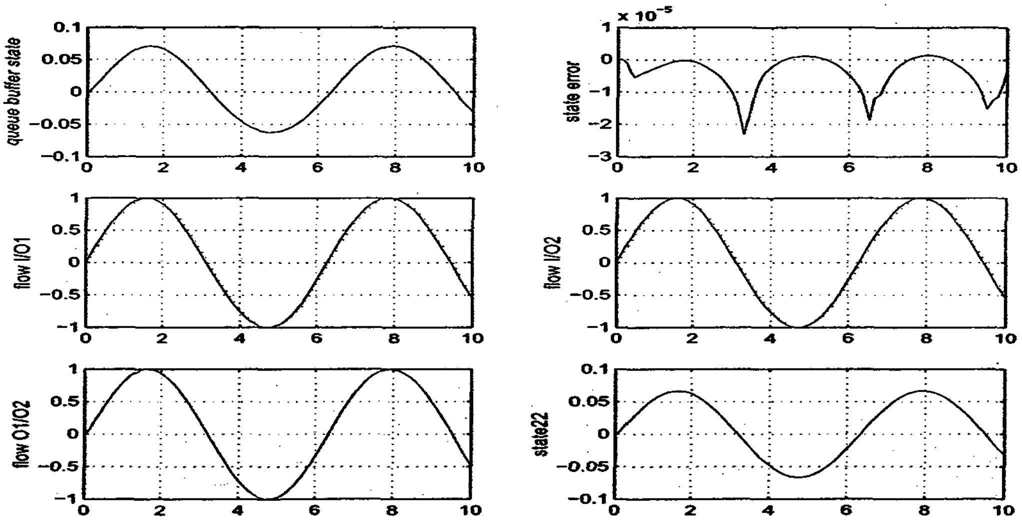


Figure 6.1: Validation of the second-order model (6.2) due to a sinusoidal input traffic.

The second step required for obtaining the generalized observable canonical form (GOCF) is to derive a corresponding differential equation for model (6.2). The elimination of the state from the model (6.2) can be performed first by computing the derivative of y_{hf} as $\dot{y}_{hf} = \dot{x}_{1hf}$ to deduce the state $x_{2hf} = \frac{\lambda_{ihf}-y_{hf}}{C_{hf}}$ and then the derivative of the latter is equated with the second term of (6.2) to obtain the following differential equation

$$\ddot{y}_{hf} = \dot{\lambda}_{ihf} - \frac{\dot{C}_{hf}}{C_{hf}}(\lambda_{ihf} - \dot{y}_{hf}) - C_{hf}\frac{y_{hf}}{(y_{hf}+1)^2} \tag{6.3}$$

which is in the form of (2.17) since the control input u_{hf} corresponds to C_{hp} for the Premium service and to C_{hr} and λ_{ihr} for the Ordinary service, respectively, that is

$$\zeta(y_{hf}, \dot{y}_{hf}, \ddot{y}_{hf}, u_{hf}, \dot{u}_{hf}) = 0 \tag{6.4}$$

According to the rank condition (2.16), one may argue that for both the premium and the ordinary buffer dynamics, the relative degree is $r_{d,hf} = 1$, the order of the highest derivative of the output $d_{hf} = 2$, and the order of the highest derivative of the control input $\alpha_{i,hf} = 1$. Therefore, the condition $\alpha_{i,hf} = d_{hf} - r_{d,hf}$ is verified as well as the rank condition (2.16) that is also verified, as shown below

$$rank\frac{\delta(h_{s,hf}, \dot{h}_{s,hf})}{\delta x_{hf}} = rank\frac{\delta(h_{s,hf}, \dot{h}_{s,hf}, \ddot{h}_{s,hf})}{\delta x_{hf}} = 2 \tag{6.5}$$

Introducing a new set of state variables that are given according to

$$\begin{cases} z_{1hf} = y_{hf} \\ z_{2hf} = \dot{y}_{hf} = \dot{z}_{1hf} \\ \dot{z}_{2hf} = \ddot{y}_{hf} \end{cases} \tag{6.6}$$

and since the Jacobian $\frac{\delta\zeta}{\delta\ddot{y}_{hf}}$ is non singular and it is equal to identity, the transition from the implicit input-output representation to an explicit one can be accomplished.

By now substituting the new variables (6.6) into equation (6.3), the GOCF representation is obtained as follows

$$(GOCF)\begin{cases} \dot{z}_{1hf} = z_{2hf} \\ \dot{z}_{2hf} = \dot{\lambda}_{ihf} + \frac{\dot{C}_{hf}}{C_{hf}}(\lambda_{ihf} - z_{2hf}) - C_{hf}\frac{2hf}{(z_{1hf}+1)^2} \\ y_{hf} = z_{1hf} \end{cases} \tag{6.7}$$

To validate the transition to the GOCF model, the latter model given by equation (6.7) is compared with the second-order FFM model (6.2) and the results are shown in Figure 6.2. It can be observed through the first row graphs that the buffer queue corresponding to the GOCF model tracks the buffer queue quiet good corresponding to the model given by (6.2).

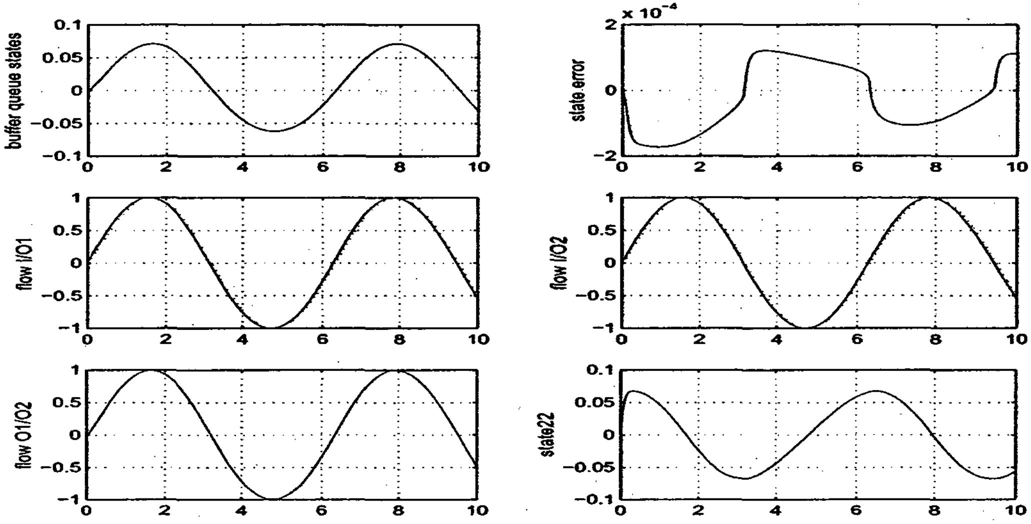


Figure 6.2: Validation of the GOCF model (6.7) due to a sinusoidal input traffic.

Now in order to design the controller in the error state space, let us consider the following error variables as shown in (6.8):

$$\begin{cases} e_{1hf} = z_{1hf} - z_{hf_{ref}} \\ \dot{e}_{1hf} = e_{2hf} = \dot{z}_{1hf} - \dot{z}_{hf_{ref}} = z_{2hf} - \dot{z}_{hf_{ref}} \\ \dot{e}_{2hf} = \dot{z}_{2hf} - \ddot{z}_{hf_{ref}} \end{cases} \tag{6.8}$$

By substituting the above state error variables into equation (6.7), we obtain the GOCF model in the error state space, that is

$$GOCF\begin{cases} \dot{e}_{1hf} = e_{2hf} \\ \dot{e}_{2hf} = \dot{\lambda}_{ihf} - \dfrac{\dot{C}_{hf}}{C_{hf}}(\lambda_{ihf} - e_{2hf} - \dot{z}_{hf_{ref}}) - C_{hf}\dfrac{e_{2hf} + \dot{z}_{hf_{ref}}}{(e_{1hf} + z_{hf_{ref}} + 1)^2} - \ddot{z}_{hf_{ref}} \\ y_{hf} = e_{1hf} + z_{hf_{ref}} \end{cases} \tag{6.9}$$

Considering the resulting error state representation of the GOCF model given by (6.9), the objective of the proposed congestion control is to ensure robustness of the closed-loop system to uncertainties and unmodeled dynamics as well as external disturbances. This will be the case provided that for the above system (6.9), a sliding mode exists in the neighborhood of the switching surface as given by (6.10a) and (6.10b)

$$S_{hf} = e_{n,hf} + \sum_{j=1}^{n-1} s_{j,hf}e_{j,hf} = s_{1hf}e_{1hf} + \dot{e}_{1hf} = s_{1hf}e_{1hf} + e_{2hf};\ (f = p, r) \tag{6.10a}$$

$$\dot{S}_{hf} = s_{1hf}\dot{e}_{1hf} + \dot{e}_{2hf} = s_{1hf}e_{2hf} + \dot{e}_{2hf} \tag{6.10b}$$

where s_{1hf}, $(f = p, r)$ designates two positive parameters corresponding to the slope of the sliding surfaces of the premium and the ordinary dynamics, respectively.

It is worth mentioning that the state representation of the system obtained with the derivatives of the control input constitutes the core of the GVS control design procedure. Using the common state representation (6.9), several GVS control approaches can be studied. Three approaches can be considered for deriving the GVS congestion control algorithm, namely

a) by solving the well-known sliding condition $S_{hf}\dot{S}_{hf} < 0$ after substitution of (6.9),

b) by considering the feedback control (2.20) that is introduced in [82], and

c) by using what is denoted as the hypersurface convergence equation

$$\dot{S}_{hf} + \mu_{hf}S_{hf} + \gamma_{hf}sgn(S_{hf}) = 0 \tag{6.11}$$

where $\gamma_{hf} = \mu_{hf}\Omega_{hf}$ with $f = p, r$. Note that Ω_{hf} is a gain designating the proportionality between γ_{hf} and μ_{hf}.

6.2.2 GVS congestion controller design using the sliding condition [15]

Using this control approach, the control variables are $\dot{C}_{hp}$ and $\dot{\lambda}_{hr}$ that should satisfy the sliding condition $S_{hf}\dot{S}_{hf} < 0$, where S_{hf} and $f = p, r$ is the sliding surface given by (6.10a), such that the control law designed according to (6.12) holds. Note that by taking into account the leftover capacity $C_{hr}(t) = C_{serv} - C_{hp}(t)$, the second equation will clearly indicate the control coupling, namely

$$\begin{cases} \dot{C}_{hp} = \frac{C_{hp}}{(\lambda_{ihp} - e_{2hp} - \dot{z}_{hp_{ref}})}[\dot{\lambda}_{ihp} - C_{hp}\frac{e_{2hp} + \dot{z}_{hp_{ref}}}{(e_{1hp} + z_{hp_{ref}} + 1)^2} - \ddot{z}_{hp_{ref}} + s_{1hp}e_{2hp} + b_{hp}sign(S_{hp})] \\ \dot{\lambda}_{ihr} = \frac{\dot{C}_{hr}}{C_{hr}}[\lambda_{ihr} - (e_{2hr} + \dot{z}_{hr_{ref}})] + C_{hr}\frac{e_{2hr} + \dot{z}_{hr_{ref}}}{(e_{1hr} + z_{hr_{ref}} + 1)^2} + \ddot{z}_{hr_{ref}} - s_{hr}e_{2hr} - b_{hr}sgn(S_{hr}) \end{cases} \tag{6.12}$$

In the sliding mode, the error dynamics driven by the canonical form (GOCF) (6.9) and the control law given by equation (6.12) result in the closed-loop in the so-called reduced-order and free error dynamics. Indeed, as expressed by equation (6.13), the resulting dynamics depends only on the stable sliding hyperplane

$$\begin{cases} \dot{e}_{1hf} = e_{2hf} \\ \dot{e}_{2hf} = -s_{1hf}e_{2hf} - b_{hf}sgn(S_{hf}) \end{cases} \tag{6.13}$$

Theorem 6.1. *For a differentiated-services network where the h^{th} node is described by the Fluid Flow Model* (6.1), *by adopting the decentralized control approach, the congestion control law governed by equation* (6.12) *guarantees the asymptotic stability of the error dynamics* (6.13) *provided that s_{1hf}, b_{hf} $(f = p, r)$ are selected as positive design parameters.*

Proof: Choose a candidate Lyapunov function as $V(S_{hf}, t) = \frac{S_{hf}^2(t)}{2}$ such that for all $S_{hf}(t) \neq 0 \Rightarrow V(S_{hf}) > 0$. The time derivative of V along the trajectories of

(6.13) can be obtained as follows

$$\dot{V}(S_{hf},t) = S_{hf}\dot{S}_{hf} < 0 \tag{6.14}$$

Corresponding to the sliding condition, $\dot{V}(S_{hf})$ is strictly negative definite provided that the design parameters s_{1hf}, b_{hf} $(f = p, r)$ are selected to be positive, and hence this completes the proof. ▲

6.2.3 GVS congestion controller design using the feedback control introduced in [82]

Let us consider for the nonlinear second-order FFM model (6.7) the feedback control (2.20) that can be written for a second order system as follows

$$\dot{z}_{2hf} = -\sum_{j=1}^{2} a_{j,hf} z_{j,hf} - \sum_{j=1}^{2} b_{j,hf} v_{j,hf} = -a_{1hf} z_{1hf} - a_{2hf} z_{2hf} - b_{hf} v_{hf} \tag{6.15}$$

where $v_{hf} = sgn(S_{hf}) = sgn(s_{1hf} e_{1hf} + \dot{e}_{1hf})$ is the new input, which permits us to deduce the control law for both the premium and the ordinary traffics given by

$$\begin{cases} \dot{C}_{hp} = \frac{C_{hp}}{(\lambda_{ihp} - e_{2hf} - \dot{z}_{hp_{ref}})}[\dot{\lambda}_{ihp} - C_{hp}\frac{e_{2hp} + \dot{z}_{hp_{ref}}}{(e_{1hp} + z_{hp_{ref}} + 1)^2} - \ddot{z}_{hp_{ref}} + a_{1hp} e_{1hp} + a_{2hp} e_{2hp} + b_{hp} v_{hp}] \\ \dot{\lambda}_{ihr} = \frac{\dot{C}_{hr}}{C_{hr}}(\lambda_{ihr} - e_{2hr} - \dot{z}_{hr_{ref}}) + C_{hr}\frac{e_{2hr} + \dot{z}_{hr_{ref}}}{(e_{1hr} + z_{hr_{ref}} + 1)^2} + \ddot{z}_{hr_{ref}} - a_{1hr} e_{1hr} - a_{2hr} e_{2hr} - b_{hr} v_{hr} \end{cases} \tag{6.16}$$

The closed-loop error dynamics driven by the canonical form (GOCF) (6.9) and the control law given by equation (6.16) is given by

$$\dot{e}_{hf} = -A_{hf} e_{hf} - B_{hf} v_{hf} \tag{6.17}$$

where $e_{hf} = [e_{1hf}\ e_{2hf}]^T$, $B_{hf} = [0\ b_{hf}]^T$, $v_{hf} = sgn(S_{hf})$, and.

$$A_{hf} = \begin{pmatrix} 0 & 1 \\ a_{1hf} & a_{2hf} \end{pmatrix} \tag{6.18}$$

Theorem 6.2. *For a differentiated-services network where the h^{th} node is described by the Fluid Flow Model* (6.1)*, by adopting the decentralized control approach, the congestion control law governed by equation* (6.16) *guarantees the asymptotic stability of the error dynamics* (6.17) *provided that $a_{j,hf}$ and b_{hf} ($j = 1, 2,\ f = p, r$) are selected as positive design parameters.*

Proof: It is well-known that our resulting closed-loop error dynamics given by equation (6.17) corresponds to a standard linear system whose asymptotic stability is guaranteed provided that matrix A_{hf} is Hurwitz, and since the external input v_f corresponding to the signum function is always bounded. This completes the proof. ▲

6.2.4 GVS congestion controller design using the hypersurface convergence equation [17]

Among several sliding mode control approaches that we investigated in our previous research [24], the approach that uses the hypersurface convergence equation have given the best simulation and experimental results in terms of performance and disturbance rejection. The control design consists of solving for the hypersurface convergence equation (6.11) whose solution is the sliding surface S_{hf}.

This equation can be written in an explicit form by substituting in (6.11) equations (6.10a) and (6.10b), that is

$$\dot{e}_{2hf} = -s_{1hf}e_{2hf} - \mu_{hf}(\Omega_{hf}sgn(S_{hf}) + S_{hf}); \qquad (f = p, r) \qquad (6.19)$$

Let us now consider equation (6.19) as a feedback control to model (6.9), that is

$$\begin{cases} \dot{e}_{2hf} = \dot{\lambda}_{ihf} - \dfrac{\dot{C}_{hf}}{C_{hf}}(\lambda_{ihf} - e_{2hf} - \dot{z}_{hf_{ref}}) - C_{hf}\dfrac{e_{2hf} + \dot{z}_{hf_{ref}}}{(e_{1hf} + z_{hf_{ref}} + 1)^2} - \ddot{z}_{hf_{ref}} \\ \quad = -s_{1hf}e_{2hf} - \mu_{hf}(\Omega_{hf}sgn(S_{hf}) + S_{hf}) \\ \quad = -(s_{1hf} + \mu_{hf})e_{2hf} - \mu_{hf}s_{1hf}e_{1hf} - \mu_{hf}\Omega_{hf}sgn(S_{hf}) \\ \quad = \displaystyle\sum_{j=1}^{2} a_{j,hf}e_{j,hf} + v_{hf} \end{cases} \tag{6.20}$$

where $a_{1hf} = -\mu_{hf}s_{1hf}$, $a_{2hf} = -(s_{1hf} + \mu_{hf})$ and $v_{hf} = b_{hf}sgn(S_{hf})$ with $b_{hf} = -\mu_{hf}\Omega_{hf}$ and again $f = p, r$.

One may observe that the above corresponds to the feedback control (2.20) with convergence tuning parameters. Indeed, $\sum_{j=1}^{2} a_{j,hp}e_{j,hp} + v_{hp}$ and $\sum_{j=1}^{2} a_{j,hr}e_{j,hr} + v_{hr}$ are the resulting linearized systems for the Premium and the Ordinary services, respectively. The coefficients $a_{j,hf}$ with $f = p, r$ are chosen such that stability of the closed-loop system will be ensured. Consequently, by taking into account the leftover capacity $C_{hr}(t) = C_{serv} - C_{hp}(t)$, where C_{serv} is the maximum available capacity of the server, the above feedback control leads to the law as specified below

$$\begin{cases} \dot{C}_{hp} = \dfrac{C_{hp}}{\lambda_{ihp} - e_{2hp} - \dot{z}_{hp_{ref}}}[\dot{\lambda}_{ihp} - C_{hp}\alpha_{hp} - \ddot{z}_{hp_{ref}} + \beta_{hp}e_{2hp} + \xi_{hp}e_{1hp} + \gamma_{hp}sgn(S_{hp})] \\ \dot{\lambda}_{ihr} = \dfrac{\dot{C}_{hr}}{C_{hr}}(\lambda_{ihr} - e_{2hr} - \dot{z}_{hr_{ref}}) + C_{hr}\alpha_{hr} + \ddot{z}_{hr_{ref}} - \beta_{hr}e_{2hr} - \xi_{hr}e_{1hr} - \gamma_{hr}sgn(S_{hr}) \end{cases} \tag{6.21}$$

where $\alpha_{hf} = \frac{(e_{2hf} + \dot{z}_{hf_{ref}})}{(e_{1hf} + z_{hf_{ref}} + 1)^2}$, $\beta_{hf} = s_{1hf} + \mu_{hf}$, $\gamma_{hf} = \mu_{hf}\Omega_{hf}$, and $\xi_{hf} = \mu_{hf}s_{1hf}$.

Theorem 6.3. *For a differentiated-services network where the h^{th} node is described by the Fluid Flow Model* (6.1), *by adopting the decentralized control approach, the congestion control law governed by equation* (6.21) *guarantees the asymptotic stability of the error dynamics* (6.20) *provided that s_{1f}, μ_f, and Ω_f $(f = p, r)$ are selected as positive design parameters.*

Proof: In equation (6.20), we have shown that the resulting closed-loop error dynamics can be expressed by the same linear form as given by equation (6.15) which is asymptotically stable provided that s_{1f}, μ_f, and Ω_f $(f = p, r)$ are selected to be positive. This completes the proof. ▲

The above control approach that is designed on the basis of the second-order FFM model (6.9) is shown to be similar to the feedback linearization control approach with some improvement for adjusting the design parameters. However, the considered approach can be also seen as an alternative to the following classical input-output linearization control approach that is empowered by robustness capabilities. In fact, note that several issues can be investigated within the generalized variable structure context. For example, in comparison with the normal form that is utilized in nonlinear control [58] in which the internal and the external dynamics are represented separately, the canonical form of the GOCF does not represent explicitly the zero dynamics of the plant. In fact, this is expected since the GOCF exhibits the derivatives of the control inputs. One issue that would be interesting to investigate is to compare the performance and robustness capabilities of the obtained GVS congestion controller with the input-output linearization-based congestion controller when the non-minimal realization model (6.2) is considered. The latter controller is designed below.

6.2.4.1 Input-output linearization-based congestion control for the second-order fluid flow model

The obtained FFM model (6.2) can be expressed for both the premium and the ordinary dynamics by using the following conventional form

$$\begin{cases} \dot{x}_h = f_{s,h}(x_h) + g_{s,h}(x_h)U(x_h) + \delta_{sh} \\ y_h(x_h) = \dot{h}_{s,h}(x_h) \end{cases} \tag{6.22}$$

where index s designates *system* as stated in Chapter 2 (refer to equation (2.15)), $x_h = [\dot{x}_{1hp}\ \dot{x}_{2hp}\ \dot{x}_{1hr}\ \dot{x}_{2hr}]^T$, $y_h = [y_{hp}\ y_{hr}]^T$, $f_{s,h}(x_h) = [0\ \ \frac{\lambda_{1hp}}{(x_{1hp}+1)^2}\ \ -x_{2hr}C_{serv}\ \ -$

$\frac{x_{2hr}C_{serv}}{(x_{1hr}+1)^2}]^T$, $U_h = [C_{hp}\ \lambda_{ihr}]^T$, $\delta_h = [\lambda_{ihp}\ 0\ 0\ 0]^T$,

$$f_{s,h}(x_h) = \begin{pmatrix} -x_{2hp} & 0 \\ (-x_{2hp})(x_{1hp}+1)^{-2} & 0 \\ x_{2hr} & 1 \\ (x_{2hr})(x_{1hr}+1)^{-2} & (x_{1hr}+1)^{-2} \end{pmatrix}, h_{s,h}(x_h) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} x_h \tag{6.23}$$

Let us now consider the input-output linearization control for model (6.22). let $r_{d,h}$ denotes the relative degree. We have $\dot{y}_h = \dot{h}_{s,h}(x_h) \Rightarrow$

$$\begin{cases} \dot{y}_{hp} = \dot{x}_{1hp} = -C_{hp}x_{2hp} + \lambda_{1hp} \\ \dot{y}_{hr} = \dot{x}_{1hr} = -C_{hr}x_{2hr} + \lambda_{1hr} \\ \qquad = -C_{serv}x_{2hr} + C_{hp}x_{2hr} + \lambda_{1hr} \end{cases} \tag{6.24}$$

As both the control variables appear in the first derivative, then $r_{d,h} = 2 < n_h = 4$ implying the existence of an internal dynamic.

The general form of the coordinate transformation equation (6.25) in which L designates the Lie derivative is given by

$$\Phi(x_h) = \begin{pmatrix} \Phi_1(x_h) = h_{s,h}(x_h) \\ \Phi_2(x_h) = L_{f_{s,h}}h_{s,h}(x_h) \\ \dots \\ \Phi_{r_d,h}(x_h) = L_{f_{s,h}}^{r_d-1}h_{s,h}(x_h) \\ \Phi_n(x_h) \end{pmatrix} \tag{6.25}$$

which can be expressed for our 4th order system dynamics as follows

$$\Phi(x_h) = \begin{pmatrix} \Phi_1(x_h) = h_{s,h}(x_h) \\ \Phi_2(x_h) = \Phi_{r_d,h}(x_h) = L_{f_s,h}h_{s,h}(x_h) \\ \Phi_3(x_h) = \Phi_{r_d+1}(x_h) \\ \Phi_4(x_h) = \Phi_n(x_h) \end{pmatrix} \tag{6.26}$$

Let us now introduce a new set of variables as given below

$$\begin{cases} \Phi_1(x_h) = y_{hp} = z_{1h} \\ \Phi_2(x_h) = y_{hr} = z_{2h} \\ \Phi_3(x_h) = z_{3h} \\ \Phi_4(x_h) = z_{4h} \end{cases} \tag{6.27}$$

As far as the two variables z_{3h} and z_{4h} are considered, we need to find any function that will be the solution to the equations $z_{j,h} \rightarrow L_{g_{s,h}}\Phi_j = 0$, with $j = 3,4$ which are explicitly given as follows

$$L_{g_{s,h}}\Phi_j = \begin{cases} \dfrac{\partial\Phi_j}{\partial x_{1hp}} g_{11,s,h}(x_h) + \dfrac{\partial\Phi_j}{\partial x_{2hp}} g_{21,s,h}(x_h) + \dfrac{\partial\Phi_j}{\partial x_{1hr}} g_{31,s,h}(x_h) + \dfrac{\partial\Phi_j}{\partial x_{2hr}} g_{41,s,h}(x_h) = 0 \\ \dfrac{\partial\Phi_j}{\partial x_{1hp}} g_{12,s,h}(x_h) + \dfrac{\partial\Phi_j}{\partial x_{2hp}} g_{22,s,h}(x_h) + \dfrac{\partial\Phi_j}{\partial x_{1hr}} g_{32,s,h}(x_h) + \dfrac{\partial\Phi_j}{\partial x_{2hr}} g_{42,s,h}(x_h) = 0 \end{cases} \tag{6.28}$$

Now after replacing all $g_{ij,s,h}$ components in equation (6.28), the resulting equation (6.29) below

$$L_{g_s}\Phi_j = \begin{cases} -\dfrac{\partial\Phi_j}{\partial x_{1hp}} x_{2hp} - \dfrac{\partial\Phi_j}{\partial x_{2hp}} \dfrac{x_{2hp}}{(x_{1hp}+1)^2} + \dfrac{\partial\Phi_j}{\partial x_{1hr}} x_{2hr} + \dfrac{\partial\Phi_j}{\partial x_{2hr}} \dfrac{x_{2hr}}{(x_{1hr}+1)^2} = 0 \\ \dfrac{\partial\Phi_j}{\partial x_{1hr}} + \dfrac{\partial\Phi_j}{\partial x_{2hr}} (x_{1hr}+1)^{-2} = 0 \end{cases} \tag{6.29}$$

has the following solutions

$$\begin{cases} \Phi_3(x_h) = x_{2hr} + (x_{1hr}+1)^{-1} \\ \Phi_4(x_h) = x_{2hp} + (x_{1hp}+1)^{-1} + x_{2hr} + (x_{1hr}+1)^{-1} \end{cases} \tag{6.30}$$

The transformation $\phi(x_h)$ is thus completed by $z_{3h} = \phi_3(x_h)$ and $z_{4h} = \phi_4(x_h)$, whose diffeomorphism exists since the rank of the Jacobian J is full and the determinant is non zero ($det(J) = -1$).

$$J = \frac{\partial \Phi}{\partial x} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -(x_{1hr}+1)^{-2} & 1 \\ -(x_{1hp}+1)^{-2} & 1 & -(x_{1hr}+1)^{-2} & 1 \end{pmatrix} \tag{6.31}$$

This effectively confirms the existence of the transformation which can be written as

$$\begin{cases} x_{1hp} = z_{1h} \\ x_{2hp} = z_{4h} - z_{3h} - (z_{1h}+1)^{-1} \\ x_{1hr} = z_{2h} \\ x_{2hr} = z_{3h} - (z_{2h}+1)^{-1} \end{cases} \tag{6.32}$$

Using the new variables (6.32), our overall system can be expressed as follows

$$\begin{cases} \dot{z}_{1h} = -C_{hp}(z_{4h} - z_{3h} - (z_{1h}+1)^{-1}) + \lambda_{ihp} \\ \dot{z}_{2h} = -C_{serv}[z_{3h} - (z_{2h}+1)^{-1}] + C_{hp}[z_{3h} - (z_{2h}+1)^{-1}] + \lambda_{ihr} \\ \dot{z}_{3h} = 0 \\ \dot{z}_{4h} = 0 \end{cases} \tag{6.33}$$

where the two last equations obtained on the basis of solutions of (6.30) correspond to the stable internal dynamic.

In order to derive the feedback linearization control for our system given by equation (6.33), let us rearrange the external dynamics in the matrix form, that is

$$\begin{pmatrix} \dot{z}_{1h} \\ \dot{z}_{2h} \end{pmatrix} = \begin{pmatrix} \lambda_{1hp} \\ -C_{ser}[z_{3h} - (z_{2h}+1)^{-1}] \end{pmatrix} + \begin{pmatrix} -(z_{4h} - z_{3h} - (z_{1h}+1)^{-1}) & 0 \\ z_{3h} - (z_{2h}+1)^{-1} & 1 \end{pmatrix} \begin{pmatrix} C_{hp} \\ \lambda_{1hp} \end{pmatrix} \tag{6.34}$$

and considering v_{hf} with $f = p, r$ as the new inputs for the linearized system, the resulting congestion control law is as follows

$$\begin{pmatrix} C_{hp} \\ \lambda_{1hr} \end{pmatrix} = \begin{pmatrix} 1 & (z_{2h}+1)^{-1} - z_{3h} \\ 0 & -z_{4h} + z_{3h} + (z_{1h}+1)^{-1} \end{pmatrix} \begin{pmatrix} v_{hp} - \lambda_{1hp} \\ v_{hr} + C_{serv}[z_{3h} - (z_{2h}+1)^{-1}] \end{pmatrix} \tag{6.35}$$

or in the explicit form as

$$\begin{cases} C_{hp} = v_{hp} - \lambda_{1hp} + [(z_{2h}+1)^{-1} - z_{3h}][v_{hr} + C_{ser}[z_{3h} - (z_{2h}+1)^{-1}]] \\ \lambda_{1hr} = [z_{3h} - z_{4h} + (z_{1h}+1)^{-1}][v_{hr} + C_{ser}[z_{3h} - (z_{2h}+1)^{-1}]] \end{cases} \tag{6.36}$$

The feedback linearization that is utilized leads to a closed-loop error dynamics corresponding to an unstable parallel integrators. Therefore, its stabilization can be performed through the new inputs v_{hf} with $f = p, r$ implying a simple output feedback. By choosing $v_{hp} = a_{1h}(z_{1h} - z_{1h_{ref}})$ and $v_{hr} = a_{2h}(z_{2h} - z_{2h_{ref}})$, the stabilized error dynamics will be expressed as follows

$$\begin{cases} \dot{z}_{1h} = & -a_{1h}z_{1h} + a_{1h}z_{1h_{ref}} \\ \dot{z}_{2h} = & -a_{2h}z_{2h} + a_{2h}z_{2h_{ref}} \end{cases} \tag{6.37}$$

Theorem 6.4. *For a differentiated-services network where the h^{th} node is described by the Fluid Flow Model* (6.1), *by adopting the decentralized control approach, the congestion control law governed by equation* (6.36) *guarantees the asymptotic stability of the error dynamics* (6.37) *provided that a_1 and a_2 are selected as positive design parameters.*

Proof: It is well-known that our resulting closed-loop error dynamics given by equation (6.37) corresponds to a standard linear system whose asymptotic stability is guaranteed provided that a_1 and a_2 are selected to be positive. This completes the proof. ▲

6.3 Conclusion

Several robust congestion control approaches are proposed in this chapter by using sliding mode generalized variable structure control (SM-GVSC) techniques. The GVSC control consists of transformation of the system dynamics into a new type of state space representation exhibiting the derivatives of the control input and to which the output equation is associated to yield what are known as Generalized Observable and Controllable Canonical Forms (GOCF, GCCF). On the basis of these canonical forms, the control law might be derived using different approaches. The first approach consists of solving the sliding mode condition which is inspired from the equivalent sliding mode control method. The second approach consists of adding a discontinuous term to the classical feedback linearization control law for robustness purposes. The third approach utilizes the hyperplane convergence equation as a nonlinear feedback control. In effect, the third approach is more interesting since it attempts on one hand to keep the advantage of the feedback linearization and on the other hand to act on the sliding hyperplane convergence resulting in design parameters other than the hyperplane slope. Furthermore, an input-output linearization congestion controller is also investigated in this chapter.

Chapter 7

SM-GVS Congestion Control for Cascaded DiffServ Networks: Degenerate Case

7.1 Introduction

Contrary to the non-degenerate sliding mode approach utilized to design the controller considered in Chapter 6, a degenerate sliding mode approach is investigated in this chapter. The class of DiffServ networks analyzed in this chapter belongs to the cascaded networks. The extension to a fully-connected and feedbacked networks is investigated in Chapter 8.

This chapter is intended to extend our investigations conducted in Chapter 6 by (a) simplifying our proposed congestion controller design, (b) evaluating the proposed controller for a single node bottleneck in presence of delays, (c) deriving the single node error dynamics in order to guarantee stability of the closed-loop system, (d) evaluating the proposed controller for a three node cascade network in presence of time-delay, and (e) deriving the network error dynamics in order to guarantee stability of the closed-loop system.

7.2 Proposed Congestion Control Strategy

As in the congestion controller that is proposed for DiffServ network in Chapter 6, [15], and [17], the controller proposed in this chapter uses the buffer queue length as a feedback information to control locally the queue length of each buffer by acting on the server bandwidth and simultaneously it sends back to the ordinary source the allowed maximum rate. Note that the bandwidth of the ordinary server is not controlled, only a left over capacity from the premium service is allocated to the ordinary service. The controller proposed here consists of extending the approach introduced in Chapter 6, and [17] through certain simplifying modifications. In Chapter 6 and [17], the sliding mode congestion controller is designed on the basis of a non-minimal FFM realization model to exhibit the derivatives of the control variables in the generalized observable canonical form (the approach is called *non-degenerate*). In this chapter as well as in the subsequent chapter, the controller is designed on the basis of the FFM model that is given by equation (6.1). Given that the derivatives of the control are not considered for this controller, the approach is called *degenerate*. The controller designed here uses the same nonlinear feedback control approach as in [17]. The latter approach uses a convergence condition on the sliding hyperplane dynamics.

7.2.1 Proposed decentralized end-to-end congestion controller design [19, 20]

For the h^{th} node as illustrated in Figure 2.1 for a differentiated-services (DiffServ) network, the congestion controller is designed in the error state space by defining the error variables $e_{hf} = x_{hf} - x_{hf_{ref}}$, $\dot{e}_{hf} = \dot{x}_{hf} - \dot{x}_{hf_{ref}}$. By substituting them in model (6.1), the following state error representation is obtained

$$\begin{cases} \dot{e}_{hf}(t) = -C_{hf}(t)\dfrac{e_{hf}(t) + x_{hf_{ref}}}{e_{hf}(t) + x_{hf_{ref}} + 1} + \lambda_{hf} - \dot{x}_{hf_{ref}}; (f = p, r) \\ y_{hf}(t) = e_{hf}(t) + x_{hf_{ref}} \end{cases} \tag{7.1}$$

Let us now define the sliding manifold as a linear hypersurface that is $S_{hf} = e_{hf}$ and choose a positive definite Lyapunov function $V_{hf}(S_{hf}) = \frac{S_{hf}^2}{2}$; $(f = p, r)$. Note that the derivative of the Lyapunov function corresponds to the existence condition of the sliding mode, ensuring thus its definite negativeness, that is

$$\dot{V}_{hf}(S_{hf}) = S_{hf}\dot{S}_{hf} < 0; \qquad (f = p, r) \tag{7.2}$$

Let us now impose the following condition

$$\dot{V}_{hf}(S_{hf}) = -\mu_{hf}S_{hf}^2 - \mu_{hf}\Omega_{hf}S_{hf}sgn(S_{hf}); \qquad (f = p, r) \tag{7.3}$$

which makes $\dot{V}_{hf}$ always negative definite whenever both design parameters μ_{hf} and Ω_{hf} are strictly positive.

Therefore, from equations (7.2) and (7.3), the following differential equations

$$\begin{cases} \dot{S}_{hf} = -\mu_{hf}S_{hf} - \mu_{hf}\Omega_{hf}sgn(S_{hf}); \qquad (f = p, r) \\ \dot{e}_{hf} = -\mu_{hf}e_{hf} - \mu_{hf}\Omega_{hf}sgn(e_{hf}); \qquad (f = p, r) \end{cases} \tag{7.4}$$

yield the solution to the sliding hyperplane S_{hf} which after starting from $S_{hf} = S_{hf}(0)$, at time $t = 0$, reaches the condition $S_{hf} = 0$ in finite time $T_{hf} = \mu_{hf}^{-1}ln[1 + \frac{|S_{hf}(0)|}{\Omega_{hf}}]$. It is worth noting that in our approach, equation (7.4) may be seen as a feedback control of the FFM state error representation (7.1) which leads to the following closed-loop error dynamics of the controlled system

$$\dot{e}_{hf} = -A_{hf}e_{hf} - B_{hf}sgn(e_{hf}) \tag{7.5}$$

where $A_{hf} = diag[\mu_{hp} \;\; \mu_{hr}]$, $B_{hp} = diag[\mu_{hp}\Omega_{hp} \;\; \mu_{hr}\Omega_{hr}]$ and $sgn(e_{hf}) \triangleq col[sgn(e_{hp})$ $sgn(e_{hr})]$.

Consequently, by taking into account the leftover capacity $C_{hr}(t) = C_{serv} - C_{hp}(t)$, where C_{serv} is the maximum available capacity of the server, the control law may be expressed as

$$\begin{cases} C_{hp} = E_{hp}^{-1}[\mu_{hp}e_{hp} + \mu_{hp}\Omega_{hp}sgn(e_{hp}) + \lambda_{ihp} - \dot{x}_{hp_{ref}}] \\ \lambda_{hr} = C_{hr}E_{hr} - \mu_{hr}e_{hr} - \mu_{hr}\Omega_{hr}sgn(e_{hr}) + \dot{x}_{hr_{ref}} \end{cases} \tag{7.6}$$

where $E_{hf} = \frac{e_{hf}+x_{hf_{ref}}}{e_{hf}+x_{hf_{ref}}+1}$ with $f = p, r$. It may be observed that no derivative of the control input appears in (7.6), which as mentioned earlier corresponds to the so-called *degenerate case*. Note that λ_{ihp} is assumed to be measured or estimated.

Theorem 7.1. *For the h^{th} node of a Differentiated-Services network,, having a buffer queueing model as described by the Fluid Flow Model* (7.1)*, the nonlinear feedback control law* (7.6) *guarantees the asymptotic stability of the closed-loop dynamics* (7.5) *if μ_h and Ω_h are selected as positive parameters.*

Proof: Choose a candidate Lyapunov function $V(e_h, t) = \frac{e_h^T e_h}{2}$, where $e_h^T = [e_{hp} \quad e_{hr}]$ such that for all $e_h \neq 0 \Rightarrow V(e_h, t) > 0$. The time derivative of the Lyapunov function along the trajectories of (7.5) is given by

$$\dot{V}(e_h, t) = -\sum_{f=p,r} \mu_{hf}e_{hf}^2(t) - \sum_{f=p,r} \gamma_{hf}e_{hf}(t)sgn(e_{hf}(t)) \tag{7.7}$$

It follows that since $sgn(e_{hf}(t)) = \frac{e_{hf}(t)}{|e_{hf}(t)|}$ with $(f = p, r)$, $\dot{V}(e_{hf}, t)$ is strictly negative definite whenever the design parameters μ_{hf} and Ω_{hf} are selected to be positive, and hence $e_{hf}(t) \to 0$ as $t \to \infty$. This completes the proof. ▲

7.2.2 Single node control dynamics: time-delayed dependent case [19,20]

Let us derive the FFM-based congestion control dynamics in presence of unknown but constant time-delays. As illustrated in Figure 2.1, to study the sensitivity of our closed-loop control system to time-delays, we introduce three block delays, τ_p, τ_r and τ_b in the Premium, the Ordinary, and the Best Effort channels, respectively. They capture and correspond to any delay/latency in the network due to propagation, processing, transmission, etc. factors. In presence of these delays, the FFM model (7.1) on which our proposed congestion controller is designed will then be time dependent and is given by

$$\begin{cases} \dot{e}_{hf}(t) = -C_{hf}(t)\dfrac{e_{hf}(t) + x_{hf_{ref}}}{e_{hf}(t) + x_{hf_{ref}} + 1} + \lambda_{ihf}(t - \tau_{hf}) - \dot{x}_{hf_{ref}} \\ y_{hf}(t) = e_{hf}(t) + x_{hf_{ref}} \end{cases} \tag{7.8}$$

where τ_{hf} is an unknown but constant delay coefficient.

Assumption 7.1. *Given that the delay τ_{hf}, where $(f = p, r)$, corresponding to any block delay as shown in Figure 2.1 is constant and bounded, the time-delayed signals may be approximated by their first-order representation as*

$$\lambda_{ihf}(t - \tau_{hf}) = \lambda_{ihf}(t) - \tau_{hf}\dot{\lambda}_{ihf}(t)$$

Although the controller given by Theorem 7.1 does not assume any explicit information about the time-delay, by substituting the control law (7.6) into the time delayed error dynamics (7.8), it follows that in the overall closed-loop dynamics (given by (7.9)), the presence of the time-delay τ_{hf} will be different from that given by equations (7.5) corresponding to the non delayed closed-loop system, namely, we have

$$\begin{cases} \dot{e}_{hp} = -\mu_{hp}e_{hp} - \mu_{hp}\Omega_{hp}sgn(e_{hp}) - \tau_{hp}\dot{\lambda}_{ihp} \\ \dot{e}_{hr} = -D_0^{-1}\mu_{hr}e_{hr} - D_0^{-1}\mu_{hr}\Omega_{hr}sgn(e_{hr}) - \tau_{hr}D_0^{-1}P_{hr_1} \end{cases} \tag{7.9}$$

where $D_0 = (1 - \mu_{hr}\tau_{hr})$ and $P_{r_1} = -\mu_{hr}\Omega_{hr}\frac{d(sgne_{hr})}{dt} + \ddot{x}_{hr_{ref}} + \frac{d(C_{hr}E_{hr})}{dt}$.

Lemma 7.1. *Under Assumption 7.1 and using the control law given by Theorem 7.1, the error dynamics described by* (7.9) *of the time-delay dependent control system illustrated by Figure 2.1 is stable provided that the design parameters μ_{hf} and Ω_{hf} in* *the control law* (7.6) *are positive and the ordinary delay* $\tau_{hr} \leq \tau_{hr_{max}}$, *where* $\tau_{hr_{max}}$ *is the ordinary delay upper bound.*

Proof: According to the two dynamics in (7.9), the premium buffer queue dynamics corresponds to the time-delayed independent premium dynamics given in (7.5) with the presence of an additional disturbance $\delta_{hp} = -\tau_{hp}\dot{\lambda}_{ihp}$. It is well-known that such a system is stable whenever the disturbance δ_{hp} is bounded, implying that both τ_{hp} and the derivative of the premium traffic $\dot{\lambda}_{ihp}$ will be bounded.

With respect to the ordinary buffer queue dynamics in (7.9), it corresponds to the time-delay independent ordinary dynamics (7.5) that is now divided by D_0, a function of the delay τ_{hr} and μ_{hr}, and then augmented with one additional term. Note that since this additional term is nonlinear with respect to the state error e_{hr}, the overall ordinary error dynamics is now nonlinear and time-delay dependent. We may thus conclude that a) the premium error dynamics is stable whenever the premium input traffic and the delay in the premium channel are bounded, and b) the ordinary error dynamics is nonlinear and time-delay dependent.

In the following derivations, we aim to show that the ordinary error dynamics is indeed sensitive to the time-delay, but stability may be ensured if the time-delay does not exceed a certain upper bound. Stability of the time-delayed dependent ordinary dynamics may be investigated by studying the stability of the overall linearized dynamics around a certain operating point e_{hr0}.

After computing the derivative of the term $\frac{d(C_{hr}E_{hr})}{dt}$ in the error dynamic (7.9), the ordinary error dynamics can be written as follows

$$\dot{e}_{hr} = -D_1 e_{hr} - D_2 sgne_{hr} - D_3 s\dot{g}ne_{hr} - D_4 - D_5 E_{hr} - D_6 P_{hr_2} \qquad (7.10)$$

where $D_1 = \frac{\mu_{hr}}{D_0}$, $D_2 = \frac{\mu_{hr}\Omega_{hr}}{D_0}$, $D_3 = -\frac{\mu_{hr}\Omega_{hr}\tau_{hr}}{D_0}$, $D_4 = \frac{\tau_{hr}\ddot{x}_{hr_{ref}}}{D_0}$, $D_5 = -\frac{\tau_{hr}\dot{C}_{hp}}{D_0}$, $D_6 = \frac{\tau_{hr}C_{hr}}{D_0}$, $s\dot{g}n(e_{hr}) = \frac{d(sgn(e_{hr}))}{dt}$ and $P_{hr_2} = \frac{\dot{e}_{hr}+\dot{x}_{hr_{ref}}}{(e_{hr}+x_{hr_{ref}}+1)^2}$. Note that D_i, $i = 0, \ldots, 6$ contains mainly ordinary buffer delay τ_{hr}. The sign function terms in the error dynamics (7.10) may be approximated by using different approaches. In [19], $sgn(e_{hr})$ is approximated by $\frac{2}{1+e^{-e_{hr}/\theta_0}}-1$, and subsequently the resulting continuous error dynamics is linearized. Here, we may show that the same results may be obtained by substituting $sgn(e_{hr})$ by $\frac{e_{hr}}{|e_{hr}|}$.

Let us substitute in the ordinary error dynamics (7.10), $sgn(e_{hr}) = \frac{e_{hr}}{|e_{hr}|}$ and $\frac{d(sgn(e_{hr}))}{dt} = \dot{e}_{hr}[\frac{1}{|e_{hr}|} - \frac{e_{hr}^2}{|e_{hr}|^3}]$, such that the nonlinear time-delayed ordinary dynamics may be written as follows

$$\dot{e}_{hr} = L_1^{-1} \sum_{i=2}^{i=6} L_i \tag{7.11}$$

where $L_1 = D_3K_1 + D_6K_2 + K_3$, $L_2 = -D_1K_4$ with $K_4 = e_{hr}|e_{hr}|^3(e_{hr} + x_{hr_{ref}} + 1)^2$, $L_3 = -D_2K_5$ with $K_5 = K_4/|e_{hr}|$, $L_4 = -D_4K_6$ with $K_6 = K_4/e_{hr}$, $L_5 = -D_5K_7$ with $K_7 = |e_{hr}|^3(e_{hr} + x_{hr_{ref}})(e_{hr} + x_{hr_{ref}} + 1)$, $L_6 = -D_6K_8$ with $K_8 = |e_{hr}|^3\dot{x}_{hr_{ref}}$ are new variables introduced for sake of simplifying the notations.

The linearization of equation (7.11) leads to the following conventional representation

$$\dot{e}_{hr} = \frac{\delta \dot{e}_{hr}}{\delta e_{hr}}|_{e_{hr0}} e_{hr} + \Phi \tag{7.12}$$

where the Jacobian

$$\frac{\delta \dot{e}_{hr}}{\delta e_{hr}}|_{e_{hr0}} = \frac{B_1\tau_{hr}^2 + B_2\tau_{hr} + B_3}{L^2(1 - \mu_{hr}\tau_{hr})^2} \tag{7.13}$$

with

$$\begin{cases} B_1 = -\mu_{hr}\Omega_{hr}Z_1 - \mu_5 Z_2 + Z_3 \\ B_2 = -\mu_{hr}^2[\Omega_{hr}^2 Z_4 + \Omega_{hr}Z_6 + Z_7] + \mu_{hr}[\Omega_{hr}Z_5 + Z_8] + Z_9 \\ B_3 = \mu_{hr}[\Omega_{hr}Z_{10} + Z_{11}] \end{cases} \tag{7.14}$$

and where the terms Z_j with $(j = 1, ..., 11)$ are algebraic combinations of parameters μ_{hr0} and Ω_{hr} from e_{hr0}, $x_{hr_{ref}}$, C_{serv} and C_{p0}. It is worth noting that the obtained linear dynamics (7.12) is of the first-order whose Jacobian corresponds to its pole. This pole represents the sufficient condition for stability of our ordinary dynamics which is subject to the time-delay τ_{hr}. In other words, to analyze the stability of the resulting linear system, we need to determine the condition on the time-delay

τ_{hr} that makes the eigenvalue $\frac{\delta \dot{e}_{rh}}{\delta e_{hr}}|_{e_{hr0}}$ lies in the left-half of the complex plane. The negativeness of this eigenvalue implies also the negativeness of the numerator in (7.13), i.e. $B_1\tau_{hr}^2 + B_2\tau_{hr} + B_3$, whose characteristic is parabolic in τ_{hr}. It is well-known that depending on the coefficients B_1, B_2 and B_3, such characteristic may have two roots τ_{hr1} and τ_{hr2} that are negative and positive, respectively. The numerator is negative if $\tau_{hr1} < \tau_{hr} < \tau_{hr2}$, whereas it is positive elsewhere for $B_1 > 0$. In fact, this stability condition shows that the time-delay τ_{hr} should be less than an upper bound $\tau_{hr_{max}} = \tau_{hr2}$ and positive. Therefore, the previous condition reduces to $0 < \tau_{hr} < \tau_{hr2}$.

One may thus conclude that the stability of the error dynamics (7.9) is ensured if the design parameters μ_{hf} and Ω_{hf} in the control law given by Theorem 7.1 are positive, the derivative of the premium traffic $\dot{\lambda}_{hp}$ is bounded, and $\tau_{hr} < \tau_{hr2} = \tau_{hr_{max}}$, where $\tau_{hr_{max}}$ is the ordinary delay upper bound. Furthermore, since τ_{hr2} is a function of the coefficients B_1, B_2 and B_3 which depends on the controller parameters μ_{hr} and Ω_{hr}, then one may conclude that there is a constraint on these parameters in order to ensure stability. This completes the proof. ▲

7.2.3 Design of the benchmark solution

In this section the proposed robust congestion controller (7.6) is compared to the equivalent control sliding mode approach that is given by

$$U = U_{eq} - bsgn(S) \tag{7.15}$$

The equivalent control component U_{eq} is derived for a general nonlinear system $\dot{x}_h = f_{s,h}(x_h) + g_{s,h}(x_h)u_h$, when the ideal sliding mode occurs such that if $S_h(x_h) = 0$ then $U_{eq,h} = -\frac{<\nabla S_h, f_{s,h}>}{<\nabla S_h, g_{s,h}>} = -\frac{L_{f_{s,h}} S_h}{L_{g_{s,h}} S_h}$.

By using the above derivation for the n^{th} node of a DiffServ network, and taking into account the leftover capacity $C_{hr}(t) = C_{serv} - C_{hp}(t)$, the equivalent sliding mode control law is expressed as follows

$$\begin{cases} C_{hp} = \lambda_{hp}\dfrac{e_{hp} + x_{hr_{ref}} + 1}{e_{hp} + x_{hr_{ref}}} - b_{hp}sgn(S_{hp}) \\ \lambda_{hr} = \dfrac{e_{hp} + x_{hr_{ref}} + 1}{e_{hp} + x_{hr_{ref}}}C_{hr} - b_{hr}sgn(S_{hr}) \end{cases} \tag{7.16}$$

7.2.4 Simulation results for a single node

A comparative performance evaluation of the proposed robust congestion controls (controls (7.6) and (7.16)) is now performed subject to (a) studying the time-delay influence on both controllers, (b) comparing their capabilities and robustness to presence of premium traffic stimuli, and (c) determining the relationship between the time-delay and the controller parameters.

For all the simulations presented below the sampling time is set to $T_s = 1$ ms and the proposed control parameters are set to $\mu_{hp} = 1000$, $\mu_{hr} = 500$, and $\Omega_{hp} = \Omega_{hr} = 10^{-4}$. For the equivalent control approach, the parameters are set to $b_{hp} = 8000$ and $b_{hr} = 5$ x10^4. The premium random input traffic λ_{ihp} has a mean of 3000 packet/s (step signal) and a variance of 1000 packet/s, and it is bounded such that $0 \leq \lambda_{ihp} \leq 4000$ packet/s. The premium and the ordinary buffer capacities are finite and their queues are bounded according to the following constraints $0 \leq x_{hp} \leq 128$ packet and $0 \leq x_{hr} \leq 1024$ packet, respectively. Finally, the constraints on the maximum available service capacity are $0 \leq C_{hf} \leq 40000$ packet/s for both services $f = p, r$. With respect to the time-delay, and as illustrated in Figure 2.1, two delay blocks are introduced between each source-buffer to simulate and reflect any system time-delay due to either propagation, transmission, processing delays, etc. The corresponding simulation results are presented in three steps.

First, the premium and the ordinary buffer queue references are set to $x_{hp} =$ 100 packet and $x_{hr} = 1000$ packet, respectively. Figure 7.1 shows a set of buffer queue step responses that are obtained for both controllers by varying the time-delay from 0 to 30 ms when the equivalent congestion controller is used (Figure 7.1

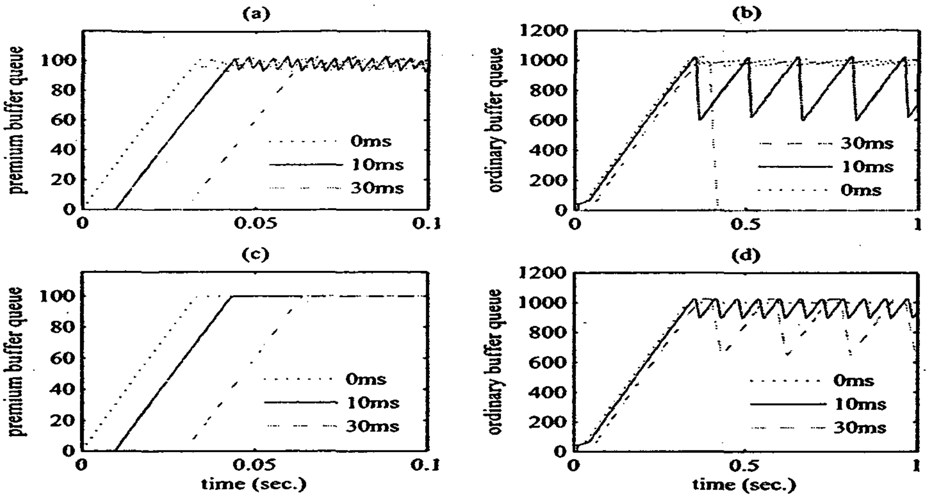


Figure 7.1: Effects of the delay on the system response: (a), (b) equivalent congestion controller (unstable for 30 ms and 160 ms delays); (c), (d) proposed congestion controller.

a and b) and from 0 to 160 ms when the proposed congestion controller is considered (Figure 7.1 c and d). By comparing the graphs in Figure 7.1 a and c, we can argue that except the delay in the transmission there is no influence on the premium traffic channel. Furthermore, for the premium buffer queue graphs, we may also conclude that our proposed congestion controller is quite smooth when compared to the oscillatory equivalent control approach.

On the other hand, the plots in Figure 7.1 b and d corresponding to the ordinary buffer queues that are obtained by using the equivalent control and our proposed congestion control approaches, respectively, show the sensitivity of the ordinary service dynamics to the time-delay variations. Indeed, one may observe that our proposed congestion controller is less sensitive in comparison to the equivalent control congestion controller which becomes **unstable** for a time-delay of 30 ms and larger. It is worth noting that these simulations do indeed confirm the stability analysis presented in the previous section.

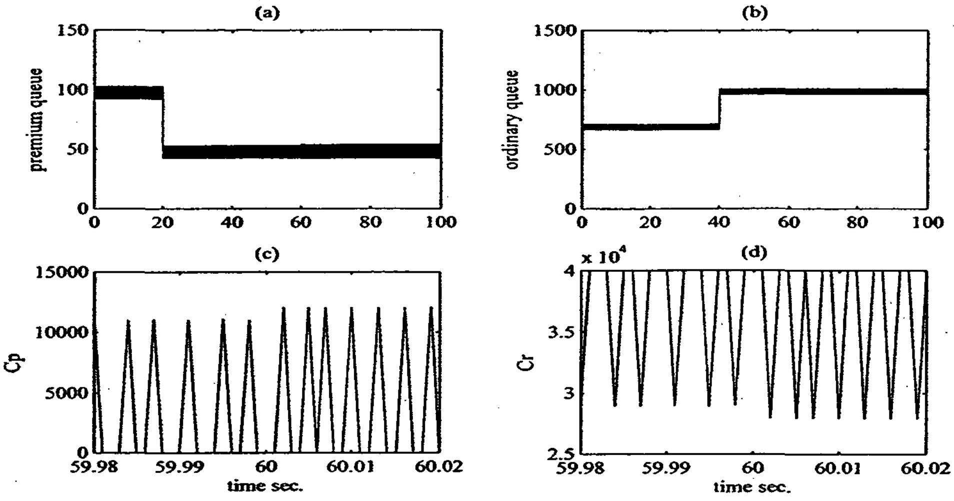


Figure 7.2: Simulation results in the tracking mode with equivalent congestion controller in presence of the premium traffic stimuli (without time-delay). Plots (c) and (d) are zoomed to show oscillatory nature of the control.

In another set of simulations that are depicted in Figures 7.2-7.5, the congestion control characteristics are shown to evolve for a wide range of operating points in presence of premium traffic stimuli and time-delays. The simulation results are given corresponding to time-delays of 0 and 10 ms.

Figures 7.2 and 7.3 correspond to the equivalent control and the proposed congestion control approaches, respectively, when no delays are added, whereas Figures 7.4 and 7.5 correspond to the controllers when a time-delay block of 10ms is added. In each figure, plots (a) and (b) illustrate the premium and the ordinary buffer queues, respectively, and (c) and (d) show the premium and the ordinary service capacities.

By setting the premium and the ordinary queue reference states to $x_{hp} = \{100, 50\}$ packet and $x_{hr} = \{800, 1000\}$ packet, respectively, and the mean of the input traffic to $\lambda_{ihp} = \{3000, 6000\}$ packet/s, one may observe that either with or without time-delay and in both control approaches and services the states (a) and (b)

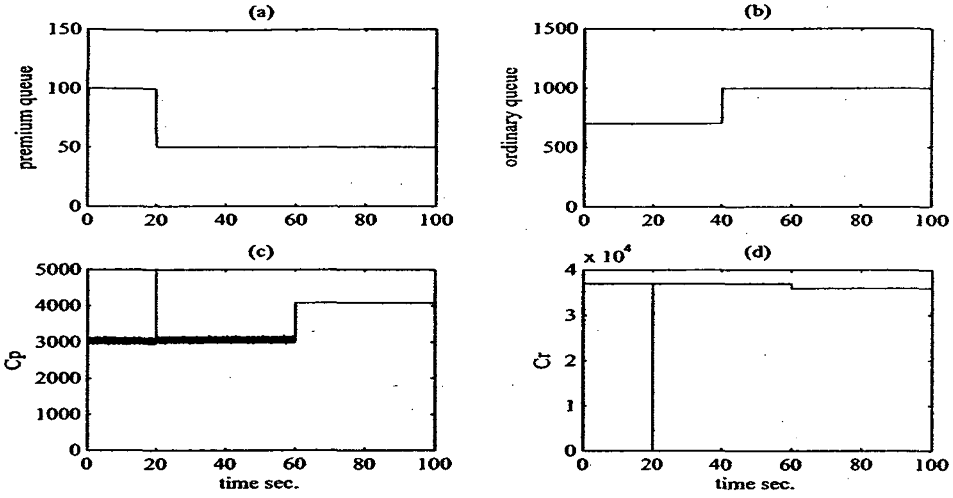


Figure 7.3: Simulation results in the tracking mode with the proposed congestion controller in presence of the premium traffic stimuli (without time-delay).

converge to their desired values. In terms of performance and time domain response, the equivalent control approach is quite oscillatory when compared to the proposed congestion controller that shows less chatter in the ordinary service dynamics in the presence of time-delay and does not show any chatter in the time-delay independent case. In virtue of the sliding mode robustness it is expected that both controllers address the added stimuli in the premium traffic.

The last simulations consist of determining the effects of the controller parameter μ_{hr} on the overall closed-loop stability for a fixed time-delay. By setting the time-delay to 30 ms and varying the controller parameter μ_{hr} from 28 to 1400, one may note from Figure 7.6a (corresponding to the premium buffer queue step responses) that no effect is observed and the stability is always guaranteed. This confirms that both the premium and the ordinary dynamics are decoupled. On the other hand by increasing μ_{hr}, we increase the amplitude of the oscillation of the ordinary buffer queue step responses as shown in Figure 7b. Note that the

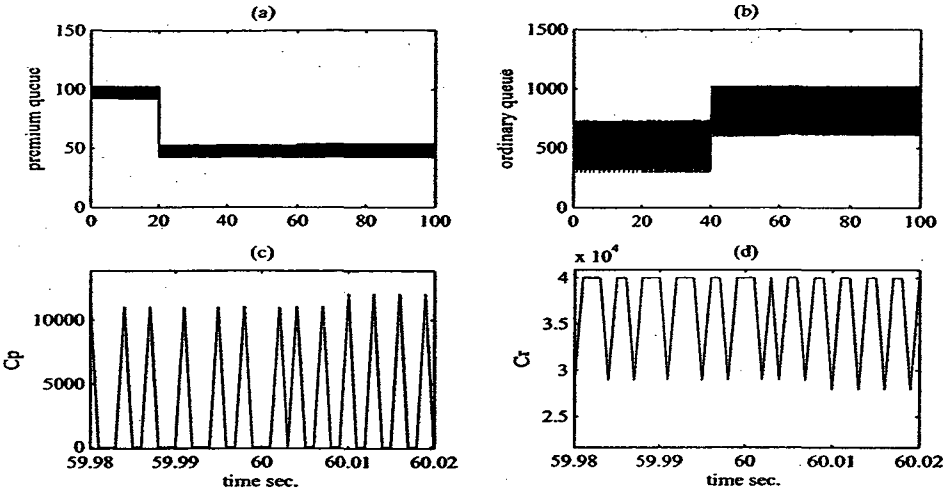


Figure 7.4: Simulation results in the tracking mode with equivalent congestion controller in presence of the premium traffic stimuli (with time delay of 10 ms). Plots (c) and (d) are zoomed to show oscillatory nature of the control.

lower value of μ_{hr} of 28 corresponds to the minimum value that gives enough energy to ensure the convergence of the ordinary buffer queue state to the desired reference. From these results we may conclude that μ_{hr} belongs to a range of values in which stability is ensured. These simulation results show that the upper bound of time-delay for ensuring stability is inversely proportional to μ_{hr}. This inverse proportionality might actually be seen in the stability condition $0 < \tau_{hr} < 1/\mu_{hr}$ of the ordinary buffer queue dynamics (7.9) when this dynamics is approximated by its linear representation $-D_1\mu_{hr}e_{hr}$.

7.2.5 Nodes cascade in differentiated-services network: Time-delay independent case [20]

The implementation of our proposed decentralized end-to-end congestion control strategy as illustrated in Figure 7.7 consists of first controlling locally the bandwidth

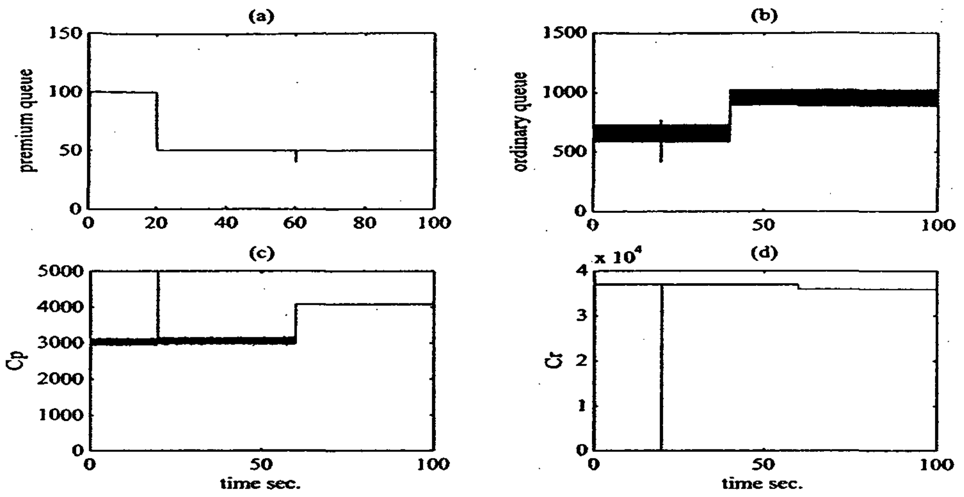


Figure 7.5: Simulation results in the tracking mode with the proposed congestion controller in presence of the premium traffic stimuli (with time-delay of 10 ms).

of both the premium and the ordinary buffers and then controlling the ordinary flow rates by notifying all upstream sources sharing this buffer regarding the maximum allowed transmission rate.

Let us first extend the state space equation of our FFM (7.1) to the time-delay independent two nodes cascade as shown below

$$\dot{e}_{hf}(t) = -C_{hf}(t)E_{hf} + \lambda_{hf}(t) - \dot{x}_{hf_{ref}}; (f = p, r; h = s, d) \qquad (7.17)$$

where $E_{hf} = \frac{e_{hf}(t)+x_{hf_{ref}}}{e_{hf}(t)+x_{hf_{ref}}+1}$, s and d designate the sensor and the decision-maker, respectively, as shown in Figure 7.7.

Remark 7.1. *It should be noted that the results can be generalized to any arbitrary number of cascaded nodes, however, for sake of notational simplicity we consider without loss of generality a two nodes cascade.*

By using the proposed congestion controller that is given by equation (7.6), the closed-loop error dynamics of the overall network as depicted by Figure 7.7 can

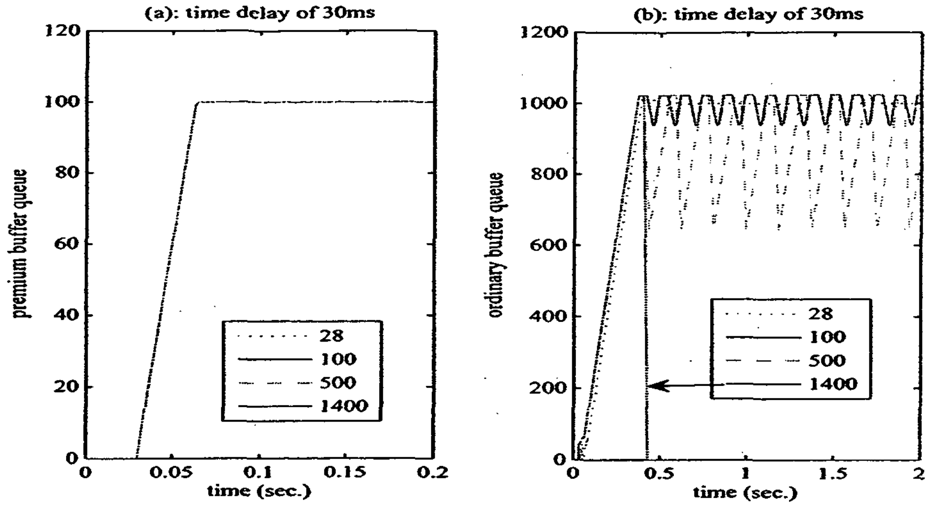


Figure 7.6: Influence of the controller parameters on performance of congestion control system in presence of time-delay of 30 ms using the proposed congestion controller.

be described as

$$\begin{cases} \dot{e}_{hp} = -\mu_{hp}e_{hp} - \mu_{hp}\Omega_{hp}sgn(e_{hp}); \; (h = s, d) \\ \dot{e}_{hr} = -\mu_{hr}e_{hr} - \mu_{hr}\Omega_{hr}sgn(e_{hr}) \\ \dot{e}_{\bar{h}r} = \dot{e}_{hr} + C_{hr}E_{hr} - C_{\bar{h}r}E_{\bar{h}r} + \dot{x}_{hr_{ref}} - \dot{x}_{\bar{h}r_{ref}} \end{cases} \qquad (7.18)$$

where $\bar{h}$ is the complement of h.

Lemma 7.2. *For a time-delay independent Differentiated-Services network shown by Figure 7.7 where the h^{th} node is described by the open-loop dynamics* (7.17)*, the decentralized end-to-end congestion control law governed by equation* (7.6) *guarantees the stability of the error dynamics* (7.18) *provided that μ_{hf} and Ω_{hf} are selected as positive design parameters.*

Proof: It is readily seen that the premium and the ordinary error dynamics that are given by (7.18) for the buffer h are the same as that for the stable single node error dynamics given in (7.5). Regarding the ordinary error dynamics by (7.18)

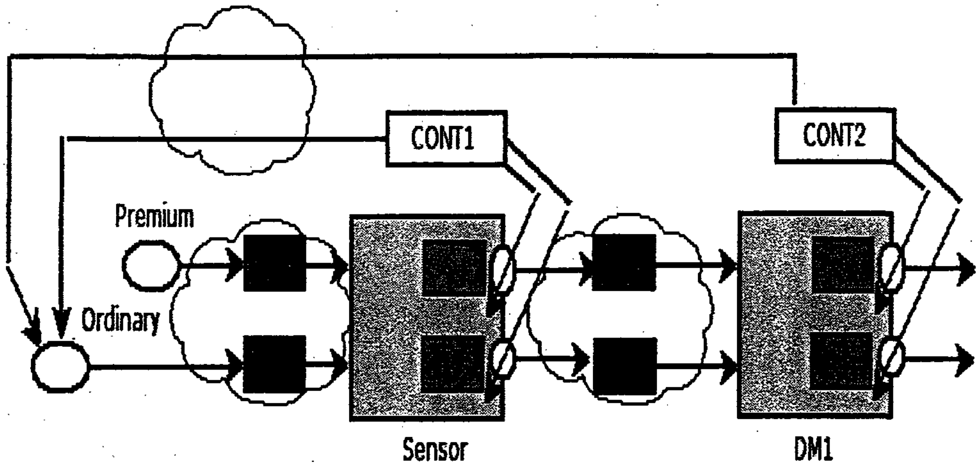


Figure 7.7: Two nodes cascaded in a differentiated-service network.

corresponding to the buffer $\bar{h}$, let us first rewrite it in a concise form given as

$$\dot{e}_{\bar{h}r} = -C_{\bar{h}r}\frac{e_{\bar{h}r}(t) + x_{\bar{h}r_{ref}}}{e_{\bar{h}r}(t) + x_{\bar{h}r_{ref}} + 1} + \Delta_{1h} \tag{7.19}$$

where $\Delta_{1h} = -\mu_{hr}e_{hr} - \mu_{hr}\Omega_{hr}sgn(e_{hr}) + C_{hr}E_{hr} + \dot{x}_{hr_{ref}} - \dot{x}_{\bar{h}r_{ref}}$. Note that Δ_{1h} is bounded, since it mainly depends on the error dynamics h which is already shown to be stable and the reference signal $\dot{x}_{\bar{h}r_{ref}}$ which is also bounded. The stability of (7.19) may be verified through its linearized dynamics around $e_{\bar{h}r0}$, whose Jacobian is $-C_{\bar{h}r}(e_{\bar{h}r0} + x_{\bar{h}r_{ref}})^{-2}$, and which is always negative. This completes the proof of the theorem. ▲

7.2.6 Nodes cascade in differentiated-services network: Time-delay dependent case

In presence of delays, the open-loop error dynamics of each node of our network is expressed as follows

$$\dot{e}_{hf} = -C_{hf}E_{hf} + \lambda_{hf}(t - \tau_{hf}) - \dot{x}_{hf_{ref}}; (f = p, r; h = s, d) \tag{7.20}$$

where $E_{hf} = \frac{e_{hf}(t)+x_{hf_{ref}}}{e_{hf}(t)+x_{hf_{ref}}+1}$, s and d denote the sensor and the decision-maker, respectively, as shown in Figure 7.7.

By now using the proposed congestion controller that is given by Lemma 7.2 and under Assumption 7.1, the overall controlled network dynamics is governed as shown below

$$\begin{cases} \dot{e}_{hp} = -\mu_{hp}e_{hp} - \gamma_{hp}sgn(e_{hp}) - \tau_{hp}\dot{\lambda}_{ihp}, \quad (h = s, d) \\ \dot{e}_{hr} = -D_{0h}^{-1}\mu_{hr}e_{hr} - D_{0h}^{-1}\gamma_{hr}sgn(e_{hr}) - D_{0h}^{-1}\tau_{hr}D_{1h} \\ \dot{e}_{\bar{h}r} = -\bar{D}_{0h}^{-1}\mu_{hr}e_{hr} - \bar{D}_{0h}^{-1}\gamma_{hr}sgn(e_{hr}) + \bar{D}_{0h}^{-1}D_{2h} - \tau_{\bar{h}r}\bar{D}_{0h}^{-1}D_{3h} \end{cases} \tag{7.21}$$

where $D_{0h} = (1 - \mu_{hr}\tau_{hr})$, $\bar{D}_{0h} = (1 - \mu_{hr}\tau_{\bar{h}r})$, $D_{1h} = -\mu_{hr}\Omega_{hr}\frac{d}{dt}sgn(e_{hr}) + \ddot{x}_{hr_{ref}} + \frac{d(C_{hr}E_{hr})}{dt}$, $D_{2h} = C_{hr}E_{hr} - C_{\bar{h}r}E_{\bar{h}r} + \dot{x}_{hr_{ref}} - \dot{x}_{\bar{h}r_{ref}}$, $D_{3h} = -\mu_{hr}\Omega_{hr}\frac{d}{dt}sgn(e_{hr}) + \ddot{x}_{hr_{ref}} + \frac{d(C_{hr}E_{hr})}{dt}$ and $\gamma_{hf} = \mu_{hf}\Omega_{hf}$ with $(f = p, r)$. It should be noted that Remark 7.1 also holds for this problem.

Lemma 7.3. *For a time-delay dependent Differentiated-Services network shown as in Figure 7.7 where the h^{th} node is described by the open-loop dynamics* (7.20), *the decentralized end-to-end congestion control law governed by equation* (7.6) *guarantees the stability of the error dynamics* (7.21) *provided that Assumption 7.1 holds, the delay of the ordinary channel h satisfies $\tau_{hr} < \tau_{hr_{max}}$ where $\tau_{hr_{max}}$ is its upper bound, the delay of the ordinary channel $\bar{h}$ satisfies $\tau_{\bar{h}r} < \mu_{\bar{h}r}^{-1}$, and μ_{hf} and Ω_{hf} are selected as positive design parameters.*

Proof: First note that as expected, the premium error dynamics remains stable whenever the design parameters μ_{hp}, and Ω_{hp} are positive and the term $\tau_{hp}\dot{\lambda}_{hp}$ is bounded as in the single node case. As far as the ordinary error dynamics is concerned, the h case remains the same as that of a single node error dynamics whose stability is subject to boundedness of τ_{hr}, that is $\tau_{hr} < \tau_{hr_{max}}$. Now for the remaining error dynamics $\bar{h}$, it readily follows that it may be written in the same form as in (7.19), that is

$$\dot{e}_{\bar{h}r} = -\bar{D}_{0h}^{-1} C_{\bar{h}r} \frac{e_{\bar{h}r}(t) + x_{\bar{h}r_{ref}}}{e_{\bar{h}r}(t) + x_{\bar{h}r_{ref}} + 1} + \Delta_{2h} \tag{7.22}$$

where

$$\Delta_{2h} = -\bar{D}_{0h}^{-1}\mu_{hr}e_{hr} - \bar{D}_{0h}^{-1}\mu_{hr}\Omega_{hr}sgn(e_{hr}) + \bar{D}_{0h}^{-1}C_{hr}E_{hr} + \dot{x}_{hr_{ref}} - \dot{x}_{\bar{h}r_{ref}} - \tau_{\bar{h}r}\bar{D}_{0h}^{-1}D_{3h}$$

Under Assumption 7.1, and provided that the above stability condition $\tau_{hr} < \tau_{hr_{max}}$ holds for the error dynamics h, the term Δ_{2h} is also bounded. Therefore, stability of the dynamics (7.22) corresponds to stability of its linearized error dynamics whose Jacobian corresponds to the one for (7.19) multiplied by $\bar{D}_{0h}^{-1}$. Hence, stability of the ordinary dynamics $\bar{h}$ is ensured whenever the condition $\tau_{\bar{h}r} < \mu_{\bar{h}r}^{-1}$ ensures the negativeness of the corresponding Jacobian. ▲

7.2.7 Simulation results for a three node cascaded network

In Section 7.2.4, our proposed congestion controller was first evaluated for a single node. The controller has demonstrated good performance for a wide range of operating points and in the presence of delay and external stimuli. The extension of the controller evaluation for a three nodes cascaded network is performed in this section below. By adopting the decentralized end-to-end congestion control approach, two sets of simulations are conducted. Note that in the end-to-end control approach, each controller notifies the upstream ordinary sources regarding the maximum allowed traffic rate that they can send over the next control interval.

For our proposed controller simulations below, the sampling rate is set to $T_s = 1$ ms, the design parameters are selected as $\mu_{1p} = 200$, $\mu_{hp} = 100$ with $h = (2,3)$, $\mu_{hr} = 50$ with $h = (1-3)$, $\Omega_{1p} = 10^{-3}$, $\Omega_{hr} = 1$ with $h = (2,3)$, $\Omega_{hr} = 10^{-3}$ with $h = (1-3)$. The network premium traffic random input λ_{ihp} has a mean of 1000 packet/s and a variance of 100 packet/s, and it is bounded such that

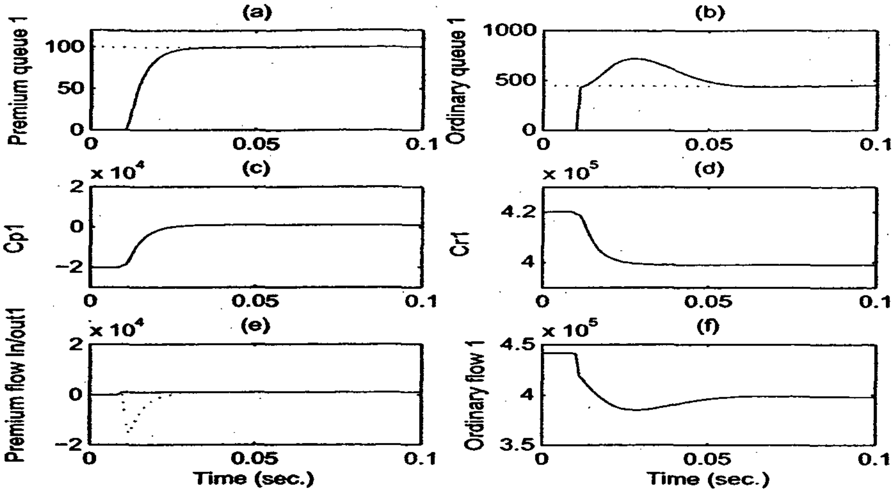


Figure 7.8: Simulation results for the sensor using the proposed congestion controller in the presence of delay of 10 ms and for the same ordinary operating points.

$0 \leq \lambda_{ihp} \leq 3000$ packet/s. The premium and ordinary buffer capacities are finite and their queues are bounded according to the following constraints $0 \leq x_{hp} \leq 128$ packet and $0 \leq x_{hr} \leq 1024$ packet, respectively. Finally, the constraints on the maximum available service capacity are $0 \leq C_{hf} \leq 4\text{x}10^5$ packet/s with $f = (p, r)$. With respect to the time-delay between two nodes of the network, a delay block of 10 ms is introduced to simulate and reflect any system time-delay due to either propagation, transmission, or processing delays, etc.

By setting the node operating points to [100, 110, 90] packets for the Premium service and to [450, 450, 450] packets for the Ordinary service, the first set simulation results show that a) the buffer queues depicted by the graphs (a) and (b) of Figures 7.8, 7.9, and 7.10 corresponding to the Premium and the Ordinary services, respectively, all converge to their respective reference sets, and b) through graph (e) of Figures 7.8, 7.9, and 7.10, the premium output traffic flows converge to the premium input traffic flows according to the fluid flow model conservation principle.

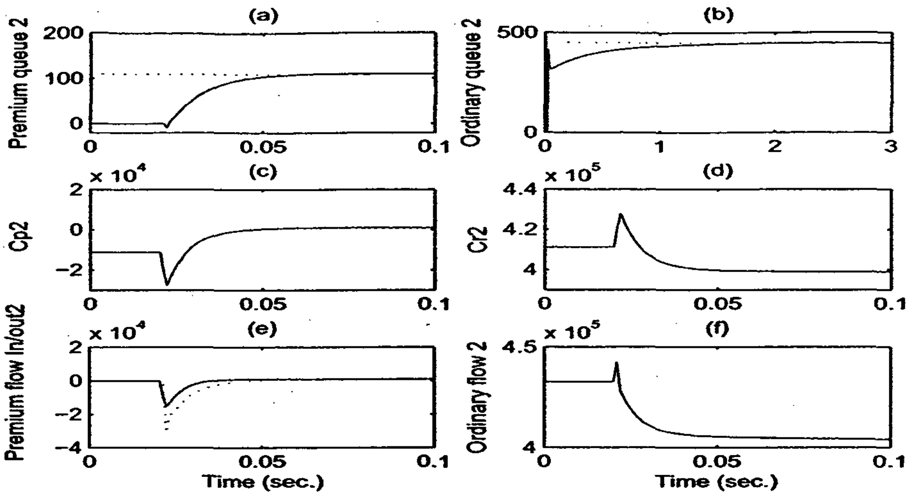


Figure 7.9: Simulation results for the decision-maker using the proposed congestion controller in the presence of delay of 10 ms and for the same ordinary operating points.

In the above three figures, graphs (c) and (d) are devoted to the premium and the ordinary capacities, respectively, while graph (f) corresponds to the ordinary traffic flow.

By now setting the node operating points to $[100, 110, 90]$ packets for the Premium service and to $[450, 430, 420]$ packets for the Ordinary service, it can be observed through the second set of simulations illustrated by Figures 7.11, 7.12, and 7.13 that a) for the Premium service, the buffer queues converge to their respective reference sets, and b) for the Ordinary service, although the buffer queues converge to their respective reference sets, oscillatory responses inducing a steady state error can be observed.

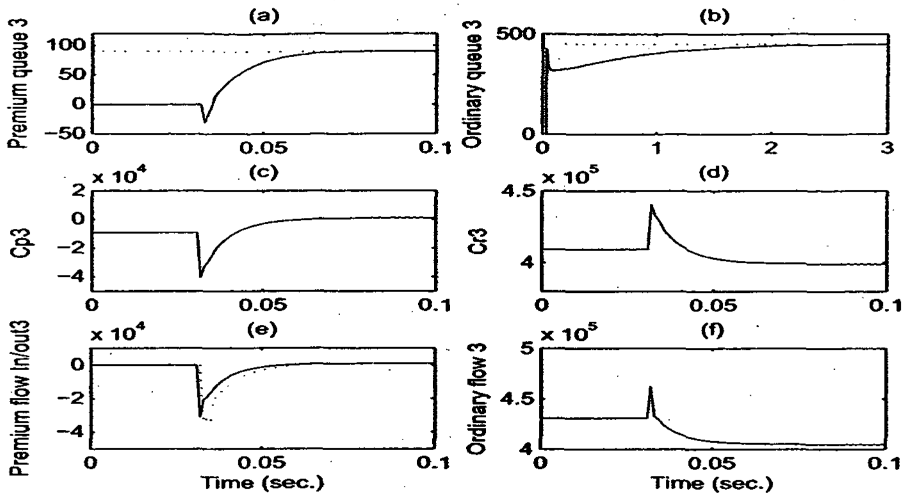


Figure 7.10: Simulation results for the actuator using the proposed congestion controller in the presence of delay of 10 ms and for the same ordinary operating points.

7.3 Conclusion

By using a first-order fluid flow model, our robust congestion control approach is first extended to a time-delayed dependent Differentiated-Services network. The proposed controller is designed using sliding mode variable structure control (SM-VSC) concepts with a convergence condition on the hyperplane dynamics.

Second, the time-delayed dependent congestion control dynamics is derived and studied analytically for a single node in order to guarantee stability of the closed-loop system. It is observed that the premium buffer queue dynamics which is the most important traffic in the differentiated services network is insensitive to the time-delay, and a condition on the time-delay upper bound should be satisfied for the ordinary buffer queue dynamics to ensure stability. These results are confirmed through a number of simulations for different unknown but constant time-delay values. By comparing our proposed congestion controllers to the well-known equivalent control sliding mode approach, the performance of our proposed algorithms and their

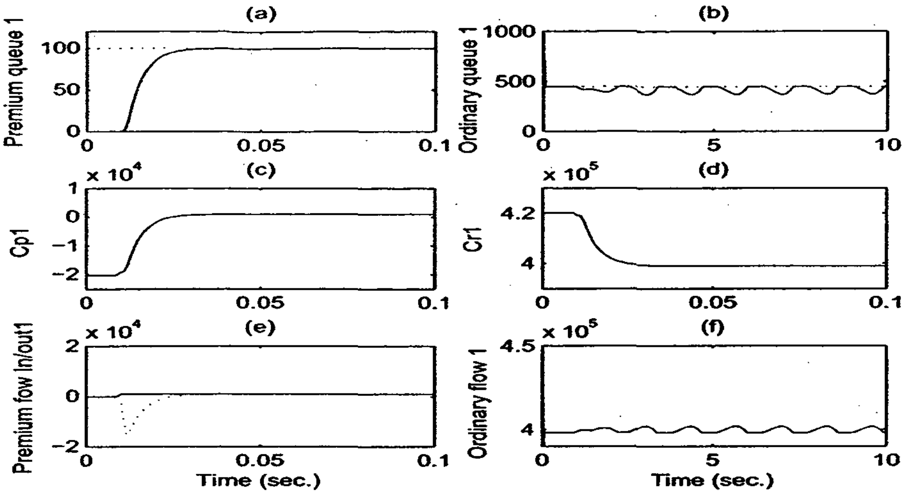


Figure 7.11: Simulation results for the sensor using the proposed congestion controller in the presence of delay of 10 ms and for different ordinary operating points.

robustness are validated for a wide range of operating conditions and time-delays.

Third, the implementation of our proposed congestion controller is performed for a multi-node (cascaded) network. The time-delayed dependent congestion control dynamics is derived analytically in order to guarantee stability of the closed-loop system eventhough the approach is conservative. Simulations results have shown that for the Premium service, the buffer queues converge to their respective reference sets and for the Ordinary service, the boundedness of the buffer queues can be reached but with oscillatory responses inducing a steady state error.

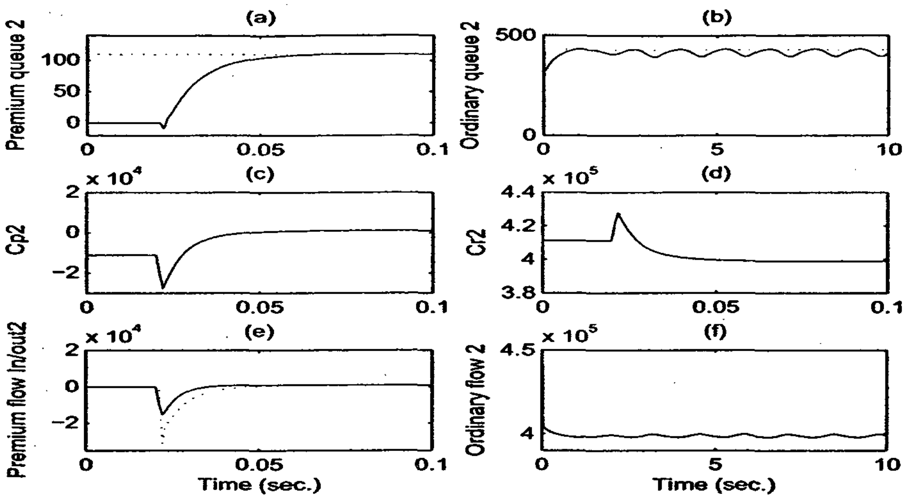


Figure 7.12: Simulation results for the sensor using the proposed congestion controller in the presence of delay of 10 ms and for different ordinary operating points.

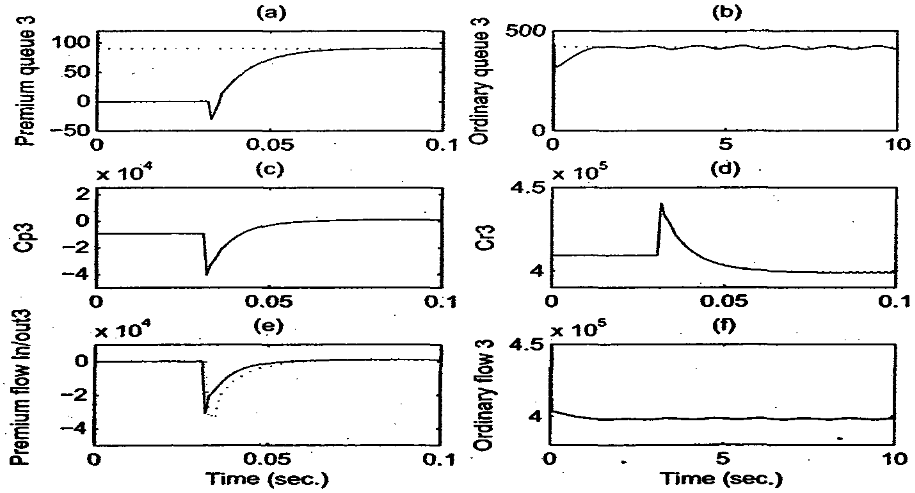


Figure 7.13: Simulation results for the actuator using the proposed congestion controller in the presence of delay of 10 ms and for different ordinary operating points.

Chapter 8

SM-GVS Congestion Control for Feedbacked DiffServ Networks: Degenerate Case

8.1 Introduction

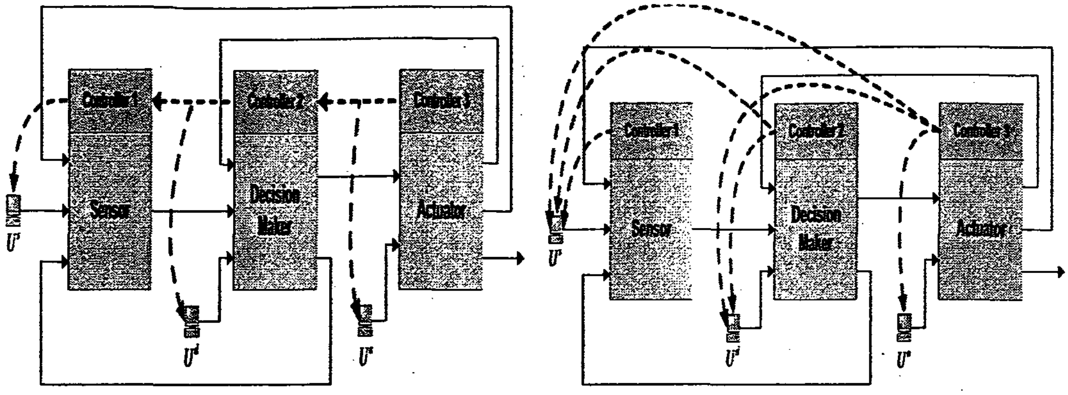
The aim of this chapter is to extend the results on cascaded DiffServ networks that was proposed in Chapter 7 as well as in [19] and [20] to feedbacked networks. The extension consists of (a) developing our congestion control strategy for a network that consists of multiple feedback loops, (b) improving and reformulating the proposed congestion controller design by adding the traffic estimation and the ordinary bandwidth allocation control, (c) implementing and evaluating the proposed congestion control algorithm on a realistic DiffServ network using the end-to-end as well as the hop-by-hop principles, (d) deriving the error dynamics of the overall controlled network, and (e) analyzing the stability of the time-delay dependent system using a less conservative method.

The remainder of this chapter is structured as follows. After stating the proposed congestion control strategies for a feedbacked network in section 2, two congestion control strategies are designed according to the semi-decentralized end-to-end as well as the distributed hop-by-hop approaches in Sections 8.3 and 8.4, respectively. In Section 8.5, the distributed hop-by-hop congestion controller is evaluated on a general loopless DiffServ mesh network (internet) and then on an unmanned system network (NUS) in Section 8.6. Finally, the conclusions are stated in Section 8.7.

8.2 Proposed Congestion Control Strategies

Our proposed robust congestion control strategies consist of (a) assigning a controller to each network node, (b) locally controlling each premium and ordinary buffer queue length by acting on the server bandwidth, and (c) simultaneously regulating the ordinary traffic rate that is flowing through the ordinary buffer by adopting either the *semi-decentralized* end-to-end approach in the sense of Definition 8.1 or the *distributed* hop-by-hop approach in the sense of Definition 8.2. Furthermore, note that a decentralized end-to-end approach is adopted in the congestion control strategy (IDCC) proposed in [89] as well as in our previous work [20].

As illustrated in Figure 8.1(b), in the semi-decentralized end-to-end congestion control approach, the ordinary traffic rate control is accomplished by notifying all the upstream sources/users that are using the node regarding the maximum allowed ordinary rate, while only the immediate upstream sources/users and/or nodes are notified in the distributed hop-by-hop congestion control approach as shown in Figure 8.1(a). Note that in the latter approach, the notification of the sending node is accomplished via its controller. Following the hop-by-hop choke packets concept,



(a) Proposed distributed hop-by-hop based congestion control approach for our three node cluster

(b) Semi-decentralized congestion control approach for our three node cluster motivated from [89]

Figure 8.1: Reduction in the traffic flow of feedback signaling by using our proposed hop-by-hop congestion control approach (solid line: traffic flow, dashed line: feedback signaling).

the explicit information sent through the upstream node controllers is repeated successfully all the way up to the involved sources (see Figure 8.1(a)). Thus, by communicating with only the immediate upstream node and/or the source, the amount of information exchanged is clearly reduced. In fact, several work have already shown the benefits of the hop-by-hop congestion control approach when compared to the end-to-end approach [83], [124].

As far as the notification is concerned, note that depending on the technology selected, different feedback signaling schemes may be adopted. For instance, to convey the feedback signaling, a special field in the cell or packet is updated such as the resource management (RM) cells in the ATM Available Bite Rate (ABR) service or the receiver window in the TCP header.

Definition 8.1. *As illustrated in Figure 8.1(b), the congestion control strategy is called semi-decentralized end-to-end if at each node h, without sharing any information with other node controllers, the controller maintains locally the averaged buffer queue size of node h close to the desired value and regulates the incoming ordinary* *traffic rate to node h by informing all upstream sources that are using node h regarding the maximum allowed rate.*

Definition 8.2. *As illustrated in Figure 8.1(a), the congestion control strategy is called distributed hop-by-hop if at each node h, by communicating with only immediate neighboring node controllers assigned to nodes that are exchanging traffic with node h, the controller maintains locally the averaged buffer queue size of node h close to the desired value and regulates the incoming ordinary traffic rate to node h by informing the immediate upstream sources that use node h as well as the controllers assigned to immediate nodes that are using node h regarding the maximum allowed rate.*

As mentioned above, note that our proposed congestion controller is designed first for a single node and then implemented on each node of the entire network as described in the following sections.

8.3 Proposed Semi-Decentralized End-To-End Congestion Control Design

The robustness capabilities of the sliding mode variable structure control (SM-VSC) are utilized to design on the basis of an inaccurate/uncertain queueing model our new congestion control algorithm. The utilized FFM (2.1) is of a first-order and simpler than a detailed (albeit more accurate) Markovian queueing models.

For the l^{th} output port of the node h, our controller is designed in the error state space coordinates by first defining the state error variables (refer to equation (2.1)) as $e_{l_h f} = x_{l_h f} - x_{l_h f_{ref}}$ and $\dot{e}_{l_h f} = \dot{x}_{l_h f} - \dot{x}_{l_h f_{ref}}$, where $x_{l_h f_{ref}}(t)$ is the reference "trajectory" or the desired requirement of the buffer queue length. In view of equation (2.1), the representation can now be rewritten as follows:

$$\begin{cases} \dot{e}_{l_h f}(t) = -C_{l_h f}(t)\dfrac{e_{l_h f}(t) + x_{l_h f_{ref}}(t)}{e_{l_h f}(t) + x_{l_h f_{ref}}(t) + 1} + \lambda_{il_h f}(t) - \dot{x}_{l_h f_{ref}}(t) \\ y_{l_h f}(t) = e_{l_h f}(t) + x_{l_h f_{ref}}(t); \ (f = p, r) \end{cases} \tag{8.1}$$

The proposed congestion control law that consists of two components is derived and expressed according to

$$C^{ac}_{l_h f}(t) = max\{0, min[C^{max}_{l_h f}(t), C_{l_h f}(t)]\}; \ (f = p, r) \tag{8.2a}$$

$$\lambda^{max,ac}_{il_h r}(t) = max\{0, min[C^{max}_{l_h r}(t), \lambda^{max}_{il_h r}(t)]\} \tag{8.2b}$$

where equations (8.2a) and (8.2b) are dedicated to the ***bandwidth allocation control*** of both services and the ***flow rate control*** of the ordinary service, respectively. The superscript *ac* designates the ***actual value***. By applying the SM-VSC machinery to equation (8.2a), the buffer capacity is then set to

$$C_{l_h f}(t) = E^{-1}_{l_h f}(\mu_{l_h f} e_{l_h f} + \gamma_{l_h f} sgn(e_{l_h f}) + \hat{\lambda}_{il_h f} - \dot{x}_{l_h f_{ref}}) \tag{8.3}$$

where $E_{l_h f} = \frac{x_{l_h f}}{x_{l_h f}+1} = \frac{e_{l_h f}+x_{l_h f_{ref}}}{e_{l_h f}+x_{l_h f_{ref}}+1}$ is the buffer occupancy, $\mu_{l_h f}$ and $\gamma_{l_h f}$ are design parameters, and $\hat{\lambda}_{il_h f}$ is the estimate of the input traffic. The estimate of the traffic is obtained by using one of the following update laws

$$\dot{\hat{\lambda}}_{il_h f} = \Gamma_{l_h f} e_{l_h f}(t) \tag{8.4a}$$

$$\dot{\hat{\lambda}}_{il_h f} = \Gamma_{l_h f} e_{l_h f}(t) - \sigma_{l_h f} \hat{\lambda}_{il_h f} \tag{8.4b}$$

where equation (8.4b) corresponds to a σ-modification approach in the adaptive control domain [57,64], and $\sigma_{l_h f} > 0$ and $\Gamma_{l_h f}$ are the design parameters.

The buffer capacity $C^{ac}_{l_h f}$ is bounded by its maximum value $C^{max}_{l_h f}$ which is equal to C_{serv} for the premium buffer while for the ordinary buffer it is equal to $C_{serv} - C^{ac}_{l_h p}$. In equation (8.2b), the term

$$\lambda^{max}_{il_h r} = (C_{serv} - C^{ac}_{l_h p})E_{l_h r} - \mu_{l_h r} e_{l_h r} - \gamma_{l_h r} sgn(e_{l_h r}) + \dot{x}_{l_h r_{ref}} \tag{8.5}$$

is the maximum ordinary traffic flow rate that is allowed by the controller and whose actual value $\lambda_{il_h r}^{max,ac}(t)$ is bounded by $C_{serv} - C_{l_h p}^{ac}$.

It is worth noting that our proposed control approach above can be interpreted as a dynamic feedback control of the FFM (8.1) that is empowered by robustness capabilities. The resulting error dynamics of the l^{th} closed-loop controlled output port of node h is now given by

$$\dot{e}_{l_h} = -A_{l_h} e_{l_h} - B_{l_h} sgn(e_{l_h}) + \tilde{\lambda}_{il_h} \tag{8.6}$$

where $e_{l_h}^T = [e_{l_h p} \quad e_{l_h r}]$, $A_{l_h} = diag[\mu_{l_h p}\ \mu_{l_h r}]$, $B_{l_h} = diag[\gamma_{l_h p}\ \gamma_{l_h r}]$, $sgn(e_{l_h}) \triangleq col[sgn(e_{l_h p}) \quad sgn(e_{l_h r})]$, and $\tilde{\lambda}_{il_h} = [\tilde{\lambda}_{il_h p}\ \tilde{\lambda}_{il_h r}]^T$ with $\tilde{\lambda}_{il_h f} = \lambda_{il_h f} - \hat{\lambda}_{il_h f}$. The characteristics of the closed-loop system (8.6) and (8.4a) is now described by the following theorem.

Theorem 8.1. *For a differentiated-services network where the l^{th} output port of node h is described by the Fluid Flow Model* (8.1), *by adopting the semi-decentralized end-to-end control approach, the congestion control law governed by equations* (8.2a), (8.2b), *and* (8.4a) *guarantees the asymptotic stability of the error dynamics* (8.6) *provided that $\mu_{l_h f}$, $\gamma_{l_h f}$, $\Gamma_{l_h f}$ $(f = p, r)$ are selected as positive design parameters.*

Proof: Let us choose a candidate Lyapunov function for the closed-loop system (8.6) and (8.4a) as $V(e_{l_h}, \tilde{\lambda}_{il_h}) = \frac{e_{l_h}^T e_{l_h}}{2} + \frac{\tilde{\lambda}_{il_h}^T \Gamma_{l_h} \tilde{\lambda}_{il_h}}{2}$, where $\tilde{\lambda}_{il_h}^T = [\tilde{\lambda}_{il_h p} \quad \tilde{\lambda}_{il_h r}]$, $\tilde{\lambda}_{il_h f} = \lambda_{il_h f} - \hat{\lambda}_{il_h f}$ and $\Gamma_{l_h} = diag[\Gamma_{l_h p}^{-1} \quad \Gamma_{l_h r}^{-1}]$ such that $V(e_{l_h}, \tilde{\lambda}_{il_h})$ is positive definite in $\mathbb{D} = \{(e_{l_h}, \tilde{\lambda}_{il_h}) \in \Re^4\}$. The time derivative of the Lyapunov function along the trajectories of (8.6) and (8.4a) is given by

$$\dot{V}(e_{l_h}, \tilde{\lambda}_{il_h}) = -\sum_{f=p,r} \mu_{l_h f} e_{l_h f}^2 - \sum_{f=p,r} \gamma_{l_h f} e_{l_h f} sgn(e_{l_h f}) \tag{8.7}$$

Given that $e_{l_h f}(t) sgn(e_{l_h f}(t))$ is always positive since

$$sgn(e_{l_h f}(t)) = \begin{cases} 1 \ if \ \ e_{l_h f}(t) > 0 \\ 0 \ if \ \ e_{l_h f}(t) = 0 \\ -1 \ if \ \ e_{l_h f}(t) < 0 \end{cases} \tag{8.8}$$

It follows that $\dot{V}$ is negative semi-definite. Using the La Salle's invariance principle [68], let us define $\mathbb{S} = \{(e_{l_h}, \tilde{\lambda}_{il_h}) \in \mathbb{D} | \dot{V}(e_{l_h}, \tilde{\lambda}_{il_h}) = 0\}$. For $\dot{V}(e_{l_h}, \tilde{\lambda}_{il_h}) = 0$, $e_{l_h} = 0$ $\Rightarrow \mathbb{S} = \{(e_{l_h}, \tilde{\lambda}_{il_h}) \in \mathbb{D} | e_{l_h} = 0\}$. Assuming that $(e_{l_h}, \tilde{\lambda}_{il_h}) \ \in \ \mathbb{S} \ \forall \ t \Rightarrow e_{l_h}(t) = 0 \ \ \forall \ \ t$, and $\dot{e}_{l_h}(t) \equiv 0 \Rightarrow 0 = -A_{l_h} e_{l_h} - B_{l_h} sgn(e_{l_h}) + \tilde{\lambda}_{il_h}$ which implies that $e_{l_h}(t) = 0$ and $\tilde{\lambda}_{il_h}(t) = 0$. Therefore, the only invariant solution in $\mathbb{S}$ is the trivial solution $e_{l_h}(t) = 0$ and $\tilde{\lambda}_{il_h}(t) = 0$. Hence, the equilibrium point $e_{l_h} = 0$ and $\tilde{\lambda}_{il_h} = 0$ is asymptotically stable. ▲

It is widely known that in certain circumstances the update law (8.4a) considered in Theorem 8.1 may suffer from drift errors [57]. Indeed, a small error in the output, may cause $\tilde{\lambda}_{il_h}$ to become unbounded. To overcome this, the update law (8.4a) can be modified by adding a damping term where the resulting approach is known as the σ-modification [64] that is given by expression (8.4b). We now have the following theorem corresponding to the resulting update law.

Theorem 8.2. *For a differentiated-services network where the l^{th} output port of node h is described by the Fluid Flow Model* (8.1)*, by adopting the semi-decentralized end-to-end control approach, the congestion control law governed by equations* (8.2a), (8.2b) *and* (8.4b) *guarantees the ultimate boundedness of the error dynamics* (8.6) *provided that $\sigma_{l_h f}$, $\mu_{l_h f}$, $\gamma_{l_h f}$, and $\Gamma_{l_h f}$, $(f = p, r)$ are selected as positive design parameters.*

Proof: Let us choose a candidate Lyapunov function for the closed-loop system (8.6) and (8.4b) as $V(e_{l_h}, \tilde{\lambda}_{il_h}) = \frac{e_{l_h}^T e_{l_h}}{2} + \frac{\tilde{\lambda}_{il_h}^T \Gamma_{l_h} \tilde{\lambda}_{il_h}}{2}$, where $\Gamma_{l_h} = diag[\Gamma_{l_h p}^{-1} \quad \Gamma_{l_h r}^{-1}]$ such that

$V(e_{l_h}, \tilde{\lambda}_{il_h})$ is positive definite in $\mathbb{D} = \{(e_{l_h}, \tilde{\lambda}_{l_h}) \in \Re^4\}$. The time derivative of the Lyapunov function along the trajectories of (8.6) and (8.4b) is given by

$$\begin{aligned}
\dot{V} &= -\sum_{f=p,r} \mu_{l_hf} e^2_{l_hf} - \sum_{f=p,r} \gamma_{l_hf} e_{l_hf} sgn(e_{l_hf}) + \Delta_{l_h}(t) \\
&\leq -\sum_{f=p,r} \mu_{l_hf} e^2_{l_hf} - \sum_{f=p,r} \Gamma^{-1}_{l_hf} \sigma_{l_hf} [\tilde{\lambda}^2_{il_hf} - \lambda_{il_hf,max} \tilde{\lambda}_{il_hf}] \\
&= -\Pi_1 - \sum_{f=p,r} \Gamma^{-1}_{l_hf} \sigma_{l_hf} [(\tilde{\lambda}_{il_hf} - \frac{\lambda_{il_hf,max}}{2})^2 - \frac{\lambda^2_{il_hf,max}}{4}] \\
&= -\Pi_1 - \Pi_2 + \sum_{f=p,r} \Gamma^{-1}_{l_hf} \sigma_{l_hf} \frac{\lambda^2_{il_hf,max}}{4}
\end{aligned} \tag{8.9}$$

where $\Delta_{l_h}(t) = \sum_{f=p,r} (\tilde{\lambda}_{il_hf} e_{l_hf} + \Gamma^{-1}_{l_hf} \tilde{\lambda}_{il_hf} \dot{\tilde{\lambda}}_{il_hf})$, $\Pi_1 = \sum_{f=p,r} \mu_{l_hf} e^2_{l_hf}$ and $\Pi_2 = \sum_{f=p,r} \Gamma^{-1}_{l_hf} \sigma_{l_hf} (\tilde{\lambda}_{il_hf} - \frac{\lambda_{il_hf,max}}{2})^2$. It follows that $\dot{V}(e_{l_h}, \tilde{\lambda}_{il_h})$ is negative semi-definite for the states e_{l_h} and λ_{il_h} outside the region defined by

$$\sum_{f=p,r} K_{l_hf_1} e^2_{l_hf} + \sum_{f=p,r} K_{l_hf_2} (\tilde{\lambda}_{il_hf} - \frac{\lambda_{il_hf,max}}{2})^2 \leq 1 \tag{8.10}$$

where $K_{l_hf_1} = \frac{\mu_{l_hf} \Gamma_{l_hf}}{\sigma_{l_hf}} K_{l_hf_2}$, $K_{l_hf_2} = \frac{4}{\lambda^2_{il_hf,max}}$ and $\lambda_{il_hf,max}$ is the upper bound on the input traffic and $\dot{V} > 0$ for the states inside that region. This implies the ultimate boundedness of the solution of the error dynamics (8.6) and this completes the proof of the theorem. ▲

8.3.1 Simulation results obtained with the semi-decentralized end-to-end congestion controller

The proposed semi-decentralized end-to-end congestion controller is evaluated on the network given by Figure 8.2. This is a five nodes network that communicate in cascade as a conventional DiffServ network. Indeed, as in internet, the network of Figure 8.2 does not consider any feedback traffic. To each node of our network,

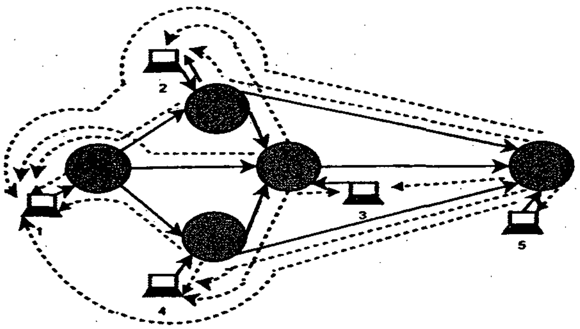


Figure 8.2: DiffServ network adopting the proposed semi-decentralized end-to-end congestion control approach (solid line: traffic flow, dashed and dotted lines: feedback signaling).

one connects a source/user that generates a random DiffServ traffic with a mean of 3000 packets/s and a variance of 1000 packets/s. The premium traffic is assumed to be bounded such that $0 \leq \lambda_{il_hp} \leq 4000$ packets/s. The premium and ordinary buffer capacities are finite and their queues are bounded according to the following constraints: $0 \leq x_{l_hp} \leq 128$ packets and $0 \leq x_{l_hr} \leq 1024$ packets, respectively. According to Assumption 3.3 the constraints on the maximum available service capacity are $0 \leq C_{l_h} \leq 40000$ packets/s for both services. The premium and ordinary references of the buffer queues are given in Table 8.1 while the sampling time is $T_s = 1$ ms.

Node	$x_{P_{ref}}$ [packets]	$x_{r_{ref}}$ [packets]
node 1	50	140
node 2	30	90
node 3	70	60
node 5	10	40
node 4	30	70

Table 8.1: Premium and ordinary buffer queue references.

The simulation results corresponding to the buffer queue step responses of all nodes are shown in Figure 8.3. The left and the right columns are devoted to the

Premium and the Ordinary services, respectively, whereas each row is dedicated to one node of the network.

By inspecting the plots presented in Figure 8.3, one may readily argue that for the Premium service all the buffer queues converge to their respective references while for the Ordinary service the queues are bounded but responses are oscillatory.

Remark 8.1. *It is worth noting that our proposed semi-decentralized end-to-end congestion controller is evaluated in simulations on the five nodes cascade configuration and oscillatory responses are obtained. This shows the weak scalability property of the end-to-end approach.*

8.4 Proposed Distributed Hop-By-Hop Congestion Control Design

For the l^{th} output port of node h, our distributed hop-by-hop congestion controller is designed in the error state coordinates using the same state error variables and representation (8.1) as in the semi-decentralized end-to-end congestion approach studied in the previous section.

Our proposed distributed hop-by-hop congestion control law that consists of two components is derived and expressed according to

$$C_{l_h f}^{ac}(t) = max\{0, min[C_{l_h f}^{max}(t), C_{l_h f}(t)]\}; \; (f = p, r) \tag{8.11a}$$

$$\lambda_{il_h r}^{max,ac}(t) = max\{0, min[C_{l_h r}^{max}(t), \lambda_{il_h r}^{max}(t)]\} \tag{8.11b}$$

where equations (8.11a) and (8.11b) are dedicated to the *bandwidth allocation control* of both services and the *flow rate control* of the ordinary service, respectively. The superscript *ac* designates the *actual value.* By applying the SM-VSC machinery to equation (8.11a), the buffer capacity is then set to

$$C_{l_h f}(t) = E_{l_h f}^{-1}(\mu_{l_h f} e_{l_h f} + \gamma_{l_h f} sgn(e_{l_h f}) + \hat{\lambda}_{il_h f} - \dot{x}_{l_h f_{ref}}) \tag{8.12}$$

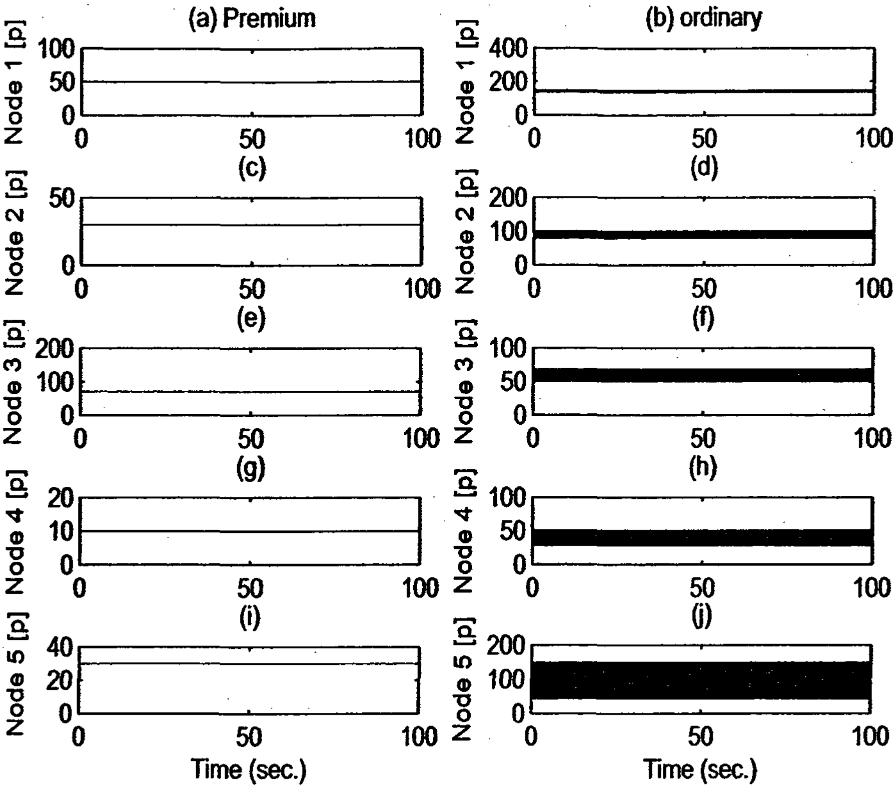


Figure 8.3: Simulation results for the five node network by using our proposed semi-decentralized end-to-end congestion controller.

where $E_{l_hf} = \frac{x_{l_hf}}{x_{l_hf}+1} = \frac{e_{l_hf}+x_{l_hf_{ref}}}{e_{l_hf}+x_{l_hf_{ref}}+1}$ is the buffer occupancy, μ_{l_hf} and γ_{l_hf} are design parameters, and $\hat{\lambda}_{il_hf}$ is the estimate of the input traffic that is obtained by using one of the update laws given by either (8.4a) or (8.4b).

The buffer capacity $C^{ac}_{l_hf}$ is bounded by its maximum value $C^{max}_{l_hf}$ which is equal to C_{serv} for the premium buffer while for the ordinary buffer it is equal to $C^{max}_{l_hr}$ as given by (8.13)

$$C^{max}_{l_hr} = max\{0, min[C_{serv} - C^{ac}_{l_hp}, \frac{\lambda^{max}_{i1r}(t-\tau)}{M_1}, \dots, \frac{\lambda^{max}_{il_jr}(t-\tau)}{M_{l_j}}]\} \quad (8.13)$$

The term $\lambda^{max}_{il_jr}$ is the maximum traffic flow rate that is allowed by the control law corresponding to the l^{th} output port of the j^{th} immediate downstream node, and M_{l_j} is the number of immediate upstream senders (nodes and/or sources) that use the output port l_j. In equation (8.11b), the term

$$\lambda_{il_hr}^{max} = C_{l_hr}^{max} E_{l_hr} - \mu_{l_hr} e_{l_hr} - \gamma_{l_hr} sgn(e_{l_hr}) + \dot{x}_{l_hr_{ref}} \tag{8.14}$$

is the maximum ordinary traffic flow rate that is allowed by the controller and whose actual value $\lambda_{il_hr}^{max,ac}(t)$ is bounded by $C_{l_hr}^{max}$.

As in the semi-decentralized end-to-end congestion controller, our distributed hop-by-hop congestion controller can also be seen as a dynamic feedback control of the FFM (8.1) enjoying robustness capabilities. The resulting error dynamics of the l^{th} closed-loop controlled output port of node h has the same form as (8.6) that is

$$\dot{e}_{l_h} = -A_{l_h} e_{l_h} - B_{l_h} sgn(e_{l_h}) + \tilde{\lambda}_{il_h} \tag{8.15}$$

where $e_{l_h}^T = [e_{l_hp} \quad e_{l_hr}]$, $A_{l_h} = diag[\mu_{l_hp}\ \mu_{l_hr}]$, $B_{l_h} = diag[\gamma_{l_hp}\ \gamma_{l_hr}]$, $sgn(e_{l_h}) \triangleq col[sgn(e_{l_hp})\ sgn(e_{l_hr})]$ and $\tilde{\lambda}_{il_h} = [\tilde{\lambda}_{il_hp}\ \tilde{\lambda}_{il_hr}]^T$ with $\tilde{\lambda}_{il_hf} = \lambda_{il_hf} - \hat{\lambda}_{il_hf}$. The behavior of the closed-loop system (8.15) is described by the following theorem.

Theorem 8.3. *For a differentiated-services network where the l^{th} output port of node h is described by the Fluid Flow Model* (8.1), *by adopting the distributed hop-by-hop control approach, the congestion control law governed by equations* (8.11a), (8.11b), *and* (8.4a) *guarantees the asymptotic stability of the error dynamics* (8.15) *provided that μ_{l_hf}, γ_{l_hf}, Γ_{l_hf} $(f = p, r)$ are selected as positive design parameters.*

Proof: Since the error dynamics of equation (8.15) is similar to the one given by (8.6), therefore the details of the proof of Theorem 8.3 that uses the update law (8.4a) is similar to that of Theorem 8.1, and hence the details are omitted. ▲

As mentioned in the semi-decentralized end-to-end congestion controller, the update law (8.4a) considered in Theorem 8.3 can be inefficient in some circumstances, therefore, the σ-modification should lead to better results. For example, our preliminary results have demonstrated a poor convergence of the update law (8.4a) when the proposed distributed hop-by-hop congestion controller is implemented on the entire network. Therefore, the update law (8.4a) is modified by adding a damping term $-\sigma_{l_h f}\hat{\lambda}_{il_h f}$ as given by (8.4b). By now considering the update (8.4b), our proposed distributed hop-by-hop congestion control strategy is modified and is now described by the following theorem. Note that the modified controller is from now utilized in this chapter in stability analysis and simulations.

Theorem 8.4. *For a differentiated-services network where the l^{th} output port of node h is described by the Fluid Flow Model* (8.1), *by adopting the distributed hop-by-hop control approach, the congestion control law governed by equations* (8.11a), (8.11b), *and* (8.4b) *guarantees the ultimate boundedness of the error dynamics* (8.15) *provided that $\sigma_{l_h f}$ and $\mu_{l_h f}$, $\gamma_{l_h f}$, and $\Gamma_{l_h f}$, $(f = p, r)$ are positive design parameters.*

Proof: Since the error dynamics of equation (8.15) is the same as the one given by (8.6), therefore the proof of Theorem 8.4 is the same as that of Theorem 8.2, and hence the details are omitted. ▲

Remark 8.2. *In compliance with the bandwidth (capacity) allocation control that is used in the above proposed congestion controllers, it is worth noting that the server can allocate a certain capacity to forward packets from the Best Effort buffer such that $C_{serv} = C^{ac}_{hp} + C^{ac}_{hr} + C^{ac}_{hb}$, where C^{ac}_{hb} is the actual capacity allocated to the Best Effort service [105].*

8.5 Proposed Distributed Hop-By-Hop Congestion Controller for a DiffServ Loopless Network (Internet) [23]

The proposed distributed hop-by-hop congestion controller that is designed for a single node h is now implemented on each node of the network enjoying thus the scalability property.

Our proposed robust congestion controller is developed for a DiffServ architecture [13], [40], [51], and is evaluated analytically on an nxn nodes network as illustrated in Figure 8.4. First, note that to demonstrate the capabilities of our proposed congestion control strategy, the network is assumed to have a mesh topology. As discussed in Chapter 3, compared to the ring or the star topologies, the mesh topology is on one hand well-known to be one of the most complex and on the other

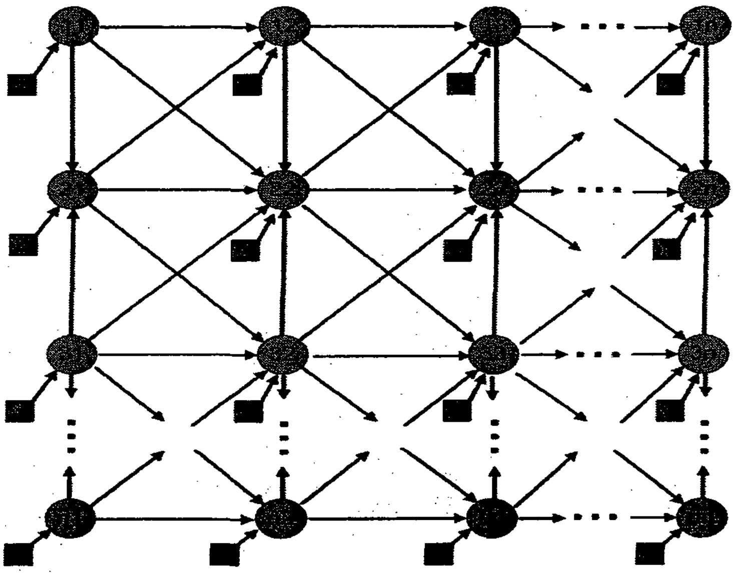


Figure 8.4: Proposed nxn nodes mesh network.

hand is appropriate for distributed hop-by-hop control strategies.

It is worth noting that the mesh network depicted in Figure 8.4 corresponds to the network given by Figure 3.3 in Chapter 3 in which the feedback traffics are disabled. Note also that since the network of Figure 8.4 is not studied as a NUS but as an Internet network, the notation given in Figure 3.3 are simplified, e.g. indices s, d, and a corresponding to the sensor, the decision-maker and the actuator, respectively are not considered in this case.

As shown in Figure 8.4, we have assumed that nodes communicate in diagonal directions. In order to make the nodes that are located at the core of the network differently fed along the vertical direction, the nodes of the even rows are assumed to receive information flows while those of the odd rows are sending information flows. For example, by considering that n is an odd number, the nodes of the last row are hence sending vertically information to the nodes of the above row $n-1$. To each node of our network, one connects a source/user traffic (denoted as U_h, $h = 11, 12, \ldots, 1n, \ldots, n1, n2, \ldots, nn$).

8.5.1 Error dynamics of the overall network

Independent of the topology, the overall network can be seen as a compartmental system [62] in which each node can be considered as a conceptual buffer while the overall system is governed by conservation of flow law and whose state variables (buffer queue lengths) are constrained to remain non-negative.

Furthermore, among the three services that constitute our differentiated-services network, the opportunistic Best Effort service does not intervene with the procedure and constraint of the control design strategy eventhough a small capacity can be allocated to it. Therefore, the network error dynamics will be obtained for the premium and the ordinary services. The overall network dynamics can be derived by interconnecting the nodes dynamics through their traffic inputs. Note that, the premium and the ordinary dynamics are both similar in the open-loop since they are modeled using the FFM (8.1) and in the closed-loop since they follow the same bandwidth allocation law given by Theorem 8.4. Using this bandwidth allocation scheme, any h^{th} node of our network can be described in closed-loop by the following error dynamics,

$$\dot{e}_{h,f} = -\mu_{h,f} e_{h,f} - \gamma_{h,f} sgn(e_{h,f}) - \hat{\lambda}_{ih,f} + \sum_{k=1}^{k_h} \lambda_{ik,f} \qquad (8.16)$$

where $\lambda_{ik,f}$ is the k^{th} input traffic to node h with i denoting the input, k_h is the number of inputs at node h, $h = 11, 12, \ldots, 1n, \ldots, n1, n2, \ldots, nn$ and $f = p, r$. Any k^{th} input traffic at node j that is the immediate downstream node to node h corresponds to the delayed output traffic of node h (its sender). As mentioned earlier the sender can be a source (user) or the immediate upstream node h. As far as the latter node is concerned, its output traffic may be expressed by (8.17) according to the conservation principle of the FFM on which its buffer is modeled, namely

$$\lambda_{ij_k,f}(t) = \lambda_{oh,f}(t - \tau(t)) = C_{h,f}(t - \tau(t)) E_{h,f}(t - \tau(t)) \qquad (8.17)$$

where h designates the immediate upstream node to node j and o denotes the output.

After interconnecting all the nodes and by substituting their inputs according to (8.17), one obtains the n_1^{th}-order error dynamics of the entire network represented in the matrix form in (8.18) with $n_1 = n^2$, and $f = p, r$ as shown below,

$$\dot{e}_f = -A_{1,f} e_f + A_{2,f} e_f(t-\tau) - A_{3,f} sgn(e_f) + A_{4,f} sgn(e_f(t-\tau)) + W_f + \lambda^U_{i,f} \quad (8.18)$$

where $W_f = -\hat{\lambda}_{i,f} + B_{1,f}[\hat{\lambda}_{i,f}(t-\tau) - \dot{x}_{f_{ref}}(t-\tau)]$, and for $m = 1, \ldots, n$, $e_f = [e_{11,f}\ e^T_{1,f}\ e_{21,f}\ e^T_{2,f}\ \ldots\ e_{n1,f}\ e^T_{n,f}]^T$ with $e_{m,f} = [e_{m2,f}\ e_{m3,f}\ \ldots\ e_{mn,f}]^T$, $\lambda^U_{i,f} = [\lambda^U_{i11,f}\ (\lambda^U_{i1,f})^T\ \lambda^U_{i21,f}\ (\lambda^U_{i2,f})^T\ \ldots\ \lambda^U_{in1,f}\ (\lambda^U_{in,f})^T]^T$ with $\lambda^U_{im,f} = [\lambda^U_{im2,f}\ \lambda^U_{im3,f}\ \ldots\ \lambda^U_{imn,f}]^T$, $\hat{\lambda}_{i,f} = [\hat{\lambda}_{i11,f}\ \hat{\lambda}^T_{i1,f}\ \hat{\lambda}_{i21,f}\ \hat{\lambda}^T_{i2,f}\ \ldots\ \hat{\lambda}_{in1,f}\ \hat{\lambda}^T_{in,f}]^T$ with $\hat{\lambda}_{im,f} = [\hat{\lambda}_{im2,f}\ \hat{\lambda}_{im3,f}\ \ldots\ \hat{\lambda}_{imn,f}]^T$, $\dot{x}_{f_{ref}} = [\dot{x}_{11,f_{ref}}\ \dot{x}^T_{1,f_{ref}}\ \dot{x}_{21,f_{ref}}\ \dot{x}^T_{2,f_{ref}}\ \ldots\ \dot{x}_{n1,f_{ref}}\ \dot{x}^T_{n,f_{ref}}]^T$ with $\dot{x}_{mf_{ref}} = [\dot{x}_{m2,f_{ref}}$

$\dot{x}_{m3,f_{ref}} \ \cdots \ \dot{x}_{mn,f_{ref}}]^T$, $A_{1,f} = diag[\mu_{11,f}\ \mu_{1,f}\ \mu_{21,f}\ \mu_{2,f}\ \cdots\ \mu_{n1,f}\ \mu_{n,f}]$ with $\mu_{m,f} = diag[\mu_{m2,f}\ \mu_{m3,f}\ \cdots\ \mu_{mn,f}]$, $A_{3,f} = diag[\gamma_{11,f}\ \gamma_{1,f}\ \gamma_{21,f}\ \gamma_{2,f}\ \cdots\ \gamma_{n1,f}\ \gamma_{n,f}]$ with $\gamma_{m,f} = diag[\gamma_{m2,f}\ \gamma_{m3,f}\ \cdots\ \gamma_{mn,f}]$,

$$A_{2,f} = \begin{bmatrix} A^1_{2f} & & * & & * \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ * & \cdots & A^2_{2f} & \cdots & * \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ * & \cdots & * & \cdots & A^3_{2f} \end{bmatrix} \tag{8.19}$$

with the matrix blocks A^1_{2f} and A^3_{2f} that are given in (8.20), respectively, corresponding to the nodes of the first and the last rows of the network as illustrated in Figure 8.4, respectively

$$A^1_{2f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \mu'_{11,f} & \mathbb{M}_{11f} & \mu'_{21f} & \mathbb{M}_{21f} \end{pmatrix}; A^3_{2f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \mu'_{n-1,1,f} & \mathbb{M}_{n-1,1,f} & \mu'_{n,1,f} & \mathbb{M}_{n,1,f} \end{pmatrix} \tag{8.20}$$

In the matrix $A_{2,f}$, the block A^2_{2f} is expressed by (8.21), where the first and the last two rows correspond to the nodes of the m^{th} and the $(m+1)^{th}$ rows of the network, respectively $(m = 1, \ldots, n)$

$$A^2_{2f} = \begin{bmatrix} \mu_{m-1,1f} & 0 & 0 & 0 & \mu_{m+1,1f} & 0 & 0 & 0 \\ \mu'_{m-1,1f} & \mathbb{M}'_{m-1,1f} & \mu'_{m,1f} & \mathbb{M}_{m,1f} & \mu'_{m+1,1f} & \mathbb{M}'_{m+1,1f} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mu'_{m,1f} & \mathbb{M}_{m,1f} & \mu'_{m+1,1f} & \mathbb{M}_{m+1,1f} & \mu'_{m+2,1f} & \mathbb{M}_{m+2,1f} \end{bmatrix} \tag{8.21}$$

with $\mu'_{m1f} = [\mu_{m1f}\ 0^T_{(n-1)x1}]^T$,

$$\mathbb{M}_{m1,f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \mu_{m2f} & 0 & 0 & 0 \\ 0 & \mu_{m3f} & 0 & 0 \\ 0 & 0 & \mu_{m4f} & 0 \end{pmatrix}; \ and \ \mathbb{M}'_{m1,f} = \begin{pmatrix} \mu_{m2f} & 0 & 0 & 0 \\ \mu_{m2f} & 0 & 0 & 0 \\ 0 & \mu_{m3f} & 0 & 0 \\ 0 & 0 & \mu_{m4f} & 0 \end{pmatrix} \tag{8.22}$$

Corresponding to the error dynamics (8.18), the matrix $A_{4,f}$ is given by

$$A_{4,f} = \begin{bmatrix} A^1_{4f} & & * & & * \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ * & \cdots & A^2_{4f} & \cdots & * \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ * & \cdots & * & \cdots & A^3_{4f} \end{bmatrix} \tag{8.23}$$

where the matrix blocks A^1_{4f} and A^3_{4f} given in (8.24), respectively, correspond to the nodes of the first and the last rows of the network as illustrated in Figure 8.4, respectively

$$A^1_{4f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \gamma'_{11,f} & \mathbb{G}_{11f} & \gamma'_{21f} & \mathbb{G}_{21f} \end{pmatrix} ; A^3_{2f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \gamma'_{n-1,1,f} & \mathbb{G}_{n-1,1,f} & \gamma'_{n,1,f} & \mathbb{G}_{n,1,f} \end{pmatrix} \tag{8.24}$$

In the matrix $A_{4,f}$, the block A^2_{4f} is expressed by (8.25), where the first and the last two rows correspond to the nodes of the m^{th} and the $(m+1)^{th}$ rows of the network, respectively $(m = 1, \ldots, n)$,

$$A^2_{4f} = \begin{bmatrix} \gamma_{m-1,1f} & 0 & 0 & 0 & \gamma_{m+1,1f} & 0 & 0 & 0 \\ \gamma'_{m-1,1f} & \mathbb{G}'_{m-1,1f} & \gamma'_{m,1f} & \mathbb{G}_{m,1f} & \gamma'_{m+1,1f} & \mathbb{G}'_{m+1,1f} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma'_{m,1f} & \mathbb{G}_{m,1f} & \gamma'_{m+1,1f} & \mathbb{G}_{m+1,1f} & \gamma'_{m+2,1f} & \mathbb{G}_{m+2,1f} \end{bmatrix} \tag{8.25}$$

with $\gamma'_{m1f} = [\gamma_{m1f}\ 0^T_{(n-1)x1}]^T$,

$$\mathbb{G}_{m1,f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \gamma_{m2f} & 0 & 0 & 0 \\ 0 & \gamma_{m3f} & 0 & 0 \\ 0 & 0 & \gamma_{m4f} & 0 \end{pmatrix} ; \mathit{and}\ \mathbb{G}'_{m1,f} = \begin{pmatrix} \gamma_{m2f} & 0 & 0 & 0 \\ \gamma_{m2f} & 0 & 0 & 0 \\ 0 & \gamma_{m3f} & 0 & 0 \\ 0 & 0 & \gamma_{m4f} & 0 \end{pmatrix} \tag{8.26}$$

The matrix $B_{1,f}$ in equation (8.18) is expressed as

$$B_{1,f} = \begin{bmatrix} B^1_{1f} & & * & & * \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ * & \cdots & B^2_{1f} & \cdots & * \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ * & \cdots & * & \cdots & B^3_{1f} \end{bmatrix} \tag{8.27}$$

where matrix blocks B^1_{1f} and B^3_{1f} given in (8.28), respectively, correspond to the nodes of the first and the last rows of the network as illustrated in Figure 8.4, respectively

$$B^1_{1f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ I' & \mathbb{U}_{11f} & I' & \mathbb{U}_{21f} \end{pmatrix}; B^3_{1f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ I' & \mathbb{U}_{n-1,1,f} & I' & \mathbb{U}_{n,1,f} \end{pmatrix} \tag{8.28}$$

In the matrix $B_{1,f}$, the block B^2_{1f} is expressed by (8.29), where the first and the last two rows correspond to the nodes of the m^{th} and the $(m+1)^{th}$ rows of the network, respectively $(m = 1, \ldots, n)$

$$B^2_{1f} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ I' & \mathbb{U}'_{m-1,1f} & I' & \mathbb{U}_{m,1f} & I' & \mathbb{U}'_{m+1,1f} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I' & \mathbb{U}_{m,1f} & I' & \mathbb{U}_{m+1,1f} & I' & \mathbb{U}_{m+2,1f} \end{bmatrix} \tag{8.29}$$

with $I' = [1\ 0^T_{(n-1)x1}]^T$,

$$\mathbb{U}_{m1,f} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}; \ and\ \mathbb{U}'_{m1,f} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{8.30}$$

The traffic in our network is estimated according to the updated law (8.4a). Consequently, the error dynamics given by (8.18) may be extended to incorporate the estimator dynamics as shown below

$$\dot{\eta}_f = \mathbb{A}_{1,f}\eta_f + \mathbb{A}_{2,f}\eta_f(t-\tau) + \mathbb{A}_{3,f}sgn(e_f) + \mathbb{A}_{4,f}sgn(e_f(t-\tau)) + \mathbb{B}_{1,f}\dot{x}_{f,ref}(t-\tau) + \mathbb{B}_{2,f}\lambda^{U}_{i,f} \tag{8.31}$$

where $\eta_f = [e_f^T \ \hat{\lambda}^T_{i,f}]^T$,

$$\mathbb{A}_{1,f} = \begin{pmatrix} -A_{1,f} & -I \\ \Gamma_f & -\sigma_f \end{pmatrix}; \ and \ \mathbb{A}_{2,f} = \begin{pmatrix} A_{2,f} & B_{1,f} \\ 0 & 0 \end{pmatrix} \tag{8.32}$$

with $\Gamma_f = diag[\Gamma_{11f} \ \Gamma_{1f} \ \dots \ \Gamma_{n1f} \ \Gamma_{nf}]$, $\Gamma_{mf} = diag[\Gamma_{m2f} \ \Gamma_{m3f} \ \dots \ \Gamma_{mnf}]$ $\sigma_f = diag[\sigma_{11f} \ \sigma_{1f} \ \dots \ \sigma_{n1f} \ \sigma_{nf}]$, $\sigma_{mf} = diag[\sigma_{m2f} \ \sigma_{m3f} \ \dots \ \sigma_{mnf}]$ $\mathbb{A}_{3,f} = [-A^T_{3,f} \ 0]^T$, $\mathbb{A}_{4,f} = [A^T_{4,f} \ 0]^T$, $\mathbb{B}_{1,f} = [-B^T_{1,f} \ 0]^T$, and $\mathbb{B}_{2,f} = [I \ 0]^T$.

It is worth noting that the dynamics (8.18) with index $f = p, r$ corresponds to the Premium and the Ordinary services, respectively. In other words, one may argue that both services are described by the same error dynamics. In fact, this is expected since the resulting dynamics is derived on the basis of the same bandwidth allocation procedure for both services. The ordinary traffic flow control as mentioned earlier is accomplished by regulating the ordinary traffic rate at the sender (node and/or the source). One may readily observe that the derived error dynamics (8.18) still contains the input vector $\lambda^U_{i,f}(t)$ which implies that the ordinary flow control corresponding to the sources is not considered yet. In order to separate the premium and the ordinary error dynamics it is judicious to substitute the input vector $\lambda^U_{i,f}(t)$ by $\lambda^U_{o,p}(t-\tau(t))$ and $\lambda^U_{o,r}(t-\tau(t))$, respectively, as shown below.

8.5.1.1 Premium error dynamics of the overall network

The Premium service is concerned solely with the bandwidth allocation. This implies that the premium *sources* are not controlled, therefore, in their error dynamics (8.18) or (8.31) we have $\lambda^U_{i,p}(t) = \lambda^U_{o,p}(t-\tau)$. The premium source output vector is defined as $\lambda^U_{o,p} = [\lambda^{U_{11}}_{o,p} \ (\lambda^{U_1}_{o,p})^T \ \dots \ \lambda^{U_{n1}}_{o,p} \ (\lambda^{U_n}_{o,p})^T]^T$ where $\lambda^{U_m}_{o,p} = [\lambda^{U_{m2}}_{o,p} \ \lambda^{U_{m3}}_{o,p} \ \dots \ \lambda^{U_{mn}}_{o,p}]^T$ with

$m = 1, \ldots, n$.

8.5.1.2 Ordinary error dynamics of the overall network

The ordinary state error dynamics is obtained by invoking both the bandwidth allocation $C^{ac}_{l_h,r}$ and the flow control $\lambda^{max}_{il_h,r}$ schemes as given by equation (8.11b). The flow rate control acts as a constraint on the source traffic in which a time-varying upper bound is imposed by the controller. In fact, the ordinary source traffic reduces its rate only when a congestion is detected within the network and the constraint on the ordinary traffic flow rate can be seen as its maximum limiting case.

In order to take into account the effects of the source flow control on the state error dynamics (8.18) or (8.31), one needs to substitute the ordinary traffic flow vector $\lambda^{U}_{i,r}(t)$ by the delayed ordinary source output vector $\lambda^{U,ac}_{o,r}(t-\tau)$. Note that $\lambda^{U,ac}_{o,r} = [\lambda^{U_{11},ac}_{o,r}\ (\lambda^{U_1,ac}_{o,r})^T\ \ldots\ \lambda^{U_{n1},ac}_{o,r}\ (\lambda^{U_n,ac}_{o,r})^T]^T$ where $\lambda^{U_m,ac}_{o,r} = [\lambda^{U_{m2},ac}_{o,r}\ \lambda^{U_{m3},ac}_{o,r}\ \ldots\ \lambda^{U_{mn},ac}_{o,r}]^T$ with $m = 1, \ldots, n$.

Following our proposed hop-by-hop congestion control strategy given by the Theorem 8.4, one may observe that whenever the congestion occurs in the network, the source output ordinary traffic $\lambda^{U}_{o,r}(t)$ at node h is limited by the maximum allowed rate $\lambda^{max}_{ih,r}$ that is computed by the controller, which should clearly be less or equal to the maximum capacity $C^{max}_{h,r}$ of the server.

Note that $C^{max}_{h,r}$ is imposed by the allowed ordinary rate calculated by the downstream node controller j, which in turn is bounded by its downstream node controller and that which is also bounded by its corresponding downstream node all the way to the last node at the edge of the network. By applying to the network nodes proper series of constraints, a switching control law can be obtained as a coupling between the premium and the ordinary error dynamics. The resulting coupling depends mainly on the bandwidth control law $C_{h,p}$ of the Premium service given by Theorem 8.4 which in turn depends on the premium queueing state error and from now on does not depend on the ordinary queueing state error. Note that the derivation of this switching coupling law becomes more complex as the dimension of the network increases. This complexity is ultimately due to the fact that the number of paths spanning any h^{th} node at the core of the network with the last node at the edge of the network is large for a large value of n_1. For a fixed $n_1 = 5$, the switching control law is subsequently given by equation (8.41) in the simulation subsection 8.5.3 [21].

8.5.2 Stability Analysis of the Delayed Network Dynamics

The premium and the ordinary dynamics of the overall controlled network are shown to be governed by similar and delay dependent models as described by equations (8.18) and (8.31). By demonstrating the stability properties of system (8.31) it will be sufficient to ensure the same property for both the premium and the ordinary dynamics, and consequently the stability of the entire network dynamics. However, stability of system (8.31) is subject to boundedness of its external inputs $\lambda^{U}_{i,f}(t)$. As far as the premium service is concerned its external inputs are assumed to be bounded, whereas as far as the ordinary service is concerned its source traffic is controlled. For the ordinary service dynamics, it turns out the assumption on the external input can be relaxed and the following lemma can be stated.

Lemma 8.1. *The external input to the ordinary traffic error dynamics described by equations* (8.18) *or* (8.31) *remains bounded if the premium error dynamics of the entire network is guaranteed to be stable.*

Proof: As discussed earlier, the external input to the ordinary error dynamics is the result of the coupling effects between the premium and the ordinary queueing dynamics. However, the boundedness of this external input $\lambda^{U,ac}_{o,r}(t-\tau)$ is guaranteed if the premium error dynamics is stable. The stability of the premium dynamics ensures the boundedness of the premium control $C_{l_h,p}$ which is present in $\lambda_{o,r}^{U,ac}$. In other words, the boundedness of $\lambda_{o,r}^{U,ac}$ depends on:

(a) the commands $C_{h,p}$ with $h = 11, 12, \ldots, 1n, \ldots, n1, n2, \ldots, nn$ remaining bounded whenever the premium error dynamics is stable and the estimator traffic converges to the bounded premium source traffic, and

(b) the server capacity C_{serv} which is finite according to Assumption 3.3.

The above shows that provided the premium error dynamics is stable, the external input to the ordinary error dynamics will also remain bounded. ▲

Note that our network dynamics belongs to a switching system. In fact two switching processes can be identified that occur in the premium and the ordinary error dynamics. The first process is due to signum function which is used in our sliding mode control strategy. The second switching process is due to the nature of the rule that is used in the switching control law. This switching is present only in the ordinary error dynamics and is induced by the ordinary traffic flow control through the ordinary traffic inputs. According to the error dynamics (8.31), only the first switching process influences the network stability as long as the ordinary traffic input remains bounded according to Lemma 8.1.

The conventional procedure for verifying stability of switched time-delay systems consists of showing that all the subsystems are stable under the same constraints by using a common Lyapunov function candidate [70]. We may show that the switching due to the signum function can be treated as an external input stimuli. However, in (8.31) the signum function defined by (8.8) may take three possible values depending on $e_{h,f}(t)$. Hence, by taking into account the delayed signum function, our system switches among $3^2 = 9$ possible sub-dynamics. Since the output of the signum function does not contain $e_{h,f}(t)$, the corresponding terms can be added to the external input term, and therefore the resulting error dynamics can be expressed as follows

$$\dot{\eta}_f = \mathbb{A}_{1,f}\eta_f + \mathbb{A}_{2,f}\eta_f(t-\tau) + \bar{\lambda}^U_{i,f} \tag{8.33}$$

where $\bar{\lambda}^U_{i,f} = \mathbb{B}_{2,f}\lambda^U_{i,f}(t) + H_f$ with

$$H_f = \begin{cases} \mathbb{B}_{1,f}\dot{x}_{f,ref}(t-\tau(t)) & \pm \quad \mathbb{A}_{3,f} \quad \pm \quad \mathbb{A}_{4,f} \\ \mathbb{B}_{1,f}\dot{x}_{f,ref}(t-\tau(t)) & \pm \quad \mathbb{A}_{3,f} \\ \mathbb{B}_{1,f}\dot{x}_{f,ref}(t-\tau(t)) & \pm \quad \mathbb{A}_{4,f} \\ \mathbb{B}_{1,f}\dot{x}_{f,ref}(t-\tau(t)) & \end{cases} \tag{8.34}$$

The following observations related to the network error dynamics (8.33) can be stated:

(a) The information matrix $\mathbb{A}_{1,f}$ corresponding to the non-delayed term is Hurwitz whenever the design parameters $\mu's$ and $\gamma's$ are selected as positive values, and

(b) the information matrix $\mathbb{A}_{2,f}$ corresponding to the delayed term depends on the design parameter $\mu's$ and the network configuration. Note that in the absence of delay this matrix will be added to $\mathbb{A}_{1,f}$, and therefore for both delayed and non-delayed systems stability will be subject to not only the positiveness of the design parameters but also to the network configuration (the location of the matrices terms change according the network configurations), and

(c) according to Lemma 3.1 in [102], in the presence of the time-varying delay the system (8.33) is globally uniformly asymptotically stable for all $\bar{\lambda}^U_{i,f} \in \mathcal{L}_2[0,\infty)$.

It is worth noting that the network traffic $\bar{\lambda}^U_{i,f}$ is practically non-vanishing, therefore to conclude the stability analysis for the time-delayed system (8.33), let us assume that the time derivative of the delay function $\tau(t)$ satisfies

$$\dot{\tau}(t) \leq \varphi < 1 \quad \forall t \in [0,\infty) \tag{8.35}$$

Furthermore, assume that the input signal $\bar{\lambda}^U_{i,f}(t)$ is an arbitrary signal in $\mathcal{L}_\infty[0,\infty)$ with zero initial conditions and $\bar{\lambda}^U_{i,f}(t) \equiv 0, \ \ \forall t < 0$. Let us consider the performance index $\mathcal{J}$ that is given by

$$\mathcal{J} = \|\eta_f\|_2^2 - \nu\|\bar{\lambda}^U_{i,f}\|_2^2 \tag{8.36}$$

where ν is a positive scalar. The objective is to find conditions that will ensure $\mathcal{J} \leq 0, \forall \bar{\lambda}^U_{i,f} \in \mathcal{L}_\infty[0,\infty)$, and hence ultimate boundedness of system (8.33). These results are formalized in the following theorem.

Theorem 8.5. *Let φ represent the upper bound of the rate of change the time-delay $\tau(t)$ that satisfies* (8.35)*. The network error dynamics governed by equation* (8.33) *is $\mathcal{L}_\infty$ stable and the performance index $\mathcal{J}$ of* (8.36) *is non-positive for all $\bar{\lambda}^U_{i,f} \in \mathcal{L}_\infty[0,\infty)$ provided that there exist symmetric positive definite matrices P and Q that satisfy the following linear matrix inequality*

$$\mathbb{L}_1 \triangleq \begin{pmatrix} R & P\mathbb{A}_{2,f} & P\mathbb{B}_{2,f} \\ \mathbb{A}_{2,f}^T P & (\varphi - 1)Q & 0 \\ \mathbb{B}_{2,f}^T P & 0 & -\nu^2 I \end{pmatrix} \leq 0 \tag{8.37}$$

where $R \triangleq \mathbb{A}_{1,f}^T P + P\mathbb{A}_{1,f} + Q + I$.

Proof: Given P and Q as symmetric positive definite matrices, let us choose the Lyapunov-Krasovskii functional candidate [102] as

$$V(t) = \eta_f^T(t)P\eta_f(t) + \int_{t-d(t)}^{t} \eta_f^T(\theta)Q\eta(\theta)d\theta \tag{8.38}$$

The corresponding Hamiltonian can now be expressed as follows:

$$\begin{aligned} \mathcal{J}_1 &= \dot{V} + \eta_f^T(t)\eta_f(t) - \nu^2(\bar{\lambda}^U_{i,f})^T\bar{\lambda}^U_{i,f} \\ &= \eta_f^T(t)(\mathbb{A}_{1,f}^T P + P\mathbb{A}_{1,f} + Q + I)\eta_f(t) \\ &+ 2\eta_f^T(t)P\mathbb{A}_{2,f}\eta_f(t-\tau) - (1-\dot{\tau})\eta_f^T(t-\tau)Q\eta_f(t-\tau) \\ &+ 2\eta_f^T PB_1\bar{\lambda}^U_{i,f} - \nu^2(\bar{\lambda}^U_{i,f})^T\bar{\lambda}^U_{i,f} \\ &= \Xi^T\,\mathbb{L}_1\,\Xi \leq 0 \end{aligned} \tag{8.39}$$

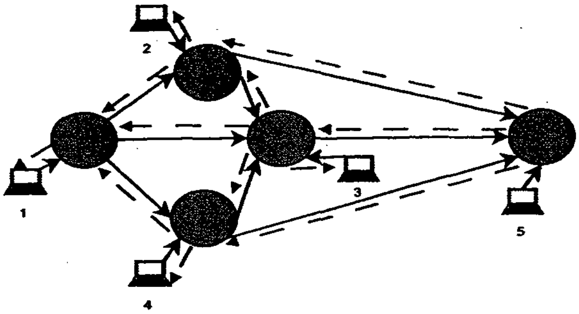


Figure 8.5: A DiffServ network adopting our proposed distributed hop-by-hop congestion control approach (solid line: traffic flow, dashed and dotted lines: feedback signaling, U_h and N_h, $h = 1, \ldots, 5$ denote the user/source and node, respectively).

where $\Xi = [\eta_f^T(t)\ \eta_f^T(t-\tau)\ (\bar{\lambda}_{i,f}^U)^T]^T$ and $\mathbb{L}_1$ is given by (8.37). The above shows that $\mathcal{J}_1$ is non-positive, and hence $\mathcal{J}$ is also non-positive according to the results in [117]. This completes the proof of the theorem. ▲

8.5.3 Simulation results for a five node loopless network using the proposed distributed hop-by-hop congestion controller

In order to evaluate in simulation our proposed distributed hop-by-hop congestion controller, we simplify our network that is given in Figure 8.4 so that n is set to 3, implying a 3x3 network. Assuming further that sources corresponding to the nodes located at the network corners do not generate any traffic, we make these nodes inactive so that simplifies our network into a 5 nodes network as illustrated in Figure 8.5. It is worth noting that the DiffServ network of Figure 8.5 is the same as that given by Figure 8.2. The latter network that is used for evaluating the semi-decentralized end-to-end congestion controller is thus chosen on purpose to fit with the network of Figure 8.4.

The error dynamics of the resulting five node network can be expressed according to the same state representations as given by (8.18) and (8.31) where $A_{1,f} = diag[\mu_{1,f}\ \ \mu_{2,f}\ \mu_{3,f}\ \mu_{4,f}\ \mu_{5,f}]$, $A_{3,f} = diag[\gamma_{1,f}\ \gamma_{2,f}\ \gamma_{3,f}\ \gamma_{4,f}\ \gamma_{5,f}]$,

$$B_{1,f} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}, \tag{8.40}$$

$A_{2,f} = B_{1,f}A_{1,f}$ and $A_{4,f} = B_{1,f}A_{3,f}$. As far as the associated switching control law $\lambda^{U}_{o,r}(t - \tau(t))$ is concerned, it is given by (8.41) as follows

$$\begin{cases} \lambda^{U_1,ac}_{o,r}=min\{\lambda^{U_1}_{o,r},C_{serv}-C_{1p}(t-\tau),\frac{C_{serv}-C_{5p}(t-4\tau)}{32},\frac{C_{serv}-C_{3p}(t-3\tau)}{8},\frac{C_{serv}-C_{2p}(t-2\tau)}{2}, \\ \qquad\qquad \frac{C_{serv}-C_{3p}(t-2\tau)}{4},\frac{C_{serv}-C_{4p}(t-2\tau)}{2},\frac{C_{serv}-C_{5p}(t-3\tau)}{16}\} \\ \lambda^{U_2,ac}_{o,r}=min\{\lambda^{U_2}_{o,r},C_{serv}-C_{2p}(t-\tau),\frac{C_{serv}-C_{5p}(t-3\tau)}{16},\frac{C_{serv}-C_{3p}(t-2\tau)}{4},\frac{C_{serv}-C_{5p}(t-2\tau)}{4}\} \\ \lambda^{U_3,ac}_{o,r}=min\{\lambda^{U_3}_{o,r},C_{serv}-C_{3p}(t-\tau),\frac{C_{serv}-C_{5p}(t-2\tau)}{4}\} \\ \lambda^{U_{4,5},ac}_{o,r}=min\{\lambda^{U_4}_{o,r},C_{serv}-C_{4p}(t-\tau),\frac{C_{serv}-C_{5p}(t-3\tau)}{16},\frac{C_{serv}-C_{3p}(t-2\tau)}{4},\frac{C_{serv}-C_{5p}(t-3\tau)}{4}\} \end{cases} \tag{8.41}$$

Our proposed distributed hop-by-hop congestion controller is now evaluated in presence of an unknown time-delay varying sinusoidally with maximum amplitudes of 1 ms and 70 ms according to Assumption 3.2. Note that the values of delays are twice these values for the ordinary flow control channel because of the round trip time of the feedback signaling (see Figure 2.1). Furthermore, in view of the distributed control principle that is investigated, it is interesting to verify the "plug and play" notion here as our controller is designed for one node and then implemented for the other network nodes without changing the design parameters.

In the simulations presented below (a) the sampling time is set to $T_s = 1$ ms, (b) the control parameters are designed to be $\mu_{h,p} = 10^3$, $\mu_{h,r} = 500$, $\gamma_{h,p} = 0.1$, $\gamma_{h,r} = 5 * 10^{-2}$, and (c) the estimation parameters are set to $\Gamma_{h,p} = 10^4$, $\Gamma_{h,r} = 10^5$, $\sigma_{h,p} = 0$ and $\sigma_{h,r} = 10$ with indices $f = p, r$ and $h = 1, \ldots, 5$. The premium and the ordinary traffics are generated by the sources/users that are dynamic. Each source/user generates a premium random traffic with a mean varying sinusoidally from 0 to 2000 packets/s, a variance of 100 packets/s and a frequency of 0.5 rad/s. The premium traffic is assumed to be bounded such that $0 \leq \lambda_{il_hp} \leq 4000$ packets/s. The sources also generate an ordinary random traffic with a mean that varies sinusoidally from 0 to 200 packet/s, a variance of 50 packets/s and a frequency of 0.5 rad/s.

The premium and ordinary buffer capacities are finite and their queues are bounded according to the following constraints: $0 \leq x_{l_hp} \leq 128$ packets and $0 \leq x_{l_hr} \leq 1024$ packets, respectively. According to Assumption 3.3 the constraints on the maximum available service capacity are $0 \leq C_{l_h} \leq 40000$ packets/s for both services.

The reference buffer queues are set as follows: For node 1, the premium and ordinary references are set to $[100, 140]$ packets/s respectively, for nodes 2 and 3 the premium and ordinary queues references are $[70, 60]$ packets/s, respectively, and for the last two nodes 4 and 5, their premium and ordinary queues references are set to $[30, 90]$ packets/s. The results are presented in two figures.

Figure 8.6 illustrates the buffer queue step responses of nodes 1-5 obtained by using our proposed distributed hop-by-hop congestion controller. In Figure 8.6, the left and the right columns are devoted to the Premium and the Ordinary services, respectively, whereas each row is dedicated to one node of the network.

By inspecting the plots presented in Figure 8.6, one may readily argue that for both services our proposed distributed hop-by-hop controller stabilizes the network despite the presence of time-delay of 70 ms amplitude (dashed line) and a dynamic traffic source. Indeed, it can be seen that not only all the buffer queue lengths converge to their respective references, but also the transient responses are also fast especially for the premium service. Note that the percentage overshoots can

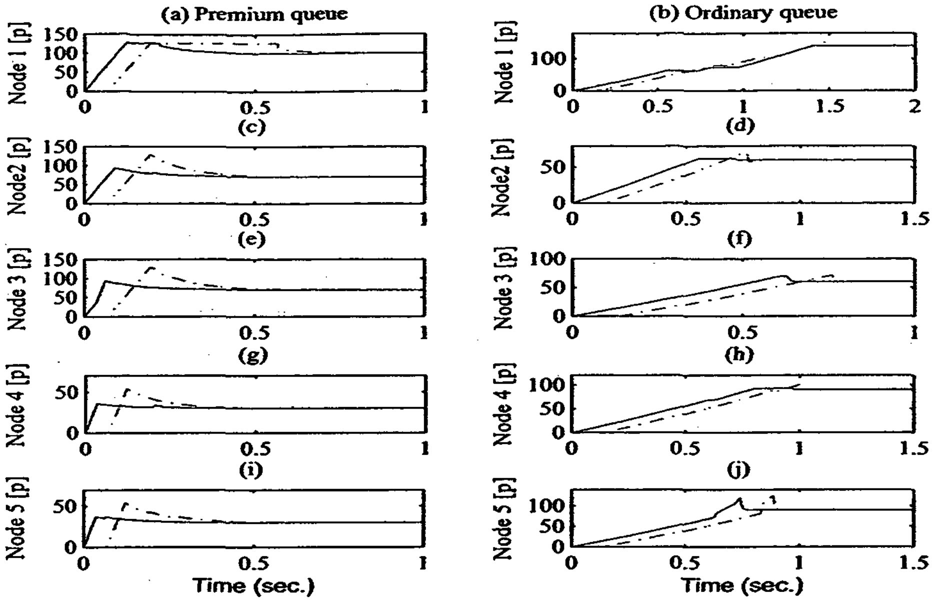


Figure 8.6: Simulation results obtained by using our proposed robust congestion control strategy (solid line: 1 ms delay, dashed line:70 ms delay).

be reduced further by adjusting the design parameters. This can be achieved by decreasing the control gains $\mu_{h,f}$ and the sliding surface slope [19], or the estimation gain $\Gamma_{h,f}$. To avoid the counter effects of stretching the settling time, a trade off between the speed and the overshoot of the step response has to be carefully chosen. In addition and according to [54], the σ-modification scheme can be improved by using the well-known normalized gains, therefore, we believe that this improvement would reduce the overshoots due to the estimation.

Figure 8.7 illustrates the behavior of the buffer characteristics of node 5 (other than the buffer queue). The left and the right columns of Figure 8.7 are devoted to the Premium and the Ordinary services, respectively. The first row plots (a) and (b) depict the server capacities, whereas the second row plots (c) and (d) show the actual buffer input and the output traffics as well as the estimate of the input traffic. Finally, the third row plots (e) and (f) correspond to the zoomed in transients of plots (c) and (d), respectively.

From the last two rows of Figure 8.7, one can readily observe that the buffer operates properly since the actual output traffic converges to the actual input traffic according to the conservation principle of the FFM while the estimate of the input traffic converges to the actual dynamic input traffic.

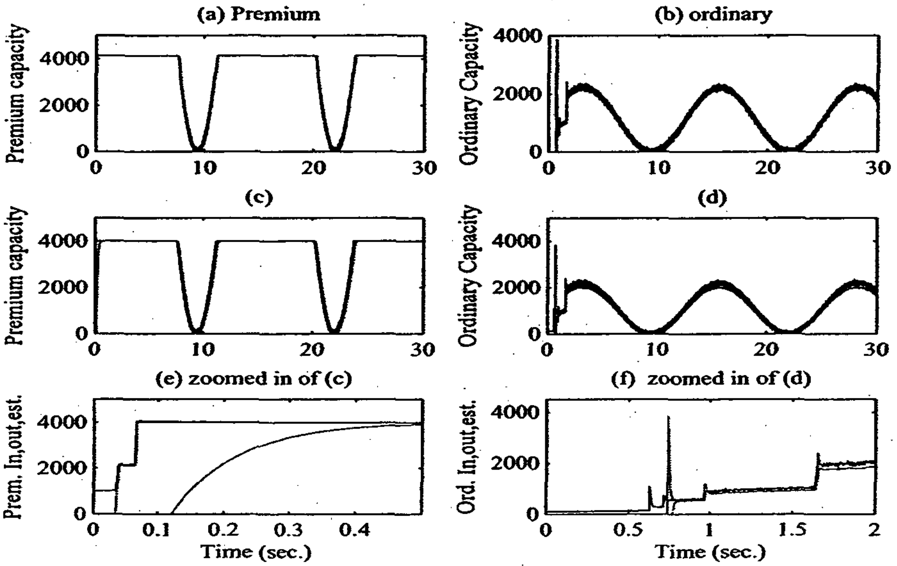


Figure 8.7: Buffer characteristics of node 5.

8.6 Proposed Distributed Hop-By-Hop Congestion Controller for a DiffServ NUS [22]

After demonstrating in the above section the good performance and the capabilities of our proposed distributed hop-by-hop congestion controller on a loopless network (Internet), let us now evaluate it on a NUS system that is illustrated by Figure 3.4. As mentioned earlier, our proposed controller that is designed for a single node h is now implemented on each node of the cluster.

In this section, we first derive the time-delay dependent network error dynamics, then analyze the stability of the resulting dynamics and finally demonstrate the effectiveness of the simulations conducted when compared with the IDCC controller [89].

8.6.1 Error dynamics of the overall network

The overall congestion control strategy for our differentiated services cluster will be based on only the Premium and the Ordinary services since the opportunistic Best Effort service does not intervene with the procedure and constraints of the control design. Below, we first derive the overall cluster dynamics by augmenting all the nodes dynamics through their input and output channels and by taking into account and incorporating in the formal model the delays $\tau_{lk,f}^{hj}$ and the process functions $F_{lk,f}^{hj}$. It is worth noting that at each node output port, the premium and the ordinary dynamics are similar in open-loop as they are modeled according to the FFM (8.1). It follows that this also holds in the closed-loop since these dynamics follow the same bandwidth allocation control law that is given by Theorem 8.4.

Consequently, by using the bandwidth allocation scheme proposed in Theorem 8.4, the network cluster dynamics corresponding to the premium and the ordinary services, as shown in Figure 3.4, are governed by

$$\begin{cases} \dot{e}_f^s = & -\mu_f^s e_f^s - \gamma_f^s sgn(e_f^s) - \hat{\lambda}_{i,f}^s + S_{k_s}^s \lambda_{i,f}^s \\ \dot{e}_f^d = & -\mu_f^d e_f^d - \gamma_f^d sgn(e_f^d) - \hat{\lambda}_{i,f}^d + S_{l_d k_d}^d \lambda_{i,f}^d \\ \dot{e}_f^a = & -\mu_f^a e_f^a - \gamma_f^a sgn(e_f^a) - \hat{\lambda}_{i,f}^a + S_{l_a k_a}^a \lambda_{i,f}^a \end{cases} \tag{8.42}$$

where $e_f^s \in \Re$, $e_f^d = [e_{1,f}^d \ e_{2,f}^d]^T \in \Re^2$, and $e_f^a = [e_{1,f}^a \ e_{2,f}^a \ e_{3,f}^a]^T \in \Re^3$ are the queue length state errors corresponding to the sensor, the decision-maker, and the actuator, respectively, and $sgn(e_f^d) \triangleq [sgn(e_{1f}^d) \ sgn(e_{2f}^d)]^T$ and $sgn(e_f^a) \triangleq [sgn(e_{1f}^a)$

$sgn(e^a_{2f})\ sgn(e^a_{3f})]^T$ are signum functions. The state $\hat{\lambda}^j_{if}$ with $j \in N_1$ and $N_1 = \{s, d, a\}$ is the buffer input traffic estimate which is governed by the update law (8.4b). For the decision-maker and the actuator, the traffic estimates are defined as follows: $\hat{\lambda}^d_{i,f} = [\hat{\lambda}^d_{i1,f}\ \hat{\lambda}^d_{i2,f}]^T$ and $\hat{\lambda}^a_{i,f} = [\hat{\lambda}^a_{i1,f}\ \hat{\lambda}^a_{i2,f}\ \hat{\lambda}^a_{i3,f}]^T$. The terms $\lambda^s_{i,f} = [\lambda^s_{i1,f}\ \lambda^s_{i2,f}\ \lambda^s_{i3,f}]^T$, $\lambda^d_{i,f} = [\lambda^d_{i1,f}\ \lambda^d_{i2,f}\ \lambda^d_{i3,f}]^T$ and $\lambda^a_{i,f} = [\lambda^a_{i1,f}\ \lambda^a_{i2,f}]^T$ are the actual node input traffic flows, $S^s_{k_s} = [S^s_1\ \ S^s_2\ \ S^s_3]$, $S^d_{l_d k_d} = [(S^d_1)^T\ \ (S^d_2)^T]^T$ with $S^d_j = [S^d_{j1}\ \ S^d_{j2}\ \ S^d_{j3}]$ $(j = 1, 2)$ and $S^a_{l_a k_a} = [(S^a_1)^T\ \ (S^a_2)^T\ \ (S^a_3)^T]^T$ with $S^a_j = [S^a_{j1}\ \ S^a_{j2}]$ $(j = 1, \ldots, 3)$ are the switching matrices corresponding to the sensor, the decision-maker, and the actuator, respectively and μ^s_f, μ^d_f and μ^a_f are positive design parameters.

According to the cluster network schematic depicted in Figure 3.4, any k^{th} input traffic flow at node j corresponds to the delayed output traffic of its immediate sender node h. As mentioned earlier, the sender can be a source (user) or a node. Regarding the latter case, its output traffic flow is expressed by equation (8.43) according to the conservation principle of the FFM, namely

$$\lambda^j_{ik,f}(t) = F^{hj}_{lk,f}\lambda^h_{ol,f}(t - \tau^{hj}_{lk,f}) = F^{hj}_{lk,f}C_{l_h,f}(t - \tau^{hj}_{lk,f})E_{l_h,f}(t - \tau^{hj}_{lk,f}) \qquad (8.43)$$

After augmenting all the nodes dynamics and by substituting their inputs according to (8.43) and invoking Assumption 3.2, the 6^{th}-order error dynamics for the entire network cluster is represented according to

$$\dot{e}_f = -A_{1,f}e_f + A_{2,f}e_f(t-\tau) - A_{3,f}sgn(e_f) + A_{4,f}sgn(e_f(t-\tau)) + W_f + B_2\lambda^U_{i,f}(t) \quad (8.44)$$

where $e_f = [e^s_f\ e^d_{1,f}\ e^d_{2,f}\ e^a_{1,f}\ e^a_{2,f}\ e^a_{3,f}]^T$ is the state error vector $(f = p, r)$, $sgn(e_f) \triangleq [sgn(e^s_f)\ sgn(e^d_{1,f})\ sgn(e^d_{2,f})\ sgn(e^a_{1,f})\ sgn(e^a_{2,f})\ sgn(e^a_{3,f})]^T$, $W_f = -\hat{\lambda}_{i,f} + B_{1,f}[\hat{\lambda}_{i,f}(t-\tau) - \dot{x}_{f,ref}(t-\tau)]$, $\hat{\lambda}_{i,f} = [\hat{\lambda}^s_{i,f}\ \hat{\lambda}^d_{i1,f}\ \hat{\lambda}^d_{i2,f}\ \hat{\lambda}^a_{i1,f}\ \hat{\lambda}^a_{i2,f}\ \hat{\lambda}^a_{i3,f}]^T$, $\lambda^U_{i,f} \doteq [\lambda^s_{i2,f}\ \lambda^d_{i3,f}\ \lambda^a_{i2,f}]^T$, matrices $A_{1,f}$ and $A_{3,f}$ are defined as $A_{1,f} = diag[\mu^s_f\ \mu^d_{1,f}\ \mu^d_{2,f}\ \mu^a_{1,f}\ \mu^a_{2,f}\ \mu^a_{3,f}]$, $A_{3,f} =$

$diag[\gamma_f^s\ \gamma_{1,f}^d\ \gamma_{2,f}^d\ \gamma_{1,f}^a\ \ \gamma_{2,f}^a\ \gamma_{3,f}^a]$, with $\mu's$ and $\gamma's$ are positive design parameters,

$$B_{1,f}=\begin{pmatrix} 0 & 0 & S_3^sF_{23}^{ds} & 0 & S_1^sF_{21}^{as} & 0 \\ S_{12}^dF_{12}^{sd} & 0 & 0 & S_{11}^dF_{11}^{ad} & 0 & 0 \\ S_{22}^dF_{12}^{sd} & 0 & 0 & S_{21}^dF_{11}^{ad} & 0 & 0 \\ 0 & S_{11}^aF_{11}^{ad} & 0 & 0 & 0 & 0 \\ 0 & S_{21}^aF_{11}^{ad} & 0 & 0 & 0 & 0 \\ 0 & S_{31}^aF_{11}^{ad} & 0 & 0 & 0 & 0 \end{pmatrix}, B_{2,f}=\begin{pmatrix} S_2^s & 0 & 0 \\ 0 & S_{13}^d & 0 \\ 0 & S_{23}^d & 0 \\ 0 & 0 & S_{12}^a \\ 0 & 0 & S_{22}^a \\ 0 & 0 & S_{32}^a \end{pmatrix}, \tag{8.45}$$

$A_{2,f} = B_{1,f}A_{1,f}$ and $A_{4,f} = B_{1,f}A_{3,f}$.

The traffic flow in our cluster is estimated according to the update law (8.4b), however the error dynamics given by (8.44) may be extended to incorporate the estimator dynamics as shown below

$$\dot{\eta}_f = \mathbb{A}_{1,f}\eta_f+\mathbb{A}_{2,f}\eta_f(t-\tau)+\mathbb{A}_{3,f}sgn(e_f)+\mathbb{A}_{4,f}sgn(e_f(t-\tau))+\mathbb{B}_{1,f}\dot{x}_{f,ref}(t-\tau)+\mathbb{B}_{2,f}\lambda_{i,f}^U(t) \tag{8.46}$$

where $\eta_f = [(e_f)^T\ (\hat{\lambda}_{i,f})^T]^T$,

$$\mathbb{A}_{1,f}=\begin{pmatrix} -A_{1,f} & -I \\ \Gamma_f & -\sigma_f \end{pmatrix}; \mathbb{A}_{2,f}=\begin{pmatrix} A_{2,f} & B_{1,f} \\ 0 & 0 \end{pmatrix}=\begin{pmatrix} B_{1,f} \\ 0 \end{pmatrix}\begin{pmatrix} A_{1,f} & I \end{pmatrix} \tag{8.47}$$

with $\Gamma_f = diag[\Gamma_f^s\ \Gamma_{1,f}^d\ \Gamma_{2,f}^d\ \Gamma_{1,f}^a\ \Gamma_{2,f}^a\ \Gamma_{3,f}^a]$ and $\sigma_f = diag[\sigma_f^s\ \sigma_{1,f}^d\ \sigma_{2,f}^d\ \sigma_{1,f}^a\ \sigma_{2,f}^a\ \sigma_{3,f}^a]$, $\mathbb{A}_{3,f} = [-(A_{3,f})^T\ 0]^T$, $\mathbb{A}_{4,f} = [(A_{4,f})^T\ 0]^T$, $\mathbb{B}_{1,f} = [-(B_{1,f})^T\ 0]^T$, and $\mathbb{B}_{2,f} = [(B_{2,f})^T\ 0]^T$.

It is worth noting that at this stage, the node inputs corresponding to the delayed source (user) outputs have not yet been substituted. In effect, one may readily observe that the resulting error dynamics (8.44) or (8.46) still contains the node input vector $\lambda_{i,f}^U(t)$.

By substituting $\lambda_{i,f}^U(t)$ with the corresponding delayed source outputs $\lambda_{o,f}^U(t-\tau) = [\lambda_{o1,f}^{u^s}(t-\tau)\ \lambda_{o1,f}^{u^d}(t-\tau)\ \lambda_{o1,f}^{u^a}(t-\tau)]^T$ according to (8.43), the network cluster error dynamics (8.44) or (8.46) will be associated with different inputs depending on the service. In other words, the input vector $\lambda^U_{i,f}(t)$ is substituted with $\lambda^U_{o,p}(t-\tau)$ for the Premium service and with $\lambda^{U,ac}_{o,r}(t-\tau)$ for the Ordinary service. Note that, due to the ordinary flow control, the actual value of the output ordinary traffic is bounded by the delayed maximum allowed rate $\lambda^{max}_{l_h r}(t-\tau)$. Therefore, the index ac is designating the *actual* value of the ordinary source. This does lead to a weak coupling between the premium and the ordinary error dynamics as illustrated below.

8.6.1.1 Premium error dynamics for the network cluster

The premium state error dynamics is obtained by invoking only the bandwidth allocation $C^{ac}_{l_h r}$, implying that the premium sources are not controlled. This means also that in the queueing error dynamics (8.44) or (8.46) corresponding to the Premium service we have $\lambda^U_{i,p}(t) = \lambda^U_{o,p}(t-\tau)$ where the premium source output vector is given by $\lambda^U_{o,p} = [\lambda^{u^s}_{o1,p}\ \lambda^{u^d}_{o1,p}\ \lambda^{u^a}_{o1,p}]^T$.

8.6.1.2 Ordinary error dynamics for the network cluster

The state error dynamics is obtained by applying both the bandwidth allocation $C^{ac}_{l_h r}$ and the flow control $\lambda^{max}_{il_h r}$ schemes as given by equations (8.11a) and (8.11b). As shown in Figure 8.1(a) the flow control law $\lambda^{max}_{il_h r}$ imposes a time-varying upper bound to the ordinary source in order to reduce its traffic rate. In other words, the actual output traffic rate generated by the ordinary source is therefore $\lambda^{U,ac}_{o,r}(t)$. In order to take into account the effects of the source flow control on the state error dynamics (8.44) or (8.46), one needs to substitute the ordinary traffic flow vector $\lambda^U_{i,r}(t)$ by the delayed ordinary source output vector $\lambda^{U,ac}_{o,r}(t-\tau)$, where $\lambda^{U,ac}_{o,r}(t) = [\lambda^{u^s,ac}_{o1,r}\ \lambda^{u^d,ac}_{o1,r}\ \lambda^{u^a,ac}_{o1,r}]^T$.

According to our proposed distributed hop-by-hop congestion control strategy that is given by Theorem 8.4, one may observe that whenever the congestion occurs in the cluster, the ordinary source output traffic $\lambda^{U,ac}_{o,r}(t)$ at node h is limited by the delayed value of the maximum allowed rate $\lambda_{il_h r}^{max,ac}$ that is computed by the controller, which should clearly be less or equal to the maximum capacity $C_{l_h r}^{max}$ of the server. Note that $C_{l_h r}^{max}$ is imposed by the allowed ordinary traffic flow rate that is calculated by the downstream node controller j, which in turn is bounded by its downstream node controller and that which is also bounded by its corresponding downstream node all the way to the last node at the edge of the network. By applying to the network cluster nodes proper series of constraints the resulting switching control law (8.48) is obtained that is given by

$$\begin{cases} \lambda_{olr}^{u^s,ac}=min\{\lambda_{olr}^{u^s},C_{serv}-C_{sp}(t-\tau),\frac{C_{serv}-C_{1_a p}(t-3\tau)}{4},\frac{C_{serv}-C_{2_a p}(t-3\tau)}{4},\frac{C_{serv}-C_{3_a p}(t-3\tau)}{4}, \\ \qquad \frac{C_{serv}-C_{1_d p}(t-2\tau)}{2},\frac{C_{serv}-C_{2_d p}(t-2\tau)}{2}\} \\ \lambda_{olr}^{u^d,ac}=min\{\lambda_{olr}^{u^d},\frac{C_{serv}-C_{1_a p}(t-2\tau)}{4},\frac{C_{serv}-C_{2_a p}(t-2\tau)}{4},\frac{C_{serv}-C_{3_a p}(t-2\tau)}{4},\frac{C_{serv}-C_{1_d p}(t-\tau)}{2},\frac{C_{serv}-C_{2_d p}(t-\tau)}{2}\} \\ \lambda_{olr}^{u^a,ac}=min\{\lambda_{olr}^{u^a},\frac{C_{serv}-C_{1_a p}(t-\tau)}{2},\frac{C_{serv}-C_{2_a p}(t-\tau)}{2},\frac{C_{serv}-C_{3_a p}(t-\tau)}{2}\} \end{cases} \tag{8.48}$$

As expressed in (8.48), depending on the congestion state of the network, the actual value of each ordinary source rate $\lambda_{olr}^{u^j,ac}(t)$ $(j = s, d, a)$ is computed as the minimum of a number of input values corresponding to the maximum traffics allowed by the downstream node controllers. Consequently, at each control interval the ordinary source rate $\lambda_{olr}^{u^j,ac}(t)$ may switch from one value to another.

One can now readily observe that the resulting law (8.48) depends mainly on the premium queueing state error and does not depend on the ordinary queueing error. This implies that the control law (8.48) shows the existence of a weak coupling between the premium and the ordinary error dynamics.

Remark 8.3. *Note that the error dynamics corresponding to the NUS and the mesh networks are different eventhough their models are similar as described by equations* (8.44) *and* (8.18) *or* (8.46) *and* (8.31)*, respectively*

Remark 8.4. *Note that the opportunistic Best Effort service does not intervene with the procedure and constraint of the above control design strategy eventhough to* *practically avoid overall network problems a certain capacity will need to be allocated to it. Therefore, in the above analysis only the error dynamics corresponding to the Premium and Ordinary services are obtained.*

8.6.2 Stability analysis of the delayed network cluster dynamics

Similarly to the stability analysis of the delayed mesh network, one may assert that a) the two dynamics constituting the overall network cluster dynamics are shown to be governed by similar and delay dependent models as described by equations (8.44) or (8.46), b) by demonstrating the stability properties of system (8.46) it will be sufficient to ensure the same property for both the premium and the ordinary dynamics, and consequently the stability of the entire network cluster dynamics, c) and the stability of (8.46) is subject to boundedness of its external inputs.

Since the external inputs to the premium error dynamics are all assumed to be bounded, the boundedness of the external inputs to the ordinary error dynamics and the stability of the overall network cluster error dynamics (8.46) can be shown by using Lemma 8.1 and Theorem 8.5, respectively. According to Lemma 8.1, the boundedness of the external inputs $\lambda_{o,r}^{U,ac}(t-\tau)$ to the ordinary dynamics is guaranteed if the premium error dynamics is stable. In fact, the boundedness of equation (8.48) is guaranteed since:

(a) the commands $C_{l_h,p}$ with $h = s, d, a$ that are present in $\lambda_{o,r}^{U,ac}$, remain bounded whenever the premium error dynamics is stable and the estimator traffic converges to the bounded premium source traffic, and

(b) the server capacity C_{serv} is finite according to Assumption 3.3.

As far as the overall error dynamics given by equation (8.46) is concerned, following the same reasoning as for the stability of the mesh network error dynamics, we may show that a) the error dynamics (8.46) includes two switching mechanisms,

b) one process is due to the nature of the rule that is used in (8.48) and the other process is due to the signum function which is used in our sliding mode control strategy, and c) the latter switching process can be treated as an external input stimuli that can be added to the external input term $\lambda^U_{i,f}(t)$ in equation (8.46) and therefore the resulting error dynamics can be expressed as

$$\dot{\eta}_f = \mathbb{A}_{1,f}\eta_f + \mathbb{A}_{2,f}\eta_f(t-\tau) + \bar{\lambda}^U_{i,f} \tag{8.49}$$

Note that the resulting error dynamics (8.49) is similar to the error dynamics (8.33) corresponding to the mesh network. Therefore, the following observations related to the network cluster error dynamics (8.49) can be stated:

(a) The information matrix $\mathbb{A}_{1,f}$ corresponding to the non-delayed term is Hurwitz whenever the design parameters $\mu' s$ and $\gamma' s$ are selected as positive values,

(b) the information matrix $\mathbb{A}_{2,f}$ in (8.47) corresponding to the delayed term depends on the design parameter μ_f, the cluster configuration and the process function through $S^{hj}_{lk,f}$ and $F^{hj}_{lk,f}$, respectively. For our considered three node cluster configuration, the resulting matrix is unstable. Note that in the absence of delay this matrix will be added to $\mathbb{A}_{1,f}$, and therefore for both delayed and non-delayed systems stability will be subject to not only the positiveness of the design parameters but also to the network configuration and the process function $F^{hj}_{lk,f}$,

(c) the matrices $\mathbb{A}_{1,f} + \mathbb{A}_{2,f}$ and $\mathbb{A}_{1,f} - \mathbb{A}_{2,f}$ constructed from the system matrices are both Hurwitz (their eigenvalues belong to the left-half-plane (LHP) as shown in Tables 8.2 and 8.3, respectively). Therefore, according to [86], [87], our time-delay system can be asymptotically stable independent of the magnitude of the delay,

(d) according to Lemma 3.1 in [102], in the presence of the time-varying delay the system (8.49) is globally uniformly asymptotically stable for all $\bar{\lambda}^U_{i,f} \in \mathcal{L}_2[0,\infty)$, and

(e) according to Theorem 8.5, the network cluster error dynamics governed by

-1.5899	-1.3607	-0.0146 + 0.0089i	-0.0146 - 0.0089i	-0.0101	-0.0101
-0.0101	-0.9899	-0.9899	-0.9899	-0.0101	-0.0101

Table 8.2: Eigenvalues of $\mathbb{A}_1 + \mathbb{A}_2$ times 10^3.

-1.9608	-0.6190	-0.3897	-0.0103	-0.0102	-0.0101
-0.0101	-0.9899	-0.9899	-0.9899	-0.0101	-0.0101

Table 8.3: Eigenvalues of $\mathbb{A}_1 - \mathbb{A}_2$ times 10^3.

(8.49) is $\mathcal{L}_\infty$ stable for all input traffic satisfying $\bar{\lambda}^U_{i,f} \in \mathcal{L}_\infty[0, \infty)$

Remark 8.5. *The network configuration as well as the process function $F^{hj}_{lk,f}$ have major influences on the system stability. For example, for a non-delayed system where the system matrix in* (8.49) *is $\mathbb{A}_{1,f} + \mathbb{A}_{2,f}$, we can make the following three observations: (1) by configuring the network cluster to have no positive feedback and by setting the process function to $F^{hj}_{lk,f} = 1$, we may easily verify that all the eigenvalues of matrices $\mathbb{A}_{j,f}$, $j = 1, 2$ are negative, (2) by reconfiguring the network cluster as a fully-connected network and by setting the process function to $F^{hj}_{lk,f} = 1$, both matrices $\mathbb{A}_{j,f}$, $j = 1, 2$ become unstable, and (3) the stability of the fully-connected network cluster can be achieved whenever the following condition is satisfied, namely $F^{hj}_{lk,f} \leq 0.6$. Simulation results given in the next section do indeed confirm that the above observations also hold in the presence of delays.*

Remark 8.6. *Using Matlab software, we have numerically verified the existence of a solution to the stability conditions obtained above. Specifically, by setting $\varphi = 0$, there exists two 12x12 positive definite and symmetric matrices P and Q and $\nu = 11.1$ for which the LMI condition* (8.37) *is negative definite whenever the process function $F^{hj}_{lk,f}$ is selected to be $F^{hj}_{lk,f} \leq 0.6$.*

8.6.3 Simulation results for a NUS network using the proposed distributed hop-by-hop congestion controller

The simulation results presented in this section are intended to demonstrate: a) the complexity of solving the congestion control problem for a DiffServ network that is characterized by feedback path traffics as present in the NUS system, and b) the effectiveness and capabilities of our proposed distributed congestion control strategy as a viable solution for such networks. Two sets of simulations are conducted for the network cluster where our proposed congestion controller is compared and evaluated with a benchmark scheme, namely the Integrated Dynamic Congestion Control (IDCC) method that is proposed in [89] (see Chapter 2).

In the first set of simulations, our network cluster is configured as a conventional (cascaded) DiffServ network (without any feedback traffic) by disabling all the loops. The simulation results show that: a) the benchmark scheme successfully addresses the congestion control problem when it is implemented as in [89] and by using the control strategy that is shown in Figure 8.1(b), in which the coupling between the ordinary traffic inputs is not considered and also the sources/users are assumed to generate a random traffic with a time-invariant mean, b) the benchmark scheme cannot maintain the boundedness of the overall network if the coupling effects are taken into account and the sources/users are assumed to generate a random traffic with a mean varying periodically, and c) our proposed controller successfully addresses the congestion control problem even if the coupling effects are taken into account and the sources/users are assumed to generate a random traffic with a mean varying periodically.

In the second set of simulations our network cluster is reconfigured as a fully-connected network by incorporating all the loops (with feedback traffic). The results show that: a) the IDCC approach cannot solve the congestion control problem in the sense that convergence and stability of the fully-connected network cannot be achieved even after extensive fine tuning of the link gains that correspond to the process functions $F^{hj}_{kl,r}$, and b) even in the presence of large delays (e.g. of the order of 1 sec.), our proposed distributed congestion control strategy converges and remains stable for the fully-connected DiffServ network provided that the process function gains are properly selected and satisfy $F^{hj}_{kl,f} \leq 0.5$ (this observation is further confirmed through our stability analysis). Note that since in this study we have not considered any dynamics associated with $F^{hj}_{kl,f}$, therefore, an associated gain of less or equal to unity for $F^{hj}_{kl,f}$ can be interpreted as dropping of packets.

For our proposed distributed congestion controller simulations below, the sampling rate is set to $T_s = 1$ ms, whereas for the simulations in which the IDCC scheme is implemented the sampling rate that yields the best results is selected as $T^{id}_s = 0.1$ ms. Furthermore, the process function gains corresponding to the premium service are set to $F^{hj}_{kl,p} = 1$. The design parameters for our proposed distributed congestion control strategy are selected as $\mu_p = 10^3$, $\mu_r = 500$, $\gamma_p = 0.1$ $\gamma_r = 5 * 10^{-2}$, $\sigma_p = 0$, and $\sigma_r = 5$. For the node controllers the estimator gains are set to $\Gamma_f = 10^4$ except for the sensor controller whose ordinary traffic gain is set to $\Gamma_r = 5 * 10^4$. In the IDCC controller, the parameters are set as follows: $\alpha^{id}_p = 10^4$, $\alpha^{id}_r = 100$, $k^{id}_p(0) = C_{serv}/2$ and $\hat{k}^{id}_p = 3 * 10^3$ [89].

Two types of random input traffics are considered in our simulations. The first random traffic is denoted as *static* since its mean is time-invariant and the second random traffic is denoted as *dynamic* (non stationary) since its mean changes periodically. The dynamic traffic is used in order to excite the high frequency modes of the system. Specifically, the mean and the variance of the static traffic are set to 4x10^3 packets/s and 100 packets/s, respectively, whereas for the dynamic traffic they are generated as follows: the premium sources generate random traffic with a mean varying periodically from 0 to 6x10^3 packets/s, a variance of 10^3 packets/s and a frequency of 0.5 rad/s. The ordinary traffic sources mean varies periodically

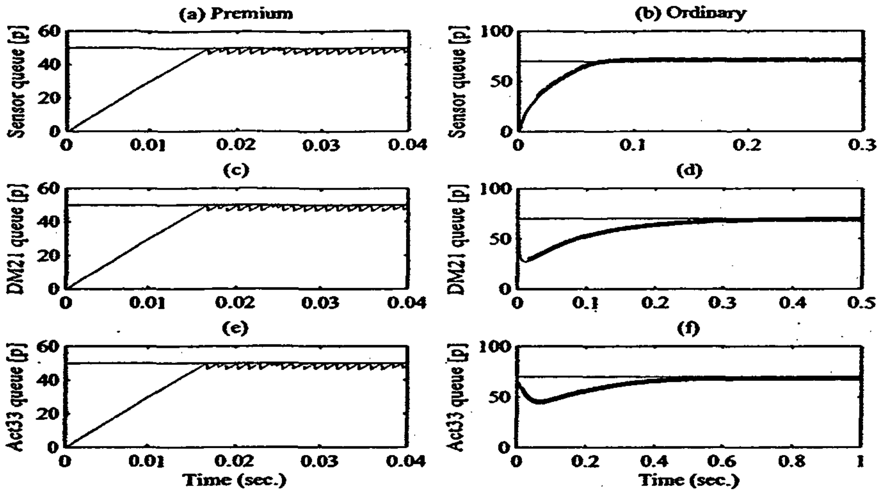


Figure 8.8: Simulation results for a network configured as a three nodes cascade (with no data feedback paths) in presence of the time-delay of $\tau_{kl,f}^{hj} = 1$ ms using the congestion control strategy (IDCC) proposed in [89].

from 0 to 200 packet/s, have a variance of 50 packets/s and a frequency of 0.5 rad/s.

In the first set of simulations, the IDCC controller is implemented with the static input traffic. In the second set of simulations, the IDCC controller is implemented using the dynamic input traffic. In all the simulations our proposed congestion controller is evaluated subject to dynamic input traffic.

According to Assumption 3.1, the buffer queue sizes are constrained to the bounds $0 \leq x_{l_h p} \leq 128$ packets and $0 \leq x_{l_h r} \leq 1024$ packets, respectively. According to Assumption 3.3, the constraint on the maximum available service capacity is given as $0 \leq C_{l_h} \leq 4 * 10^4$ packets/s for both services.

For the first set of simulations the network cluster has a three nodes configuration. Furthermore, the process functions $F_{lk,f}^{hj}$ $(f = p, r)$ are set equal to unity in all the links. The IDCC scheme results are shown in Figures 8.8 and 8.9 when the sources/users generate static traffics, and in Figure 8.10 when they generate dynamic traffics. Figures 8.8 and 8.10 correspond to the time delays of $\tau_{lk,f}^{hj} = 1$ ms, whereas

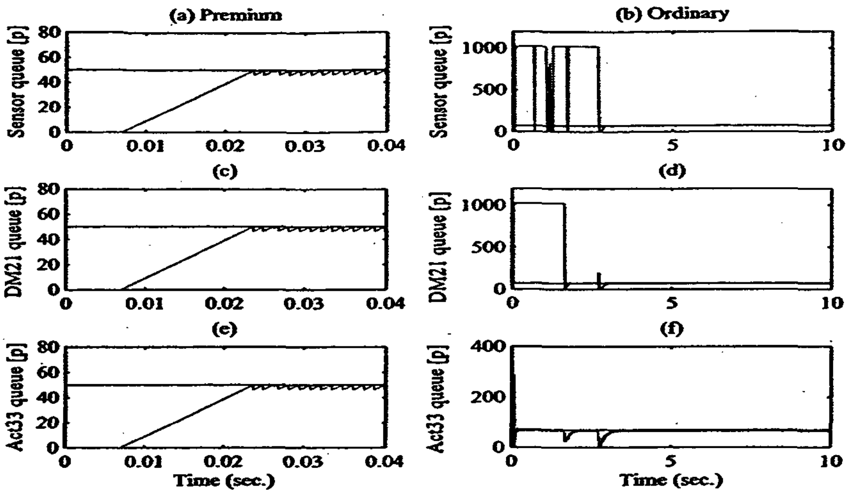


Figure 8.9: Simulation results for a network configured as a three nodes cascade (with no data feedback paths) in presence of the time delay of 70 ms using the congestion control strategy (IDCC) proposed in [89].

Figure 8.9 corresponds to $\tau^{hj}_{lk,f} = 70$ ms. The rows in Figures 8.8 and 8.10 depict the buffer queue step responses corresponding to the sensor, the decision maker DM21, and the actuator Act33, respectively. The left hand side column shows the premium service and the right hand side column shows the ordinary service results.

The results in Figures 8.8 and 8.9 clearly demonstrate that when the sources/users generate static traffics and provided that one considers that the ordinary buffer inputs are not coupled, all the buffer queues converge to their respective reference sets of 50 packets and 70 packets for the premium and the ordinary services, respectively. It can also be observed that: a) the premium service step responses for both time-delays are faster than that of the ordinary service, and b) although the ordinary service corresponding to the time-delay of 70 ms has significant overshoot, the queues nevertheless converge to their desired references in steady state. The results in Figure 8.10 show that when the sources/users generate dynamic traffics and provided that one assumes that the ordinary buffer inputs are coupled to one

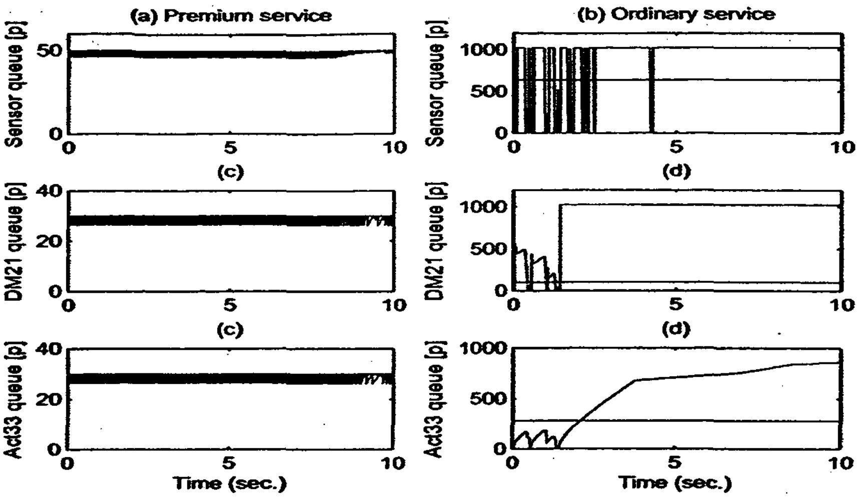


Figure 8.10: Simulation results for a network configured as a three nodes cascade (with no data feedback paths) in presence of the time-delay of $\tau_{kl,f}^{hj} = 1$ ms and dynamic source traffics using the congestion control strategy (IDCC) proposed in [89].

another, then all the ordinary buffer queues diverge from their respective references, implying that the network is globally unstable.

Given the above network cluster configuration, simulation results for our proposed congestion controller are shown in Figure 8.11 for the time delays of 1 ms (solid line) and 70 ms (dashed line). This figure illustrates that even in presence of 70 ms (dotted line) time-delay and a dynamic source traffic our proposed controller stabilizes the network quiet well. Indeed, we may observe that: a) the sensor, the decision maker DM21, and the actuator Act33 buffer queue lengths all converge to their respective references which are set to $[50, 30, 30]$ packets for the premium service and to $[640, 100, 280]$ packets for the ordinary service, respectively, b) the transient responses except for the premium service that experiences some overshoot are acceptable, and c) better control performance is obtained by using our proposed controller when compared to the IDCC scheme as shown in the results of Figures

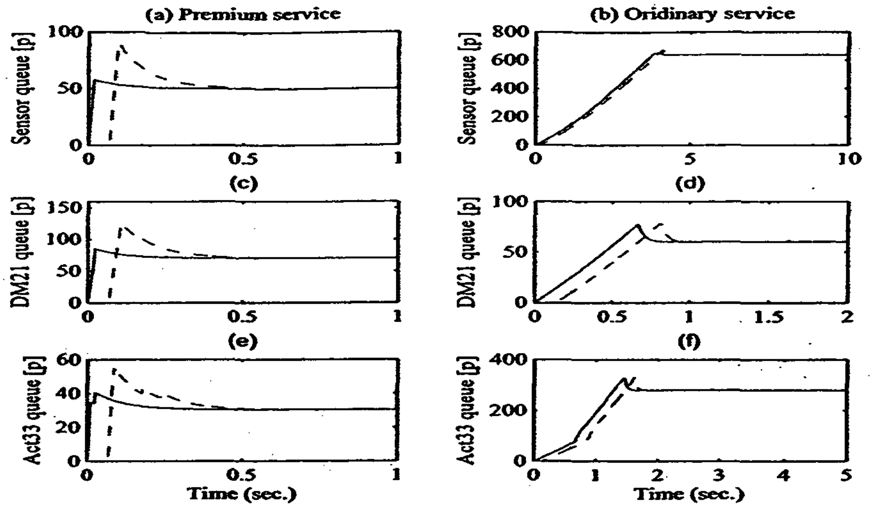


Figure 8.11: Simulation results for a network configured as a three nodes cascade (with no data feedback paths) in presence of time delays (solid line: 1 ms, dashed line: 70 ms) and dynamic source traffics using our proposed congestion control strategy.

8.8-8.10.

In the second set of simulations, we consider a network cluster configuration that incorporates full feedback paths. The fully-connected network cluster is configured by selecting the following switching combinations, namely $S_k^s = [1\ 1\ 1]$, $S_{kl}^d = [1\ 1\ 1; 1\ 1\ 1]$ and $S_{kl}^a = [1\ 1\ ;\ 1\ 1\ ;\ 1\ 1]$.

The buffer queue references for the premium and the ordinary services are set as follows: for the sensor node we have $[50, 640]$ packets, for the output port DM21 corresponding to λ_{o1}^d in Figure 3.4 we have $[70, 60]$ packets, and for the output port DM22 corresponding to λ_{o2}^d in Figure 3.4 we have $[30, 100]$ packets. Also for the actuator node the output port *Act*31 corresponding to λ_{o1}^a has the premium and ordinary queue references set to $[50, 70]$ packets, respectively, the premium and the ordinary references of the output port *Act*32 corresponding to λ_{o2}^a are set to $[70, 50]$ packets, respectively and finally the premium and the ordinary queue references of

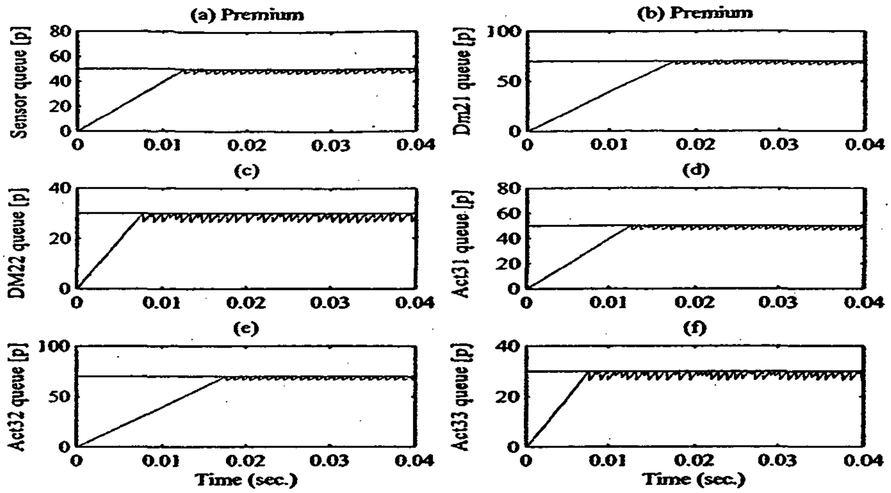


Figure 8.12: Simulation results for the premium service traffic obtained for the fully-connected network in the presence of the delay $\tau^{hj}_{kl,f} = 1$ ms and the dropping packet gain of $F^{hj}_{kl,r} = 1$ by using the congestion control strategy (IDCC) proposed in [89].

the output port *Act*33 corresponding to λ^{a}_{o3} are set to [30, 280] packets, respectively. Note that since the ordinary services have lower priority than the premium services, their assigned buffer size is greater, and consequently the ordinary references are set to higher values when compared to the premium references.

The simulation results for the fully-connected network using the IDCC controller are shown in Figures 8.12-8.14. It can be observed that due to the existence of feedback paths, a) the step responses for the premium service traffics still converge to their desired references as shown in Figure 8.12 whenever the premium input traffic is limited to $\lambda_{l_h p} \leq 3 * 10^3$ packets/s and become unstable beyond this limit, and b) the step responses for the ordinary service traffics are unstable even if the link process gains $F^{hj}_{kl,r}$ are selected as small as $F^{hj}_{kl,r} = 0.1$ (refer to Figures 8.13 and 8.14).

The above results do clearly demonstrate and identify the restrictive limitations of the IDCC controller for a network cluster with feedback traffics. Below, we

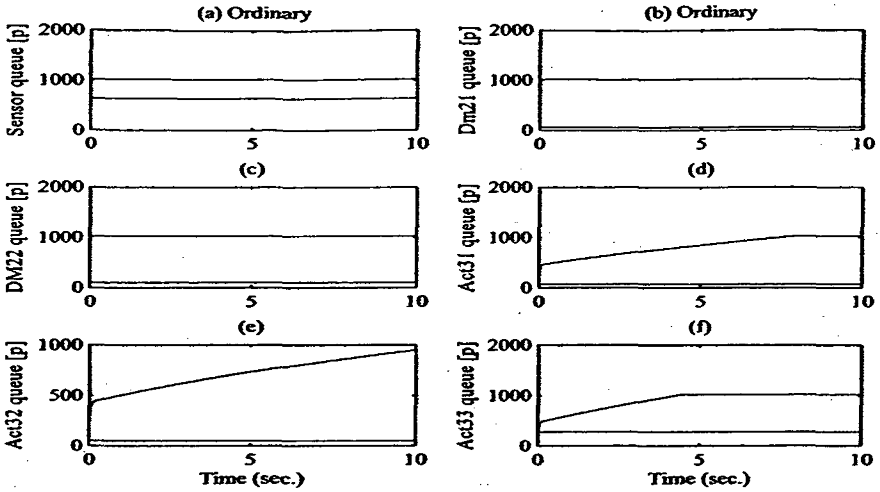


Figure 8.13: Simulation results for the ordinary service traffic obtained for the fully-connected network in the presence of the delay $\tau^{hj}_{kl,f} = 1$ ms and the dropping packet gain of $F^{hj}_{kl,r} = 1$ by using the congestion control strategy (IDCC) proposed in [89].

will show that by utilizing our proposed controller it will be possible to overcome and remedy these problems. The simulations are considered for a fully-connected network cluster with all the links process gains set to $F^{hj}_{kl,f} = 0.5$. Note that this value is close to the numerical bound that is obtained in Remark 8.6, namely $F^{hj}_{kl,f} \leq 0.6$. In other words, the smaller the gains $F^{hj}_{kl,f}$, the better the performance of the controller and its convergence property. By setting the process gains to $F^{hj}_{kl,f} = 0.5$, one observes from results of Figures 8.15 and 8.16 that all the buffer queue lengths do indeed converge to their respective references for both services.

The last simulation results are shown in Figure 8.17 to demonstrate the effectiveness and capabilities of our proposed congestion control strategy in presence of a "large" delay of 1 s. In addition, to show the characteristic behavior of a typical node, here chosen as the sensor output port, we depict the queue lengths, the actual input and output traffics, and the estimate of the input traffic and the controlled

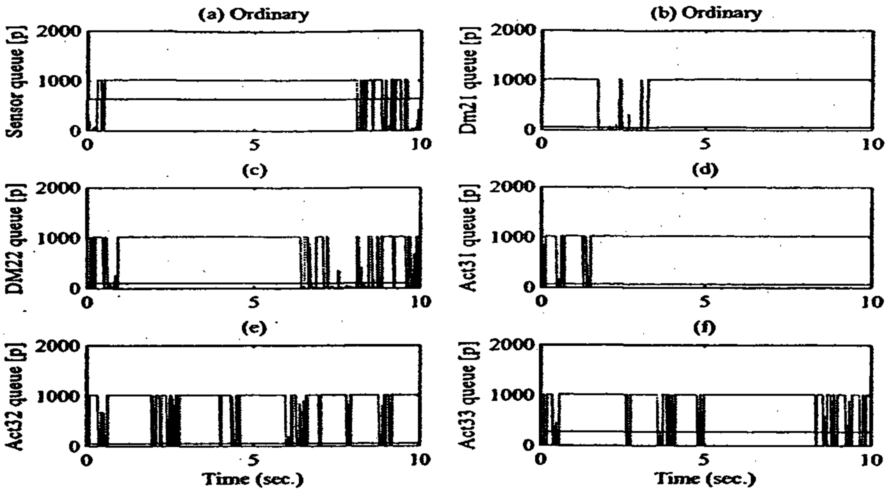


Figure 8.14: Simulation results for the ordinary service traffic obtained for the fully-connected network in the presence of the delay $\tau_{kl,f}^{hj} = 1$ ms and the dropping packet gain of $F_{kl,r}^{hj} = 0.1$ by using the congestion control strategy (IDCC) proposed in [89].

capacities for both services. From the first row of Figure 8.17, it follows that despite the presence of 1 s delay both the premium and the ordinary queue lengths do indeed converge to their corresponding references. This confirms, as argued in Remark 8.5, that by setting $F_{kl,f}^{hj} = 0.5$, the time delayed closed-loop dynamics is stable for sufficiently large delay. The second row in Figure 8.17 shows the premium and the ordinary capacities that are used as control variables. The last two rows of Figure 8.17 depict the actual buffer input and output traffics as well as the estimate of the input traffic. Note that the actual output traffic converges to the actual input traffic according to the FFM conservation principle, whereas the estimate of the input traffic on which the control computation is based on converges to the actual dynamic input traffic. The plots of the last row of Figure 8.17 are the zoomed in respective plots of the third row of the same figure, given here to readily observe the convergence of the three plots. It is worth noting that the spikes that appear in the behavior of the capacities and the traffics are due to the estimation errors

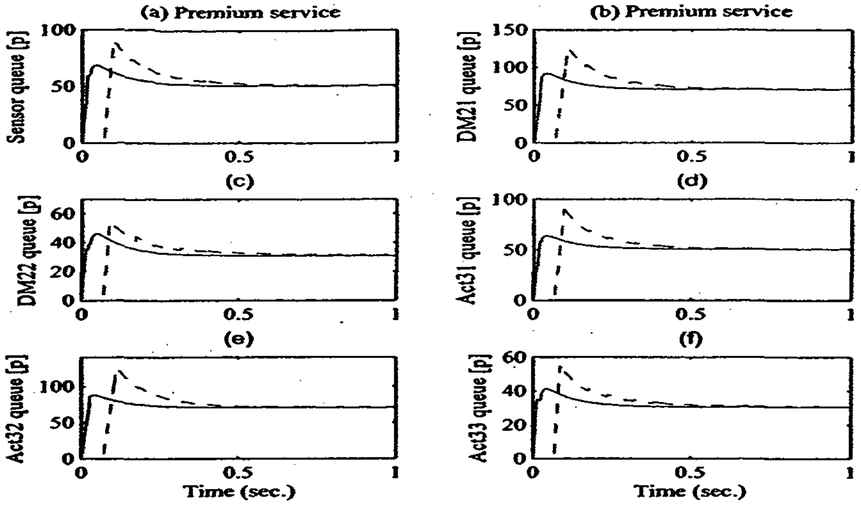


Figure 8.15: Simulation results for the premium service traffic obtained for the fully-connected network in the presence of the delay (solid line: 1ms, dashed line: 70 ms) and the dropping packet gain of $F^{hj}_{kl,f} = 0.5$ by using our proposed congestion control strategy.

introduced by the large delay that is present in the network. By assuming that all the input traffics are known *a priori* to the controllers the spikes will all disappear.

8.7 Conclusion

In this chapter, the robust congestion control strategy for a cascaded DiffServ network proposed in Chapter 7 is extended. The extension consists mainly of adding the traffic estimation and the ordinary bandwidth allocation control. For our robust congestion strategy, two controllers are proposed by adopting the semi-decentralized end-to-end and the distributed hop-by-hop control approaches. The distributed hop-by-hop control approach is considered to reduce the information required in the congestion control scheme. Indeed, each controller communicates the maximum

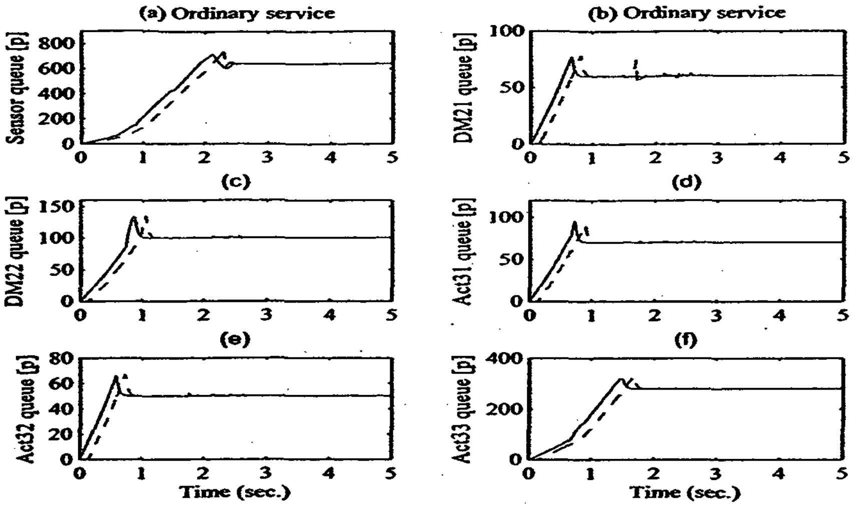


Figure 8.16: Simulation results for the ordinary service obtained for the fully-connected network in the presence of the delay (solid line: 1ms, dashed line: 70 ms) and the dropping packet gain of $F^{hj}_{kl,f} = 0.5$ by using our proposed congestion control strategy.

allowed flow rate only with its immediate upstream node and/or source.

The semi-decentralized end-to-end congestion controller is implemented on a five nodes cascade network and the simulation results have shown that the buffer queue length responses corresponding to the ordinary service are oscillatory. The distributed hop-by-hop congestion controller is investigated first on a general mesh network and later on a NUS cluster which consists of a sensor, decision-maker, and actuator. Note that contrary to the mesh network, the NUS cluster is mainly characterized by traffic feedback.

For each network, the resulting closed-loop network cluster error dynamics is studied as a time-delay system and stability conditions are derived that are only dependent on the configuration of the network. Specifically, we have shown that in presence of time-varying and unknown but bounded delays, the error dynamics of the controlled network is $\mathcal{L}_\infty$ stable for an $\mathcal{L}_\infty$ input traffic. The analytical results

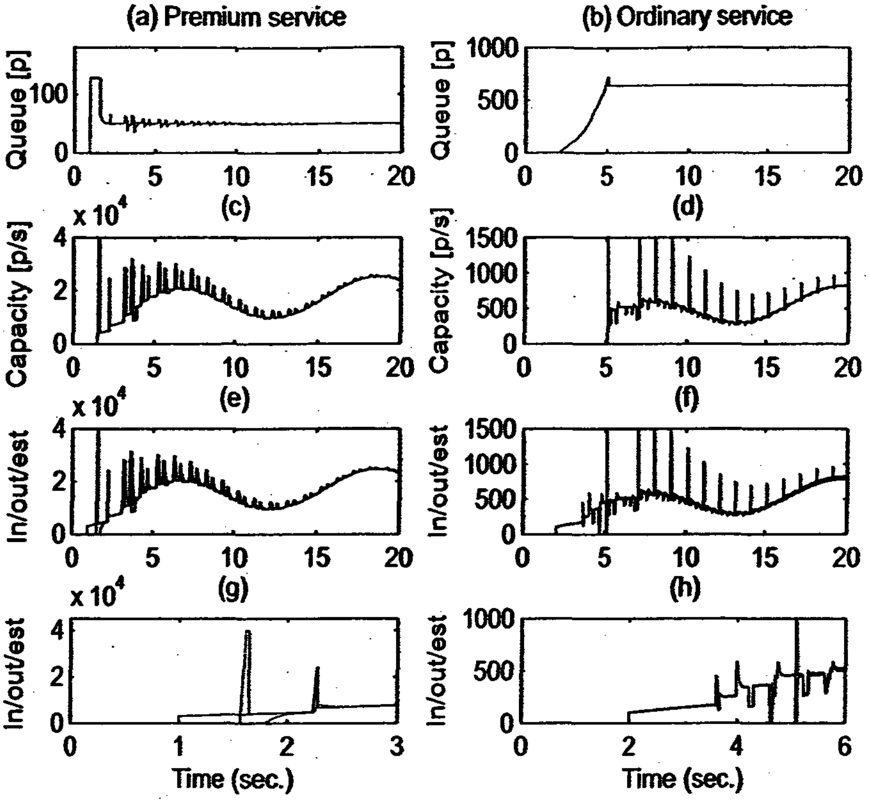


Figure 8.17: Simulation results for the sensor output port in the fully-connected network in the presence of the delay $\tau_{kl,f}^{hj} = 1$ s and the dropping packet gain of $F_{kl,f}^{hj} = 0.5$) by using our proposed congestion control strategy.

are confirmed through a number of simulation case studies. A comparative study is conducted on our network cluster to demonstrate and illustrate the advantages and superiority of our proposed congestion controller when compared to a conventional IDCC method (which in general results in an unstable closed-loop system).

Chapter 9

Conclusions and Future Research

In this thesis, we have presented resource allocation and congestion control strategies for networked unmanned systems (NUS). The research problem is divided into three parts. In the first part that is covered in Chapter 4, the bandwidth allocation problem is considered and a robust nonlinear control algorithm is proposed. In the second part that is covered in Chapter 5, three robust congestion control strategies that are integrating the bandwidth allocation and the flow rate control problems are proposed. Finaly, the third part that comprises Chapters 6-8, propose several robust congestion control strategies for differentiated-services (DiffServ) NUS systems.

Before discussing the main contributions of each of the aforementioned parts and outlining our future research directions, it is worth noting that some algorithms and formulations that are background information in this research are reviewed in Chapter 2. Furthermore, a DiffServ NUS system on which we have evaluated our proposed control algorithms is introduced and described in Chapter 3. In Chapter 2, fluid flow model (FFM)-based resource allocation and congestion control benchmark solutions are first discussed. Subsequently, then sliding mode-based generalized variable structure control (SM-GVSC) concepts are reviewed. In Chapter 3, a general NUS system that can be seen as a fully-connected DiffServ network is presented.

Each NUS cluster is defined as a group of three nodes, namely, a sensor, a decision-maker, and an actuator. Using the (FFM) model, the overall dynamics of our cluster is derived as a time-delay dependent system.

The first part of the thesis dealt with the problem of resource (bandwidth) allocation control within our proposed network for different configurations in terms of possible interconnections of various nodes. To demonstrate the difficulty of allocating the bandwidth to different nodes of a NUS, a standard PID controller is implemented and the simulation results show that depending on interconnections between the network nodes, the behavior of the closed-loop delayed system vary considerably such that the network might become even unstable. Using the sliding mode machinery, a robust bandwidth allocation controller is proposed and evaluated on our NUS system. For unknown but upper bounded time-delays, the resulting fully-connected control dynamics is shown analytically and through simulations to be stable. In addition, our proposed robust bandwidth allocation controller is evaluated when node mobility is considered and good performance and stability are demonstrated for a wide range of operating points provided that the delay is upper bounded.

The second part of this thesis has focused on the integration of the bandwidth allocation and the flow rate control problems. Three robust control strategies are proposed where the results can be outlined as follows: a) the first and second strategies do not need any knowledge about the incoming traffic in order to compute their commands while the third strategy uses the measured or estimated input traffic in its control law, b) using the first and third strategies for the network cluster that is configured as a cascade or fully-connected network, all buffer queue lengths converge to their respective references even in presence of time-delays of 60 ms, c) the buffer queue length responses obtained using the first strategy are very oscillatory while those obtained using the third strategy are oscillation-free, and d) using the third strategy, the results obtained when the input traffic is assumed to be measurable are obviously better than those obtained when the input traffic is estimated (presence of overshoots and longer settling time). We believe that by adding the σ-modification to the update law, the third strategy would constitute a significant step toward an efficient robust integrated control algorithm.

The third and the last part of this thesis constitutes the core of our research. Using SM-GVS control techniques, several robust congestion control strategies are proposed. These strategies are categorized into two classes, namely, the *non degenerate* and the *degenerate*. In fact, the GVS control technique is recently introduced to alleviate the undesirable chattering phenomenon by designing a control law that switches on the derivatives of the control input (*non degenerate* case) rather than on the control input itself as in the classical variable structure (CVS). In some cases, when the GVS design may lead to a control law that doesn't exhibit the derivatives of the control input, it is called *degenerate case*. In this thesis, three *non degenerate* GVS controllers are proposed on the basis of a non-minimal realization of the FFM model and two versions of *degenerate* GVS controllers are designed using the original first-order FFM model, resulting in the simplicity of the latter controllers.

As far as the *non degenerate* case is concerned, it is observed through a number of simulations that for both the premium and the ordinary services, the convergence of our proposed technique results in good performance in addition to robustness with respect to modeling inaccuracy, propagation delays, and variations of the premium traffic.

Consistent with the first version of the *degenerate* GVS controllers, the controllers are first implemented on a single bottleneck and then on a multi-node (cascade) network. For a single node, the time-delayed dependent congestion control dynamics is derived analytically and studied in order to guarantee stability of the closed-loop system. It is observed that the premium buffer queue dynamics corresponding to the most important traffic in the DiffServ network is insensitive to time-delays, and a condition on the time-delay upper bound should be satisfied for the ordinary buffer queue dynamics to ensure stability. For a multi-node (cascade) network, the time-delayed dependent congestion control dynamics is derived analytically in order to guarantee stability of the closed-loop system eventhough the approach is conservative. Simulation results have shown that for the premium service, the buffer queues converge to their respective reference sets and for the ordinary service, the convergence of the buffer queues can be reached but oscillations are present in the ordinary step responses inducing thus steady state errors.

To remedy the oscillatory responses obtained in the ordinary service using the above first version of the d*egenerate* GVS controller, the latter controller needed to be improved by adding the traffic estimation and the ordinary bandwidth allocation control. This results in the second version of the *degenerate* GVS controller that is investigated by adopting the semi-decentralized end-to-end as well as the distributed hop-by-hop control approaches. The distributed hop-by-hop control approach is considered to reduce the information that is required in the congestion control scheme. Indeed, each controller communicates the maximum allowed flow rate only with its immediate upstream node and/or source.

Although the semi-decentralized end-to-end congestion controller has demonstrated good performance and stability on a three node (cascade) network, oscillatory responses are observed one a five node network, showing the weak scalability property of the end-to-end approach. As far as the distributed hop-by-hop congestion controller is concerned, it is studied first on a general mesh network (Internet) and then on a NUS system cluster. For each network, the resulting closed-loop network cluster error dynamics is investigated as a time-delay system and stability conditions are derived that are only dependent on the configuration of the network. Specifically, we have shown that in presence of time-varying and unknown but bounded delays, the error dynamics of the controlled network is $\mathcal{L}_\infty$ stable for an $\mathcal{L}_\infty$ input traffic. The analytical results are confirmed through a number of simulation case studies. A comparative study is conducted on our network cluster to demonstrate and illustrate the advantages and superiority of our proposed congestion controller when compared to a conventional method (which in general results in an unstable closed-loop system).

Our future research will deal with the extension of our proposed algorithms and their implementations. On one hand, simulations will be conducted using networking-oriented software such as network simulator 2 (NS2) or QUALNET and on the other hand, experimentations will follow by using first a network of computers (laptops will be more appropriate for mobility purpose) and then using small mobile robots.

Our future work will focus on the scalability property of our proposed distributed hop-by-hop congestion controller when implemented on a larger NUS system (three clusters).

Our future investigations will also consider a non-commensurate and non identical time-varying delays. Note that preliminary simulation results on non-identical time-delays have demonstrated the effectiveness and performance of our proposed distributed hop-by-hop congestion control laws.

The last issue that will constitute our future research will focus on the proposed distributed approach. Note that in our proposed distributed hop-by-hop congestion control strategy, the controllers communicate with one another in one direction (i.e. from the downstream controller node to the upstream controllers). Preliminary investigations on the bidirectional case are worth pursuing and appear to provide additional benefits and advantages.

Bibliography

[1] C. Agnew. Dynamic modelling and control of congestion-prone. *Operations Research*, 24, No.3:400–419, 1976.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communication Magazine*, 40, issue 8:102–114, August 2002.

[3] I. F. Akyildiz, X. Xudong, and W. Wang. A survey on wireless mesh networks. *IEEE Communications Magazine*, 43:s23–s30, Setember 2005.

[4] I. F. Akyildiz, X. Xudong, and W. Wang. Wireless mesh networks: a survey. *Computer Network, Elsevier Science Publishers*, 47:445–487, March 2005.

[5] A. T. Al-Hammouri, M. S. Branicky, V. Liberatore, and S. M. Phillips. Decentralized and dynamic bandwidth allocation in networked control systems. In *International Workshop on Parallel and Distributed Real-Time Systems*, April 2006.

[6] A. T. Al-Hammouri and V. Liberatore. Transversal issues in real-time sense-and-respond systems. In *EESR05*, 2005.

[7] M. Baines, B. Nandy, P. Pieda, N. Seddigh, and M. Devetsikiotis. Using tcp models to understand bandwidth assurance in a differentiated services network. Technical report, Nortel Technical Report, July 2000.

[8] G. Bartolini, A. Ferrara, and E. Usai. Chattering avoidance by second order sliding mode control. *IEEE Transactions on Automatic Control*, 43, No. 2:241–246, February 1998.

[9] G. Bastin and V. Guffens. Congestion control in compartmental network systems. *Systems and Control Letters, Elsevier*, 55:689–696, 2006.

[10] M. Belhocine, M. Hamerlain, and K. Bouyoucef. Tracking trajectory for robot using variable structure control. In *Proc. of the 4th ECPD, International Conference on Advanced Robotics, Intelligent Automation and Active Systems*, pages 207–212, Moscow, 24-26 August 1998.

[11] L. Benmohamed and Y. T. Yang. A control-theoretic ABR explicit rate algorithm for ATM switches with per-vc queueing. *Procedings of IEEE Joint Conference on Computer and Communications Societies*, 1:183–191, 1998.

[12] H. Bin-Abbas and D. Tipper. A call level adaptive bandwidth allocation scheme based on lyapunov control theory. In *Proc. of 11th IEEE Symposium on Computers and Communications ISCC'06*, pages 655–661, June 2006.

[13] D. Black, S. Blake, M. Clarson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated services. *Network Working Group*, RFC2475, 1998.

[14] K. Bouyoucef and K. Khorasani. A network-centric input-output feedback linearization-based control strategy for unmanned systems. In *Proc. of SPIE, Defense Transformation and Network-Centric Systems*, pages 160–171, Orlando, USA, May 2005.

[15] K. Bouyoucef and K. Khorasani. Variable structure-based nonlinear congestion control for networked controlled systems. In *Proc. of IEEE International*

Conference on Physics and Control, pages 583–586, Saint-Petersburg, Russia, August 2005.

[16] K. Bouyoucef and K. Khorasani. Chattering-alleviated generalized variable structure control for experimental robot manipulators. In *Proc. of the 38th IEEE Southeastern Symposium on System Theory*, pages 152–156, Tennessee, USA, 5-7 March 2006.

[17] K. Bouyoucef and K. Khorasani. Robust feedback linearization-based congestion control using a fluid flow model. In *Proc. of IEEE 2006 American Control Conference*, pages 4883–4887, Mineapolis, USA, 14-16 June 2006.

[18] K. Bouyoucef and K. Khorasani. A network-centric robust resource allocation strategy for unmanned systems: stability analysis. In *Proc. of SPIE'07, Defense Transformation and Network-Centric Systems*, Orlando, USA, 9-13, April 2007.

[19] K. Bouyoucef and K. Khorasani. A robust congestion control strategy for delay dependent differentiated-services networks. In *Proc. of 2007 IEEE Internaltional Symposium on Industrial Electronics*, Vigo, Spain, June, 4-7 2007.

[20] K. Bouyoucef and K. Khorasani. A sliding mode-based congestion control for time delayed differentiated-services networks. In *Proc. of IEEE Mediteraneen on Control and Automation*, pages 1–6, Athens, Greece, June, 27-29 2007.

[21] K. Bouyoucef and K. Khorasani. A distributed congestion control strategy for differentiated-services networks. In *Proc. of International Conference on Communication Theory, Reliability, and Quality of Service*, Bucharest, Romania, 29 June-5 July 2008.

[22] K. Bouyoucef and K. Khorasani. A Distributed Differentiated-Services Congestion Control of a Networked Sensor, Decider, and Actuator System. *submitted to IEEE/ACM Transactions on Networking*, 2008.

[23] K. Bouyoucef and K. Khorasani. A Robust Distributed Congestion Control Strategy for Differentiated-Services Network. *provisionally accepted at IEEE Transactions on Industrial Electronics*, 2008.

[24] K. Bouyoucef, K. Khorasani, and M. Hamerlain. *Experimental results on variable structure control for an uncertain robot model.* Book chapter, I-Tech Publishing Company, 2008.

[25] L. Brakmo and L. Peterson. TCP vegas: end-to-end congestion avoidance on a global internet. *IEEE J. Select. Areas Commun.*, 13, No. 8:1465–1480, Oct. 1995.

[26] J. Chen, F. Paganini, R. Wang, M.Y. Sanadidi, and M. Gerla. Fluid-flow analysis of tcp westwood with red. *Computer Network Elsevier*, 50:1302–1326, June 2006.

[27] M. Chen and A. Zakhor. Rate control for streaming and video over wireless. In *IEEE Procedings Transaction on Multimedia*, Hongkong, China, March 2004.

[28] M. Chen and A. Zakhor. AIO-TFRC: a light-weighted rate control scheme for streaming over wireless. In *IEEE WirelessCom Symposium on multimedia over wireless*, June 2005.

[29] M. Chen and A. Zakhor. Flow control over wireless network and application layer implementation. In *Proceeding of INFOCOM'06*, Barcelona, Spain, April 2005.

[30] M. Chen and A. Zakhor. Multiple TFRC connections based rate control for wireless networks. *IEEE Transaction on Multimedia*, 8, issue 5:1045–1062, October 2006.

[31] N. Christin, J. Liebeherr, and T. F. Abdelzaher. A quantitative assured forwarding service. In *Proc. IEEE INFOCOM'02*, New York, NY, June 2002.

[32] C. Chrysostomou, A. Pitsillides, G. Hadjipollas, M. Polycarpou, and A. Sekercioglu. Congestion control in differentiated-services networks using fuzzy logic. In *43rd Conference on Decision and Control*, pages 543–548, December 2004.

[33] C. Chrysostomou, A. Pitsillides, G. Hadjipollas, A. Sekercioglu, and M. Polycarpou. Fuzzy explicit marking for congestion control in differentiated-services networks. In *8th IEEE Symposium on Computers and Communications*, pages 312–319, 2003.

[34] G. Conte, C. H. Moog, and A. Perdon. Un theoreme sur la representation entree-sortie d'un systeme non lineaire. *C. R. Acad. Sci. Paris*, 307, serie I:363–366, 1988.

[35] S. Deb and R. Srikant. Rate-based versus queue-based models of congestion control. *ACM Transactions on Modeling and Computer Simulation*, 51:606–619, April 2006.

[36] A. Demers, S. Keshav, and S. Shenker. Analysis and simulation of a fair queueing algorithm. In *ACM Special Interest Group on Data Communications*, pages 1–12, 1998.

[37] S-X. Du and S-F. Han. Congestion control of multimedia stream for ATM multiplexers using neural networks. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 2206–2211, Oct. 2005.

[38] C. T. Ee and R. Bajcsy. Congestion control and fairnes for many-to-many routing in sensors networks. In *SenSys'04*, Baltimore, Maryland, USA, November 2004.

[39] A. Eryilmaz and R. Srikant. Joint control, routing and MAC for stability and fairness in wireless networks. *IEEE Journal on Selected Areas in Communications*, 2006.

[40] B. Davie et al. An expedited frowarding PHB Per-Hop-Behabior. *Network Working Group*, RFC3246, March 2001.

[41] N. G. Duffield et al. A flexible model for resource managment in virtual private networks. In *Proc. ACM SIGCOMM'99*, pages 95–108, Cambridge, MA, Oct. 1999.

[42] J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicles formations. *IEEE Transaction on Automatic Control*, 49, No. 9:1465–1476, September 2004.

[43] J. Filipiak. *Modeling and control of dynamics flows in communication networks.* New York: Springer-Verlag, 1992.

[44] M. Fliess. Generalized controller canonical forms for linear and nonlinear dynamic. *IEEE Transactions on Automatic Control*, 35, No.1:994–1001, 1990.

[45] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transaction on Networking*, 1, No.4:397–413, Aug. 1993.

[46] K. Furuta, K. Kosuge, and K. Kobayshi. VSS-type self-tuning control of direct drive motor. In *Proc. of the of IEEE/IECON89*, pages 281–286, Philadelphia, November 1989.

[47] V. Guffens and G. Bastin. Optimal adaptive feedback control of a network. In *IEEE American Conference on Control*, pages 1835–1840, Portland, June 2005.

[48] V. Guffens, G. Bastin, and H. Mounier. Fluid ow network modeling for hop-by-hop feedback control design and analysis. In *Proceedings of Internetworking*, San Jose, June 2003.

[49] M. Hamerlain, K. Bouyoucef, and M. Belhocine. Sliding mode control for a robot SCARA. In *Proceeding of IFAC/ACE97*, pages 153–157, Istambul, Turkey, 14-16 July 1997.

[50] E. Harashima, H. Hashimoto, and K. Maruyama. Practical robust control of robot arm using variable structure systems. In *Proceeding of IEEE, International Conference on Robotics and Automation*, pages 532–538, San Francisco, 1986.

[51] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski. Assured Frowarding PHB Group. *Network Working Group*, RFC2597, June 1999.

[52] J. L. Hillman, S. D. Jones, R. A. Nichols, and I. J. Wang. Communications network architectures for the army future combat system and objective force. In *IEEE MILCOM'02, Proceedings*, pages Vol.2 1417–1421, October 7-10 2002.

[53] I. Hsu and J. Walrand. Dynamic bandwidth allocation for ATM switches. *J. Applied Probability*, 33, No. 3, 1996.

[54] L. Hsu and R. Costa. Analysis and design of I/O based variable structure adaptive control. *IEEE Transactions on Automatic Control*, 39, No.1:4–21, 1994.

[55] W. Hu, G. Liu, and D. Rees. Event-driven networked predictive control. *IEEE Transactions on Industrial Electronics*, 54, No.3:1603–1613, June 2007.

[56] B. Hull, K. Jamieson, and H. Balakrishnan. Mitigating congestion in wireless sensor networks. In *Proc. ACM Sensys'04*, Baltimore, MD, November 2004.

[57] P. Ioannou and P.V. Kokotovic. Instability analysis and improvement for robustness of adaptive control. *Automatica*, 20:583–594, 1984.

[58] A. Isidori. *Nonlinear Control Systems.* Addison-Wesley Publishing Company, 1992.

[59] Y. G. Iyer, S. Gandham, and S. Venkatesan. STCP: a generic transport layer protocol for wireless sensor networks. In *IEEE Procedings of International Conference on Computer Communications and Networks*, San Diego, October 2005.

[60] V. Jacobson. Congestion avoidance and control. In *Symp. proc. Communication Architectures and Protocols*, pages 314–329, Stanford, CA, 1988.

[61] V. Jacobson, R. Braden, and D. Borman. TCP extensions for high performance. *Network Working Group*, pages 1465–1480, RFC1323 1992.

[62] J. A. Jacquez and C. P. Simon. Qualitative theory of compartmental systems with lags. *Mathematical Biosciences, Elsevier*, 180:329–362, November 2006.

[63] M. Karaliopoulos, R. Tafazolli, and B. G. Evans. Providng differentiated services to TCP flows over bandwidth on demand geostationary satellite networks. *IEEE Journal on Selected Area in Communications*, 22, No.2:333–347, Feb. 2004.

[64] H. Kaufman, I. Barkana, and K. Sobel. *Direct Adaptive Control Algorithms, Theory and Applications.* New York, 1998.

[65] F. Kelly. Fairness and stability of end-to-end congestion control. *European Journal of Control*, 9:159–176, 2003.

[66] G. Kesidis. Bandwidth adjustments using online packet-level measurements. In *SPIE conf. Performance and Control of Networks Systems*, Boston, MA, Sept. 1999.

[67] K. M. Kevin, B. E. Mullins, G.W.P. York, and R.O. Baldwin. Impact of limited communications on a cooperative search algorithm for multiple UAVs. In *IEEE international conference on Networking, Sensing and Control*, pages 572–577, April 2006.

[68] H. K. Khalil. *Nonlinear systems.* Prentice-Hall, Englewood Cliffs, NJ, 3rd ed. edition, 2002.

[69] M. Kim and Y. Mun. Distributed control scheme for diffserv in heterogeneous mobile IP networks. In *Proc. of International Conference on Computational Sciences and Its Applications*, pages 60–64, June-July 2008.

[70] S. Kim, S. A. Campbell, and X. Liu. Stability of a class of linear switching systems with time delay. *IEEE Transactions on Circuits and Systems*, 53, No.2:384–393, February 2006.

[71] J. F. Kurose and K. W. Ross. *Computer Networking: A Top Down Approach Featuring the Internet.* Addison-Wesley, 2001.

[72] T. J. Kwon, M. Gerla, V.K. Varma, M. Barton, and T.R. Hsing. Efficient flooding with passive clustering-an overhead-free selective forward mechanism for as hoc/sensor networks. *Procedings of the IEEE*, 91, No.8:1210–1220, August 2003.

[73] J. Lakkakorpi, O. Standberg, and J. Salonen. Adaptive connection admission control for differentiated services access networks. *IEEE Journal on Selected Area in Communications*, 23, No.10:1963–1972, Oct. 2005.

[74] K. P. Lebarteaux, C. E. Rohrs, and P. J. Antsaklis. A practical controller for explicit rate congestion control. *IEEE Transactions on Automatic Control*, 47, issue6:960–978, June 2002.

[75] B. K. Lee, L. K. John, and E. John. Architectural enhancements for network congestion control applications. *IEEE Transaction on Very Large Scale Systems*, 14, No.6:609–615, June 2006.

[76] K. C. Lee, S. Lee, and M. H. Lee Rees. Worst case communication delay of real-time industrial switched ethernet with multiple levels. *IEEE Transactions on Industrial Electronics*, 53, No.5:1669–1676, October 2006.

[77] A. Levant and L. Alelishvili. Integral high-order sliding modes. *IEEE Transactions on Automatic Control*, 5, No. 7:12781282, July 2007.

[78] V. Liberatore, W. S. Newman, and K. Bashin. IP communication and distributed agents for unmanned autonomous vehicles. In *AIAA-UAV'03*, 2003.

[79] G. P. Liu, Y. Xia, J. Chen, D. Rees, and W. Hu. Networked predictive control of systems with random network delay in both forward and feedback channel. *IEEE Transactions on Industrial Electronics*, 54, No.3:1282–1297, June 2007.

[80] Y.C. Liu and C. Douligeris. Rate regulation with feedback controller in ATM networks-A neural network approach. *IEEE Journal in special areas in communication*, 15, No.2:200–208, February 1997.

[81] M. Mahramian, H. Taheri, and M. Haeri. AMPCS: Adaptive model predictive control scheduler for guaranteed delay in diffserv architecture. *International Journal of Communication Systems*, 21-3:233–249, 2008.

[82] E. Messager. *Sur la stabilisation discontinue des systemes*. PhD thesis, Orsay, Paris, 1992.

[83] P. P. Mishra, H. Kanakia, and S. T. Tripathi. On hop-by-hop rate-based congestion control. *IEEE/ACM Transaction on Networking*, 4, no.2:224–239, Apr. 1996.

[84] P. Newman. Backward explicit congestion notification for ATM local area networks. In *Proc. GLOBECOM'98*, pages 719–723, 1993.

[85] K. Nichols, S. Blake, F. Baker, and D. Black. Definition of the differentiated services field (DS Field) in the IPv4 and IPv6 header. *Network Working Group*, RFC2475, 1998.

[86] S. I. Niculescu. Stability and hyperbolicity of linear systems with delayed state: a matrix-pencil approach. *IMA Journal of Mathematical Control and Information*, 15:331–347, 1998.

[87] S. I. Niculescu, L. Dugard, and J. M. Dion. *Stability et stabilization robustes des systems a retards.* Hermes, Paris, 285-314, 1995.

[88] C. E. Palazzi, C. Roseti, M. Luglio, M. Gerla, M. Y. Sanadidi, and J. Stepanek. Satellite coverage in urban areas using unmanned airborne vehicles UAVs. In *59th IEEE Conference on vehicular technology*, pages 2886–2890, May 17-19 2004.

[89] A. Pitsillides, P. Ioannou, M. Lestas, and L. Rossides. Adaptive Nonlinear Congestion Controller for a Differentiated-Services Framework. *IEEE Transactions on Networking*, 13, No.1:94–107, February 2005.

[90] A. Pitsillides, P. Ioannou, and L. Rossides. Congestion control for differentiated-services using nonlinear control theory. In *6th EEE Symposium on Computers and Communications*, pages 726–733, 3-5, July 2001.

[91] A. Pitsillides, P. Ioannou, and D. Tipper. Integrated control of connection admission, flow rate, and bandwidth for ATM based networks. In *INFOCOM'96*, pages 785–793, March 1996.

[92] A. Pitsillides and J. Lambert. Adaptive congestion control in ATM based networks: quality of service with utilization. *J. Comput. Commun.*, 20:1239–1258, 1997.

[93] A. Pitsillides, J. Lambert, and D. Tipper. Dynamic bandwidth allocation in broadband-ISDN using a multilevel optimal control approach. In *INFOCOM'95*, pages 1086–1094, April 1995.

[94] A. Pitsillides and A. Sekercioglu. Fuzzy logic based effective congestion control. In *TCD workshop on applications of computational Intelligence to Telecommunications*, London, May 1999.

[95] H. Sira Ramirez. A dynamical variable structure control strategy in asymptotic output tracking problems. *IEEE Transactions on Automatic Control*, 38, No. 4:615–620, April 1993.

[96] S. Rathinam, R. Sengupta, and S. Darbha. A resource allocation algorithm for multivehicle systems with nonholonomic constraints. *IEEE Transaction on Automation Science and Engineering*, 99:1–1, 2006.

[97] C. E. Rohrs, R. A. Berry, and S. J. O'halek. A control engineer look at ATM congestion avoidance. In *Proc. GLOBECOM'95*, 1995.

[98] L. Rossides, A. Pitsillides, and P. Ioannou. Non-linear congestion control: comparison of a fluid flow based model with OPNET simulated ATM switch model. *Department of Computer Science, University of Cyprus*, TR-99-1, 1999.

[99] A. Ryan, M. Zennaro, A. Howell, R. Sengupta, and J.K. Hedrick. An overview of emerging results in cooperative UAV control. In *43rd IEEE conference on Decision and Control*, pages 14–17, Atlantis, Paradise Islan, Bahamas, December 2004.

[100] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. Sastry. Foundation of control and estimation over lossy networks. In *Proceedings of IEEE*, February 2007.

[101] A. Segall. The modelling of adaptive routing in data communication networks. *IEEE Transactions on on communications*, 25, No.1:85–95, January 1977.

[102] U. Shaked, I. Yaesh, and C. E. de Souza. Bounded real criteria for linear time-delay systems. *IEEE Transactions on Automatic Control*, 43, No.7:1016–1022, July 1998.

[103] S. Sharma and D. Tipper. Approximate models for the study of nonstationary queues and their applications to communication networks. In *Procedings of IEEE International Conference Communications*, pages 352–358, May 1993.

[104] S. Shenker, L. Zhang, and D. D. Clark. Some observation on the dynamics of a congestion control algorithms. *Comput. Commun. Rev.*, pages 30–39, Oct. 1990.

[105] J. Shu and P. Varaiya. Smart pay access control via incentive alignment. *IEEE Journal on Selected Areas in Communications*, Vol. 24, NO. 5:1051–1060, May 2006.

[106] P. Siripongwutikom, S. Banergee, and D. Tipper. Adaptive bandwidth control for efficient aggregate QoS provisioning. In *Proc. IEEE GLOBECOM'02*, Taipei, Taiwan, Nov. 2002.

[107] P. Siripongwutikom, S. Banergee, and D. Tipper. A survey of adaptive bandwidth control algorithms. *IEEE Communications Surveys and Tutorials*, 5, No. 1:14–26, 2003.

[108] R. Sivakumar, T. Kim, N. Venkitaraman, and V. Bharghavan. Achieving perflow weighted rate fairness in a core stateless network. In *IEEE International Conference on Distributed Computing Systems*, pages 188–196, Tapei, Taiwan, April 2000.

[109] J. J. E. Slotine and J. A. Coetsee. Adaptive sliding controller synthesis for nonlinear systems. *International Journal of Control*, 43 No. 6:1631–1651, November 1986.

[110] R. Srikant. *Models and methods for analyzing internet congestion control algorithms*. Lecture notes in control and information sciences edition, 2004.

[111] D. Stiliadis and A. Varma. Efficient fair-queing algorithms for packet-switched networks. *IEEE/ACM Transaction on Networking*, 6, No.2:175–185, April 1998.

[112] N. O. Tippenhauer. A multi-robot architecture for autonomous cooperative behaviours. Technical Report Research Project- Final Report, University of Waterloo, August 2005.

[113] D. Tipper, Y. Qian, and S. Banergee. Modeling the time varying behavior of mobile ad-hoc networks. In *Procedings of International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Venezia, Italy, October 2004.

[114] D. Tipper and M. K. Sundareshan. Numerical methods for modeling computer networks under nonstationary conditions. *IEEE Journal on Selected Areas in Communications*, 8, No. 9:1682–1695, December 1990.

[115] D. Tipper and M.K. Sundareshan. Numerical methods for modelling computer networks under non stationary conditions. *IEEE Journal of Selected Areas in Communications*, December 1990.

[116] V. I. Utkin. *Sliding Mode in Control Optimization.* Springer-Verlag, Berlin, 1995.

[117] A. J. Van and D. Schaft. L2-gain analysis of nonlinear systems and nonlinear state feedback H1 control. *IEEE Transactions on Automatic Control*, 37, No.6:770–784, June 1992.

[118] C. Wang, K. Sohraby, V. Lawrence, B. Li, and H. Yueming. Priority-based congestion control in wireless sensor networks. In *IEEE Int. Conf. on Sensor Networks, Ubiquitous and Trustworthy Computing*, pages 22–31, June, 05-07 2006.

[119] C. Wang, K. Sohraby, B. Li, M. Daneshmand, and H. Yueming. A survey of transport protocols for wireless sensor networks. *IEEE Networks*, 20, issue 3:34–40, May-June 2006.

[120] J. Wang, W. Yurcik, Y. Yang, and J. Hester. Multiring techniques for scalable battlespace group communication. *IEEE Communication Magazine*, 43, issue 11:124–133, November 2005.

[121] S. Wang, D. Xuan, R. Bettati, and W. Zhao. Providng absolute differentiated services for real-time applications in static-priority scheduling networks. *IEEE/ACM Transaction on Networking*, 12, No.2:326–339, Apr. 2004.

[122] T. C. Yang. Networked control systems: a brief survey. In *IEE Proceedings in Control Theory and Applications*, pages 403–412, July 2006.

[123] I. Yeom and A. Reddy. Realizing throughput guarantees in a differentiated services network. In *Proceedings of ICMCS*, pages 372–376, Italy, June 1994.

[124] Y. Yi and S. Sakkottai. Hop-by-hop congestion control over a wireless multi-hop network. *IEEE/ACM Transaction on Networking*, 15, no.1:133–144, Feb. 2007.

[125] N. Zhang, Y. Jing, M. Yang, and S. Zhang. Robust AQM controller for diff-serv network using sliding mode control. In *Proc. of the International Federation of Automation Control*, pages 5635–5639, Seoul, July 2008.

[126] N. Zhang, Y. Jing, Y. Zhou, M. Yang, and S. Zhang. Backstepping based variable structure controller design for diff-serv network. In *IEEE American Conference on Control*, pages 4451–4455, Seattle, June 2008.

[127] W. Zhang. *Stability Analysis of Networked Control Systems.* Case Western Reserve University, 2001.

[128] X. Zheng, N. Zhang, G. M. Dimirovski, and Y. Jing. Aadaptive sliding mode congestion control for diff-serv network. In *Proc. of the International Federation of Automation Control*, pages 12983–12987, Seoul, July 2008.