

https://dspace.org/

Congestion control in packet switch networks

Kamga, Morgan

2008-12-10T11:08:01Z
http://hdl.handle.net/10539/5884

Downloaded from a DSpace repository. https://dspace.org/

CONGESTION CONTROL IN PACKET SWITCH NETWORKS

Morgan M. KAMGA
School of Computational and Applied Mathematics
University of the Witwatersrand

A dissertation submitted to the Faculty of Science
in fulfillment of the requirements of the degree of
Masters of Sciences in
Computational and Applied Mathematics

Under the Supervision of

Prof. K. H. WONG and **Mr. C. MYBURGH**
School of Computational and Applied Mathematics
University of the Witwatersrand

Declaration

I declare that this dissertation is my own, unaided work. It is being submitted for the Degree of Master of Science to the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination to any other University.

(Signature)

(Date)

Abstract

We consider a *congestion control* problem in *computer networks*. The problem is posed as an optimal control problem and reduced to a problem of finding solutions to delay differential equations. Systems involving time delays in the dynamics are actually very difficult to model and therefore very difficult to solve. We consider three approaches in our congestion control problem: an *elastic queue approach* leading to an optimal control problem with a state–dependent delay differential equation; three approaches in flow models (also leading to systems containing delay differential equations), precisely the *dual control approach*, the *primal–dual control approach* and the *control approach based on queueing delay*. The elastic queue approach is not explored due to the lack of software good enough to solve optimal control problems involving delay differential equations.

In flow models, we consider the standard case, that is where the feedback from sources to links is exact and the network behaves perfectly well (without any unexpected event). We also consider some non–standard cases such as the case where this feedback contains errors (for example overestimation, underestimation or noise), and the case where one link breaks in the network. We numerically solve the delay differential equations obtained and use the results we get to determine all the considered dynamics in the network. This is followed by an analysis of the results. We also explore the stability of some simple cases in the dual control approach, with weaker conditions on some network parameters, and discuss some fairness conditions in some simple cases in all the flow model approaches. Non–standard cases are also solved numerically and the results can be compared with those obtained in the standard case.

Acknowledgements

Without the help, support and contribution of many people, this work would not have been possible.

I would first of all like to thank the *CAM Staff* for offering me a place in the School of Computational and Applied Mathematics.

I am also grateful to *Prof. K. H. WONG* for accepting to supervise my work and for giving me this topic from which I learnt and continue to learn some mathematics.

I would also like to thank *Mr. Colin MYBURGH* for his availability, for some interesting discussions and for co–supervising the current work.

A special thanks to the Head of School *Prof. D. SHERWELL* for his support, advice and encouragements.

I don't forget to thank my friends *Naval ANDRIANJAFINANDRASANA, Mihaja RAMANANTOANINA, Charles Lebon MBERI KIMPOLO, Gasper MWANGA* and *Musa GABERE* for their support, discussions and collaboration.

I am very grateful to my family for their unconditional support and gracious assistance despite the distance between us.

I finally express my gratitude to all the people who in one way or another contributed to the achievement of this work.

Contents

Abstract **ii**

Introduction **1**

1 Preliminaries **3**
1.1 Computer Networks 3
1.2 Flow Models 4
1.3 Laplace Transforms 7
1.4 Optimal Control Theory 8

2 Literature Review **10**
2.1 Congestion Avoidance 10
2.2 An Elastic Queue Approach to Congestion Avoidance 10
2.3 An Optimisation Approach to Flow Control 12
2.4 Some Flow Control Schemes 13
2.4.1 Dual Control Laws 14
2.4.2 Primal–Dual Control Laws 15
2.4.3 Control Laws Based on Queueing Delay 16
2.5 Stability Analysis of Flow Control Schemes 17

3 An Overview of Delay Differential Equations **18**
3.1 An Introduction 18
3.2 Solving Delay Differential Equations 19
3.2.1 Existence and Uniqueness of Solutions 21
3.2.2 Euler's Method for Delay Differential Equations . . . 22
3.2.3 Heun's Method for Delay Differential Equations 23
3.3 Illustrative Examples 25
3.4 Remark 28

4 Exploration of Some Control Schemes **29**
4.1 The Dual Control Approach 29
4.1.1 Formulation 29
4.1.2 Stability Analysis 31
4.1.3 Conditions for Fairness in some Simple Cases 37

4.2 The Primal–Dual Control Approach 39
4.2.1 Formulation 39
4.2.2 Stability Analysis 41
4.2.3 Dynamics of Source Rates 41
4.2.4 Conditions for Fairness in some Simple Cases 41
4.3 The Control Approach based on Queueing Delay 43
4.3.1 Formulation 43
4.3.2 Dynamics of Source Rates 45
4.3.3 Conditions for Fairness in some Simple Cases 45

5 Computer Simulations **47**
5.1 The Dual Control Approach 47
5.1.1 One link and one source 48
5.1.2 One link and three sources 48
5.1.3 Three links and one source 49
5.1.4 Three links and four sources 52
5.2 The Primal–Dual Control Approach 54
5.2.1 One link and one source 56
5.2.2 One link and three sources 56
5.2.3 Three links and four sources 58
5.3 The Control Approach based on Queueing Delay 59
5.3.1 One link and one source 60
5.3.2 One link and three sources 62
5.3.3 Three links and four sources 63

6 Some Non–Standard Cases **68**
6.1 Case With Errors 68
6.1.1 The Dual Control Approach 71
6.1.2 The Primal–Dual Control Approach 72
6.1.3 The Control Approach based on Queueing Delay . . . 74
6.2 Impact of Errors on the Network Dynamics 75
6.3 When a Link Breaks in the Network 75
6.3.1 The Dual Control Approach 76
6.3.2 The Primal–Dual Control Approach 79
6.3.3 The Control Approach based on Queueing Delay . . . 79

General Conclusion **82**

A Graphical Representations (Non–Standard Cases) **83**
A.1 Dual Control 83
A.2 Primal–Dual Control 93
A.3 Control based on Queueing Delay 103

Bibliography **113**

List of Figures

1 Source rate (Slow Start) 1
2 Source rate (A I M D) 1

1.1 A network composed of 3 links and 4 nodes 3
1.2 Some physical network topologies 4
1.3 A network composed of 3 links and 4 sources 5

3.1 Poor approximation to the solution of a DDE. 20
3.2 Ordinary and delay differential equations 1 25
3.3 Ordinary and delay differential equations 2 26
3.4 Ordinary and delay differential equations 3 27

4.1 Conic Sector 36

5.1 A network composed of 1 link and 1 source 48
5.2 Link price, case 1 with 1 link and 1 source (Dual) 48
5.3 Link price, case 2 with 1 link and 1 source (Dual) 48
5.4 Source rate, case 1 with 1 link and 1 source (Dual) 49
5.5 Source rate, case 2 with 1 link and 1 source (Dual) 49
5.6 Queue size, case 1 with 1 link and 1 source (Dual) 49
5.7 Queue size, case 2 with 1 link and 1 source (Dual) 49
5.8 A network composed of 1 link and 3 sources 49
5.9 Link price, simple case with 1 link and 3 sources (Dual) 50
5.10 Link price, fairness case with 1 link and 3 sources (Dual) 50
5.11 Source rates, simple case with 1 link and 3 sources (Dual) 50
5.12 Source rates, fairness case with 1 links and 3 sources (Dual) 51
5.13 Queue size, simple case with 1 link and 3 sources (Dual) 51
5.14 Queue size, fairness case with 1 link and 3 sources (Dual) 51
5.15 A network composed of 3 links and 1 source 51
5.16 Link prices, case 1 with 3 links and 1 source (Dual) 52
5.17 Link prices, case 2 with 3 links and 1 source (Dual) 52
5.18 Source rate, case 1 with 3 links and 1 source (Dual) 53
5.19 Source rate, case 2 with 3 links and 1 source (Dual) 53
5.20 Queue sizes, case 1 with 3 links and 1 source (Dual) 53
5.21 Queue sizes, case 2 with 3 links and 1 source (Dual) 53

5.22 A network composed of 3 links and 4 sources 53
5.23 Link prices with 3 links and 4 sources (Dual) 54
5.24 Link prices with 3 links (same capacity) and 4 sources (Dual) 54
5.25 Source rates with 3 links and 4 sources (Dual) 55
5.26 Source rates with 3 links (same capacity) and 4 sources (Dual) 55
5.27 Queue sizes with 3 links and 4 sources (Dual) 56
5.28 Queue sizes with 3 links (same capacity) and 4 sources (Dual) 56
5.29 Link price, case 1 with 1 link and 1 source (P D C) 56
5.30 Link price, case 2 with 1 link and 1 source (P D C) 56
5.31 Source rate, case 1 with 1 link and 1 source (P D C) 57
5.32 Source rate, case 2 with 1 link and 1 source (P D C) 57
5.33 Queue size, case 1 with 1 link and 1 source (P D C) 57
5.34 Queue size, case 2 with 1 link and 1 source (P D C) 57
5.35 Link price, simple case with 1 link and 3 sources (P D C) 57
5.36 Link price, fairness case with 1 link and 3 sources (P D C) 57
5.37 Source rates, simple case with 1 link and 3 sources (P D C) 58
5.38 Source rates, fairness case with 1 links and 3 sources (P D C) 58
5.39 Queue size, simple case with 1 link and 3 sources (P D C) 59
5.40 Queue size, fairness case with 1 link and 3 sources (P D C) 59
5.41 Link prices with 3 links and 4 sources (P D C) 59
5.42 Link prices with 3 links (same capacity) and 4 sources (P D C) 60
5.43 Source rates with 3 links and 4 sources (P D C) 60
5.44 Source rates with 3 links (same capacity) and 4 sources (P D C) 61
5.45 Queue sizes with 3 links and 4 sources (P D C) 61
5.46 Queue sizes with 3 links (same capacity) and 4 sources (P D C) 61
5.47 Link price, case 1 with 1 link and 1 source (C Q D) 61
5.48 Link price, case 2 with 1 link and 1 source (C Q D) 61
5.49 Source rate, case 1 with 1 link and 1 source (C Q D) 62
5.50 Source rate, case 2 with 1 link and 1 source (C Q D) 62
5.51 Queue size, case 1 with 1 link and 1 source (C Q D) 62
5.52 Queue size, case 2 with 1 link and 1 source (C Q D) 62
5.53 Link price, simple case with 1 link and 3 sources (C Q D) 62
5.54 Link price, fairness case with 1 link and 3 sources (C Q D) 62
5.55 Source rates, simple case with 1 link and 3 sources (C Q D) 63
5.56 Source rates, fairness case with 1 links and 3 sources (C Q D) 63
5.57 Queue size, simple case with 1 link and 3 sources (C Q D) 64
5.58 Queue size, fairness case with 1 link and 3 sources (C Q D) 64
5.59 Link prices with 3 links and 4 sources (C Q D) 64
5.60 Link prices with 3 links (same capacity) and 4 sources (C Q D) 65
5.61 Source rates with 3 links and 4 sources (C Q D) 65
5.62 Source rates with 3 links (same capacity) and 4 sources (C Q D) 66
5.63 Queue sizes with 3 links and 4 sources (C Q D) 66
5.64 Queue sizes with 3 links (same capacity) and 4 sources (C Q D) 67

6.1 An example of network where a link breaks 76
6.2 Source rates with 3 links and 4 sources (Dual) 77
6.3 Queue size with 3 links and 4 sources (Dual) 77
6.4 Source rates (with noise) with 3 links and 4 sources (Dual) 78
6.5 Queue size (with noise) with 3 links and 4 sources (Dual) 78
6.6 Source rates with 3 links and 4 sources (C Q D) 80
6.7 Queue size with 3 links and 4 sources (C Q D) 80
6.8 Source rates (with noise) with 3 links and 4 sources (C Q D) 81
6.9 Queue size (with noise) with 3 links and 4 sources (C Q D) 81

A.1 Link price, case 1 with 1 link and 1 source (Dual) 83
A.2 Link price, case 2 with 1 link and 1 source (Dual) 83
A.3 Link price, case 3 with 1 link and 1 source (Dual) 84
A.4 Link price, case 4 with 1 link and 1 source (Dual) 84
A.5 Source rate, case 1 with 1 link and 1 source (Dual) 84
A.6 Source rate, case 2 with 1 link and 1 source (Dual) 84
A.7 Source rate, case 3 with 1 link and 1 source (Dual) 84
A.8 Source rate, case 4 with 1 link and 1 source (Dual) 84
A.9 Queue size, case 1 with 1 link and 1 source (Dual) 85
A.10 Queue size, case 2 with 1 link and 1 source (Dual) 85
A.11 Queue size, case 3 with 1 link and 1 source (Dual) 85
A.12 Queue size, case 4 with 1 link and 1 source (Dual) 85
A.13 Link price, case 1 with 1 link and 3 sources (Dual) 85
A.14 Link price, case 2 with 1 link and 3 sources (Dual) 85
A.15 Link price, case 3 with 1 link and 3 sources (Dual) 86
A.16 Link price, case 4 with 1 link and 3 sources (Dual) 86
A.17 Source rates, case 1 with 1 link and 3 sources (Dual) 86
A.18 Source rates, case 2 with 1 link and 3 sources (Dual) 86
A.19 Source rates, case 3 with 1 link and 3 sources (Dual) 87
A.20 Source rates, case 4 with 1 link and 3 sources (Dual) 87
A.21 Queue size, case 1 with 1 link and 3 sources (Dual) 87
A.22 Queue size, case 2 with 1 link and 3 sources (Dual) 87
A.23 Queue size, case 3 with 1 link and 3 sources (Dual) 88
A.24 Queue size, case 4 with 1 link and 3 sources (Dual) 88
A.25 Link price, case 1 with 3 links and 4 sources (Dual) 88
A.26 Link price, case 2 with 3 links and 4 sources (Dual) 88
A.27 Link price, case 3 with 3 links and 4 sources (Dual) 89
A.28 Link price, case 4 with 3 links and 4 sources (Dual) 89
A.29 Source rates, case 1 with 3 links and 4 sources (Dual) 90
A.30 Source rates, case 2 with 3 links and 4 sources (Dual) 91
A.31 Source rates, case 3 with 3 links and 4 sources (Dual) 91
A.32 Source rates, case 4 with 3 links and 4 sources (Dual) 92
A.33 Queue size, case 1 with 3 links and 4 sources (Dual) 92
A.34 Queue size, case 2 with 3 links and 4 sources (Dual) 92

A.35 Queue size, case 3 with 3 links and 4 sources (Dual) 92
A.36 Queue size, case 4 with 3 links and 4 sources (Dual) 92
A.37 Link price, case 1 with 1 link and 1 source (P D C) 93
A.38 Link price, case 2 with 1 link and 1 source (P D C) 93
A.39 Link price, case 3 with 1 link and 1 source (P D C) 93
A.40 Link price, case 4 with 1 link and 1 source (P D C) 93
A.41 Source rate, case 1 with 1 link and 1 source (P D C) 93
A.42 Source rate, case 2 with 1 link and 1 source (P D C) 93
A.43 Source rate, case 3 with 1 link and 1 source (P D C) 94
A.44 Source rate, case 4 with 1 link and 1 source (P D C) 94
A.45 Queue size, case 1 with 1 link and 1 source (P D C) 94
A.46 Queue size, case 2 with 1 link and 1 source (P D C) 94
A.47 Queue size, case 3 with 1 link and 1 source (P D C) 94
A.48 Queue size, case 4 with 1 link and 1 source (P D C) 94
A.49 Link price, case 1 with 1 link and 3 sources (P D C) 95
A.50 Link price, case 2 with 1 link and 3 sources (P D C) 95
A.51 Link price, case 3 with 1 link and 3 sources (P D C) 95
A.52 Link price, case 4 with 1 link and 3 sources (P D C) 95
A.53 Source rates, case 1 with 1 link and 3 sources (P D C) 95
A.54 Source rates, case 2 with 1 link and 3 sources (P D C) 96
A.55 Source rates, case 3 with 1 link and 3 sources (P D C) 96
A.56 Source rates, case 4 with 1 link and 3 sources (P D C) 97
A.57 Queue size, case 1 with 1 link and 3 sources (P D C) 97
A.58 Queue size, case 2 with 1 link and 3 sources (P D C) 97
A.59 Queue size, case 3 with 1 link and 3 sources (P D C) 97
A.60 Queue size, case 4 with 1 link and 3 sources (P D C) 97
A.61 Link price, case 1 with 3 links and 4 sources (P D C) 98
A.62 Link price, case 2 with 3 links and 4 sources (P D C) 98
A.63 Link price, case 3 with 3 links and 4 sources (P D C) 99
A.64 Link price, case 4 with 3 links and 4 sources (P D C) 99
A.65 Source rates, case 1 with 3 links and 4 sources (P D C) 100
A.66 Source rates, case 2 with 3 links and 4 sources (P D C) 100
A.67 Source rates, case 3 with 3 links and 4 sources (P D C) 101
A.68 Source rates, case 4 with 3 links and 4 sources (P D C) 101
A.69 Queue size, case 1 with 3 links and 4 sources (P D C) 101
A.70 Queue size, case 2 with 3 links and 4 sources (P D C) 101
A.71 Queue size, case 3 with 3 links and 4 sources (P D C) 102
A.72 Queue size, case 4 with 3 links and 4 sources (P D C) 102
A.73 Link price, case 1 with 1 link and 1 source (C Q D) 103
A.74 Link price, case 2 with 1 link and 1 source (C Q D) 103
A.75 Link price, case 3 with 1 link and 1 source (C Q D) 103
A.76 Link price, case 4 with 1 link and 1 source (C Q D) 103
A.77 Source rate, case 1 with 1 link and 1 source (C Q D) 103
A.78 Source rate, case 2 with 1 link and 1 source (C Q D) 103

A.79 Source rate, case 3 with 1 link and 1 source (C Q D) 104
A.80 Source rate, case 4 with 1 link and 1 source (C Q D) 104
A.81 Queue size, case 1 with 1 link and 1 source (C Q D) 104
A.82 Queue size, case 2 with 1 link and 1 source (C Q D) 104
A.83 Queue size, case 3 with 1 link and 1 source (C Q D) 104
A.84 Queue size, case 4 with 1 link and 1 source (C Q D) 104
A.85 Link price, case 1 with 1 link and 3 sources (C Q D) 105
A.86 Link price, case 2 with 1 link and 3 sources (C Q D) 105
A.87 Link price, case 3 with 1 link and 3 sources (C Q D) 105
A.88 Link price, case 4 with 1 link and 3 sources (C Q D) 105
A.89 Source rates, case 1 with 1 link and 3 sources (C Q D) 105
A.90 Source rates, case 2 with 1 link and 3 sources (C Q D) 106
A.91 Source rates, case 3 with 1 link and 3 sources (C Q D) 106
A.92 Source rates, case 4 with 1 link and 3 sources (C Q D) 107
A.93 Queue size, case 1 with 1 link and 3 sources (C Q D) 107
A.94 Queue size, case 2 with 1 link and 3 sources (C Q D) 107
A.95 Queue size, case 3 with 1 link and 3 sources (C Q D) 107
A.96 Queue size, case 4 with 1 link and 3 sources (C Q D) 107
A.97 Link price, case 1 with 3 links and 4 sources (C Q D) 108
A.98 Link price, case 2 with 3 links and 4 sources (C Q D) 108
A.99 Link price, case 3 with 3 links and 4 sources (C Q D) 109
A.100 Link price, case 4 with 3 links and 4 sources (C Q D) 109
A.101 Source rates, case 1 with 3 links and 4 sources (C Q D) 110
A.102 Source rates, case 2 with 3 links and 4 sources (C Q D) 110
A.103 Source rates, case 3 with 3 links and 4 sources (C Q D) 111
A.104 Source rates, case 4 with 3 links and 4 sources (C Q D) 111
A.105 Queue size, case 1 with 3 links and 4 sources (C Q D) 111
A.106 Queue size, case 2 with 3 links and 4 sources (C Q D) 111
A.107 Queue size, case 3 with 3 links and 4 sources (C Q D) 112
A.108 Queue size, case 4 with 3 links and 4 sources (C Q D) 112

Introduction

Congestion control is a research area using tools from applied mathematics such as optimal control theory, differential equations and numerical analysis, to solve some problems in computer networks. Due to the fact that the Internet is a very useful tool for communication and research and is thus used by a wide range of people, it is almost impossible to satisfy all the Internet customers. One advantage of congestion control is that it can help to improve the network performance, and therefore satisfy more customers. Congestion problems were observed and increased considerably on the Internet in the late $1980's$. Researchers such as Van Jacobson and Michael Karels [6] contributed in improving congestion avoidance and congestion control models. Models developed at that time did not involve differential equations, but sources just had to adjust their congestion window (amount of information sent by the source during a round–trip time). However, the network resources were not optimally utilised. We can consider for example the slow start and the AIMD (Additive Increase Multiplicative Decrease) models in a single link and single source network, where the link capacity is 500 *packets/sec*; the network utilisation is about 45% for the first case and 75% for the second case, and the source rates are presented in figures 1 and 2.

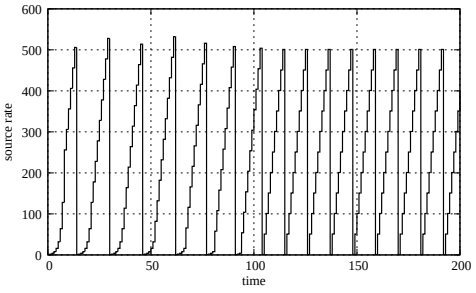


Figure 1: Source rate (Slow Start)

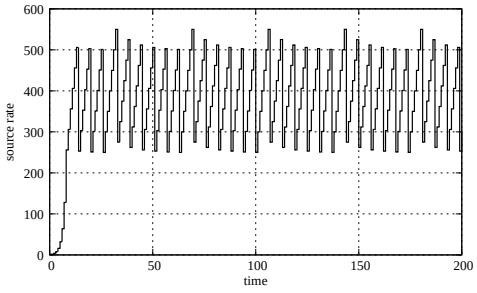


Figure 2: Source rate (A I M D)

In both cases, the congestion window starts at 1 *packet/sec* and is opened exponentially until a certain threshold, and then increased linearly as long as there is no congestion. In the AIMD case, the congestion window is halved as long as there is congestion, otherwise increased linearly if it is above the threshold, exponentially if it is below the threshold, and the cycle repeats. In the slow start case, the threshold is halved as long as there is congestion, and the congestion window set to 1 *packet/sec* and the cycle repeats. Not only the network is not optimally utilised, but in a network composed of many links, the change of window for sources using the first link has a big influence on the subsequent links in the network; and the more links the network has, the further from optimal the network utilisation is. Congestion control models involving differential equations [9, 11, 10] solve these problems, bring the network to a better (optimal) utilisation and also to a much better general behaviour.

The current work is organised as follows: In Chapter 1, we present some basics in different areas that can be of some help in the congestion control problem; in Chapter 2, we present some previous work done by some network researchers; in Chapter 3, we lay out some basics in the area of delay differential equations, mainly some examples and some numerical methods for solving some of them; in Chapter 4, we explore some control schemes in flow models, discuss fairness conditions in some simple cases and also stability analysis in the dual control scheme; in Chapter 5, we perform some computer simulations to illustrate the numerical solutions of the equations involved in our models; in Chapter 6, we start exploring some non–standard cases so that the results obtained can be compared with those obtained in the standard case studied in Chapters 4 and 5; we conclude the work with a final discussion and some directions to possibly explore in the future.

Chapter 1

Preliminaries

In this chapter we lay out some preliminary notions that will be used in the subsequent chapters.

1.1 Computer Networks

A *computer network* is composed of *nodes* (computers, printers, ...) and *links* between nodes. Links are used to send information from one node to another. Information are measured in *bits* and sent in *packets*. The *capacity* of a link is the maximum amount of information that can be sent through that link per unit time, and is measured in *bits/sec* or in *packets/sec*. The link is said to be *congested* (or a *bottleneck*) if the amount of information sent through that link per unit time attempts to be greater than the link capacity. A picture to illustrate computer networks is given in Figure 1.1.

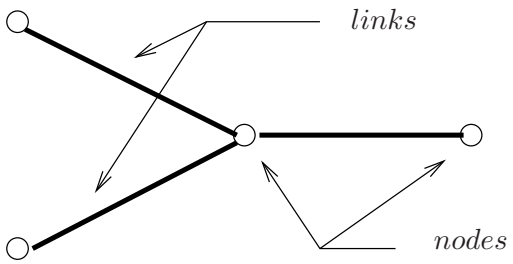


Figure 1.1: A network composed of 3 links and 4 nodes

In networking, the term *topology* refers to how devices are connected in a network. In other words, the network (physical) topology is the pattern

of links connecting pairs of nodes in the network. There are many types of topologies in networks, such as the *star* topology, the *ring* topology, the *bus* topology, just to name the few. Some of them are illustrated in Figure 1.2, taken from the Internet (http://en.wikipedia.org/wiki/Network_topology).

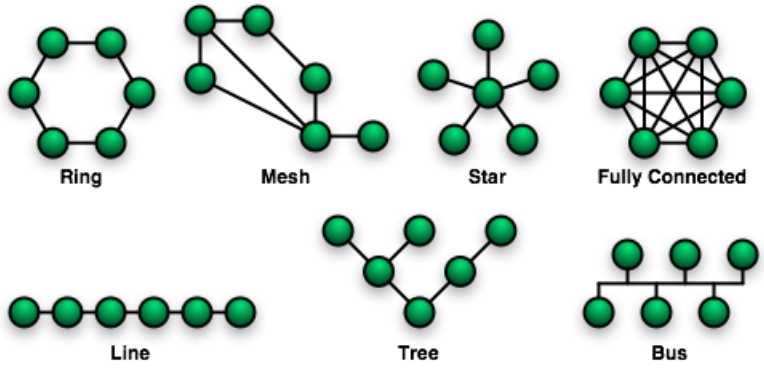


Figure 1.2: Some physical network topologies

Congestion control aims to avoid congestion by controlling various parameters in the network. We consider a model of congestion control in which we manipulate the source rates, (that is, the amount of information sent by each source per unit time) so that eventually, the links are almost fully utilised but not congested. In other words, the sum of source rates per link is equal to (or slightly below) the link capacity.

Three main objectives of congestion avoidance are:

1. Maximise the utilisation of the available bandwidth;
2. Allocate the bandwidth to multiple connections in a possibly fair way;
3. Avoid buffer overflow.

1.2 Flow Models

Flow models [9, 11, 10] are congestion control models in computer networks, where a *source* is not a node, but a flow between two nodes (in one direction) through one or many links of the network. An example of network to illustrate the flow models is given in Figure 1.3.

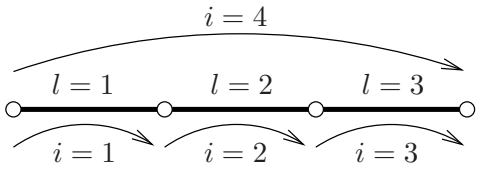


Figure 1.3: A network composed of 3 links and 4 sources

We consider a network composed of a set $\mathcal{L} = \{1, 2, \ldots, L\}$ of unidirectional links shared by a set $\mathcal{S} = \{1, 2, \ldots, S\}$ of sources. Each link is used at least by one source, and each source uses at least one link; but given a source i and a link l, link l need not be used by source i.

Any link $l \in \mathcal{L}$, at time t, is characterised by:

- a capacity denoted by c_l and defined above, measured in *packets/sec*;
- an aggregate rate of flow $y_l(t)$ in *packets/sec*;
- a *price signal* or congestion measure $p_l(t)$;
- a *buffer* or *queue* with occupancy b_l such that $0 \leq b_l \leq B_l$ where $B_l \leq \infty$ is the maximum buffer occupancy (or maximum size). The queue is assumed unbounded.

Any source $i \in \mathcal{S}$, is characterised by:

- its *transmission rate* x_i in *packets/sec*;
- a *minimum transmission rate* $x_{\min,i}$ assumed non–negative and measured in *packets/sec*;
- a *maximum transmission rate* $x_{\max,i}$ assumed positive and finite, in *packets/sec*;
- a *utility function* $U_i : \mathbb{R}_+ \longrightarrow \mathbb{R}$ that is assumed increasing and strictly concave [8] with argument x_i; source i is said to attain a utility of $U_i(x_i)$ if it transmits at a rate x_i such that $x_{\min,i} \leq x_i \leq x_{\max,i}$;
- a *path* $\mathcal{L}(i) \subseteq \mathcal{L}$ defined to be the set of all links used by source i;
- an *aggregate price* $q_i(t)$ of all links used by the source;
- a *round–trip time (RTT)* denoted by τ_i.

For each $l \in \mathcal{L}$, we can also define the set $\mathcal{S}(l)$ of all sources using link l, that is $\mathcal{S}(l) = \{i \in \mathcal{S} : l \in \mathcal{L}(i)\}$.

The dynamics (in discrete time) of the buffer b_l [5] is given by:

$$b_l(t+1) = \left[b_l(t) + \sum_{i \in \mathcal{S}(l)} x_i(t) - c_l \right]_0^{B_l}$$

where $[x]_a^b = \min\big(\max(x,a), b\big)$ provided $a \leq b$. And if $b = \infty$, then $[x]_a^b = \max(x,a)$. It follows that if the buffer is assumed unbounded, its dynamics becomes:

$$b_l(t+1) = \left[b_l(t) + \sum_{i \in \mathcal{S}(l)} x_i(t) - c_l \right]^+$$

where $[x]^+ = \max(x, 0)$.

Since $l \in \mathcal{L}(i) \iff i \in \mathcal{S}(l)$, it is more convenient to work out the dynamics of the network using a *routing matrix*. In a network composed of L links shared by S sources, the *routing matrix* is the $L \times S$ matrix R defined as follows [9, 11, 10, 13]:

$$R_{li} = \begin{cases} 1 & \text{if source } i \text{ uses link } l \\ 0 & \text{otherwise} \end{cases}$$

and is assumed constant.

The *aggregate rate of flow* $y_l(t)$ of link l is obtained by adding the transmission rates of all the sources using link l, that is

$$y_l(t) = \sum_{i=1}^{S} R_{li} x_i \left(t - \tau_{li}^f\right) \tag{1.1}$$

where the τ_{li}^f are the *forward transmission delays* from the sources to the link.

The *aggregate price* $q_i(t)$ of the links used by source i is given by:

$$q_i(t) = \sum_{l=1}^{L} R_{li} p_l \left(t - \tau_{li}^b\right) \tag{1.2}$$

where the τ_{li}^b are the *backward transmission delays* from the links to the source.

The buffer dynamics (in continuous time, with delay and no upper bound) can then be written as follows:

$$\dot{b}_l(t) = \begin{cases} y_l(t) - c_l & \text{if } b_l(t) > 0 \\ \max\{0, y_l(t) - c_l\} & \text{if } b_l(t) = 0 \end{cases}$$

where c_l is the link capacity.

The *round–trip time* τ_i for each source i is given by:

$$\tau_i = \sum_{l=1}^{L} R_{li} \left(\tau_{li}^f + \tau_{li}^b \right).$$

As an example, for the network illustrated in Figure 1.3, the routing matrix is given as follows:

$$R = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}.$$

1.3 Laplace Transforms

In Mathematics, the *Laplace transform* is a useful tool formulated to solve a wide variety of initial–value problems. Differential equations that are most of the time very difficult to solve, can then be transformed into algebraic problems, and those algebraic problems can be solved more easily. To obtain the solution to the initial differential equation, the *inverse Laplace transform* is used.

Given a function $f : [0, \infty) \to \mathbb{R}$, the *Laplace transform* of f is the function F defined as follows:

$$F(s) = \mathcal{L}\{f(t)\} = \int_{0^-}^{\infty} e^{-st} f(t) dt$$

where the argument s is complex in general. The domain described by t is called the *time domain* and the domain described by s is the *Laplace domain* or the *frequency domain*. $\mathcal{L}$ is called the *Laplace operator*. The Laplace transform $F(s)$ is generally defined for all complex numbers s such that $Re(s) > a$, where a is a real constant depending on the growth behaviour of f. The *region of convergence (ROC)* or *domain of convergence* is the subset of values of s for which the Laplace transform exists.

If $F(s)$ is the Laplace transform of $f(t)$, then we say that $f(t)$ is the *inverse Laplace transform* of $F(s)$. In other words, $f(t) = \mathcal{L}^{-1}\{F(s)\}$. More precisely, the inverse Laplace transform of $F(s)$ is the complex integral

$$f(t) = \mathcal{L}^{-1}\{F(s)\} = \frac{1}{2\pi i}\int_{\gamma-i\infty}^{\gamma+i\infty} e^{st}F(s)ds$$

where γ is a real number so that the contour path of integration is in the region of convergence of $F(s)$, provided $\gamma > Re(s_p)$ for every singularity s_p of $F(s)$. If all singularities are in the left half–plane, that is $Re(s_p) < 0$ for every s_p, then γ can be set to zero and the above integral formula becomes identical to the *inverse Fourier transform*. $\mathcal{L}^{-1}$ is called the *inverse Laplace operator*.

Laplace Transforms have some very useful properties. Generally, those properties (of Laplace transforms and inverse Laplace transforms) are used, more than the initial definitions of Laplace transforms and inverse Laplace transforms, to tackle some difficult problems such as solving some complicated differential equations.

1.4 Optimal Control Theory

Optimal control theory is a mathematical field that aims to determine control policies that can be deduced using mathematical optimisation. It deals with the problem of finding a control law for a given system such that a certain optimality criterion is achieved. The problem is posed in terms of minimising a certain functional called *cost functional* or *objective functional*. The control that minimises the cost functional is called the *optimal control*. Generally, it assumes the existence of a dynamical system with input $u(t)$, output $y(t)$ and state $x(t)$. The cost functional usually takes the form of an integral over time of some function, plus a final cost that depends on the state in which the system ends up. Optimal control problems are generally of the form:

$$\min g_0(u) = \underbrace{\Phi_0(x(T|u))}_{\text{terminal state cost}} + \underbrace{\int_0^T \mathcal{L}_0(t, x(t|u), u(t))dt}_{\text{integral cost}}$$

where x is the *state variable* and u the *input* or *control variable*. The optimisation is done under some constraints and the corresponding class of

problems is solvable using the *control parametrisation technique*, which consists in creating a sequence of approximate problems to the original problem and thus transforming the original optimal control problem to a finite dimensional optimisation problem, solvable by optimisation tools such as sequential quadratic programming. A software program called MISER [7] has been developed to solve optimal control problems using control parametrisation technique and sequential quadratic programming. However, some optimal control problems cannot be solved using the control parametrisation method, and cannot be solved directly with the help of MISER. We can mention in particular, optimal control problems involving delay differential equations. Delay differential equations are studied in Chapter 3.

Conclusion

We have laid out some preliminaries on computer networks, flow models, Laplace transforms and optimal control theory. These are the main areas explored by the congestion control we discuss, especially when introducing the flow model control approaches as reviewed in the literature and summarised in Chapter 2.

Chapter 2

Literature Review

Computer networks have experienced significant growth over the past few decades. In the late $1980's$, congestion problems were observed. These problems were of different types: delays in processing, and mainly packet loss. Some network researchers started focusing more on congestion problems, and notably how to try and get rid of such problems.

2.1 Congestion Avoidance

In the late $1980's$, Van Jacobson and Michael Karels [6] contributed in that way by setting up some algorithms that can be used to bring the network to a better behaviour. About seven new congestion control algorithms [6] were developed by then: Round–trip time variance estimation, exponential retransmit timer backoff, slow start, more aggressive receiver ack policy, dynamic window sizing on congestion, Karn's clamped retransmit backoff, fast retransmit. Van Jacobson and Michael Karels focused more on the first five algorithms. This was a good point to start for subsequent studies.

2.2 An Elastic Queue Approach to Congestion Avoidance

H. Sirisena and M. Hassan developed this approach, [4, 14] where the model used is a model of elastic queues in computer networks (the case of one link and one source), with main intention to control the congestion window of the source. The *congestion window* is the amount of information sent through

the link by the source during a round–trip time. The round–trip time [14] is the sum of the link propagation delay with the queueing delay, where the queueing delay can be approximated by the quotient of the queue size and the available bandwidth. The available bandwidth here changes with time. The main idea is to keep the queue size at a reference value and minimise the variance of queue size from the reference value, in order to minimise the probability of buffer overflow or underflow.

Following the approach developed by H. Sirisena and M. Hassan [4, 14], the optimal control problem may be summarised as follows:
Subject to the differential equation

$$\begin{aligned}
&\frac{dq}{dt} = r\left(t - \Delta - \frac{q(t)}{b(t)}\right) - b(t), \qquad\qquad t \in [t_1, T] \\
&q(t_1) = q_1, \\
&t_1 - \Delta - \frac{q_1}{b(t_1)} = 0
\end{aligned}$$

where $q(t)$ is the queue size at time t, Δ is the round–trip propagation delay, $b(t)$ is the bandwidth available to the connection at time t, $\frac{q(t)}{b(t)}$ is the variable queueing delay, $r(t)$ is the transmission rate at time t, we wish to maximise the amount of information transmitted, denoted by

$$\int_0^T r(t)dt.$$

K. H. Wong [18] considered the case where $b(t)$ is constant and used the Pontryagin Maximum Principle[1] to deduce an optimal control. When the bandwidth is not constant, the problem becomes much more challenging.

One of the problems with going further in the direction of elastic queue approach is that there is no software good enough to solve optimal control problems with state–dependent delays. In fact, the software MISER [7] helps to solve (numerically) optimal control problems in canonical form, hence with no delay in the dynamics. However, in some very simple cases, a model transformation can be possible in order to obtain an optimal control problem in canonical form and then use MISER to solve the transformed problem numerically. This uses sophisticated techniques such as the *control parametrisation*. These two challenges, namely the design of an improved software to solve optimal control problems with state–dependent delays, as

[1] This can be found in many documents on Optimal Control Theory.

well as a model transformation and the application of methods such as the control parametrisation when the bandwidth $b(t)$ is not constant, may be an object of further research.

2.3 An Optimisation Approach to Flow Control

In the late $1990's$, Steven Low and David Lapsley [8] proposed an optimisation approach to flow control, where the objective is to maximise the aggregate source utility over their transmission rates. Considering flow models as described in section 1.2, the objective at the beginning is to choose source rates $(x_i)_{i\in\mathcal{S}}$ so as to maximise the network utility, subject to network capacity constraints, that is:

$$\begin{aligned} &\max \sum_{i=1}^{S} U_i(x_i) \\ &\text{subject to} \\ &\sum_{i=1}^{S} R_{li}x_i \leq c_l \qquad \forall l \in \mathcal{L}. \end{aligned} \tag{2.1}$$

This leads to the so called *primal solution* or *primal control* [8]. The dual problem was then investigated. For each link l, let us denote by p_l the price signal or congestion measure, that is, the price per unit bandwidth. The total price per unit bandwidth for all links on the path of source i is $q_i = \sum_{l=1}^{L} R_{li}p_l$. Source i then chooses a transmission rate x_i that maximises its own benefit, that is $U_i(x_i) - q_i x_i$. The quantity $q_i x_i$ is the bandwidth cost. The *Lagrangian* [8] is defined as follows:

$$\begin{aligned} \mathbb{L}(x,p) &= \sum_{i=1}^{S} U_i(x_i) - \sum_{l=1}^{L} p_l \cdot \left(\sum_{i=1}^{S} R_{li}x_i - c_l \right) \\ &= \sum_{i=1}^{S} \left(U_i(x_i) - x_i \sum_{l=1}^{L} R_{li}p_l \right) + \sum_{l=1}^{L} p_l c_l \\ &= \sum_{i=1}^{S} \left(U_i(x_i) - x_i q_i \right) + \sum_{l=1}^{L} p_l c_l. \end{aligned}$$

So the *objective function* is:

$$\begin{aligned}\mathbb{D}(p) &= \max \mathbb{L}(x,p) \\ &= \sum_{i=1}^{S} B_i(q_i) + \sum_{l=1}^{L} p_l c_l\end{aligned}$$

where

$$B_i(q_i) = \max\left(U_i(x_i) - x_i q_i\right)$$

and finally the dual problem is:

$$\min_{p \geq 0} \mathbb{D}(p). \tag{2.2}$$

$B_i(q_i)$ is the maximum benefit source i can achieve at the given price q_i.

The dual problem is solved using a gradient projection algorithm (in synchronous and asynchronous way). Sources select their transmission rates so as to maximise their own individual profit (that is utility minus bandwidth cost) and links adjust their prices (bandwidth prices) to coordinate the sources' decisions. Feedback delays are possibly different from each other and may also change with time. Links and sources can update at different times, and at different rates. The algorithms described illustrate a reactive flow control. From an aggregate source rate per link, (link is given feedback by sources), each link computes its price per unit bandwidth and gives the feedback to sources; the sources adjust their transmission rates so as to maximise their own profit, and the cycle repeats. For the algorithm to converge and yield the optimal rates, the intervals between updates must be bounded.

Solving the dual problem leads to the so called *dual solution* or *dual control*, which is later improved by a *primal–dual control*, and then by a *control based on queueing delay.*

2.4 Some Flow Control Schemes

Some flow control schemes have been developed by Steven Low, Fernando Paganini, John Doyle and Zhikui Wang, inspired by some ideas developed in [8].

2.4.1 Dual Control Laws

In 2001, Fernando Paganini, John Doyle and Steven Low developed a dual control law [9]. In the proposed control law, stability is maintained for arbitrary network topologies and arbitrary delays. The system is implemented in a decentralised way (that is, in a way that the change of rate of each source depends only on present and past rates of the same source). Each source adjusts its transmission rate so that the network resources are almost fully utilised, and link capacities are not exceeded (that is, links are not congested). They consider a network composed of L unidirectional links shared by S sources. Each link l has a capacity c_l, a *target capacity* $\tilde{c}_l$ assumed constant and equal to (or slightly below) the actual link capacity c_l, an aggregate rate of flow y_l, a price signal p_l; and each source i has a transmission rate x_i, an aggregate price q_i of all links used by the source, as given in flow models described in section 1.2. Data for links and sources are given at any time t. The modelling part of the work is done in the Laplace domain and takes into account the delays, that are assumed to be constant. The most important things to specify are:

- how the links fix their prices, based on link utilisation;
- how the sources fix their rates, based on their aggregate price.

Since one of the objectives of the control is to bring the system to an equilibrium, they start by studying dynamic properties around a given equilibrium point of the system. The main objectives of the control are the following:

- the target capacity must be matched at equilibrium;
- to have local dynamic stability for arbitrary network delays, link capacities and routing topologies.

Linear stability has also been studied (in the Laplace domain). A linear control law was proposed as follows:

- for source i, a *gain* κ_i between q_i and x_i given by:

$$-\kappa_i = -\frac{\alpha_i x_i^*}{M_i \tau_i}$$

- for link l, an *integrator* with gain normalised by capacity,

$$p_l = \frac{1}{\tilde{c}_l s} y_l.$$

where x_i^* is the equilibrium value of x_i, α_i a parameter (source gains, assumed positive), M_i an upper bound on the number of bottleneck links on source i's path, s the *Laplace variable* since we are in the Laplace domain.

The proposed linear control law was embedded in a global nonlinear control scheme, in the time domain. The source and link laws were inspired by a more general class of algorithms developed in [8], but now adapted to continuous time. The proposed global nonlinear control scheme is given as follows:

$$\dot{p}_l(t) = \begin{cases} \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l} & \text{if } p_l(t) > 0 \\ \max\left\{0, \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l(t) = 0 \end{cases}$$

$$x_i(t) = x_{\max,i} \exp\left(\frac{-\alpha_i}{M_i \tau_i} q_i(t)\right).$$

Denoting by x_i^* and q_i^* the respective equilibrium values of x_i and q_i, the function f_i such that $x_i = f_i(q_i)$, and therefore $x_i^* = f_i(q_i^*)$, is called a *demand curve* and the *elasticity* of the source demand curve is $\nu_i = -f_i'(q_i^*)$.

2.4.2 Primal–Dual Control Laws

In 2003, the three authors above, with the contribution of Zhikui Wang, reviewed the dual control laws proposed in [9], and proposed a control law [11, 10] that improves on the previous one. In the previous control law (dual control law), the elasticity of the demand curve is constrained by stability, since it is the same as the gain at DC. So to decouple the two gains, it was proposed in the linearised source control law, to replace the equation

$$\delta x_i = -\kappa_i \delta q_i$$

by

$$\delta x_i = -\frac{\kappa_i(s+z)}{s + z\frac{\kappa_i}{\nu_i}} \delta q_i$$

(in the Laplace domain), where

$$\kappa_i = \frac{\alpha_i x_i^*}{M_i \tau_i}$$

is the high frequency gain and $\nu_i = -f_i'(q_i^*)$ is the elasticity of the source demand based on its own demand curve $x_i^* = -f_i(q_i^*)$. The linearised control is embedded in a global nonlinear control, so as for the equilibrium to match

the desired utility function, that is, $U_i'(x_i^*) = q_i^*$, or equivalently, $f_i = (U_i')^{-1}$. The proposed global nonlinear laws are given as follows:

$$\dot{p}_l(t) = \begin{cases} \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l} & \text{if } p_l(t) > 0 \\ \max\left\{0, \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l(t) = 0 \end{cases}$$

$$x_i(t) = x_{m,i} \exp\left(\xi_i(t) - \frac{\alpha_i}{M_i \tau_i} q_i(t)\right)$$

$$\dot{\xi}_i(t) = \frac{\beta_i}{\tau_i}\big(U_i'(x_i(t)) - q_i(t)\big)$$

where β_i is a parameter.

With the proposed nonlinear control laws, there is an enhanced fairness among sources sharing the same links, and less constraints on the elasticity of demand curves. With these control laws, the links are almost fully utilised and not congested since the queues are empty (at equilibrium).

2.4.3 Control Laws Based on Queueing Delay

Without some form of *Explicit Congestion Notification*, congestion can only be inferred through some of its effects such as queueing delays, and the objective is now to make them as small as possible, and possibly bring them to zero. So a source protocol has to be found so that:

- the prices at equilibrium (the price is the same as the delay) are small;
- there is some freedom of choice in source allocation;
- and stability is achieved.

In this case, the target capacity is the same as the actual capacity for each link in the network. A modified control law is proposed [11, 10], taking into account the queueing delays. The (global nonlinear) control law is formulated as follows:

$$\dot{p}_l(t) = \begin{cases} \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l} & \text{if } p_l(t) > 0 \\ \max\left\{0, \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l(t) = 0 \end{cases}$$

$$x_i(t) = x_{m,i} \exp\left(\xi_i(t)\right) \left(\frac{d_i}{d_i + q_i(t)}\right)^{\frac{\alpha_i}{M_i}}$$

$$\dot{\xi}_i(t) = \left(\frac{\beta_i}{d_i + q_i(t)}\right)\big(U_i'(x_i(t)) - q_i(t)\big)$$

with $x_{m,i}$ as in the primal–dual control case.

2.5 Stability Analysis of Flow Control Schemes

In 2004, Antonis Papachristodoulou, John Doyle and Steven Low [13] consider the dual and primal–dual control laws, and the stability analysis developed in [15, 16] based on Lyapunov–Razumikhin functions, and more details on some Lyapunov functions as developed in other references. They use the Sums Of Squares (SOS) and semidefinite programming, to check the stability conditions in an algorithmic way. They also give conditions for stability, asymptotic stability, robust stability, uniform asymptotic stability. The stability of the dual control is investigated for a single link (or bottleneck) used by one or many sources. The stability of the primal–dual control is investigated for two cases: single link and single source, two links and three sources.

Chapter 3

An Overview of Delay Differential Equations

3.1 An Introduction

Many phenomena are modelled by ordinary differential equations (*ODEs*), that is, equations of the form

$$\dot{x}(t) = f\big(t, x(t)\big).$$

In other words, *ODEs* are equations in which the change of state depends on time and present state. For example, population growth can be modelled by the following *ODE*

$$\dot{x}(t) = k \cdot x(t) \tag{3.1}$$

where k denotes the growth rate. The solution of the above ODE is well known to be of the form

$$x(t) = A \cdot e^{kt} \quad \text{for some real constant } A.$$

If we take into account the fact that an individual is not able to reproduce as soon as he is born, we can consider the time τ it takes for an individual to be able to reproduce. Then equation (3.1) becomes

$$\dot{x}(t) = k \cdot x(t - \tau)$$

which is a particular case of a delay differential equation (DDE). A *delay differential equation (DDE)* is an equation in which the change of state

depends on time, present state and past states. We will be working with *DDEs* of the form

$$\dot{x}(t) = f\big(t, x(t), x(t-\tau)\big)$$

where $\tau > 0$ is the *delay*, and $x(t) \in \mathbb{R}^d$ is the state at time t with d a positive integer.

- τ can be constant, then we have a *constant delay differential equation*.
- τ can depend only on time t, then we have a *time–dependent delay differential equation*.
- τ can depend on the state $x(t)$, then we have a *state–dependent delay differential equation*.
- τ can depend both on time t and state $x(t)$, then we have a *time – and state–dependent delay differential equation*.

Some ODEs can be solved analytically and most of them can be solved numerically. Some numerical methods for solving ODEs are: forward Euler's method, Runge–Kutta methods, Heun's method. Some of those methods can be adapted to find the numerical solutions of delay differential equations. Most delay differential equations can only be solved numerically.

3.2 Solving Delay Differential Equations

Suppose that we have to solve for example the equation $\dot{x}(t) = -x(t-1)$. To have the solution at time $t = 0$ say, we need to have the solution at time $t = -1$; and to know the solution at time $t = 1$ we need to know the solution at time $t = 0$; therefore for a complete solution from time $t = 0$ to time $t = 1$ we need to have the whole initial solution from time $t = -1$ to time $t = 0$. Thus, such an initial solution has to be given. We call the initial solution $\Phi(t)$ say, and set for example $\Phi(t) = 1$ for $t \leq 0$. The function Φ is called the *tail function* for the delay differential equation. Now we can solve the equation for $t \in [0, 1]$; we then obtain x as a linear function of t, precisely $x(t) = -t + 1$; we can use this information to progress forward and solve the equation for $t \in [1, 2]$; we obtain x as a quadratic function of t, precisely $x(t) = \frac{1}{2}t^2 - t + \frac{1}{2}$; repeating the same process and solving the equation for $x \in [2, 3]$, we obtain $x(t) = -\frac{1}{6}t^3 + \frac{1}{2}t^2 - \frac{1}{2}t + \frac{5}{6}$; we could

iterate this process, solving the DDE as an infinite series of ODEs. In each interval $[n, n+1]$ (with $n \in \mathbb{N}$), we have a polynomial of degree $n+1$ as an approximation to the solution of the DDE. We then obtain a function which is continuous with respect to t, but not differentiable at $t = 0$. Reconsidering the equation $\dot{x}(t) = -x(t-1)$ we want to solve, it can be established that $x^{(n)}(t) = (-1)^n \cdot x(t-n)$, therefore $x^{(n+1)}(t) = (-1)^n \cdot \dot{x}(t-n)$ where $x^{(n)}$ stands for the n^{th} derivative of the function x, with respect to t. And since x is not differentiable at $t = 0$, we can deduce that the derivative of order $n+1$, when it is defined, is discontinuous at $t = n$, in general. We have described above a way to give an analytical approximation to the solution of the delay differential equation in the above example. It is indeed, as illustrated in Figure 3.1, a very poor approximation. A more accurate way

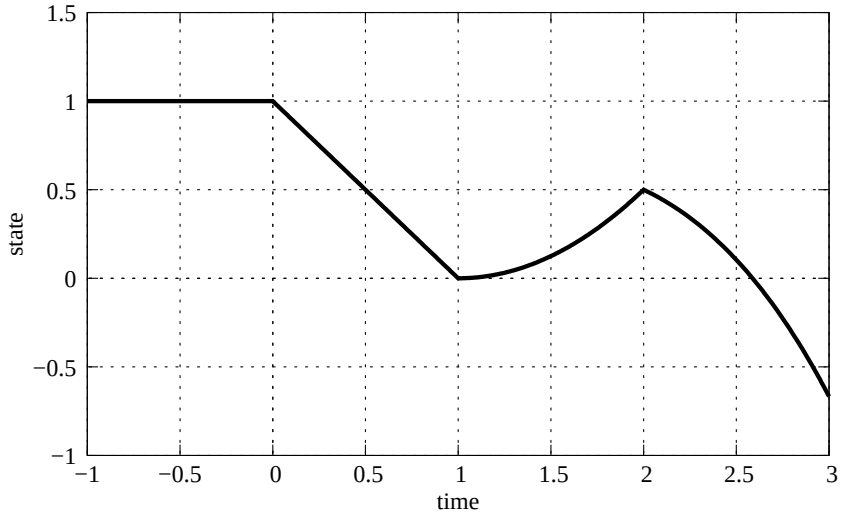


Figure 3.1: Poor approximation to the solution of a DDE.

is to use the above method but with intervals of smaller magnitude. The smaller the magnitude of the interval, the more accurate the solution is. We need to know the maximum value τ_M of the delay for all $t > 0$, so that the tail function Φ will be defined in $(-\tau_M, 0]$. If $\tau_M = +\infty$, the tail function Φ will be defined in $(-\infty, 0]$. Then we solve from time $t = 0$ the infinite series of ODEs, on intervals as small as possible. But this is not the best way to solve the given DDE, at all. When the delay is not necessarily constant, the above method does not work as well as we would like. Therefore more sophisticated numerical methods are needed.

3.2.1 Existence and Uniqueness of Solutions

We consider equations of the type

$$\dot{x}(t) = f\left(t, x(t), x\left(t-\tau\right)\right)$$

where for any time $t \in \mathbb{R}$, $x(t) \in \mathbb{R}^d$, with d a positive integer not depending on t; $\tau > 0$ is the delay. A better formulation of the above equation is as follows:

$$\begin{cases} \dot{x}(t) = f\left(t, x(t), x\left(t-\tau\right)\right) & \text{if } t_0 \leq t < t_f \\ x(t) = \Phi(t) & \text{if } t \leq t_0 \end{cases}$$

where t_0 is the initial time, t_f the final time and Φ the tail function.

The existence and uniqueness of solutions to this equation is subject to some conditions, as given in the statements to follow (more details can be found in [1]). For the time–dependent delay differential equation case, that is, when $\tau = \tau(t)$, conditions on local and global existence of solutions are given as follows:

Theorem 3.1 (Local existence)**.** *Consider the problem*

$$\begin{cases} \dot{x}(t) = f\left(t, x(t), x\left(t-\tau(t)\right)\right) & \text{if } t_0 \leq t < t_f \\ x(t_0) = x_0 \end{cases} \tag{3.2}$$

and assume that the function $f(t,u,v)$ is continuous on $A \subseteq [t_0, t_f) \times \mathbb{R}^d \times \mathbb{R}^d$ and locally Lipschitz continuous with respect to u and v. Moreover, assume that the delay function $\tau(t) \geq 0$ is continuous on $[t_0, t_f)$, $\tau(t_0) = 0$ and, for some $\xi > 0$, $t - \tau(t) > t_0$ in the interval $(t_0, t_0 + \xi]$. Then the problem 3.2 has a unique solution in $[t_0, t_0 + \delta)$ for some $\delta > 0$ and this solution depends continuously on the initial data.

Theorem 3.2 (Global existence)**.** *If under the hypotheses of Theorem 3.1, the unique maximal solution of problem 3.2 is bounded, then it exists on the entire interval $[t_0, t_f)$.*

Corollary 3.3. *Besides the hypotheses of Theorem 3.1, assume that the function $f(t,u,v)$ satisfies the condition*

$$\|f(t,u,v)\| \leq M(t) + N(t)\left(\|u\| + \|v\|\right)$$

in $[t_0, t_f) \times \mathbb{R}^d \times \mathbb{R}^d$, where $M(t)$ and $N(t)$ are continuous positive functions on $[t_0, t_f)$. Then the solution of problem 3.2 exists and is unique on the entire interval $[t_0, t_f)$.

The time – and state–dependent delay equation case is a more general case, and more restrictive than the previous case. However, we have conditions on local existence of solutions as given in the following theorem:

Theorem 3.4 (Local existence)**.** *Consider the problem*

$$\begin{cases} \dot{x}(t) = f\left(t, x(t), x\left(t - \tau(t, x(t))\right)\right) & \text{if } t \geq t_0 \\ x(t) = \Phi(t) & \text{if } t \leq t_0. \end{cases} \tag{3.3}$$

Let $U \subseteq \mathbb{R}^d$ and $V \subseteq \mathbb{R}^d$ be neighbourhoods of $\Phi(t_0)$ and $\Phi(t_0 - \tau(t_0, \Phi(t_0)))$, respectively, and assume that the function $f(t, u, v)$ is continuous with respect to t and Lipschitz continuous with respect to u and v in $[t_0, t_0 + h] \times U \times V$ for some $h > 0$. Moreover, assume that the initial function $\Phi(t)$ is Lipschitz continuous for $t \leq t_0$ and that the delay function $\tau(t, y) \geq 0$ is continuous with respect to t and Lipschitz continuous with respect to y in $[t_0, t_0 + h] \times U$. Then problem 3.3 has a unique solution in $[t_0, t_0 + \delta)$ for some $\delta > 0$ and this solution depends continuously on the initial data.

3.2.2 Euler's Method for Delay Differential Equations

We now consider the delay differential equation

$$\begin{cases} \dot{x}(t) = f\left(t, x(t), x\left(t - \tau(t, x(t))\right)\right) & \text{if } t \in [t_0, t_f) \\ x(t) = \Phi(t) & \text{if } t \leq t_0. \end{cases}$$

where for any time $t \in \mathbb{R}$, $x(t) \in \mathbb{R}$ and $\tau(t, x(t)) \geq 0$. To solve this equation numerically (using the *forward Euler's method*), we divide the time interval $[t_0, t_f)$ into small intervals $[t_n, t_{n+1})_{0 \leq n < N}$ so that $t_0 < t_1 < \ldots < t_N = t_f$, and $t_{n+1} - t_n = h$ for all $n \in \{0, 1, \ldots, N-1\}$, where h is a small positive real number. At any time t_n, we approximate $x(t)$ by x_n and $x(t - \tau(t, x(t)))$ by y_n. Considering the fact that at the beginning we have $x_0 = \Phi(t_0)$ where Φ is the tail function, x_{n+1} is deduced from x_n by the following relation:

$$x_{n+1} = x_n + f(t_n, x_n, y_n)h.$$

The algorithm is as follows: take a small positive real number h, then follow the steps below

$$\begin{aligned} x_0 &= \Phi(t_0) \\ t_n &= t_0 + nh \\ l_n &= t_n - \tau(t_n, x_n) \\ k_n &= \left[\frac{l_n}{h}\right] \\ y_n &= \begin{cases} \Phi(l_n) & l_n \leq t_0 \\ \frac{x_{k_n+1} - x_{k_n}}{h}(l_n - k_n h) + x_{k_n} & l_n > t_0 \end{cases} \\ x_{n+1} &= x_n + f(t_n, x_n, y_n)h \end{aligned} \tag{3.4}$$

where for a real number x, $[x]$ stands for the largest integer less than or equal to x.

Remark 3.5. The above method is mentioned in [2] as the *first order method*.

3.2.3 Heun's Method for Delay Differential Equations

We reconsider the delay differential equation

$$\begin{cases} \dot{x}(t) = f\left(t, x(t), x\left(t - \tau(t, x(t))\right)\right) & \text{if } t \in [t_0, t_f) \\ x(t) = \Phi(t) & \text{if } t \leq t_0. \end{cases}$$

where for any time $t \in \mathbb{R}$, $x(t) \in \mathbb{R}$ and $\tau(t, x(t)) \geq 0$. To solve this equation numerically (using the *Heun's method*), we divide the time interval $[t_0, t_f)$ into small intervals $[t_n, t_{n+1})_{0 \leq n < N}$ so that $t_0 < t_1 < \ldots < t_N = t_f$, and $t_{n+1} - t_n = h$ for all $n \in \{0, 1, \ldots, N-1\}$, where h is a small positive real number. At any time t_n, we approximate $x(t)$ by x_n and $x(t - \tau(t, x(t)))$ by y_n; we also give respective predictions (or poor approximations) $\tilde{x}_{n+1}$ to x_{n+1}, and $\tilde{y}_{n+1}$ to y_{n+1}. Considering the fact that at the beginning we have $x_0 = \Phi(t_0)$ where Φ is the tail function, x_{n+1} is deduced from x_n by the following relation:

$$x_{n+1} = x_n + \Big(f\left(t_n, x_n, y_n\right) + f\left(t_{n+1}, \tilde{x}_{n+1}, \tilde{y}_{n+1}\right)\Big)\frac{h}{2}.$$

The algorithm is as follows [2]: take a small positive real number h, then follow the steps below

$$
\begin{aligned}
x_0 &= \Phi(t_0) \\
t_n &= t_0 + nh \\
l_n &= t_n - \tau(t_n, x_n) \\
k_n &= \left[\frac{l_n}{h}\right] \\
y_n &= \begin{cases} \Phi(l_n) & l_n \le t_0 \\ \frac{x_{k_n+1} - x_{k_n}}{h}(l_n - k_n h) + x_{k_n} & l_n > t_0 \end{cases} \\
\tilde{x}_{n+1} &= x_n + f\left(t_n, x_n, y_n\right) h \\
\tilde{l}_{n+1} &= t_{n+1} - \tau(t_{n+1}, \tilde{x}_{n+1}) \\
\tilde{k}_{n+1} &= \left[\frac{\tilde{l}_{n+1}}{h}\right] \\
\tilde{y}_{n+1} &= \begin{cases} \Phi(\tilde{l}_{n+1}) & \tilde{l}_{n+1} \le t_0 \\ \frac{x_{\tilde{k}_{n+1}+1} - x_{\tilde{k}_{n+1}}}{h}(\tilde{l}_{n+1} - \tilde{k}_{n+1} h) + x_{\tilde{k}_{n+1}} & t_0 < \tilde{l}_{n+1} \le t_n \\ \frac{\tilde{x}_{n+1} - x_n}{h}(\tilde{l}_{n+1} - t_n h) + x_n & t_n < \tilde{l}_{n+1} < t_{n+1} \\ \tilde{x}_{n+1} & \tilde{l}_{n+1} = t_{n+1} \end{cases} \\
x_{n+1} &= x_n + \Big(f\left(t_n, x_n, y_n\right) + f\left(t_{n+1}, \tilde{x}_{n+1}, \tilde{y}_{n+1}\right)\Big)\frac{h}{2}
\end{aligned}
\tag{3.5}
$$

where for a real number x, $[x]$ stands for the largest integer less than or equal to x.

Remark 3.6. The accuracy of the solutions to a delay differential equation, with respect to the time step h and the method (forward Euler's method or Heun's method) has also been investigated in [2] and illustrated using some numerical results on some test problems. It has been established that with both methods, the smaller the time step h, the more accurate the solutions are; and in all the cases, Heun's method is much more accurate than the forward Euler's method.

Remark 3.7. Apart from the forward Euler's method and Heun's method for delay differential equations, *Runge–Kutta methods* and *Predictor Error Corrector methods* for delay differential equations are also investigated in [1].

3.3 Illustrative Examples

Example 1

The first example is Figure 3.2 that illustrates the difference between ordinary and delay differential equations, where in this example the ordinary differential equation to be solved is $\dot{x}(t) = -x(t)$ and the delay differential equation is $\dot{x}(t) = -x(t-1)$, with $x(t) = 1$ for $t \leq 0$ in both cases:

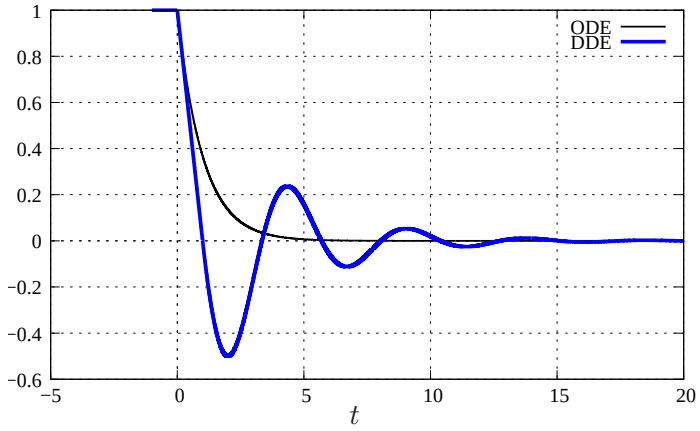


Figure 3.2: Ordinary and delay differential equations 1

The solution of the ordinary differential equation is well known to be

$$x(t) = e^{-t}.$$

Both equations were solved numerically using the (second order) Heun's method. For both cases we took $t_0 = 0$, $t_f = 20$, $\tau(t, x(t)) = 1$, $h = 0.01$; for the ordinary differential equation, we have $f_o(t, u, v) = -u$ and for the delay differential equation, we have $f_d(t, u, v) = -v$. It can be checked that all the conditions of Theorem 3.4 hold, and therefore the local existence (and uniqueness) of a solution for each of the two differential equations is guaranteed on some interval $[0, \delta)$ for some $\delta > 0$. Though we don't know the analytical expression of the solution to equation $\dot{x}(t) = -x(t-1)$, according to the graphs in Figure 3.2, we notice that the graph of the solution to the delay differential equation oscillates around the graph of the solution to the ordinary differential equation, and the amplitude of the oscillation decreases to zero as t goes to infinity.

Example 2

The second example is Figure 3.3 where the ordinary differential equation is still $\dot{x}(t) = -x(t)$ and the delay differential equation is $\dot{x}(t) = -x(t-2)$, with $x(t) = 1$ for $t \leq 0$ in both cases:

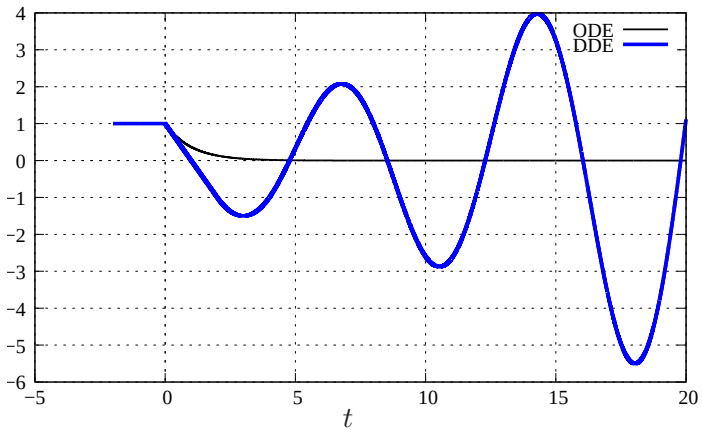


Figure 3.3: Ordinary and delay differential equations 2

The solution of the ordinary differential equation, as mentioned earlier, is well known to be

$$x(t) = e^{-t}.$$

Both equations were solved numerically using the (second order) Heun's method. For both cases we took $t_0 = 0$, $t_f = 20$, $\tau(t, x(t)) = 2$, $h = 0.01$. It can also be checked that all the conditions of Theorem 3.4 hold, and therefore the local existence (and uniqueness) of a solution for each of the two differential equations is guaranteed on some interval $[0, \delta)$ for some $\delta > 0$. Though we don't know the analytical expression of the solution to equation $\dot{x}(t) = -x(t-2)$, according to the graphs in Figure 3.3, we notice that the graph of the solution to the delay differential equation oscillates around the graph of the solution to the ordinary differential equation, and the amplitude of the oscillation goes to infinity as t goes to infinity.

Example 3

The third example is Figure 3.4 where the ordinary differential equation is again $\dot{x}(t) = -x(t)$ and the delay differential equation is now $\dot{x}(t) =$

$-x\left(t-\frac{\pi}{2}\right)$, with $x(t)=1$ for $t\leq 0$ in both cases:

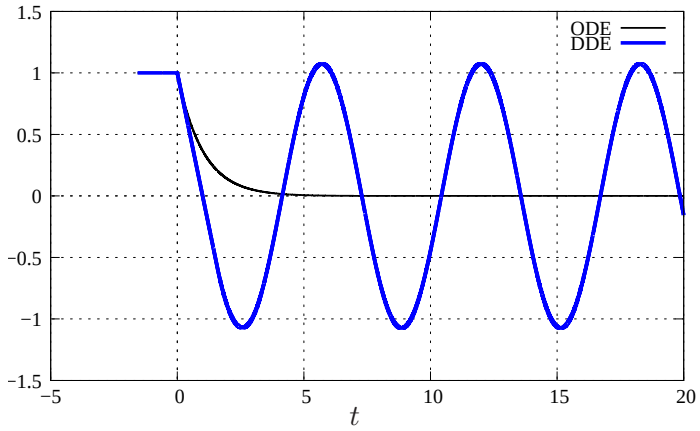


Figure 3.4: Ordinary and delay differential equations 3

The solution of the ordinary differential equation, as mentioned earlier, is

$$x(t)=e^{-t}.$$

Both equations were solved numerically using the (second order) Heun's method. For both cases we took $t_0=0$, $t_f=20$, $\tau(t,x(t))=\frac{\pi}{2}$, $h=0.01$. Any function x such that $x(t)=a\cos(t+\phi)$ where a and ϕ are real numbers with $a\cos(\phi)=1$ (since $x(0)=1$), is a solution of equation $\dot{x}(t)=-x(t-\frac{\pi}{2})$. The solution is not unique in this case, but according to the graphs in Figure 3.4, we can notice that the graph of the solution to the delay differential equation oscillates around the graph of the solution to the ordinary differential equation, and the amplitude of the oscillation is constant.

Conclusion

The three examples above show that adding a delay to an ordinary differential equation can completely change the behaviour of the solution. Generally, solutions to delay differential equations (provided they exist) have an unknown analytical expression.

3.4 Remark

The methods described in the section above can be extended to a delay differential equation with several delays [2], and can also be extended to a system of delay differential equations with one or many delays [3].

Conclusion

We have introduced the concept of delay differential equations, discussed an intuitive way to solve delay differential equations which is not indeed reasonable. We have reviewed some sufficient conditions for the existence and uniqueness of solutions for some delay differential equations, described some numerical methods that can be used to solve them, precisely the forward Euler's method and the modified Heun's method. We have also discussed using some examples, some differences between ordinary and delay differential equations.

Chapter 4

Exploration of Some Control Schemes

We consider flow models as described in section 1.2 and take all our minimum source rates $x_{\min,i}, i \in \mathcal{S}$ to be zero. We also consider the optimal control problem as given in [8], and reviewed in section 2.3.

This leads, as mentioned earlier, to the so called *dual solution* or *dual control*, which is later improved by a *primal–dual control*, and then by a *control based on queueing delay*. We recall the dynamics governing flow models described in section 1.2 for flow rates, link prices and round–trip time:

$$y_l(t) = \sum_{i=1}^{S} R_{li} x_i \left(t - \tau_{li}^f\right) \tag{4.1}$$

$$q_i(t) = \sum_{l=1}^{L} R_{li} p_l \left(t - \tau_{li}^b\right) \tag{4.2}$$

$$\tau_i = \sum_{l=1}^{L} R_{li} \left(\tau_{li}^f + \tau_{li}^b\right) \tag{4.3}$$

where the R_{li} are the components of the routing matrix.

4.1 The Dual Control Approach

4.1.1 Formulation

As proposed in [9] (for link prices and source rates), the dynamics governing the link prices, source rates and buffer sizes are given as follows, for each

$l \in \mathcal{L}$ and for each $i \in \mathcal{S}$:

$$\dot{p}_l(t) = \begin{cases} \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l} & \text{if } p_l(t) > 0 \\ \max\left\{0, \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l(t) = 0 \end{cases} \tag{4.4}$$

$$x_i(t) = x_{\max,i} \exp\left(\frac{-\alpha_i}{M_i \tau_i} q_i(t)\right) \tag{4.5}$$

$$\dot{b}_l(t) = \begin{cases} y_l(t) - c_l & \text{if } b_l(t) > 0 \\ \max\{0, y_l(t) - c_l\} & \text{if } b_l(t) = 0 \end{cases} \tag{4.6}$$

where M_i is an upper bound on the number of bottleneck links in the source's path, $x_{\max,i}$ the maximum transmission rate for source i, and $\tilde{c}_l$ the *target capacity* of link l, slightly below its actual capacity c_l. We shall also mention that the computation of the buffer sizes follows immediately from the source rates and link prices.

The above dynamics can be reformulated as follows:

$$\dot{p}_l(t) = \begin{cases} \lambda_l(t) & \text{if } p_l(t) > 0 \\ \max\{0, \lambda_l(t)\} & \text{if } p_l(t) = 0 \end{cases} \tag{4.7}$$

$$x_i(t) = x_{\max,i} \exp\left(\frac{-\alpha_i}{M_i \tau_i} \sum_{l=1}^{L} R_{li} p_l\left(t - \tau_{li}^b\right)\right) \tag{4.8}$$

$$\dot{b}_l(t) = \begin{cases} \mu_l(t) & \text{if } b_l(t) > 0 \\ \max\{0, \mu_l(t)\} & \text{if } b_l(t) = 0 \end{cases} \tag{4.9}$$

$$\lambda_l(t) = \sum_{i=1}^{S} \frac{R_{li}}{\tilde{c}_l} x_{\max,i} \exp\left(\frac{-\alpha_i}{M_i \tau_i} \sum_{m=1}^{L} R_{mi} p_m\left(t - \tau_{li}^f - \tau_{mi}^b\right)\right) - 1 \tag{4.10}$$

$$\mu_l(t) = \sum_{i=1}^{S} R_{li} x_{\max,i} \exp\left(\frac{-\alpha_i}{M_i \tau_i} \sum_{m=1}^{L} R_{mi} p_m\left(t - \tau_{li}^f - \tau_{mi}^b\right)\right) - c_l. \tag{4.11}$$

From the formulation of λ_l in equation (4.10) we can observe that equation (4.7) is a nonlinear delay differential equation and the computation of the link prices p_l at any time t determines the remaining dynamics in the network.

Remark 4.1. For all the links $l \in \mathcal{L}$ in our network to be bottlenecks, it is sufficient for us to assume the following condition:

$$\sum_{i=1}^{S} R_{li} x_{\max,i} > c_l \quad \text{for each link } l \in \mathcal{L}.$$

And since we assumed that the target capacity $\tilde{c}_l$ is (slightly) below the actual capacity c_l, it follows that for each link $l \in \mathcal{L}$,

$$\sum_{i=1}^{S} R_{li} x_{\max,i} > \tilde{c}_l.$$

Remark 4.2. For the system described above in equations (4.1) and (4.2), at equilibrium, for each link l, we have $\tilde{c}_l = y_l^*$, the equilibrium value of y_l; and therefore if $\dot{b}_l \neq 0$, then $\dot{b}_l = \tilde{c}_l - c_l$. So if we take $\tilde{c}_l = \theta c_l$ with $\theta \in (0,1)$ and θ close to 1, then $\dot{b}_l = (\theta - 1)c_l$ provided that $\dot{b}_l \neq 0$. It can be checked (through simulations) that for each link l, there exists a $\theta_0 = \theta_0(l) \in (0,1)$ such that for $\tilde{c}_l = \theta_0 c_l$, at equilibrium, the queues are empty or contain a small number of packets. So for $\theta \in (0,1)$, $\theta > \theta_0$, (for example $\theta = \frac{9+\theta_0}{10}$ or $\theta = \frac{3+\theta_0}{4}$), the queues when the system just reaches equilibrium, will be nonzero. So for $\tilde{c}_l = \theta c_l$ with $\theta > \theta_0$, we will have $\dot{b}_l = (\theta - 1)c_l < 0$ and queues b_l will decrease linearly to zero, with rate $(\theta - 1)c_l$. We will then have a sort of *two phase equilibrium*. This is the most likely to happen (with a network composed of many links) since the links do not necessarily have all the same capacity, or the same queue size all the times, especially when the system just reaches equilibrium. Another fact to mention is that at any time t_l^*, $l \in \mathcal{L}$, when the queue size b_l goes to zero, the behaviour of the system is not necessarily smooth.

4.1.2 Stability Analysis

Denoting by $\bar{R}$ the *reduced routing matrix* obtained by eliminating non–bottleneck links [9, 15, 13], the following theorem gives sufficient conditions for local stability of the dual control law.

Theorem 4.3 (Local Stability)**.** *[9, 15, 13] If $\bar{R}$ is of full row rank and $\alpha_i < \frac{\pi}{2}$, then the system described by equations (4.1), (4.2), (4.4), and (4.5) is linearly stable for arbitrary delays and link capacities.*

Analysis of some simple cases for the dual control law

Single source, single bottleneck In this case, $S = L = 1$, therefore $R = 1$. Also, $M_i = 1$. So, Equation (4.1) becomes

$$y(t) = x(t - \tau^f) \tag{4.12}$$

where τ^f is the forward delay from the source to the link; equation (4.2) becomes

$$q(t) = p(t - \tau^b) \tag{4.13}$$

where τ^b is the backward delay from the link to the source; equation (4.7) is maintained (with the subscript l dropped) while equation (4.10) becomes

$$\lambda(t) = \frac{x_{\max}}{\tilde{c}} \exp\left(\frac{-\alpha}{\tau} p(t - \tau)\right) - 1 \tag{4.14}$$

and equation (4.5) becomes

$$x(t) = x_{\max} \exp\left(\frac{-\alpha}{\tau} p(t - \tau^b)\right) \tag{4.15}$$

Performing the change of variable

$$z(t) = \frac{x_{\max}}{\tilde{c}} \exp\left(\frac{-\alpha}{\tau} q(t)\right) - 1$$

that is,

$$z(t) = \frac{x_{\max}}{\tilde{c}} \exp\left(\frac{-\alpha}{\tau} p(t - \tau^b)\right) - 1$$

we obtain [13]

$$\begin{aligned} \dot{z}(t) &= \frac{-\alpha}{\tau} \frac{x_{\max}}{\tilde{c}} \exp\left(\frac{-\alpha}{\tau} p(t - \tau^b)\right) \dot{p}(t - \tau^b) \\ &= \frac{-\alpha}{\tau} [1 + z(t)] z(t - \tau). \end{aligned}$$

with

$$-1 \leq z(t) \leq -1 + x_{\max}.$$

Linearising around the *zero* equilibrium gives

$$\dot{z}(t) = \frac{-\alpha}{\tau} z(t - \tau)$$

and the system is locally stable for $\alpha < \frac{\pi}{2}$. [13]

Many sources, single bottleneck In this case, for all $i \in \mathcal{S}$, $R_{li} = 1$ and $M_i = 1$; equation (4.1) becomes

$$y(t) = \sum_{i=1}^{S} x_i(t - \tau_i^f) \tag{4.16}$$

where τ_i^f is the forward delay from source i to the link; equation (4.2) becomes

$$q_i(t) = p(t - \tau_i^b) \tag{4.17}$$

where τ_i^b is the backward delay from the link to source i; equation (4.7) is maintained while equation (4.10) becomes

$$\lambda(t) = \sum_{i=1}^{S} \frac{x_{\max,i}}{\tilde{c}} \exp\left(\frac{-\alpha_i}{\tau_i} p(t - \tau_i)\right) - 1 \tag{4.18}$$

and equation (4.8) becomes

$$x_i(t) = x_{\max,i} \exp\left(\frac{-\alpha_i}{\tau_i} p(t - \tau_i^b)\right). \tag{4.19}$$

We then perform the change of variable

$$z_i(t) = \frac{x_{\max,i}}{\tilde{c}} \exp\left(\frac{-\alpha_i}{\tau_i} p(t)\right) - \beta_i$$

where $\beta_i > 0$ for all $i \in \{1, 2, \ldots, S\}$ and $\sum_{i=1}^{S} \beta_i = 1$. This condition on the β_i is less restrictive than the one used in [13], notably $\beta_i = \frac{1}{S}$ for all $i \in \{1, 2, \ldots, S\}$. We then obtain

$$\begin{aligned}
\dot{z}_i(t) &= \frac{x_{\max,i}}{\tilde{c}} \exp\left(\frac{-\alpha_i}{\tau_i} p(t)\right) \left(\frac{-\alpha_i}{\tau_i} \dot{p}(t)\right) \\
&= \frac{-\alpha_i}{\tau_i} [z_i(t) + \beta_i] \left(\sum_{j=1}^{S} \frac{x_{\max,j}}{\tilde{c}} \exp\left(\frac{-\alpha_j}{\tau_j} p(t - \tau_j)\right) - 1\right) \\
&= \frac{-\alpha_i}{\tau_i} [z_i(t) + \beta_i] \sum_{j=1}^{S} \left(\frac{x_{\max,j}}{\tilde{c}} \exp\left(\frac{-\alpha_j}{\tau_j} p(t - \tau_j)\right) - \beta_j\right) \\
\dot{z}_i(t) &= \frac{-\alpha_i}{\tau_i} [z_i(t) + \beta_i] \sum_{j=1}^{S} z_j(t - \tau_j).
\end{aligned}$$

with

$$-\beta_i \leq z(t) \leq -\beta_i + x_{\max,i}.$$

Under the conditions (equality constraints)

$$\left(\frac{z_i(t)}{\beta_i} + 1\right)^{\tau_i} = \left(\frac{z_j(t)}{\beta_j} + 1\right)^{\tau_j},$$

which constraints are weaker than those used in [13], linearising about equilibrium $z_i = 0 \quad \forall i \in \{1, 2, \ldots, S\}$, we have

$$1 + \frac{\tau_i}{\beta_i} z_i(t) = 1 + \frac{\tau_j}{\beta_j} z_j(t)$$

and therefore

$$z_j(t) = \frac{\tau_i}{\tau_j} \frac{\beta_j}{\beta_i} z_i(t).$$

Hence,

$$\begin{aligned}
\dot{z}_i(t) &= \frac{-\alpha_i}{\tau_i}[z_i(t) + \beta_i] \sum_{j=1}^{S} z_j(t - \tau_j) \\
&= \frac{-\alpha_i}{\tau_i}[z_i(t) + \beta_i] \sum_{j=1}^{S} \frac{\tau_i}{\tau_j} \frac{\beta_j}{\beta_i} z_i(t - \tau_j) \\
\dot{z}_i(t) &= \frac{-\alpha_i}{\beta_i}[z_i(t) + \beta_i] \sum_{j=1}^{S} \frac{\beta_j}{\tau_j} z_i(t - \tau_j).
\end{aligned}$$

And around equilibrium,

$$\dot{z}_i(t) = -\alpha_i \sum_{j=1}^{S} \frac{\beta_j}{\tau_j} z_i(t - \tau_j).$$

Hence for the particular case of two sources (on one link) we have

$$\begin{aligned}
\dot{z}_1(t) &= -\alpha_1 \frac{\beta_1}{\tau_1} z_1(t - \tau_1) - \alpha_1 \frac{\beta_2}{\tau_2} z_1(t - \tau_2); \\
\dot{z}_2(t) &= -\alpha_2 \frac{\beta_1}{\tau_1} z_2(t - \tau_1) - \alpha_2 \frac{\beta_2}{\tau_2} z_2(t - \tau_2).
\end{aligned} \tag{4.20}$$

The following statement gives sufficient conditions for asymptotic stability of the system given by equation (4.20):

Proposition 4.4. *[13] The trivial solution of*

$$\dot{x}(t) = -a_1 x(t - \tau_1) - a_2 x(t - \tau_2)$$

is asymptotically stable if $a_1\tau_1 + a_2\tau_2 < \frac{\pi}{2}$.

The system given by equation (4.20) is asymptotically stable if

$$\begin{cases}
\alpha_1\beta_1 + \alpha_1\beta_2 < \frac{\pi}{2} \\
\alpha_2\beta_1 + \alpha_2\beta_2 < \frac{\pi}{2}
\end{cases}$$

that is

$$\begin{cases} \alpha_1(\beta_1+\beta_2) < \frac{\pi}{2} \\ \alpha_2(\beta_1+\beta_2) < \frac{\pi}{2} \end{cases}$$

and since $\beta_1+\beta_2=1$, this means that the system is asymptotically stable if

$$\begin{cases} \alpha_1 < \frac{\pi}{2}; \\ \alpha_2 < \frac{\pi}{2}. \end{cases}$$

Beyond the local stability, some results on global stability have been proved in the case of a single link network. Denoting by p the link price and by $\tilde{c}$ the target link capacity, the price dynamics is given by:

$$\dot{p}(t) = \begin{cases} \frac{y(t)-\tilde{c}}{\tilde{c}} & \text{if } p(t) > 0 \\ \max\left\{0, \frac{y(t)-\tilde{c}}{\tilde{c}}\right\} & \text{if } p(t) = 0. \end{cases}$$

And at equilibrium, from $\dot{p}(t)=0$, we have

$$\sum_{i=1}^{S} \frac{x_{\max,i}}{\tilde{c}} \exp\left(\frac{-\alpha_i}{\tau_i} p^*\right) = 1 \tag{4.21}$$

where p^* is the equilibrium value of p. Then for $\delta p(t) = p(t) - p^*$, we have

$$\delta\dot{p}(t) = \sum_{i=1}^{S} \kappa_i \left(-1 + \exp\left(\frac{-\alpha_i}{\tau_i}\delta p(t-\tau_i)\right)\right) \tag{4.22}$$

where

$$\kappa_i = \frac{x_{\max,i}}{\tilde{c}} \exp\left(\frac{-\alpha_i}{\tau_i} p^*\right) \quad \text{and from above,} \quad \sum_{i=1}^{S} \kappa_i = 1.$$

Defining the function f_i by

$$f_i(u) = -1 + \exp\left(\frac{\alpha_i}{\tau_i} u\right), \quad i = 1, 2, \ldots, S$$

we have $u \le p^*$ since $\delta p \ge -p^*$. The nonlinearity of f_i is restricted to a *conic sector* [15], see Figure 4.1:

$$f_i \in \text{sec}[0, \beta_i] \quad \text{with} \quad \beta_i = \frac{\exp\left(\frac{\alpha_i}{\tau_i} p^*\right)}{p^*} \tag{4.23}$$

that is,

$$0 \le f_i(u)u \le \beta_i u^2 \quad i = 1, 2, \ldots, S.$$

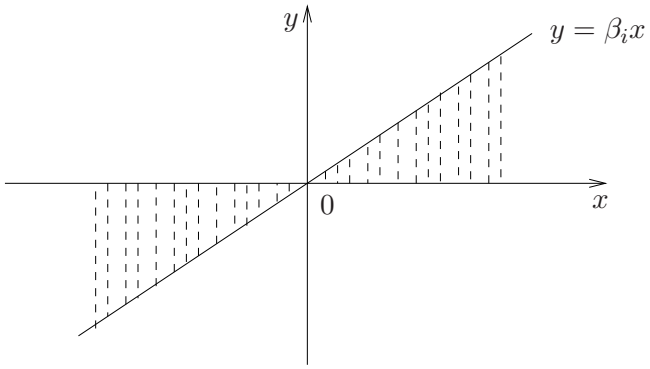


Figure 4.1: Conic Sector

Now if we consider

$$z(t) = \delta p(t), \quad \dot{z}(t) = w(t) \quad \text{and} \quad v_i(t) = \frac{1}{\tau_i} \int_{t-\tau_i}^{t} w(\sigma) d\sigma,$$

and also the fact that

$$z(t) - z(t - \tau_i) = \int_{t-\tau_i}^{t} \dot{z}(\sigma) d\sigma = \tau_i v_i(t) \tag{4.24}$$

then equation 4.22 becomes

$$\dot{z}(t) = \sum_{i=1}^{S} \kappa_i f_i(\tau_i v_i(t) - z(t)). \tag{4.25}$$

The following theorem gives sufficient conditions for global asymptotic stability of the system in the case of one link and one source. The system is said to be globally asymptotically stable if: given initial conditions $z(0) = z_0$ and $w(t) = w_0(t)$ for $t \in [-\tau, 0]$ with $w_0(t) \in \mathcal{L}_2[-\tau, 0]$,[1] we have

$$z(t)^2 \leq C_1 z_0^2 + C_2 \|w_0\|^2$$

for some positive constants C_1 and C_2, and

$$\lim_{t \to \infty} z(t) = 0.$$

Theorem 4.5 (Global Asymptotic Stability 1)**.** *[15] Let us consider the case of one source and one link, that is, when*

$$\dot{z}(t) = f(\tau v(t) - z(t)), \quad w(t) = \dot{z}(t), \quad v(t) = \frac{1}{\tau} \int_{t-\tau}^{t} w(\sigma) d\sigma$$

[1] $\mathcal{L}_2[-\tau, 0]$ is the class of functions f defined on $[-\tau, 0]$, such that f^2 is integrable

where $f(u) \in \text{sec}[0, \beta]$. *Suppose* $\beta\tau < 1$. *Then the equilibrium* $z = 0$ *is globally asymptotically stable.*

Remark 4.6. The condition $\beta\tau < 1$ in the theorem above can be equivalently written as

$$\alpha < \frac{\tilde{c}\big[\log(x_{\max}) - \log(\tilde{c})\big]}{x_{\max} - \tilde{c}}$$

where log is the natural logarithm. This can easily be checked with equations (4.21) and (4.23). This condition is indeed more restrictive on α than the one for local stability.

The next theorem gives sufficient conditions for global asymptotic stability of a system corresponding to a network composed of many sources sharing the same link.

Theorem 4.7 (Global Asymptotic Stability 2)**.** *[15] In the case of many sources sharing the same link, there exists a positive constant* Θ *such that for* $\alpha_i \leq \alpha < \Theta \quad i = 1, 2, \ldots, S$ *(where* α *is just an upper bound of the* α_i*), the origin of the corresponding system*

$$\dot{z}(t) = \sum_{i=1}^{S} \kappa_i f_i(-z(t - \tau_i)), \tag{4.26}$$

is globally asymptotically stable.

Remark 4.8. Equation (4.26) is obtained from equations (4.24) and (4.25).

A more general theorem on stability, established by *Antonis Papachristodoulou* [12] and giving sufficient conditions for asymptotic stability is given below:

Theorem 4.9 (Global Asymptotic Stability 3)**.** *[12] The equilibrium of the system described by equations (4.7), (4.8) and (4.10) is asymptotically stable for arbitrary delays, provided that* $\alpha_i < \frac{x_i^*}{x_{\max,i}}$, *and the routing matrix* R *is a fixed full row rank matrix.*

4.1.3 Conditions for Fairness in some Simple Cases

Single Link and Single Source In this case, from $\dot{q} = \dot{p} = 0$, the equilibrium source rate x^* is given by $x^* = \tilde{c}$; from equation (4.15) and the value of x^*, the equilibrium price p^* is given by

$$p^* = q^* = \frac{-\tau}{\alpha} \log\left(\frac{\tilde{c}}{x_{\max}}\right);$$

and there are no additional condition to verify for fairness.

Single Link and Many Sources In this case, from $\dot{q} = \dot{p} = 0$, the equilibrium source rates x_i^* satisfy the following condition:

$$\sum_{i=1}^{S} x_i^* = \tilde{c}$$

and the equilibrium price is given by:

$$p^* = q_i^* = \frac{-\tau_i}{\alpha_i} \log\left(\frac{x_i^*}{x_{\max,i}}\right) = \frac{\tau_i}{\alpha_i} \log\left(\frac{x_{\max,i}}{x_i^*}\right).$$

All the source rates are equal if and only if

$$x_i^* = \frac{\tilde{c}}{S} \qquad \forall i \in \{1, 2, \ldots, S\}$$

and therefore

$$p^* = q_i^* = \frac{-\tau_i}{\alpha_i} \log\left(\frac{\tilde{c}}{S x_{\max,i}}\right) \qquad \forall i \in \{1, 2, \ldots, S\}.$$

To achieve fairness, the only parameters we can control are the α_i. But in general, for all $i, j \in \{1, 2, \ldots, S\}$, we have:

$$\begin{aligned} q_i^* = q_j^* &\iff \frac{\tau_i}{\alpha_i} \log\left(\frac{x_{\max,i}}{x_i^*}\right) = \frac{\tau_j}{\alpha_j} \log\left(\frac{x_{\max,j}}{x_j^*}\right) \\ &\iff \alpha_j = \alpha_i \frac{\tau_j}{\tau_i} \frac{\log\left(\frac{x_{\max,j}}{x_j^*}\right)}{\log\left(\frac{x_{\max,i}}{x_i^*}\right)}. \end{aligned}$$

So in the particular case of fairness, the relation between the α_i is:

$$\begin{aligned} q_i^* = q_j^* &\iff \frac{\tau_i}{\alpha_i} \log\left(\frac{S x_{\max,i}}{\tilde{c}}\right) = \frac{\tau_j}{\alpha_j} \log\left(\frac{S x_{\max,j}}{\tilde{c}}\right) \\ &\iff \alpha_j = \alpha_i \frac{\tau_j}{\tau_i} \frac{\log\left(\frac{S x_{\max,j}}{\tilde{c}}\right)}{\log\left(\frac{S x_{\max,i}}{\tilde{c}}\right)}. \end{aligned}$$

So we can conveniently[2] choose α_1 for example and deduce all the α_i according to the relation above.

[2] in such a way that the system is at least locally stable with the values of α_i we obtained. The best way to do it is checking by simulation.

Case of the Network of Figure 1.3 This case is more complicated to study. From the definition of equilibrium, we have:

$$x_1^* + x_4^* = \tilde{c}_1, \quad x_2^* + x_4^* = \tilde{c}_2, \quad x_3^* + x_4^* = \tilde{c}_3, \quad q_4^* = q_1^* + q_2^* + q_3^*,$$

$$q_1^* = \frac{-\tau_1}{\alpha_1} \log\left(\frac{x_1^*}{x_{\max,1}}\right), \qquad q_2^* = \frac{-\tau_2}{\alpha_2} \log\left(\frac{x_2^*}{x_{\max,2}}\right),$$

$$q_3^* = \frac{-\tau_3}{\alpha_3} \log\left(\frac{x_3^*}{x_{\max,3}}\right), \qquad q_4^* = \frac{-3\tau_4}{\alpha_4} \log\left(\frac{x_4^*}{x_{\max,4}}\right),$$

and therefore x_4^* is a solution of the following equation:

$$\left(\frac{x_4^*}{x_{\max,4}}\right)^{\frac{-3\tau_4}{\alpha_4}} = \left(\frac{\tilde{c}_1 - x_4^*}{x_{\max,1}}\right)^{\frac{-\tau_1}{\alpha_1}} \left(\frac{\tilde{c}_2 - x_4^*}{x_{\max,2}}\right)^{\frac{-\tau_2}{\alpha_2}} \left(\frac{\tilde{c}_3 - x_4^*}{x_{\max,3}}\right)^{\frac{-\tau_3}{\alpha_3}}$$

and solving the above equation is not an easy task at all. However, if we have x_4^*, we can easily deduce x_1^*, x_2^* and x_3^*. Choosing α_1, α_2, α_3 and α_4 can not be done using a particular mathematical method. Moreover, fairness is very difficult to achieve in this case. A simple example is when the links have the same capacity, c, say. In that case we will obtain $x_1^* = x_2^* = x_3^* = x_4^* = \frac{\tilde{c}}{2}$. But still, the choice of α_1, α_2, α_3 and α_4 is not subject to a given mathematical method.

4.2 The Primal–Dual Control Approach

4.2.1 Formulation

As proposed in [11, 10] (for link prices and source rates), the dynamics governing the link prices, source rates and buffer sizes are given as follows, for each $l \in \mathcal{L}$ and for each $i \in \mathcal{S}$:

$$\dot{p}_l(t) = \begin{cases} \frac{y_l(t) - \tilde{c}_l}{\tilde{c}_l} & \text{if } p_l(t) > 0 \\ \max\left\{0, \frac{y_l(t) - \tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l(t) = 0 \end{cases} \tag{4.27}$$

$$x_i(t) = x_{m,i} \exp\left(\xi_i(t) - \frac{\alpha_i}{M_i \tau_i} q_i(t)\right) \tag{4.28}$$

$$\dot{\xi}_i(t) = \frac{\beta_i}{\tau_i}\big(U_i'(x_i(t)) - q_i(t)\big) \tag{4.29}$$

$$\dot{b}_l(t) = \begin{cases} y_l(t) - c_l & \text{if } b_l(t) > 0 \\ \max\{0, y_l(t) - c_l\} & \text{if } b_l(t) = 0 \end{cases} \tag{4.30}$$

where U_i is the source's *utility function* and is a strictly concave increasing function, β_i is a parameter, and $x_{m,i}$ is a given transmission rate for source i, not necessarily the maximum transmission rate. We notice that the quantity $x_{\max,i}$ in the dual control law has changed to $x_{m,i}\exp(\xi_i(t))$ and is not constant any more but varies exponentially.

Remark 4.10. One advantage of this control law is that the conditions on the $x_{m,i}$ are less restrictive than the corresponding conditions in the dual control law. In this case, to be realistic, we just have to take all the $x_{m,i}$ to be positive and small enough (close to zero compared to the link capacity) and therefore not greater than the link capacity as in the dual control law, so that the source rates will not jump (discontinuously) from zero when $t < t_0$ to a positive value (probably greater than the link capacity c_l) when $t = t_0$.

We can reformulate the above dynamics as follows:

$$\dot{p}_l(t) = \begin{cases} \lambda_l(t) & \text{if } p_l(t) > 0 \\ \max\{0, \lambda_l(t)\} & \text{if } p_l(t) = 0 \end{cases} \tag{4.31}$$

$$x_i(t) = x_{m,i}\exp\left(\xi_i(t) - \frac{\alpha_i}{M_i\tau_i}\sum_{l=1}^{L} R_{li}p_l\left(t-\tau_{li}^b\right)\right) \tag{4.32}$$

$$\dot{\xi}_i(t) = \frac{\beta_i}{\tau_i}\left(U_i'(x_i(t)) - \sum_{l=1}^{L} R_{li}p_l\left(t-\tau_{li}^b\right)\right) \tag{4.33}$$

$$\dot{b}_l(t) = \begin{cases} \mu_l(t) & \text{if } b_l(t) > 0 \\ \max\{0, \mu_l(t)\} & \text{if } b_l(t) = 0 \end{cases} \tag{4.34}$$

$$\lambda_l(t) = \sum_{i=1}^{S}\frac{R_{li}}{\tilde{c}_l}x_i\left(t-\tau_{li}^f\right) - 1 \tag{4.35}$$

$$\mu_l(t) = \sum_{i=1}^{S} R_{li}x_i\left(t-\tau_{li}^f\right) - c_l. \tag{4.36}$$

From now on, we will take all our utility functions (in the primal–dual control laws) to be logarithmic, in particular, $U_i(x_i) = K_i\log(x_i)$ where K_i is a positive real constant. These utility functions are therefore continuous, strictly concave and increasing, so as to ensure the existence and uniqueness of the solution to the optimisation problem behind our optimal control problem, as explained in [8].

4.2.2 Stability Analysis

We reconsider the reduced routing matrix $\bar{R}$ as defined in the dual control case. The following theorem gives sufficient conditions for local stability of the primal–dual control law.

Theorem 4.11 (Local Stability)**.** *[16] Assume that $\bar{R}$ is of full row rank and that for every source i, we have $\tau_i \leq \bar{\tau}$ for a positive constant $\bar{\tau}$. Then the primal–dual control with $\alpha_i \leq \alpha < \frac{\pi}{2}$ and $\beta = \frac{\eta\alpha_i}{M_i\bar{\tau}}$, is linearly stable for a small enough $\eta \in (0,1)$ depending only on α.*

The following theorem gives sufficient conditions for global asymptotic stability of the primal–dual control law in the case of a network composed of a single link and a single source.

Theorem 4.12 (Global Asymptotic Stability)**.** *[17] Suppose that $U'(0) < \infty$. Then there exists $\beta^* > 0$ such that the primal–dual control law for a single link and a single source is globally asymptotically stable if $\alpha < 1$ and $\beta < \beta^*$.*

4.2.3 Dynamics of Source Rates

From the equations governing the dynamics in the primal–dual control, assuming that the utility functions are given by $U_i(x_i(t)) = K_i \log(x_i(t))$, and therefore $U_i'(x_i(t)) = \frac{K_i}{x_i(t)}$, it can be established that

$$\dot{x}_i(t) = \frac{\beta_i K_i}{\tau_i} - \frac{\beta_i}{\tau_i} x_i(t) q_i(t) - \frac{\alpha_i}{M_i \tau_i} x_i(t) \dot{q}_i(t).$$

And due to the fact that the source rates (as well as the link prices) are small enough at the beginning, we have $x_i(t) \approx 0$ and then

$$\dot{x}_i(t) \approx \frac{\beta_i K_i}{\tau_i},$$

which means that the source rates grow approximately linearly at the beginning, with slopes $\frac{\beta_i K_i}{\tau_i}$.

4.2.4 Conditions for Fairness in some Simple Cases

Single Link and Single Source In this case, from $\dot{q} = \dot{p} = 0$, the equilibrium source rate x^* is given by $x^* = \tilde{c}$; from the value of x^*, the

equilibrium price p^* is given by

$$p^* = q^* = U'(\tilde{c}) = \frac{K}{\tilde{c}};$$

the equilibrium value of ξ is given by

$$\xi^* = \log\left(\frac{\tilde{c}}{x_m}\right) + \frac{\alpha}{\tau}\frac{K}{\tilde{c}};$$

and there is no additional condition required for fairness.

Single Link and Many Sources In this case, from $\dot{q} = \dot{p} = 0$, the equilibrium source rates x_i^* satisfy the following relation:

$$\sum_{i=1}^{S} x_i^* = \tilde{c};$$

and from $\dot{\xi}_i = 0$ for all $i \in \{1, 2, \ldots, S\}$, we have

$$\frac{K_i}{x_i^*} = q_i^* = p^* \qquad \forall i \in \{1, 2, \ldots, S\}.$$

So

$$\frac{K_i}{x_i^*} = \frac{K_j}{x_j^*} \qquad \forall i, j \in \{1, 2, \ldots, S\}$$

and therefore, for all $i \in \{1, 2, \ldots, S\}$, we have

$$\frac{K_i}{x_i^*} = \frac{1}{\tilde{c}}\sum_{j=1}^{S} K_j,$$

so that

$$x_i^* = \frac{\tilde{c}K_i}{\sum_{j=1}^{S} K_j} \qquad \forall i \in \{1, 2, \ldots, S\}.$$

For all $i, j \in \{1, 2, \ldots, S\}$, we have

$$x_i^* = x_j^* \Longleftrightarrow K_i = K_j,$$

and then

$$x_i^* = \frac{\tilde{c}}{S} \quad \forall i \in \{1, 2, \ldots, S\} \Longleftrightarrow K_i = K \quad \forall i \in \{1, 2, \ldots, S\}$$

where K is a positive real constant. *To achieve fairness, we need to take all the K_i in the utility functions to be the same.* It follows that in the particular case of fairness, the equilibrium price is

$$p^* = \frac{KS}{\tilde{c}}.$$

Also, the equilibrium values of the ξ_i are given by

$$\xi_i^* = \frac{\alpha_i}{\tau_i}\frac{KS}{\tilde{c}} + \log\left(\frac{\tilde{c}}{Sx_{m,i}}\right).$$

Case of the Network of Figure 1.3 This case is more complicated to study. However, at equilibrium we have:

$$x_1^* + x_4^* = \tilde{c}_1, \quad x_2^* + x_4^* = \tilde{c}_2, \quad x_3^* + x_4^* = \tilde{c}_3;$$

$$q_1^* = \frac{K_1}{x_1^*} = p_1^*, \qquad q_2^* = \frac{K_2}{x_2^*} = p_2^*, \qquad q_3^* = \frac{K_3}{x_3^*} = p_3^*, \qquad q_4^* = \frac{K_4}{x_4^*} = p_4^*;$$

and also $q_4^* = q_1^* + q_2^* + q_3^*$; so

$$\frac{K_4}{x_4^*} = \frac{K_1}{x_1^*} + \frac{K_2}{x_2^*} + \frac{K_3}{x_3^*}.$$

If all the links have the same capacity, we can achieve some fairness, making all the equilibrium source rates to be equal by taking for example $K_1 = K_2 = K_3 = \frac{K_4}{3} = K$ where K is a positive real constant, or more generally $K_4 = K_1 + K_2 + K_3$ where K_1, K_2 and K_3 are arbitrarily chosen positive real constants. But when the link capacities are different, we can also assume the same conditions on the K_i to achieve something close to fairness. When we take $K_1 = K_2 = K_3 = \frac{K_4}{3} = K$ where K is a positive real constant, the equilibrium source rates are then linked by the relation

$$\frac{3}{x_4^*} = \frac{1}{x_1^*} + \frac{1}{x_2^*} + \frac{1}{x_3^*},$$

that is,

$$\frac{3}{x_4^*} = \frac{1}{\tilde{c}_1 - x_4^*} + \frac{1}{\tilde{c}_2 - x_4^*} + \frac{1}{\tilde{c}_3 - x_4^*}.$$

If we can solve the above equation which is an equation of degree 3 for x_4^* (and it is not an easy task), then x_1^*, x_2^* and x_3^* immediately follow. The general case $K_4 = K_1 + K_2 + K_3$ is more difficult to handle.

4.3 The Control Approach based on Queueing Delay

4.3.1 Formulation

As proposed in [11, 10] (for link prices and source rates), the dynamics governing the link prices, sources rates and buffer sizes are given as follows, for each $l \in \mathcal{L}$ and for each $i \in \mathcal{S}$:

$$\dot{p}_l(t) = \begin{cases} \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l} & \text{if } p_l(t) > 0 \\ \max\left\{0, \frac{y_l(t)-\tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l(t) = 0 \end{cases} \tag{4.37}$$

$$x_i(t) = x_{m,i} \exp\left(\xi_i(t)\right) \left(\frac{d_i}{d_i + q_i(t)}\right)^{\frac{\alpha_i}{M_i}} \tag{4.38}$$

$$\dot{\xi}_i(t) = \left(\frac{\beta_i}{d_i + q_i(t)}\right) \left(U_i'(x_i(t)) - q_i(t)\right) \tag{4.39}$$

$$\dot{b}_l(t) = \begin{cases} y_l(t) - c_l & \text{if } b_l(t) > 0 \\ \max\{0, y_l(t) - c_l\} & \text{if } b_l(t) = 0 \end{cases} \tag{4.40}$$

with $x_{m,i}$, β_i and U_i as in the primal–dual control case, and d_i the fixed part of the round–trip time. Since the target capacity $\tilde{c}_l$ in this case is the same as the actual capacity c_l, the aggregate price $q_i(t)$ is exactly the queueing delay.

We can reformulate the above dynamics as follows:

$$\dot{p}_l(t) = \begin{cases} \lambda_l(t) & \text{if } p_l(t) > 0 \\ \max\{0, \lambda_l(t)\} & \text{if } p_l(t) = 0 \end{cases} \tag{4.41}$$

$$x_i(t) = x_{m,i} \exp\left(\xi_i(t)\right) \left(\frac{d_i}{d_i + q_i(t)}\right)^{\frac{\alpha_i}{M_i}} \tag{4.42}$$

$$\dot{\xi}_i(t) = \left(\frac{\beta_i}{d_i + q_i(t)}\right) \left(U_i'(x_i(t)) - q_i(t)\right) \tag{4.43}$$

$$\dot{b}_l(t) = \begin{cases} \mu_l(t) & \text{if } b_l(t) > 0 \\ \max\{0, \mu_l(t)\} & \text{if } b_l(t) = 0 \end{cases} \tag{4.44}$$

$$\lambda_l(t) = \sum_{i=1}^{S} \frac{R_{li}}{\tilde{c}_l} x_i \left(t - \tau_{li}^f\right) - 1 \tag{4.45}$$

$$\mu_l(t) = \sum_{i=1}^{S} R_{li} x_i \left(t - \tau_{li}^f\right) - c_l \tag{4.46}$$

where

$$q_i(t) = \sum_{l=1}^{L} R_{li} p_l \left(t - \tau_{li}^b\right).$$

We can also notice that $\mu_l(t) = c_l \lambda_l(t)$.

4.3.2 Dynamics of Source Rates

From the equations governing the dynamics in the control based on queueing delay, assuming that the utility functions are given by $U_i(x_i(t)) = K_i \log(x_i(t))$, and therefore $U_i'(x_i(t)) = \frac{K_i}{x_i(t)}$, it can be established that

$$\dot{x}_i(t) = \frac{\beta_i K_i}{d_i + q_i(t)} - \frac{\beta_i}{d_i + q_i(t)} x_i(t) q_i(t) - \frac{\alpha_i}{M_i(d_i + q_i(t))} x_i(t) \dot{q}_i(t)$$

where d_i is the fixed part of the round–trip time for source i. And due to the fact that the source rates (as well as the link prices) are small enough at the beginning, we have $x_i(t) \approx 0$ and then

$$\dot{x}_i(t) \approx \frac{\beta_i K_i}{d_i},$$

which means that the source rates grow approximately linearly at the beginning, with slopes $\frac{\beta_i K_i}{d_i}$.

4.3.3 Conditions for Fairness in some Simple Cases

Single Link and Single Source As in the primal–dual control case, we have $x^* = \tilde{c}$ and

$$p^* = q^* = \frac{K}{\tilde{c}},$$

but

$$\xi^* = \log\left(\frac{\tilde{c}}{x_m}\right) + \alpha \log\left(1 + \frac{K}{d\tilde{c}}\right);$$

and no additional condition is required for fairness.

Single Link and Many Sources As in the primal–dual control case, we have

$$\sum_{i=1}^{S} x_i^* = \tilde{c}, \qquad p^* = \frac{K_i}{x_i^*} = \frac{1}{\tilde{c}} \sum_{j=1}^{S} K_j, \quad \forall i \in \{1, 2, \ldots, S\}.$$

So for fairness, we take all the K_i to be the same and therefore

$$x_i^* = \frac{\tilde{c}}{S} \ \forall i \quad \text{and} \quad p^* = \frac{KS}{\tilde{c}}.$$

Moreover, the equilibrium value of ξ_i is

$$\xi_i^* = \log\left(\frac{\tilde{c}}{Sx_{m,i}}\right) + \alpha_i \log\left(1 + \frac{KS}{d_i\tilde{c}}\right).$$

Case of the Network of Figure 1.3 This case is more complicated to study. However, to achieve something close to fairness, we can also take for example, as in the primal–dual control case, $K_1 = K_2 = K_3 = \frac{K_4}{3} = K$ where K is a positive real constant, or more generally $K_4 = K_1 + K_2 + K_3$ where K_1, K_2 and K_3 are arbitrarily chosen positive real constants. If all the links have the same capacity, the choice of coefficients K_i in the utility functions as mentioned just above leads to fairness among sources sharing the same link; we actually obtain the same equilibrium rate for all the sources in the network.

Conclusion

We have presented three flow model control laws, namely the dual control law, the primal–dual control law and the control law based on queueing delay. We have reviewed the stability analysis in some simple cases of the dual control law, with some weaker conditions on some network parameters. We have also discussed some fairness conditions in the three control laws, in some simple cases.

Chapter 5

Computer Simulations

In this chapter, we present the numerical solutions for the control laws developed in Chapter 2 and formulated in Chapter 4. In all the control laws studied, the dynamics are not explicitly defined at any time. The evolution of the link prices follows a delay differential equation not solved analytically, and the rest of the dynamics depends on the link prices. We need to use numerical methods to solve these equations, and therefore determine the dynamics at any time. The numerical method used here is the modified Heun's method [2] and our codes are written in C and Python programming languages. Different configurations of the network topology are associated to different computer simulations' codes in each control law. The time t in seconds is in the range $[0, 500]$.

5.1 The Dual Control Approach

The equations governing the dynamics are given in section 4.1.1. In all the examples to follow, for a link l with capacity c_l, the virtual capacity (or target capacity) will be $\tilde{c}_l = 0.95c_l$. In our simulations, we compute the link prices p_l, the source rates x_i and the buffer (queue) sizes b_l. Account is not taken of queueing delays. We will run our simulations with arbitrary[1] values of the source gains α_i, the forward and backward delays between sources and links, the maximum source rates $x_{\max,i}$ provided their sums per link are greater than the corresponding link capacities.

[1] Arbitrary but in accordance with the control law

5.1.1 One link and one source

The network is illustrated in Figure 5.1. In this case, $S = 1$ (number of

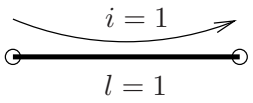


Figure 5.1: A network composed of 1 link and 1 source

sources), $L = 1$ (number of links), the routing matrix is $R = 1$, the maximum number of bottlenecks on the source's path is $M = 1$. The dual control is sensitive to the maximum source rate $x_{\max}$, the source gain α, the forward and backward delays τ^f and τ^b. We arbitrarily choose the following cases:

1) $\tau^f = 2.5s$, $\tau^b = 2s$, $x_{\max} = 250$, $\alpha = 0.75$;

2) $\tau^f = 2.5s$, $\tau^b = 2s$, $x_{\max} = 300$, $\alpha = 0.5$.

In all the cases, we consider the link capacity to be $c = 200$ *packets/s*. Figures 5.2 to 5.7 represent the link prices, the source rates and the queue sizes in both cases considered.

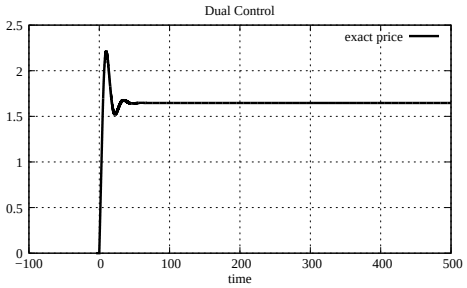


Figure 5.2: Link price, case 1 with 1 link and 1 source (Dual)

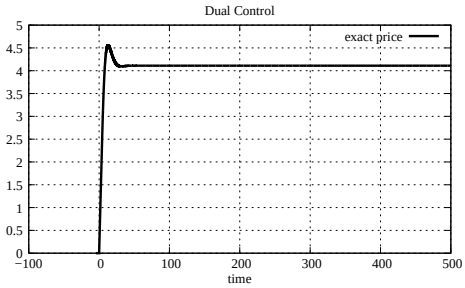


Figure 5.3: Link price, case 2 with 1 link and 1 source (Dual)

5.1.2 One link and three sources

The network is illustrated in Figure 5.8. In this case, $S = 3$ (number of sources), $L = 1$ (number of links), the routing matrix is $R_i = 1$ for $1 \leq i \leq 3$ (a column vector), the maximum number of bottlenecks on source i's path is $M_i = 1$ for $1 \leq i \leq 3$. We will explore two cases: in the first case (the simple case), we take all the source gains to be equal, precisely, $\alpha_1 = \alpha_2 = \alpha_3 = 0.5$;

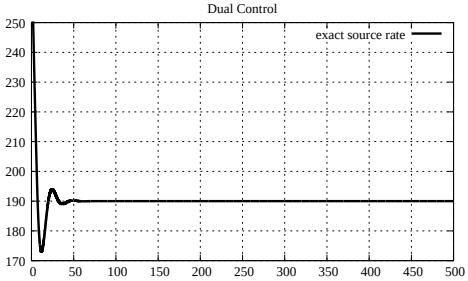


Figure 5.4: Source rate, case 1 with 1 link and 1 source (Dual)

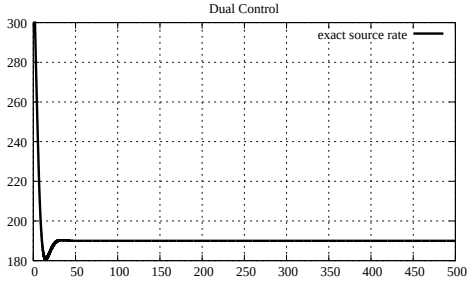


Figure 5.5: Source rate, case 2 with 1 link and 1 source (Dual)

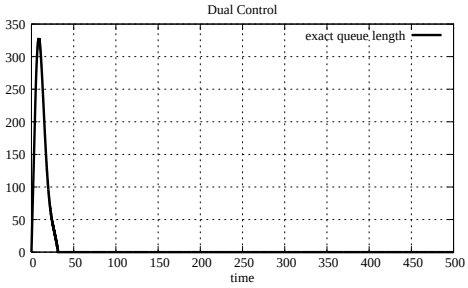


Figure 5.6: Queue size, case 1 with 1 link and 1 source (Dual)

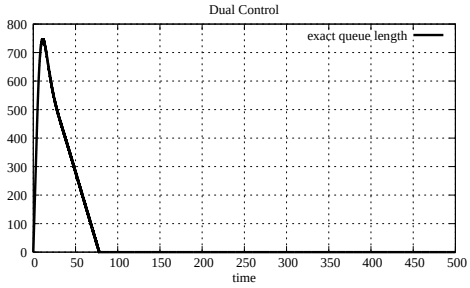


Figure 5.7: Queue size, case 2 with 1 link and 1 source (Dual)

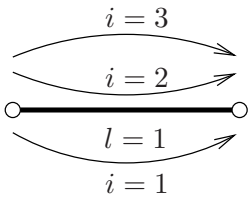


Figure 5.8: A network composed of 1 link and 3 sources

and in the second case (the fairness case, that is when the sources fairly share the links they are using) we choose $\alpha_1 = 0.5$ and according to the fairness condition explored in section 4.1.3, we obtain $\alpha_2 = 1.278657$ and $\alpha_3 = 1.153995$. The local stability conditions are respected since all the α_i are less than $\frac{\pi}{2}$. Figures 5.9 to 5.14 represent the link prices, the source rates and the queue sizes in both cases.

5.1.3 Three links and one source

The network is illustrated in Figure 5.15. In this case, $S = 1$ (number of

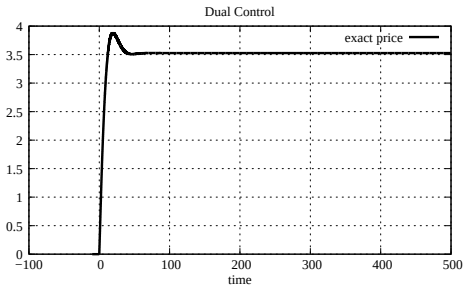


Figure 5.9: Link price, simple case with 1 link and 3 sources (Dual)

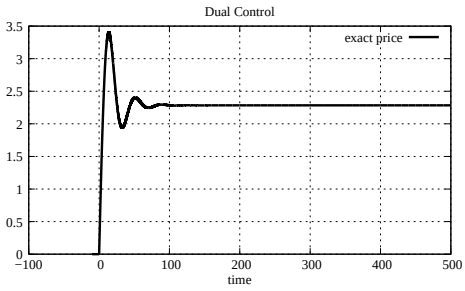


Figure 5.10: Link price, fairness case with 1 link and 3 sources (Dual)

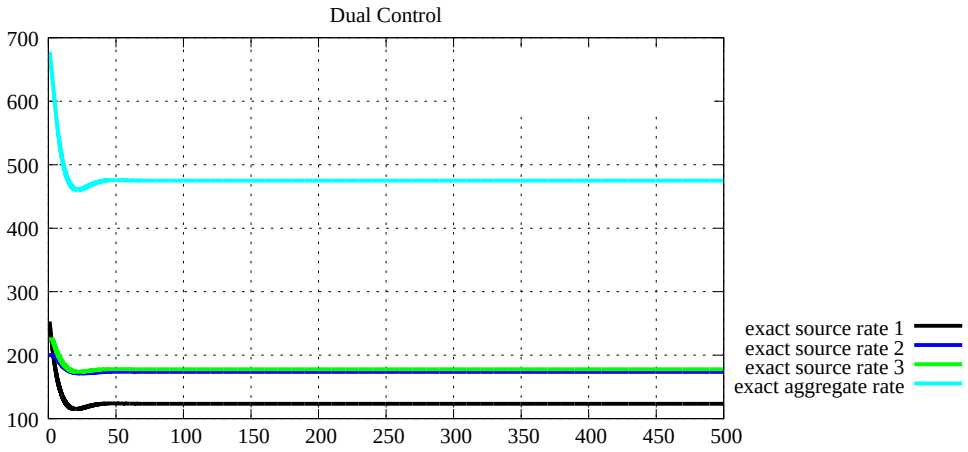


Figure 5.11: Source rates, simple case with 1 link and 3 sources (Dual)

sources), $L = 3$ (number of links), the routing matrix is $R_l = 1$ for $1 \leq l \leq 3$ (a row vector), the maximum number of bottlenecks on the source's path is $M = 3$. We can also notice that in this case, the routing matrix is not of full row rank. We will distinguish the case where all the links have the same capacity and the case where the links have different capacities. We also give the time at which two of the three links, each with capacity greater than the minimum capacity, stop being bottlenecks (when the links have different capacities). The cases analysed are the following:

1) all the links have the same capacity $c = 300$ *packets/s*;

2) links 1, 2 and 3 have respective capacities 250, 300 and 275 *packets/s*.

In all the cases, we take $\alpha = 0.75$ and $x_{\max} = 500$ *packets/s*. From the simulations, in case 1, all the links are always bottlenecks. It comes from

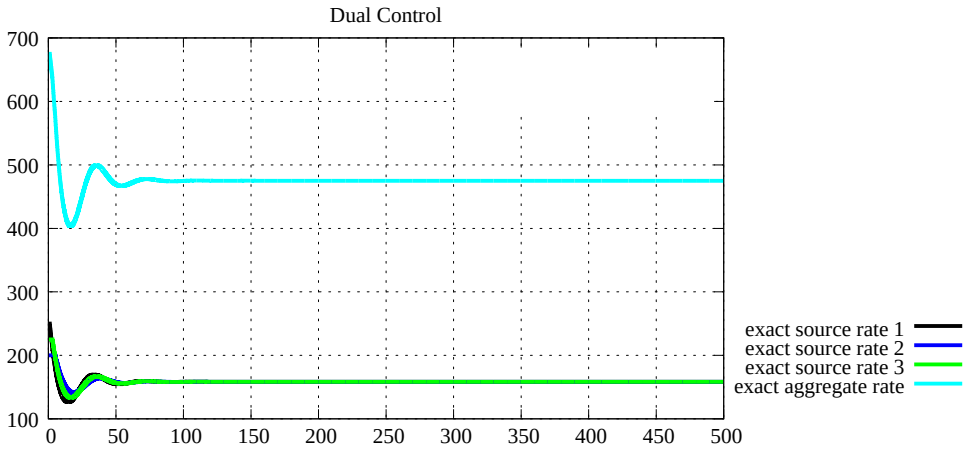


Figure 5.12: Source rates, fairness case with 1 links and 3 sources (Dual)

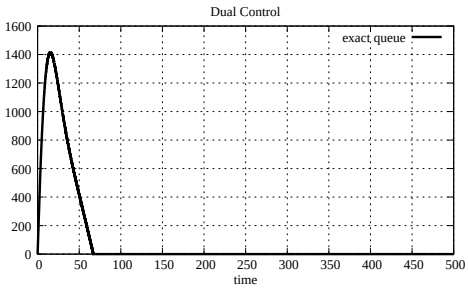


Figure 5.13: Queue size, simple case with 1 link and 3 sources (Dual)

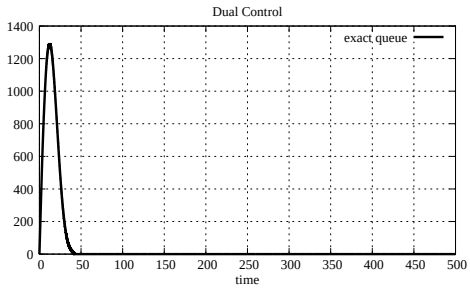


Figure 5.14: Queue size, fairness case with 1 link and 3 sources (Dual)

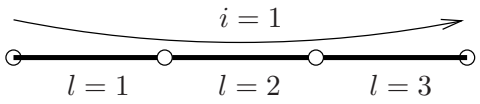


Figure 5.15: A network composed of 3 links and 1 source

the fact that they all have the same capacity. In case 2, link 2 stops being a bottleneck at $t = 94.51s$ while link 3 stops being a bottleneck at $t = 303.24s$. It is due to the fact that link 2 has the largest capacity, followed by link 3, and link 1 has the smallest capacity. Figures 5.16 to 5.21 represent the link prices, the source rates and the queue sizes in both cases.

Since the case when the links have different capacities does not agree with the maximum utilisation of the network resources, the example of three links

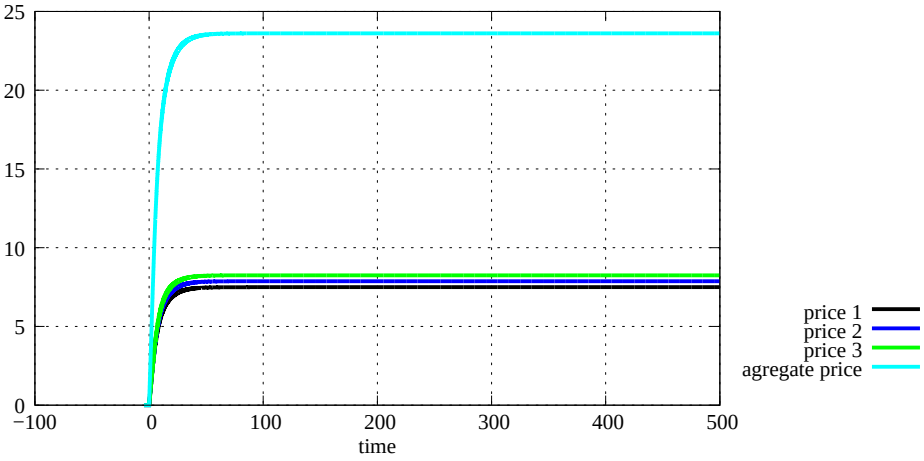


Figure 5.16: Link prices, case 1 with 3 links and 1 source (Dual)

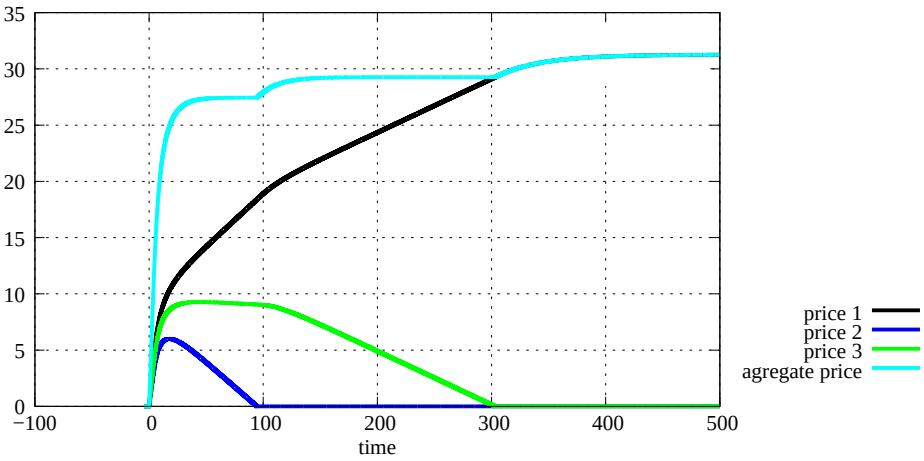


Figure 5.17: Link prices, case 2 with 3 links and 1 source (Dual)

and one source will not be considered for further analysis.

5.1.4 Three links and four sources

The corresponding network is illustrated in Figure 5.22. In this case, $S = 4$ (number of sources), $L = 3$ (number of links), the routing matrix is the one given at the end of section 1.2, the maximum number of bottlenecks on source i's path is $M_4 = 3$ and $M_i = 1$ for $1 \leq i \leq 3$. We consider the case where the links have different capacities and the one where the links

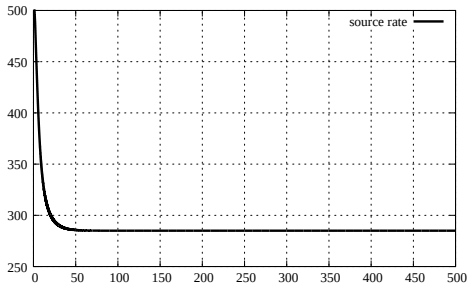


Figure 5.18: Source rate, case 1 with 3 links and 1 source (Dual)

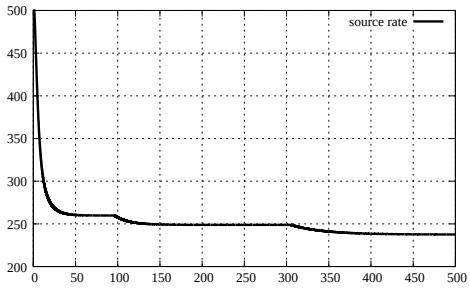


Figure 5.19: Source rate, case 2 with 3 links and 1 source (Dual)

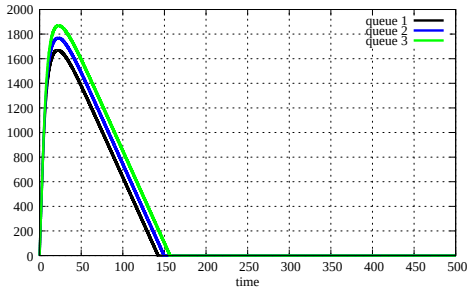


Figure 5.20: Queue sizes, case 1 with 3 links and 1 source (Dual)

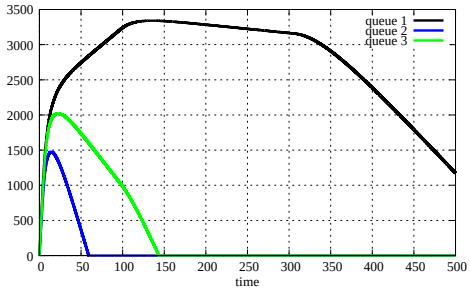


Figure 5.21: Queue sizes, case 2 with 3 links and 1 source (Dual)

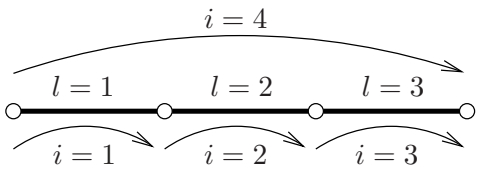


Figure 5.22: A network composed of 3 links and 4 sources

all have the same capacity. In the latter case, we set all the capacities to 300 *packets/sec* while in the other case we consider $c_1 = 300$, $c_2 = 200$ and $c_3 = 250$ *packets/sec*. Figures 5.23 to 5.28 represent the link prices, the source rates and the buffer sizes in both cases. We can see that when the links have the same capacity, sources 1, 2 and 3 have the same equilibrium rate. This is due to the fact that not only the links have the same capacity, but each of the mentioned sources is sharing the link with source 4. It follows immediately that the aggregate rates at equilibrium are the same for all the links, since they have the same virtual capacity.

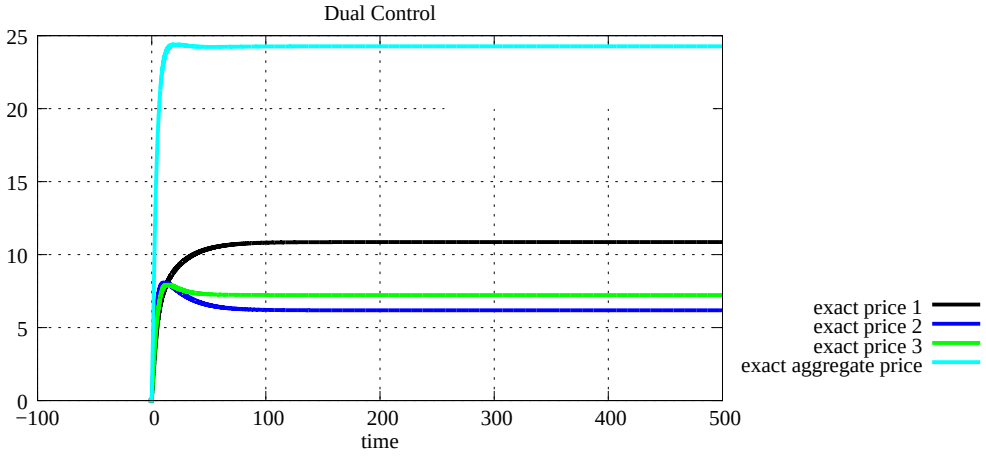


Figure 5.23: Link prices with 3 links and 4 sources (Dual)

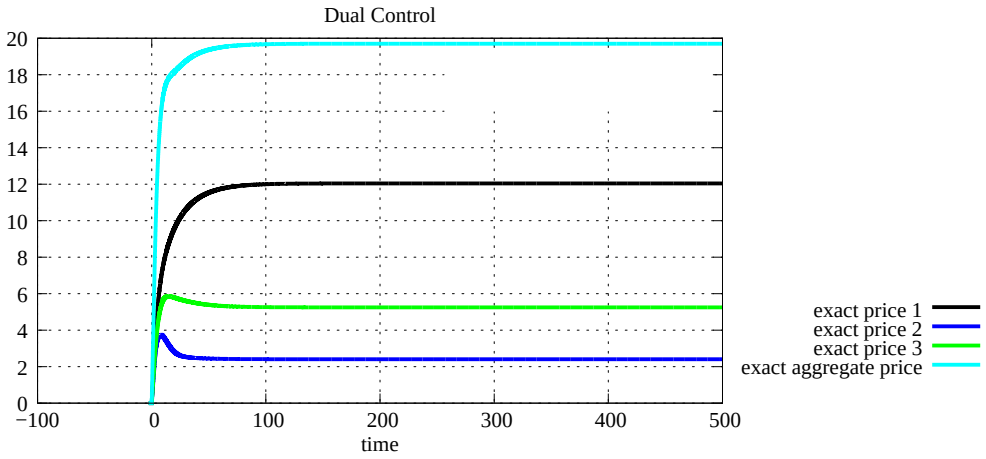


Figure 5.24: Link prices with 3 links (same capacity) and 4 sources (Dual)

5.2 The Primal–Dual Control Approach

The equations governing the dynamics are given in section 4.2.1. In all the examples to follow, for a link l with capacity c_l, the virtual capacity will be $\tilde{c}_l = 0.95c_l$. In our simulations, we compute the link prices p_l, the source rates x_i and the buffer sizes b_l. In all the computations, account is not taken of the queueing delays. We will run our simulations with arbitrary values of: the given source transmission rates $x_{m,i}$, positive and close to

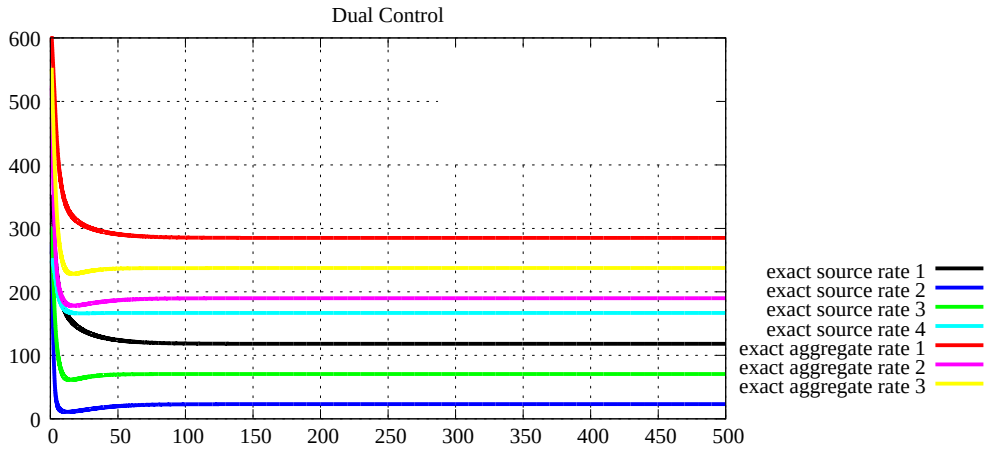


Figure 5.25: Source rates with 3 links and 4 sources (Dual)

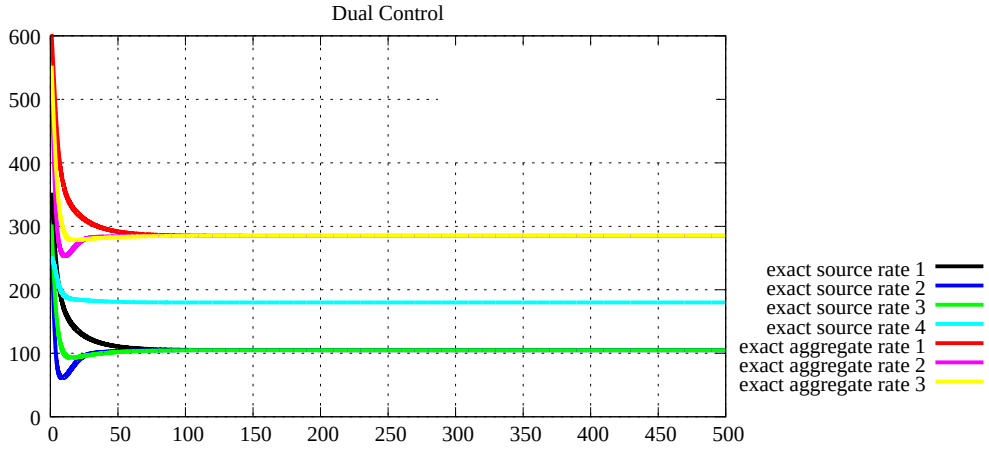


Figure 5.26: Source rates with 3 links (same capacity) and 4 sources (Dual)

zero[2] (compared to the corresponding link capacities); the gains α_i; the forward and backward delays, all the delays assumed positive. We will also consider the constants K_i in the utility functions and the parameters β_i. The primal–dual control is sensitive to all these quantities. In the captions of all the figures in this section, $P\ D\ C$ stands for *primal–dual control*.

[2]They can also have large values, but that case is less realistic

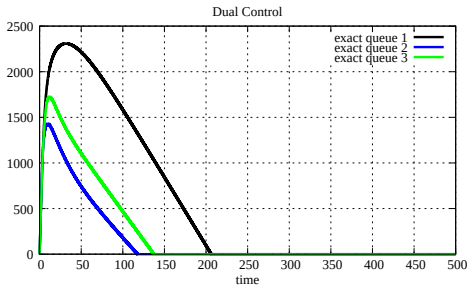


Figure 5.27: Queue sizes with 3 links and 4 sources (Dual)

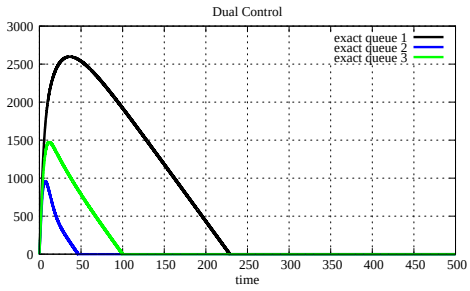


Figure 5.28: Queue sizes with 3 links (same capacity) and 4 sources (Dual)

5.2.1 One link and one source

The network is illustrated in Figure 5.1, page 48. The values S, L, R and M in this case are the same as in the corresponding case in the dual control law. The cases analysed are the following:

1) $K = 150$, $\beta = 0.05$;

2) $K = 150$, $\beta = 0.1$.

In all the cases, we consider the link capacity to be $c = 200$ *packets/s*. Figures 5.29 to 5.34 represent the link prices, the source rates and the queue sizes in both cases.

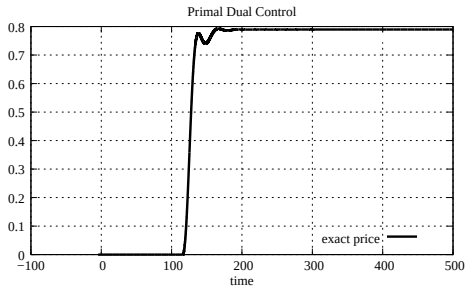


Figure 5.29: Link price, case 1 with 1 link and 1 source (P D C)

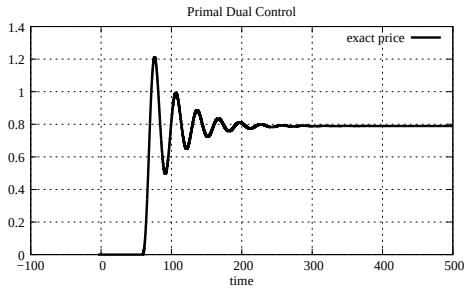


Figure 5.30: Link price, case 2 with 1 link and 1 source (P D C)

5.2.2 One link and three sources

The network is illustrated in Figure 5.8, page 49. The values S, L, R_i $(1 \leq i \leq 3)$ and M_i $(1 \leq i \leq 3)$ in this case are the same as in the corresponding case in the dual control law. The constants K_i in the utility

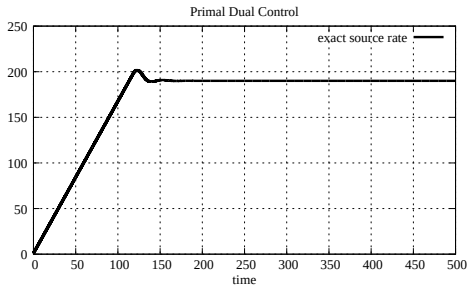


Figure 5.31: Source rate, case 1 with 1 link and 1 source (P D C)

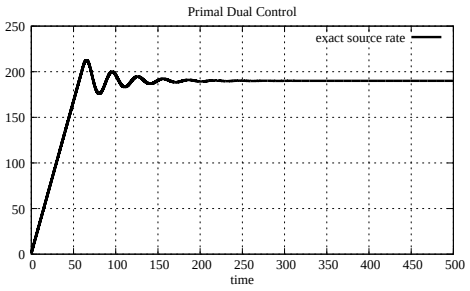


Figure 5.32: Source rate, case 2 with 1 link and 1 source (P D C)

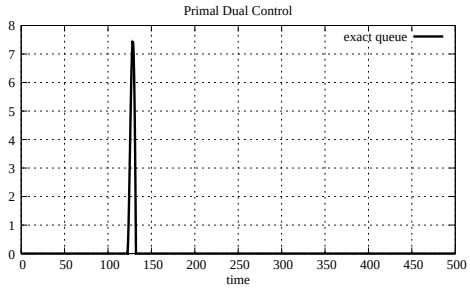


Figure 5.33: Queue size, case 1 with 1 link and 1 source (P D C)

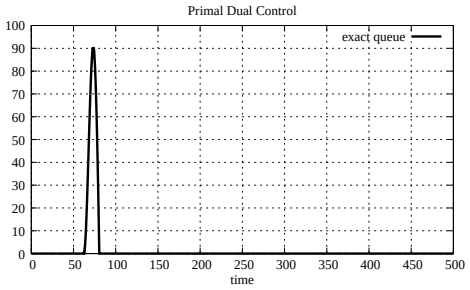


Figure 5.34: Queue size, case 2 with 1 link and 1 source (P D C)

functions are quite arbitrarily chosen but positive, then chosen in such a way to achieve fairness among the sources. Figures 5.35 to 5.40 represent the link prices, the source rates and the queue sizes in both cases.

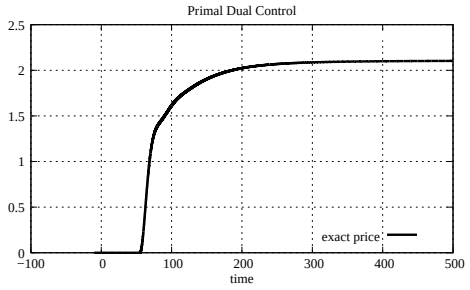


Figure 5.35: Link price, simple case with 1 link and 3 sources (P D C)

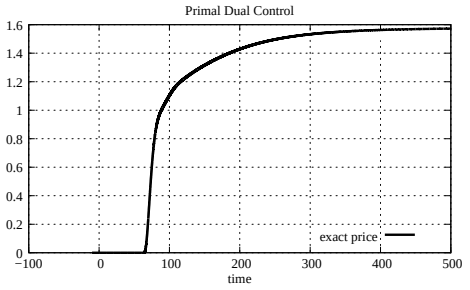


Figure 5.36: Link price, fairness case with 1 link and 3 sources (P D C)

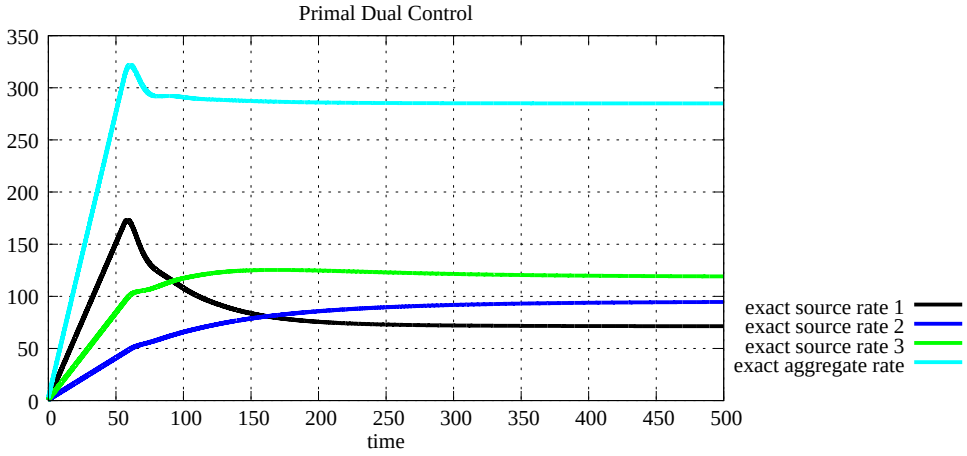


Figure 5.37: Source rates, simple case with 1 link and 3 sources (P D C)

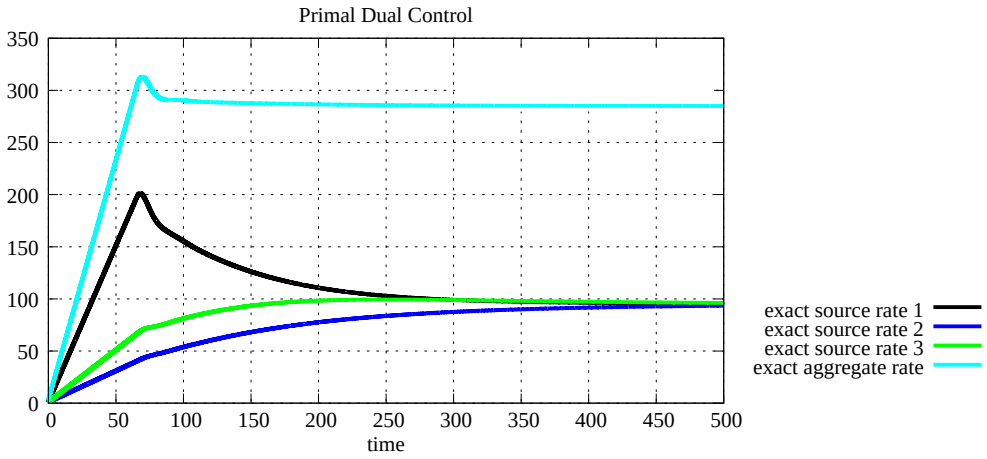


Figure 5.38: Source rates, fairness case with 1 links and 3 sources (P D C)

5.2.3 Three links and four sources

The corresponding network is illustrated in Figure 5.22, page 53. The values S, L, the routing matrix R and M_i $(1 \leq i \leq 4)$ in this case are the same as in the corresponding case in the dual control law. We consider the case where the links have different capacities and the one where the links all have the same capacity. The capacities are taken to be the same as in the corresponding cases in the dual control law. We also take $K_1 = 150$, $K_2 = 200$, $K_3 = 250$ and $K_4 = 600$. Figures 5.41 to 5.46 represent the

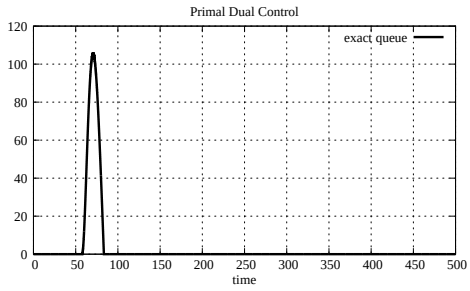


Figure 5.39: Queue size, simple case with 1 link and 3 sources (P D C)

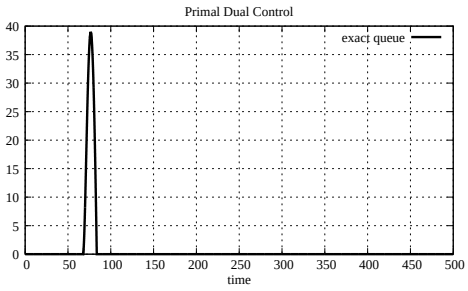


Figure 5.40: Queue size, fairness case with 1 link and 3 sources (P D C)

link prices, the source rates and the buffer sizes in both cases. We

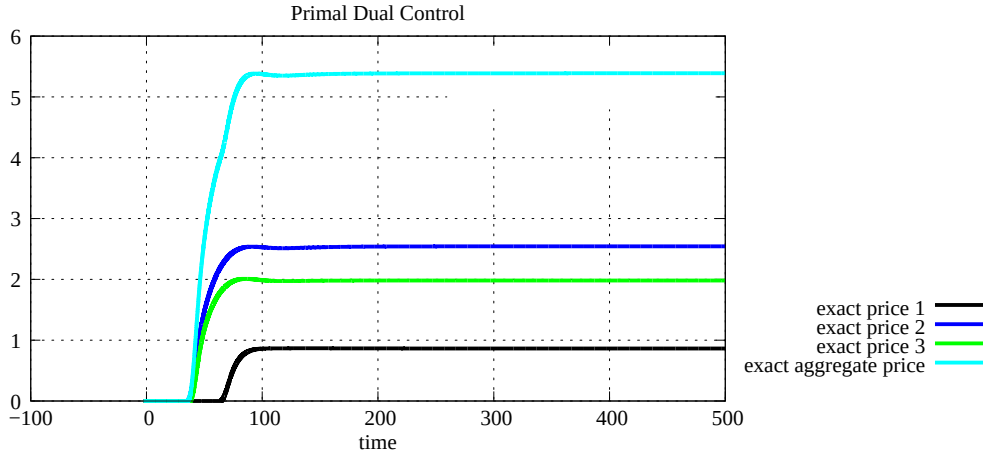


Figure 5.41: Link prices with 3 links and 4 sources (P D C)

observe that when the links have the same capacity, all the sources have the same equilibrium rate. This is comes from the values we took for the K_i as discussed in section 4.2.4.

5.3 The Control Approach based on Queueing Delay

The equations governing the dynamics are given in section 4.3.1. In this case, the virtual capacity $\tilde{c}_l$ for any link l is exactly the same as its actual capacity c_l. In our simulations we compute the link prices, the source rates and the buffer (queue) sizes. Moreover, queueing delays are taken into account in

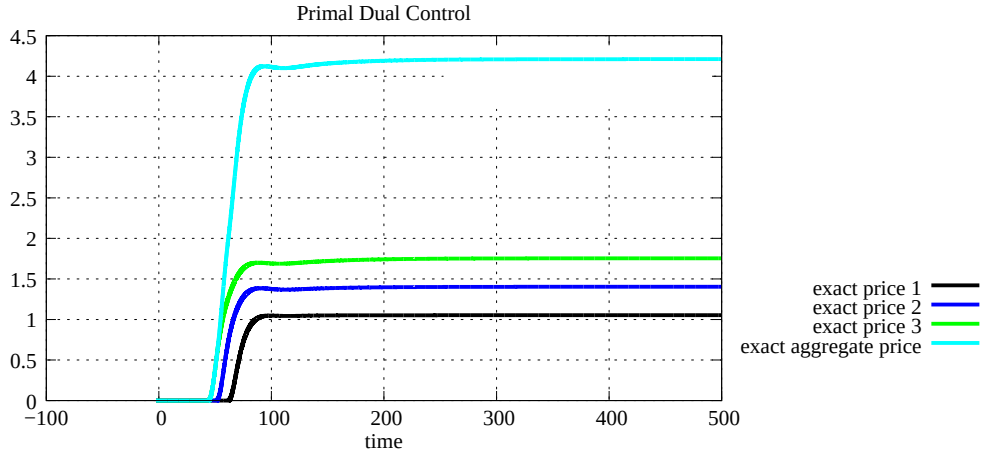


Figure 5.42: Link prices with 3 links (same capacity) and 4 sources (P D C)

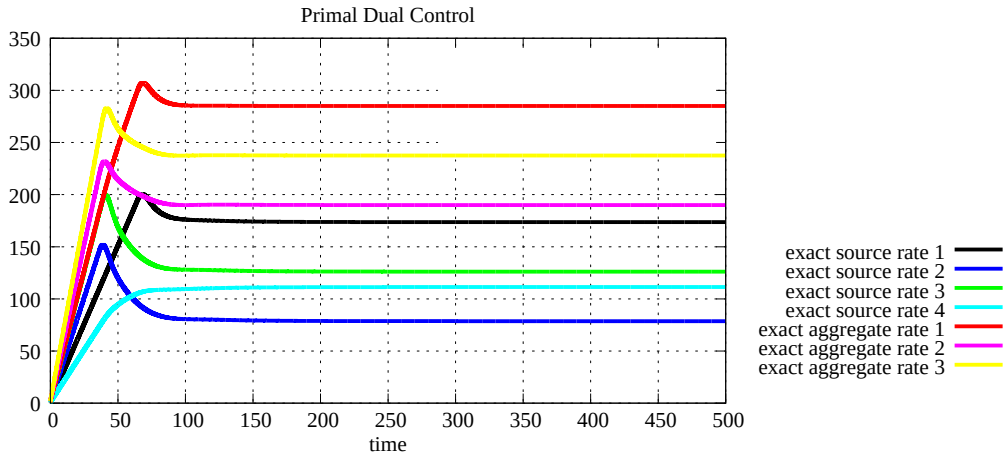


Figure 5.43: Source rates with 3 links and 4 sources (P D C)

the computations and the equilibrium queues are no longer empty. In all the captions of the figures in this section, *C Q D* stands for *control based on queueing delay*. This control is sensitive to all the quantities and parameters mentioned at the beginning of section 5.2.

5.3.1 One link and one source

The network is illustrated in Figure 5.1, page 48. In this case we follow the same considerations as in the corresponding case in the primal–dual control

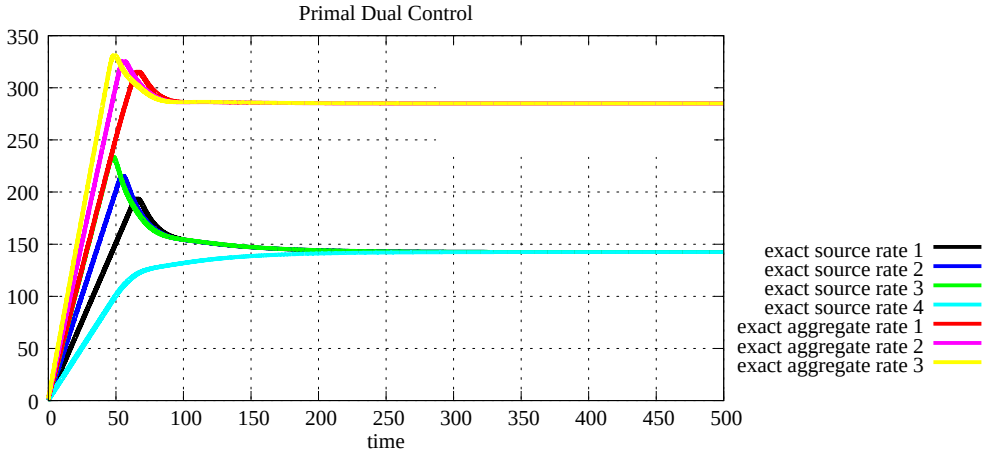


Figure 5.44: Source rates with 3 links (same capacity) and 4 sources (P D C)

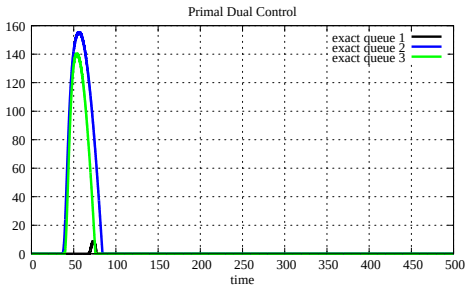


Figure 5.45: Queue sizes with 3 links and 4 sources (P D C)

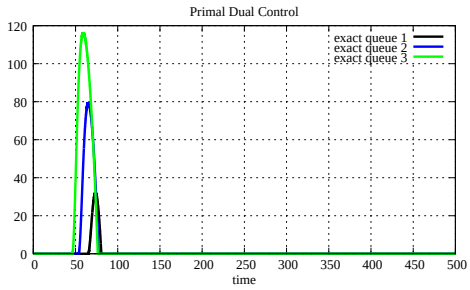


Figure 5.46: Queue sizes with 3 links (same capacity) and 4 sources (P D C)

law. Figures 5.47 to 5.52 represent the link prices, the source rates and the queue sizes in both cases.

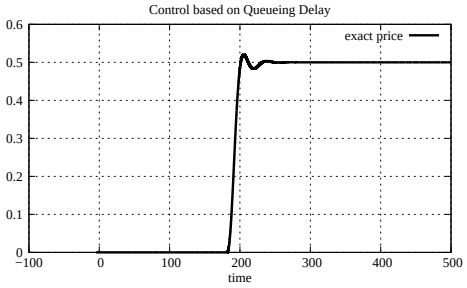


Figure 5.47: Link price, case 1 with 1 link and 1 source (C Q D)

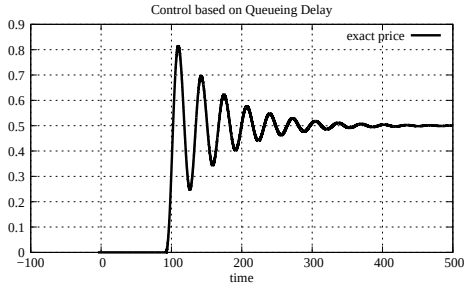


Figure 5.48: Link price, case 2 with 1 link and 1 source (C Q D)

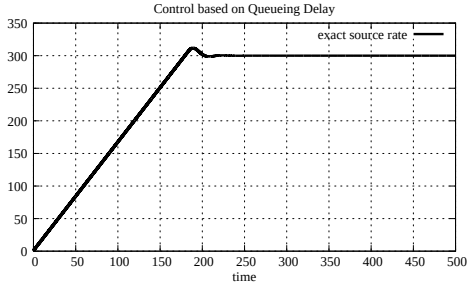


Figure 5.49: Source rate, case 1 with 1 link and 1 source (C Q D)

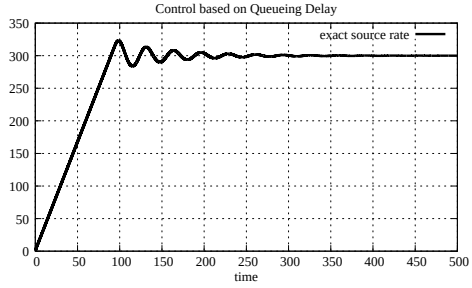


Figure 5.50: Source rate, case 2 with 1 link and 1 source (C Q D)

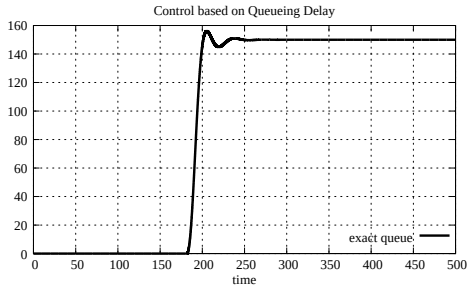


Figure 5.51: Queue size, case 1 with 1 link and 1 source (C Q D)

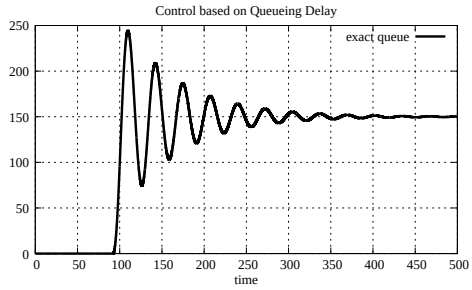


Figure 5.52: Queue size, case 2 with 1 link and 1 source (C Q D)

5.3.2 One link and three sources

The network is illustrated in Figure 5.8, page 49. In this case we follow the same considerations as in the corresponding case in the primal–dual control law. Figures 5.53 to 5.58 represent the link prices, the source rates and the queue sizes in both cases (simple and fairness cases).

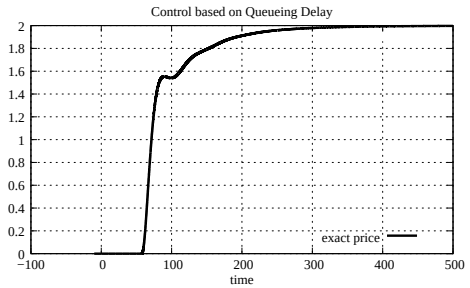


Figure 5.53: Link price, simple case with 1 link and 3 sources (C Q D)

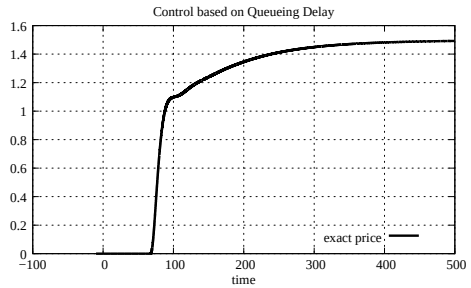


Figure 5.54: Link price, fairness case with 1 link and 3 sources (C Q D)

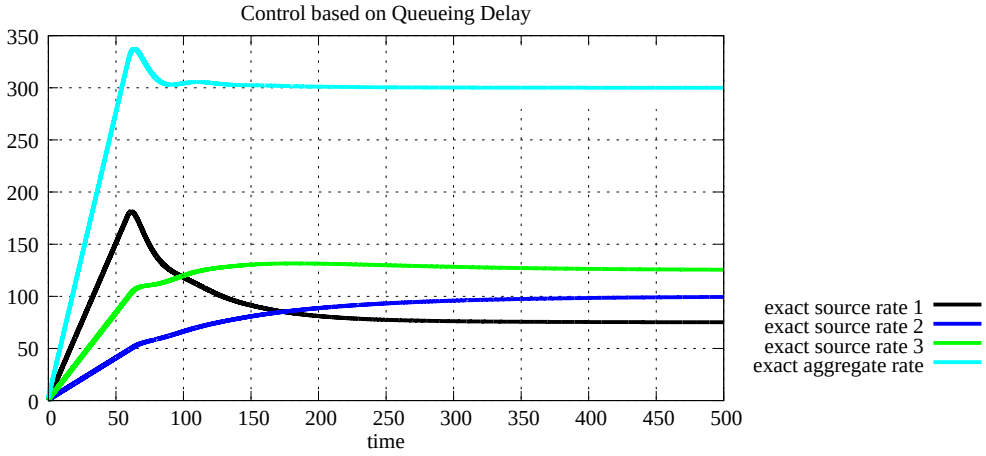


Figure 5.55: Source rates, simple case with 1 link and 3 sources (C Q D)

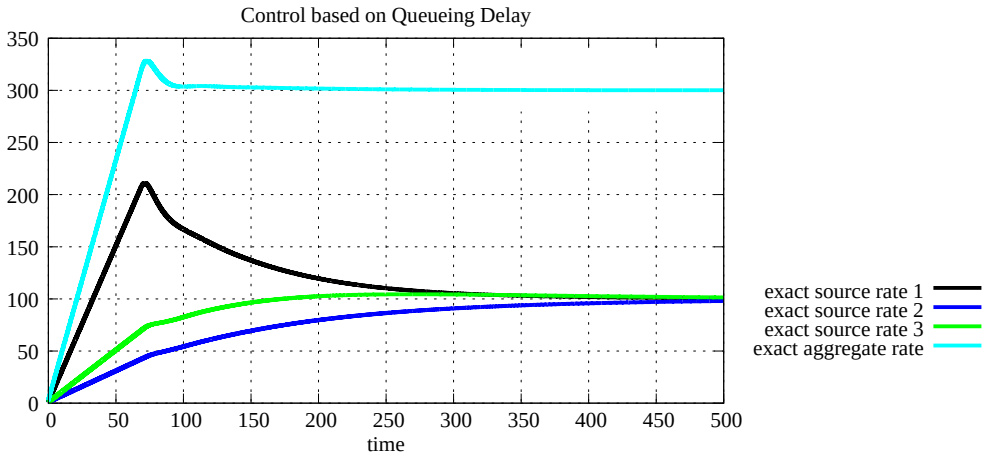


Figure 5.56: Source rates, fairness case with 1 links and 3 sources (C Q D)

5.3.3 Three links and four sources

The corresponding network is illustrated in Figure 5.22, page 53. In this case we follow the same considerations as in the corresponding case in the primal–dual control law. Figures 5.59 to 5.64 represent the link prices, the source rates and the buffer sizes in both cases (same capacity and different capacities cases).

We can observe that when the links have the same capacity, sources 1, 2 and 3 have the same equilibrium rate. This is due to the fact that the

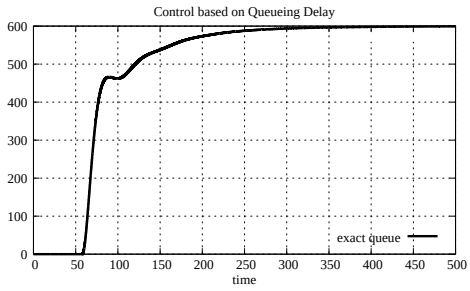


Figure 5.57: Queue size, simple case with 1 link and 3 sources (C Q D)

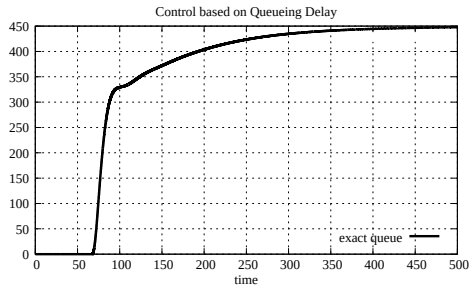


Figure 5.58: Queue size, fairness case with 1 link and 3 sources (C Q D)

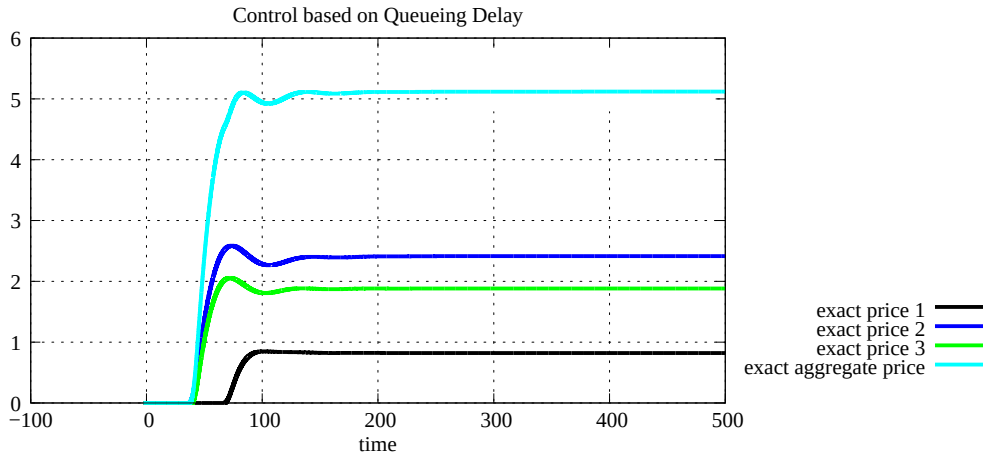


Figure 5.59: Link prices with 3 links and 4 sources (C Q D)

links have the same capacity, and each of the mentioned sources is sharing its link with source 4. It follows immediately that the aggregate rates at equilibrium are the same for all the links, since they have the same virtual capacity.

Conclusion

We have used the numerical methods described in Chapter 3 to solve the equations governing the dynamics in the three control laws presented in Chapter 4. We have computed the link prices, the source rates and the queue sizes in some simple cases for the three control laws. The solutions are given in the form of graphical representations. We observed the cases of fairness and non–fairness, the cases when the links have the same capacity

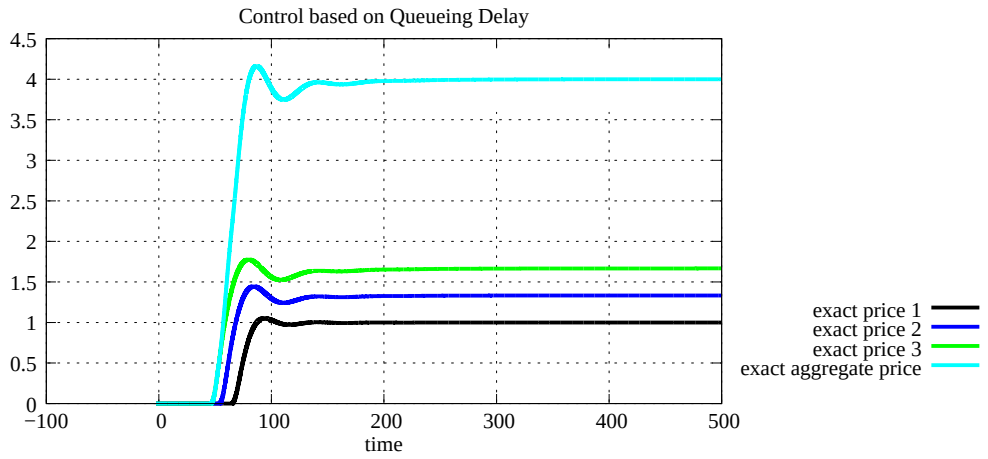


Figure 5.60: Link prices with 3 links (same capacity) and 4 sources (C Q D)

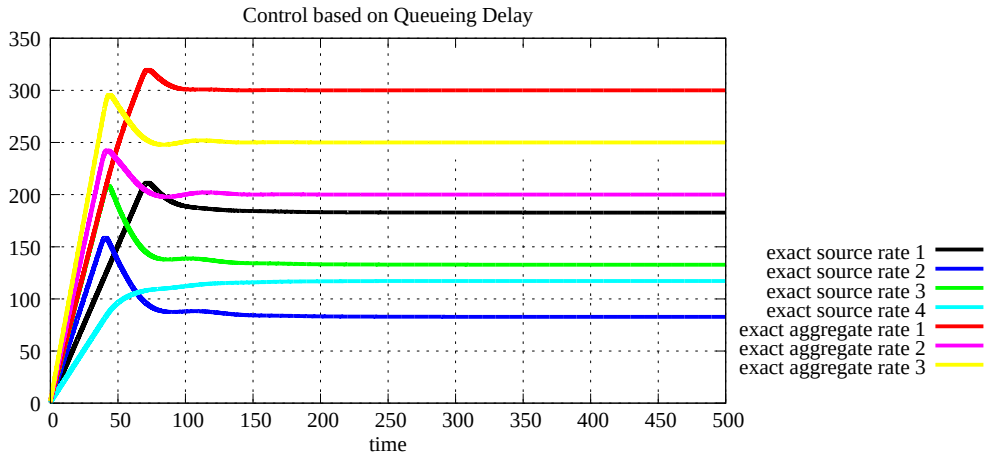


Figure 5.61: Source rates with 3 links and 4 sources (C Q D)

and when they have different capacities.

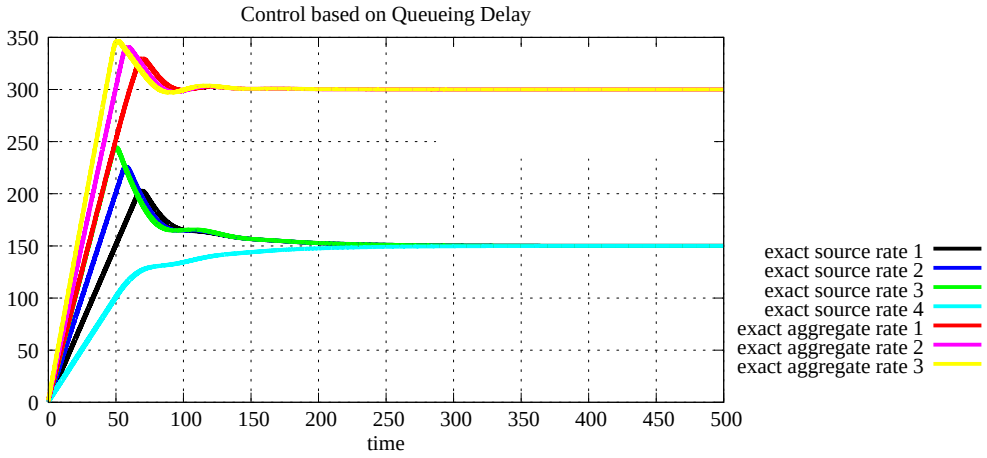


Figure 5.62: Source rates with 3 links (same capacity) and 4 sources (C Q D)

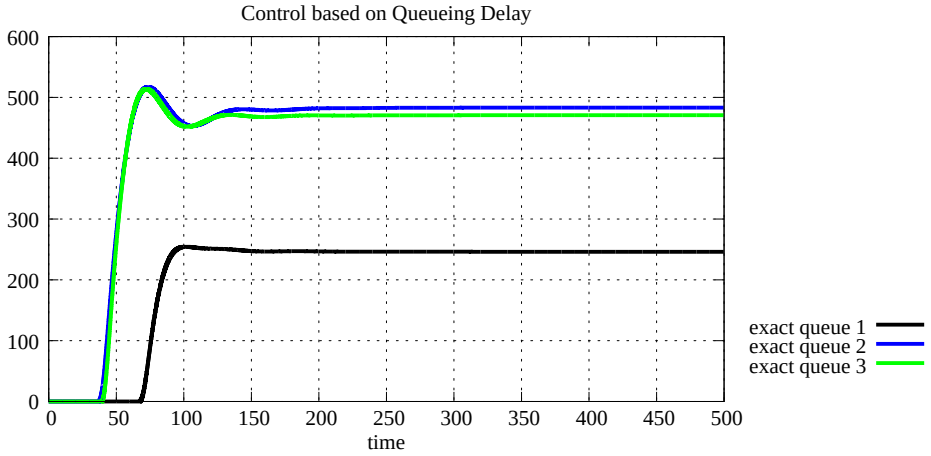


Figure 5.63: Queue sizes with 3 links and 4 sources (C Q D)

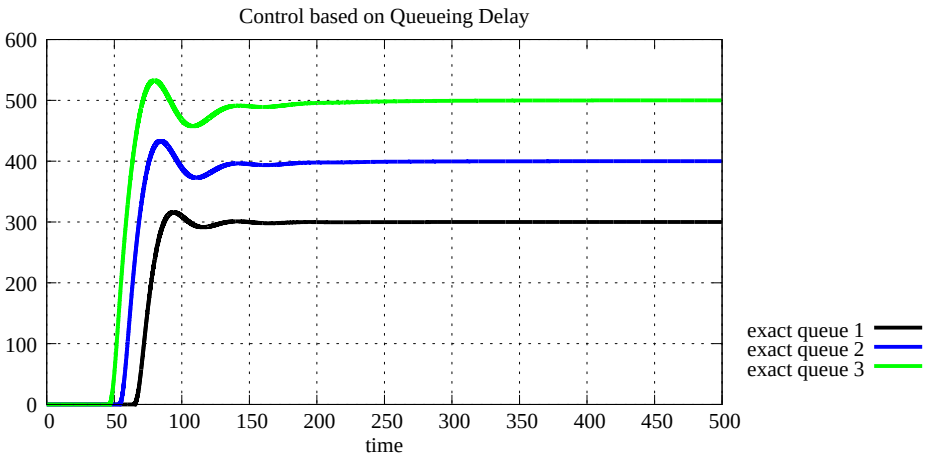


Figure 5.64: Queue sizes with 3 links (same capacity) and 4 sources (C Q D)

Chapter 6

Some Non–Standard Cases

In the standard case of flow models in congestion control as developed in Chapter 4, we consider a network composed of links and sources interacting with each other. Sources are characterised by their transmission rates, the aggregate prices of links on their paths; links are characterised by their capacities, queues, the aggregate rates of sources using these links. The idea of the control is for the sources to communicate their transmission rates to the links on their paths, so that each link will subsequently set a price and communicate it back to the sources for them to adjust their rates, and the cycle repeats; after some time, the network will be fully utilised but link capacities will not be exceeded. However, in the standard case, many things from reality are not considered. Any additional consideration likely to change the general behaviour of the network brings the dynamics from the standard case (studied in Chapters 4 and 5) to a non–standard case, which is more realistic. Some examples of such considerations are when a link breaks in the network, when there is an error in the feedback from sources to links, or from links to sources (which is a more difficult case according to the evolution of the prices in time). We will consider the situations where there is an error in the feedback from sources to links, and the situation where one link breaks in the network.

6.1 Case With Errors

In this case we suppose that the links communicate the exact values to the sources and the sources communicate values with errors to the links. We will adopt the same notations as in the standard case studied in Chapter 4,

but with the superscript e to mention that we are dealing with quantities containing errors from previous computations. The dynamics governing the flow models, for flow rates, link prices and round–trip time become:

$$y_l^e(t) = \sum_{i=1}^{S} R_{li} \left[x_i^e \left(t - \tau_{li}^f \right) + \epsilon_i \left(t - \tau_{li}^f \right) \right] \tag{6.1}$$

$$q_i^e(t) = \sum_{l=1}^{L} R_{li} p_l^e \left(t - \tau_{li}^b \right) \tag{6.2}$$

$$\tau_i = \sum_{l=1}^{L} R_{li} \left(\tau_{li}^f + \tau_{li}^b \right) \tag{6.3}$$

where the R_{li} are the components of the routing matrix, $\epsilon_i(t)$ is the error in the feedback from source i to links on its path, concerning the source rate measured at time t.

Remark 6.1. A very unlikely situation to happen is when for each link l, at any time t, we have

$$\sum_{i=1}^{S} R_{li} \epsilon_i(t) = 0;$$

and if it happens, then the network will just behave as if there is no error, since each link adjusts its price according to the aggregate flow rate of sources having the link on their path.

Remark 6.2. Sources can *underestimate* or *overestimate* their rates, that is, give to links on their respective paths a feedback of transmission rates lower or higher than the actual rates. We can have for example $x_i^e(t) + \epsilon_i(t) = k \cdot x_i^e(t)$ with $k \in (0, \infty)$. In general, if $k > 1$ (overestimation case), the price is adjusted so that the source will be constrained to transmit at a rate lower than normal; and if $k < 1$ (underestimation case), the price is adjusted so that the source will be allowed to transmit at a rate higher than normal. But in particular, at any time t:

- if $k = \theta$ where $\tilde{c}_l = \theta c_l$ with θ conveniently chosen[1], then we return to the case where the target capacity equals the actual capacity. In this case, there is a constant nonzero (positive) queue at equilibrium;
- if $k < \theta$, the length of the queue is expected to go to infinity;

[1] Chosen in the interval $(0, 1)$ and close enough to 1

- if $k \in (\theta, 1)$, the length of the queue is expected to go to zero slower than normal.
- if $k > 1$, the length of the queue is expected to go to zero faster than normal.
- if $k = 1$, there is no error in the feedback from sources to links.

However, in practice, at any time t, each source can underestimate or overestimate its rate (and give a wrong feedback to links on its path), regardless of what happened earlier, and independently of the other sources in the network. At any time t, the error $\epsilon(t)$ for a source is a randomly chosen value around 0, in some range (that is, in a closed interval centred at 0). Let us consider the sequence $t_0 < t_1 < \ldots < t_N = t_f$, where t_0 is the initial time and t_f the final time, and define, in discrete time, for any $i \in \{0, 1, 2, \ldots, N\}$, the random variable X_i on $[t_0, t_f]$ by:

$$X_i(t) = \begin{cases} x_i & \text{if } t = t_i \\ 0 & \text{otherwise} \end{cases}$$

where x_i is a randomly chosen value around 0. For any $i \in \{0, 1, 2, \ldots, N\}$, the random variable X_i is a *Bernoulli Variable* (if $x_i \neq 0$), and the error at any time t in the feedback of a source to a link on its path is a sum of independent identically distributed random variables (Bernoulli Variables), that is, $\epsilon = \sum_{i=0}^{N} X_i$. By the *Central Limit Theorem*[2], the probability distribution of ϵ tends to a *Gaussian Distribution* as N goes to infinity, that is, when we move from discrete time to continuous time. Therefore the most intuitive and reasonable type of error to investigate is the *Gaussian white noise*.

Remark 6.3. If the error is a Gaussian white noise, the dynamics are very unlikely to have a smooth behaviour. However, we can expect the source rates and link prices to fluctuate around the *exact* values, but the queues can have a different behaviour by not necessarily fluctuating around the exact values.

Remark 6.4. All the computer simulations' codes are written in Python programming language. Different configurations of the network topology are associated to different computer simulations' codes in each control law.

[2]Can be found in any standard document on Probability Theory

The parameters are the same as in the standard case, but there is an error in the dynamics. In our simulations, we compute the link prices, the source rates and the buffer sizes. We will layout the cases of overestimation, underestimation, and Gaussian white noise in the feedback from sources to links. We will also focus on the case of fairness among sources sharing the same link.

6.1.1 The Dual Control Approach

Formulation

As in the standard case, the dynamics governing the link prices, source rates and buffer sizes are given as follows, for each $l \in \mathcal{L}$ and for each $i \in \mathcal{S}$:

$$\dot{p}_l^e(t) = \begin{cases} \frac{y_l^e(t)-\tilde{c}_l}{\tilde{c}_l} & \text{if } p_l^e(t) > 0 \\ \max\left\{0, \frac{y_l^e(t)-\tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l^e(t) = 0 \end{cases} \tag{6.4}$$

$$x_i^e(t) = x_{\max,i} \exp\left(\frac{-\alpha_i}{M_i \tau_i} q_i^e(t)\right) \tag{6.5}$$

$$\dot{b}_l^e(t) = \begin{cases} \tilde{y}_l^e(t) - c_l & \text{if } b_l^e(t) > 0 \\ \max\{0, \tilde{y}_l^e(t) - c_l\} & \text{if } b_l^e(t) = 0 \end{cases} \tag{6.6}$$

$$\tilde{y}_l^e(t) = \sum_{i=1}^{S} R_{li} x_i^e\left(t - \tau_{li}^f\right). \tag{6.7}$$

We can reformulate this as follows:

$$\dot{p}_l^e(t) = \begin{cases} \lambda_l^e(t) & \text{if } p_l^e(t) > 0 \\ \max\{0, \lambda_l^e(t)\} & \text{if } p_l^e(t) = 0 \end{cases} \tag{6.8}$$

$$x_i^e(t) = x_{\max,i} \exp\left(\frac{-\alpha_i}{M_i \tau_i} \sum_{l=1}^{L} R_{li} p_l^e\left(t - \tau_{li}^b\right)\right) \tag{6.9}$$

$$\dot{b}_l^e(t) = \begin{cases} \mu_l^e(t) & \text{if } b_l^e(t) > 0 \\ \max\{0, \mu_l^e(t)\} & \text{if } b_l^e(t) = 0 \end{cases} \tag{6.10}$$

$$\lambda_l^e(t) = \sum_{i=1}^{S} \frac{R_{li}}{\tilde{c}_l}\left[x_i^e\left(t - \tau_{li}^f\right) + \epsilon_i\left(t - \tau_{li}^f\right)\right] - 1 \tag{6.11}$$

$$\mu_l^e(t) = \sum_{i=1}^{S} R_{li} x_i^e\left(t - \tau_{li}^f\right) - c_l. \tag{6.12}$$

Computer Simulations

The equations governing the dynamics are given above. We follow the same considerations as in the standard case, but taking errors into account. We consider two underestimation cases, one overestimation case and the Gaussian white noise case.

One link and one source Figures A.1 to A.12, from page 83 to page 85 represent the link prices, source rates and queue sizes in the four cases.

One link and three sources Figures A.13 to A.24, from page 85 to page 88, represent the link prices, source rates and queue sizes in the four cases.

Three links and four sources The link prices, source rates and queue sizes in the four cases are represented in figures A.25 to A.36, from page 88 to page 92.

6.1.2 The Primal–Dual Control Approach

Formulation

As in the standard case, the dynamics governing the link prices and source rates are given as follows, for each $l \in \mathcal{L}$ and for each $i \in \mathcal{S}$:

$$\dot{p}_l^e(t) = \begin{cases} \frac{y_l^e(t)-\tilde{c}_l}{\tilde{c}_l} & \text{if } p_l^e(t) > 0 \\ \max\left\{0, \frac{y_l^e(t)-\tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l^e(t) = 0 \end{cases} \tag{6.13}$$

$$x_i^e(t) = x_{m,i} \exp\left(\xi_i^e(t) - \frac{\alpha_i}{M_i \tau_i} q_i^e(t)\right) \tag{6.14}$$

$$\dot{\xi}_i^e(t) = \frac{\beta_i}{\tau_i}\big(U_i'(x_i^e(t) + \epsilon_i(t)) - q_i^e(t)\big) \tag{6.15}$$

$$\dot{b}_l^e(t) = \begin{cases} \tilde{y}_l^e(t) - c_l & \text{if } b_l^e(t) > 0 \\ \max\{0, \tilde{y}_l^e(t) - c_l\} & \text{if } b_l^e(t) = 0 \end{cases} \tag{6.16}$$

$$\tilde{y}_l^e(t) = \sum_{i=1}^{S} R_{li} x_i^e\left(t - \tau_{li}^f\right). \tag{6.17}$$

We can reformulate the above dynamics as follows:

$$\dot{p}_l^e(t) = \begin{cases} \lambda_l^e(t) & \text{if } p_l^e(t) > 0 \\ \max\{0, \lambda_l^e(t)\} & \text{if } p_l^e(t) = 0 \end{cases} \tag{6.18}$$

$$x_i^e(t) = x_{m,i} \exp\left(\xi_i^e(t) - \frac{\alpha_i}{M_i \tau_i} \sum_{l=1}^{L} R_{li} p_l^e\left(t - \tau_{li}^b\right)\right) \tag{6.19}$$

$$\dot{\xi}_i^e(t) = \frac{\beta_i}{\tau_i}\left(U_i'(x_i^e(t) + \epsilon_i(t)) - \sum_{l=1}^{L} R_{li} p_l^e\left(t - \tau_{li}^b\right)\right) \tag{6.20}$$

$$\dot{b}_l^e(t) = \begin{cases} \mu_l^e(t) & \text{if } b_l^e(t) > 0 \\ \max\{0, \mu_l^e(t)\} & \text{if } b_l^e(t) = 0 \end{cases} \tag{6.21}$$

$$\lambda_l^e(t) = \sum_{i=1}^{S} \frac{R_{li}}{\tilde{c}_l}\left[x_i^e\left(t - \tau_{li}^f\right) + \epsilon_i\left(t - \tau_{li}^f\right)\right] - 1 \tag{6.22}$$

$$\mu_l^e(t) = \sum_{i=1}^{S} R_{li} x_i^e\left(t - \tau_{li}^f\right) - c_l. \tag{6.23}$$

Computer Simulations

The equations governing the dynamics are given above. We follow the same considerations as in the standard case, and also consider errors. As in the dual control case, we consider two underestimation cases, one overestimation case and the Gaussian white noise case.

One link and one source Figures A.37 to A.48, from page 93 to page 94 represent the link prices, source rates and queue sizes in the four cases.

One link and three sources Figures A.49 to A.60, from page 95 to page 97 represent the link prices, source rates and queue sizes in the four cases.

Three links and four sources Graphical representations of the link prices, source rates and queue sizes in the four cases are given in figures A.61 to A.72, from page 98 to page 102.

6.1.3 The Control Approach based on Queueing Delay

Formulation

As in the standard case, the dynamics governing the link prices and source rates are given as follows:

$$\dot{p}_l^e(t) = \begin{cases} \frac{y_l^e(t)-\tilde{c}_l}{\tilde{c}_l} & \text{if } p_l^e(t) > 0 \\ \max\left\{0, \frac{y_l^e(t)-\tilde{c}_l}{\tilde{c}_l}\right\} & \text{if } p_l^e(t) = 0 \end{cases} \tag{6.24}$$

$$x_i^e(t) = x_{m,i} \exp\left(\xi_i^e(t)\right) \left(\frac{d_i}{d_i + q_i^e(t)}\right)^{\frac{\alpha_i}{M_i}} \tag{6.25}$$

$$\dot{\xi}_i^e(t) = \left(\frac{\beta_i}{d_i + q_i^e(t)}\right) \left(U_i'(x_i^e(t) + \epsilon_i(t)) - q_i^e(t)\right) \tag{6.26}$$

$$\dot{b}_l^e(t) = \begin{cases} \tilde{y}_l^e(t) - c_l & \text{if } b_l^e(t) > 0 \\ \max\{0, \tilde{y}_l^e(t) - c_l\} & \text{if } b_l^e(t) = 0 \end{cases} \tag{6.27}$$

$$\tilde{y}_l^e(t) = \sum_{i=1}^{S} R_{li} x_i^e \left(t - \tau_{li}^f\right). \tag{6.28}$$

We can reformulate the above dynamics as follows:

$$\dot{p}_l^e(t) = \begin{cases} \lambda_l^e(t) & \text{if } p_l^e(t) > 0 \\ \max\left\{0, \lambda_l^e(t)\right\} & \text{if } p_l^e(t) = 0 \end{cases} \tag{6.29}$$

$$x_i^e(t) = x_{m,i} \exp\left(\xi_i^e(t)\right) \left(\frac{d_i}{d_i + q_i^e(t)}\right)^{\frac{\alpha_i}{M_i}} \tag{6.30}$$

$$\dot{\xi}_i^e(t) = \left(\frac{\beta_i}{d_i + q_i^e(t)}\right) \left(U_i'(x_i^e(t) + \epsilon_i(t)) - q_i^e(t)\right) \tag{6.31}$$

$$\dot{b}_l^e(t) = \begin{cases} \mu_l^e(t) & \text{if } b_l^e(t) > 0 \\ \max\{0, \mu_l^e(t)\} & \text{if } b_l^e(t) = 0 \end{cases} \tag{6.32}$$

$$\lambda_l^e(t) = \sum_{i=1}^{S} \frac{R_{li}}{\tilde{c}_l} \left[x_i^e \left(t - \tau_{li}^f\right) + \epsilon_i \left(t - \tau_{li}^f\right)\right] - 1 \tag{6.33}$$

$$\mu_l^e(t) = \sum_{i=1}^{S} R_{li} x_i^e \left(t - \tau_{li}^f\right) - c_l. \tag{6.34}$$

where

$$q_i^e(t) = \sum_{l=1}^{L} R_{li} p_l^e \left(t - \tau_{li}^b\right).$$

Computer Simulations

The equations governing the dynamics are given above. We follow the same considerations as in the standard case, but account is taken of errors. As in the dual control case, we consider two underestimation cases, one overestimation case and the Gaussian white noise case.

One link and one source Figures A.73 to A.84, from page 103 to page 104 represent the link prices, source rates and queue sizes in the four cases.

One link and three sources Figures A.85 to A.96, from page 105 to page 107 represent the link prices, source rates and queue sizes in the four cases.

Three links and four sources Figures A.97 to A.108, from page 108 to page 112 represent the link prices, source rates and queue sizes in the four cases.

6.2 Impact of Errors on the Network Dynamics

At any time t, there is an error in the dynamics due to the feedback from sources to links, and this error is taken into account in the computation of all the subsequent values in the network dynamics. This gives rise to a *cumulative error* in link prices, source rates and buffer sizes. Precisely, the errors on the feedback from sources to links on their respective paths have an effect on the change of link prices, and therefore on the prices. The effect of the errors on the source rates is almost the same on buffer sizes. The errors obtained on link prices then affect the next computation of source rates, and the cycle repeats.

6.3 When a Link Breaks in the Network

We will explore this case in the example of a network composed of three links and four sources as illustrated in Figure 1.3 at page 5. Figure 6.1 illustrates the network described at page 5, and the reduced network when the second link breaks. We expect the flows involving the broken link to stop and the rest of the network to continue with the remaining dynamics. We

also expect a full utilisation of the remaining links in then network. Using a mathematical interpretation, let us suppose that one link, link l_0 say, breaks at time t_{l_0}. Each source i having l_0 on its path, (that is $R_{l_0 i} = 1$) will take a delay $\tau^b_{l_0 i}$ to realise that the link is broken. The transmission rate $x_i(t)$ is expected to jump (discontinuously) to zero at time $t_{l_0} + \tau^b_{l_0 i}$, and to remain zero from that time. Simultaneously, the remaining sources of the network adjust their transmission rates accordingly, that is, in such a way that the remaining links become fully utilised again, and not congested. Also, for any link l used by a source i having l_0 on its path, the buffer size b_l will be directly affected by the link breaking, and decrease (at least for a moment, eventually decrease to zero). The link breaking can be interpreted by setting its price p_{l_0} to infinity.



A network composed of 3 links and 4 sources

Reduced network when link 2 breaks

Figure 6.1: An example of network where a link breaks

Remark 6.5. If the network is just composed of one link and one or many sources, when the link breaks, obviously everything will stop. So we will focus on networks composed of more than one link and illustrate, as mentioned earlier, with the network of Figure 1.3. Computer simulations' codes are written both in C and Python programming languages. As mentioned before, different configurations of the network topology are associated to different computer simulations' codes in each control law.

6.3.1 The Dual Control Approach

Formulation

The dynamics are the same as in the standard case, but after the feedback from the broken link l_0 to sources is given, all the sources having link l_0 on their path have zero rates (from the expression of the source rates) since $p_{l_0} = \infty$. The change of link prices have the same expression, but $\dot{p}_{l_0} = -1$ and since $p_{l_0} = \infty$, the fact that $\dot{p}_{l_0} = -1$ does not change p_{l_0} from infinity.

Computer Simulations

The dynamics are the same as described at page 30, except that when a link breaks, link l_0 say (that is $p_{l_0} = \infty$), it affects the remaining dynamics after a given delay. Figures 6.2 to 6.5 represent the source rates and queue sizes, with and without noise.

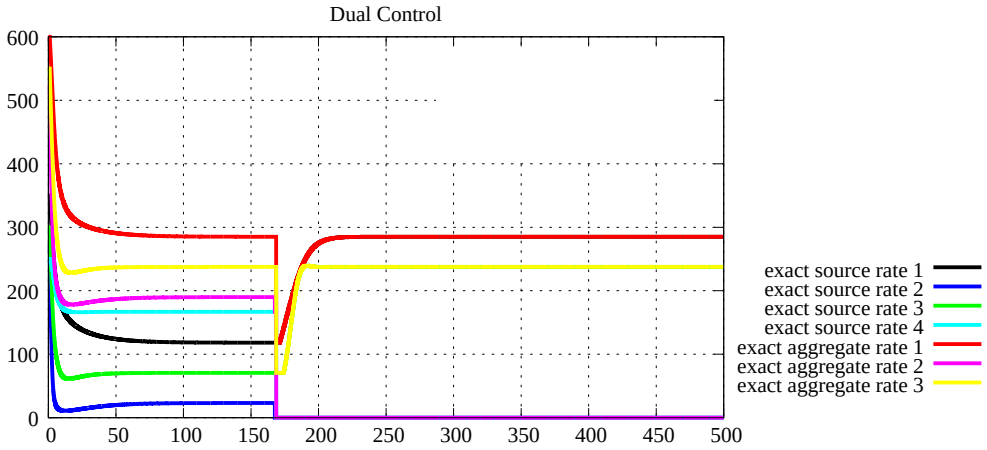


Figure 6.2: Source rates with 3 links and 4 sources (Dual)

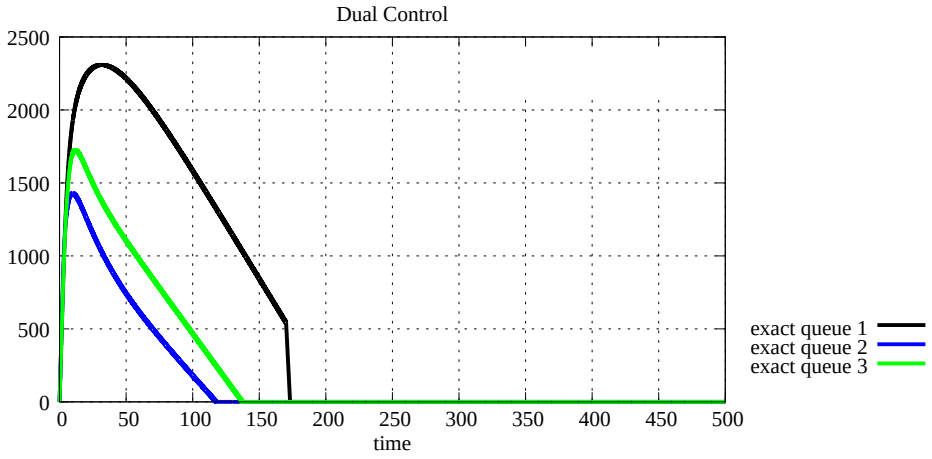


Figure 6.3: Queue size with 3 links and 4 sources (Dual)

It can be observed from the simulations that with or without errors, when link 2 breaks, the transmission rates of sources 2 and 4 (the only sources using link 2) go to zero while sources 1 and 3 adjust their rates to

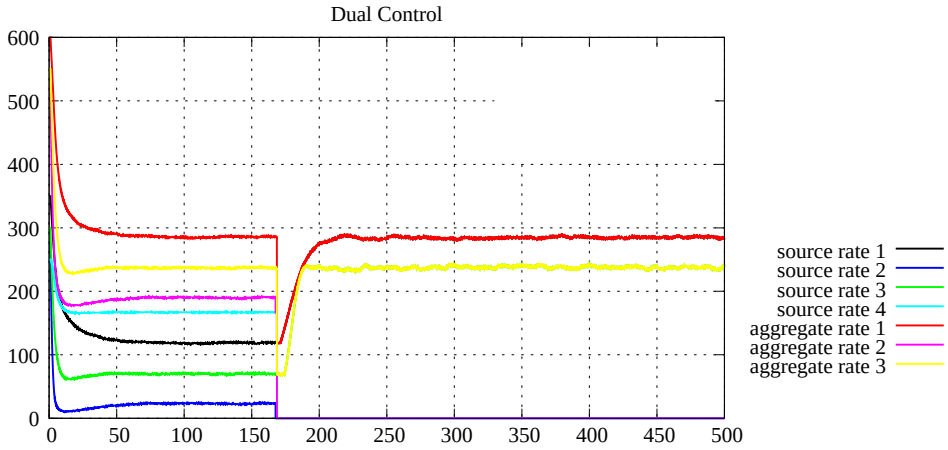


Figure 6.4: Source rates (with noise) with 3 links and 4 sources (Dual)

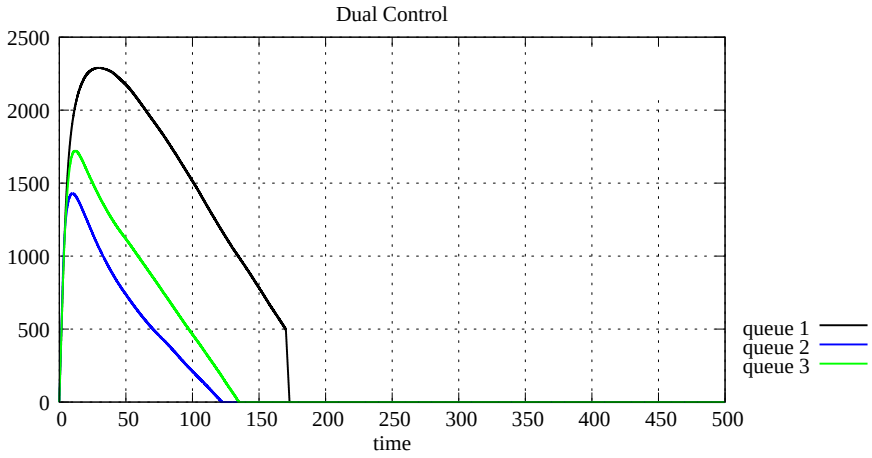


Figure 6.5: Queue size (with noise) with 3 links and 4 sources (Dual)

reach the target capacities of the respective links on their paths. We can also realise when link 2 breaks that all the queues go to zero. But depending on network parameters and on the evolution of the source rates, queues 1 and 3 can increase for a while and then come back to zero.

6.3.2 The Primal–Dual Control Approach

Formulation

From the dynamics governing the primal–dual control law, setting p_{l_0} to infinity makes the x_i and the $\dot{\xi}_i$ to be not necessarily well–defined. Indeed, from the expression of $\dot{\xi}_i$, considering $x_i = 0$ makes $\dot{\xi}_i$ to be not well–defined. Also, taking $\xi_i \neq \infty$ gives $x_i = 0$, and $\dot{\xi}_i$ not well–defined; but taking $\xi_i = \infty$ makes x_i to be not well–defined. This can also be considered as one limitation of the primal–dual control approach.

Computer Simulations

As explained above, a problem arises with some not well–defined quantities, leading to some complications in computations. This case will therefore not be explored further for the moment.

6.3.3 The Control Approach based on Queueing Delay

Formulation

As in the dual control approach, after the feedback from the broken link l_0 to sources is given, all the sources having link l_0 on their path have zero rates. The change of link prices have the same expression, but $\dot{p}_{l_0} = -1$ and $p_{l_0} = \infty$, and remains the same.

Computer Simulations

The dynamics are the same as described at page 44, except that when a link breaks, link l_0 say (that is $p_{l_0} = \infty$), it affects the remaining dynamics after a given delay. Figures 6.6 to 6.9 represent the source rates and queue sizes, with and without noise.

It can be observed from the simulations that with or without errors, when link 2 breaks, the transmission rates of sources 2 and 4 (the only sources using link 2) go to zero while sources 1 and 3 adjust their rates to reach the target capacities (same as the actual capacities) of the respective links on their paths. We can also realise when link 2 breaks that all the queues go to zero. But depending on network parameters and on the evolution of the source rates, queues 1 and 3 increase to reach the respective equilibrium

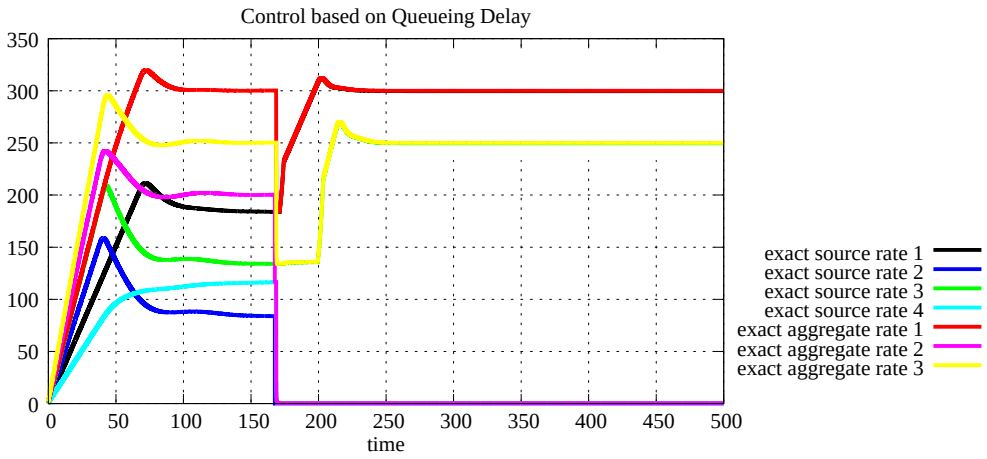


Figure 6.6: Source rates with 3 links and 4 sources (C Q D)

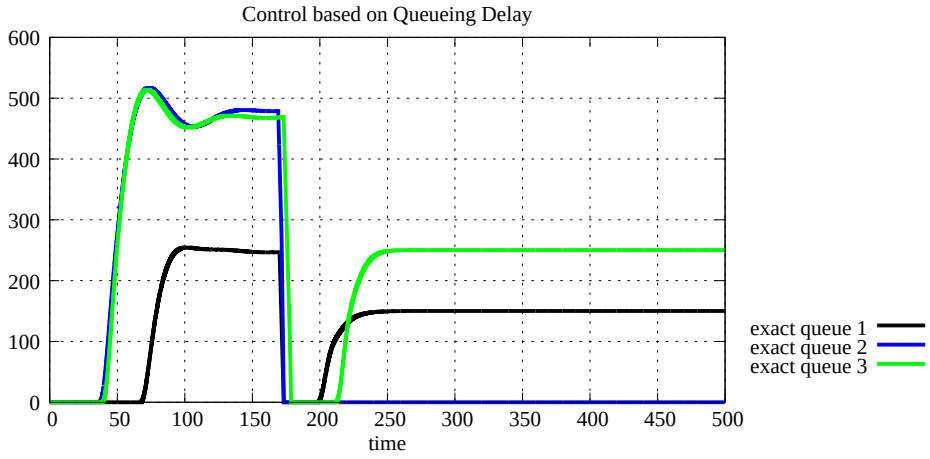


Figure 6.7: Queue size with 3 links and 4 sources (C Q D)

queues (when there is no noise), lower than the respective equilibrium queues when link 2 does not break.

Conclusion

We have formulated and numerically solved (for some simple cases) the equations governing the dynamics in the three flow model control laws when there is an error in the feedback from sources to links. The behaviour of the source rates and link prices is almost the same as in the standard case,

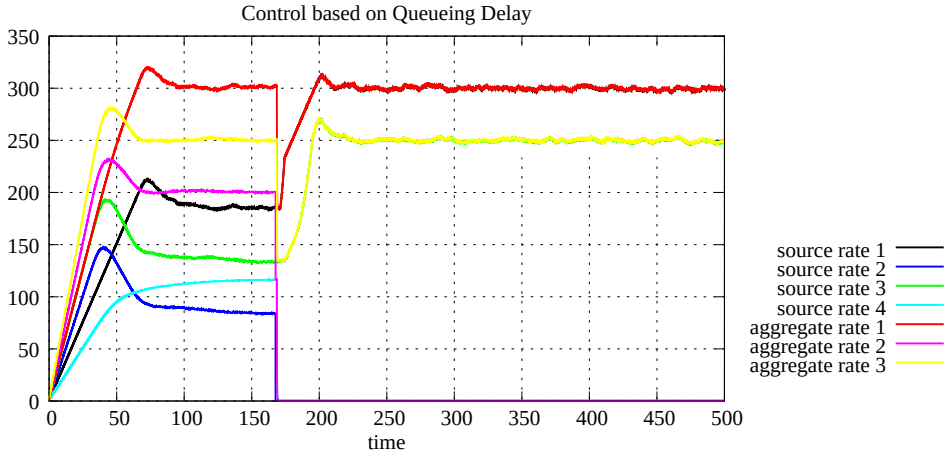


Figure 6.8: Source rates (with noise) with 3 links and 4 sources (C Q D)

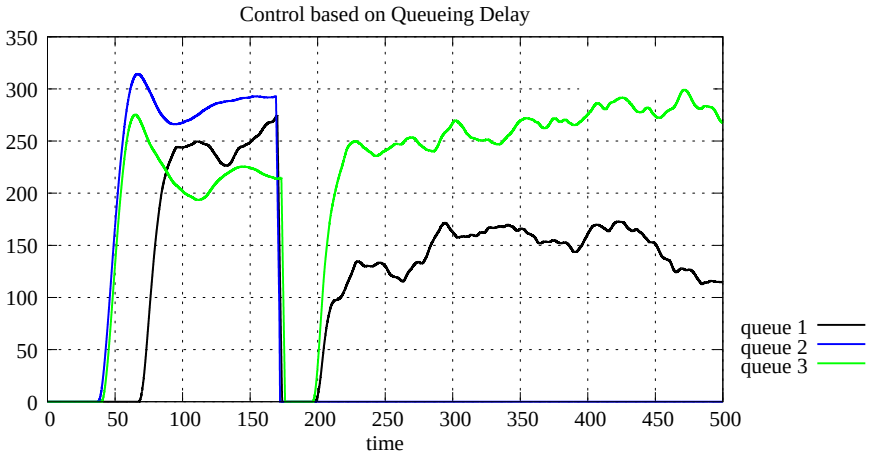


Figure 6.9: Queue size (with noise) with 3 links and 4 sources (C Q D)

depending on the type of error. However, the behaviour of the queue sizes is completely different from the corresponding behaviour in the standard case. We have also formulated the dynamics when a link breaks in the network. We have numerically solved (for some simple cases) the equations only in the dual control law and in the control law based on queueing delay, due to some complications in computation coming from some problems of not well–defined quantities. All the solutions to our equations are given in the form of graphical representations listed in the appendix (A).

General Conclusion

In this work, we explored some areas in applied mathematics useful in solving congestion control problems. We also reviewed some congestion control schemes as developed from the $1980's$. The most recent congestion control models involve delay differential equations. These congestion control models are derived from an optimal control problem, and solving the equations governing the dynamics in the derived congestion control models solves the optimal control problem mentioned earlier. This is done by using numerical methods as studied in Chapter 3, namely the modified Heun's method. In Chapter 3, we actually explored the area of delay differential equations, giving some definitions, conditions on existence and uniqueness of solutions, and describing some numerical methods we can use to solve them. We shall also mention that most of the delay differential equations can only be solved numerically. Indeed, we also studied some control schemes involving delay differential equations, reviewing the system stability in some cases, discussing also fairness conditions in all the control schemes for some simple cases. We observed that in a single link network, and in a multiple links network with links having the same capacity, some network parameters can be conveniently chosen so as to achieve fairness. Since the models studied are too perfect compared to real facts, we started studying some non–standard cases, which are more realistic.

However, computer networks in general and the Internet in particular, to satisfy more customers, need more sophisticated control mechanisms.

The elastic queue approach remains an interesting direction to explore, and a good point to start is in a single link network with one or many flows through a link with constant or time varying bandwidth.

Appendix A

Graphical Representations of the Dynamics in Non–Standard Cases

We list the figures of the link prices, source rates and queue sizes in the case of an error in the feedback from sources to links as described in chapter 6.

A.1 Dual Control

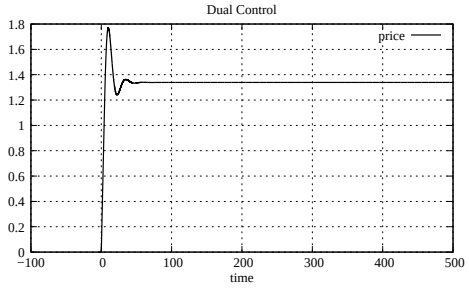


Figure A.1: Link price, case 1 with 1 link and 1 source (Dual)

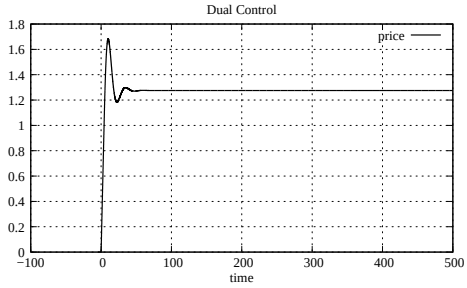


Figure A.2: Link price, case 2 with 1 link and 1 source (Dual)

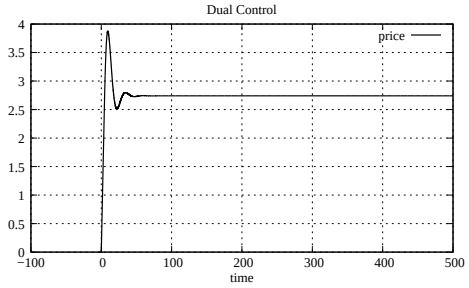


Figure A.3: Link price, case 3 with 1 link and 1 source (Dual)

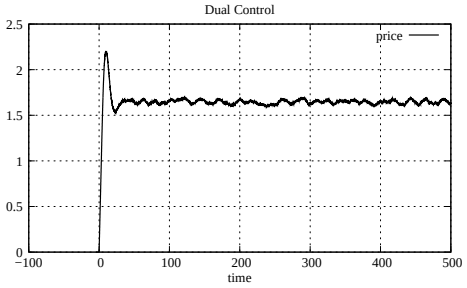


Figure A.4: Link price, case 4 with 1 link and 1 source (Dual)

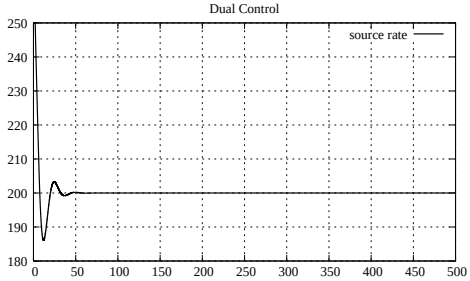


Figure A.5: Source rate, case 1 with 1 link and 1 source (Dual)

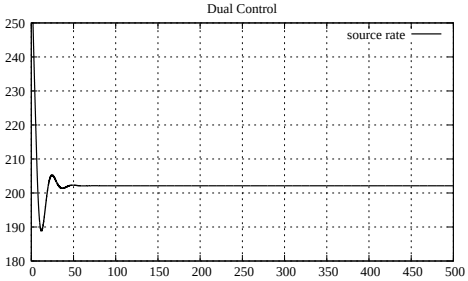


Figure A.6: Source rate, case 2 with 1 link and 1 source (Dual)

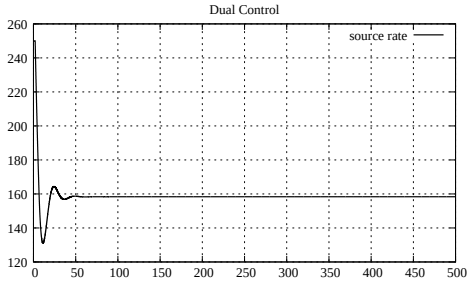


Figure A.7: Source rate, case 3 with 1 link and 1 source (Dual)

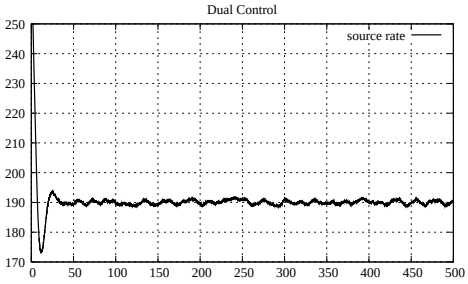


Figure A.8: Source rate, case 4 with 1 link and 1 source (Dual)

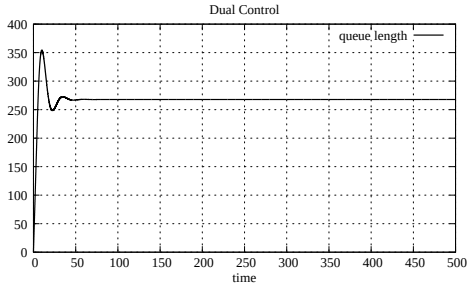


Figure A.9: Queue size, case 1 with 1 link and 1 source (Dual)

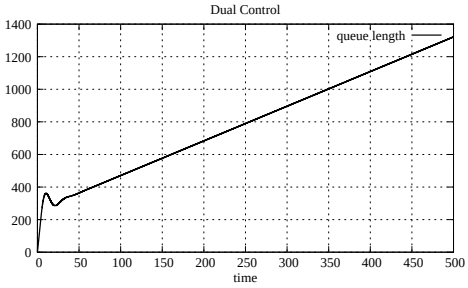


Figure A.10: Queue size, case 2 with 1 link and 1 source (Dual)

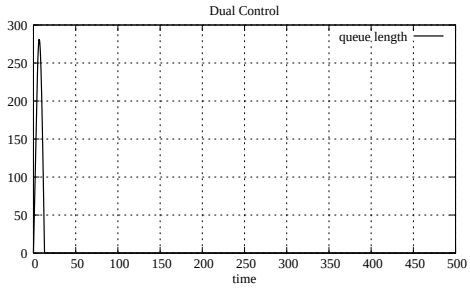


Figure A.11: Queue size, case 3 with 1 link and 1 source (Dual)

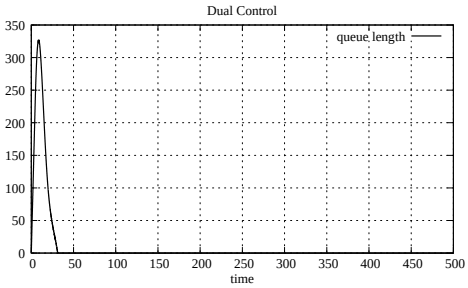


Figure A.12: Queue size, case 4 with 1 link and 1 source (Dual)

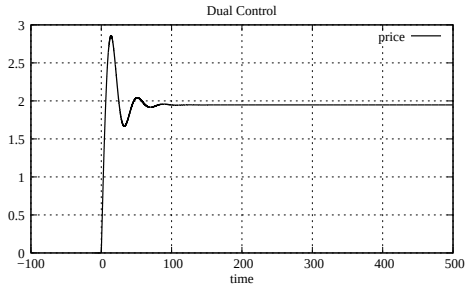


Figure A.13: Link price, case 1 with 1 link and 3 sources (Dual)

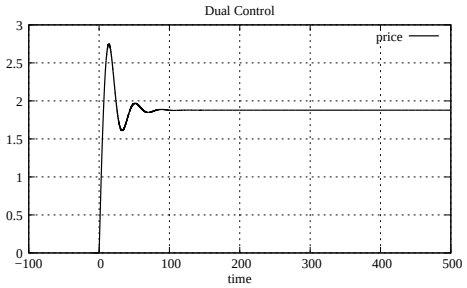


Figure A.14: Link price, case 2 with 1 link and 3 sources (Dual)

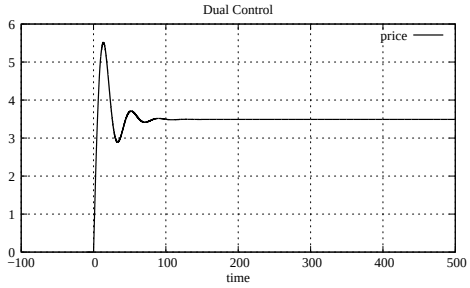


Figure A.15: Link price, case 3 with 1 link and 3 sources (Dual)

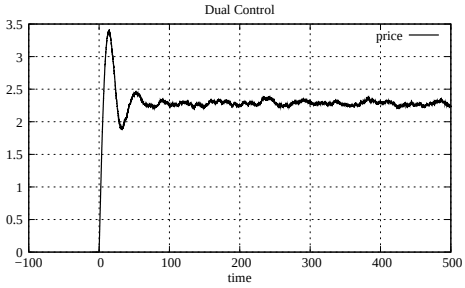


Figure A.16: Link price, case 4 with 1 link and 3 sources (Dual)

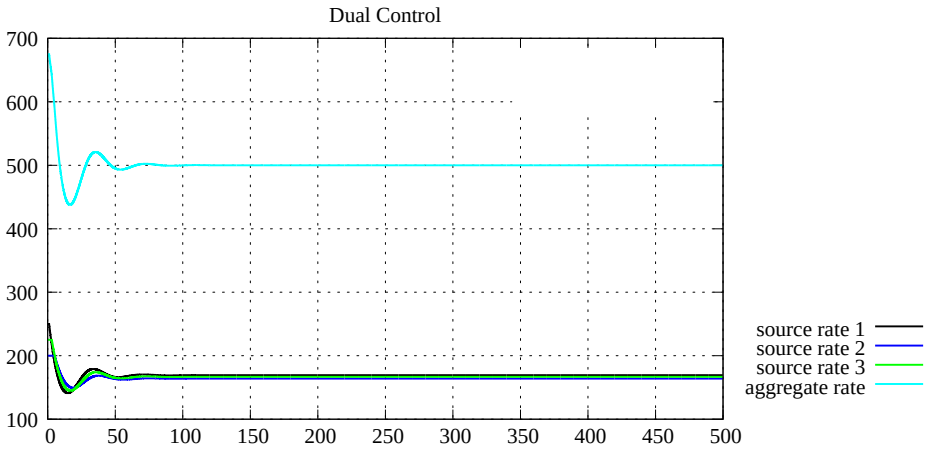


Figure A.17: Source rates, case 1 with 1 link and 3 sources (Dual)

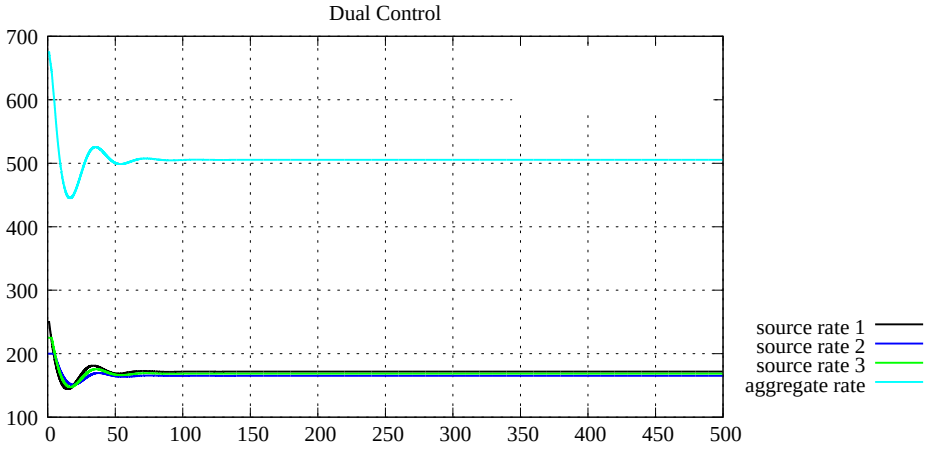


Figure A.18: Source rates, case 2 with 1 link and 3 sources (Dual)

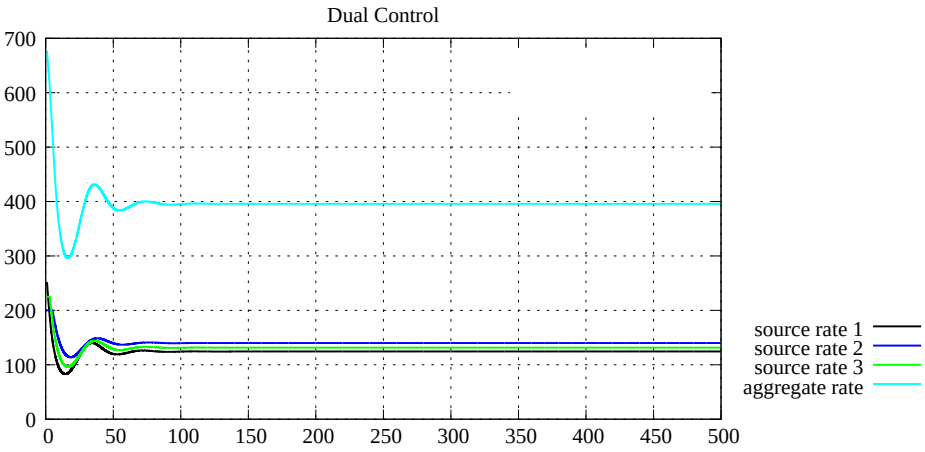


Figure A.19: Source rates, case 3 with 1 link and 3 sources (Dual)

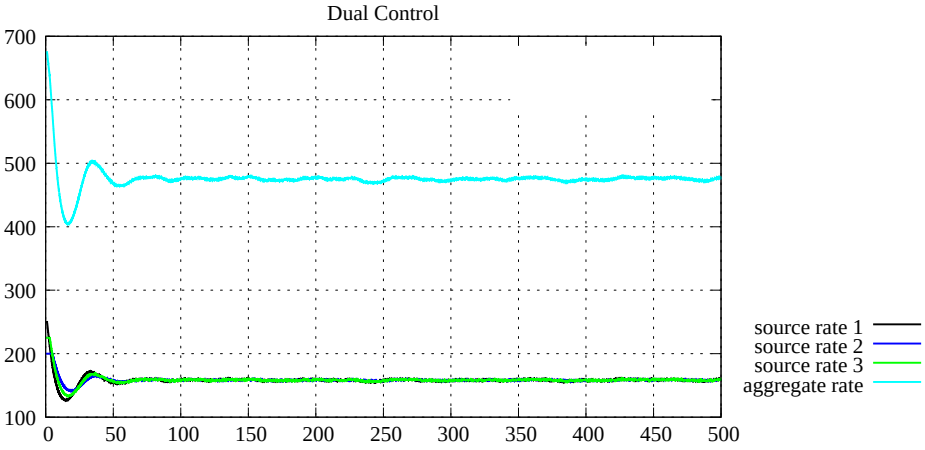


Figure A.20: Source rates, case 4 with 1 link and 3 sources (Dual)

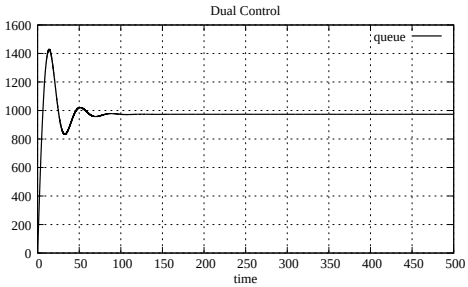


Figure A.21: Queue size, case 1 with 1 link and 3 sources (Dual)

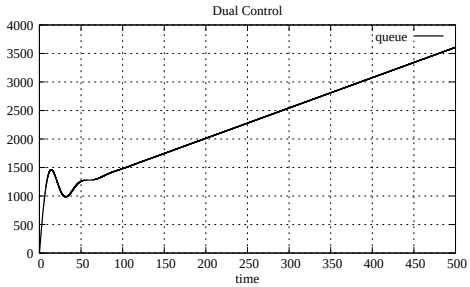


Figure A.22: Queue size, case 2 with 1 link and 3 sources (Dual)

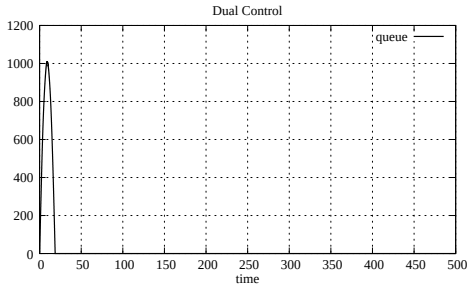


Figure A.23: Queue size, case 3 with 1 link and 3 sources (Dual)

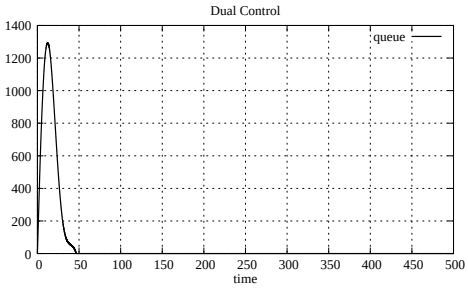


Figure A.24: Queue size, case 4 with 1 link and 3 sources (Dual)

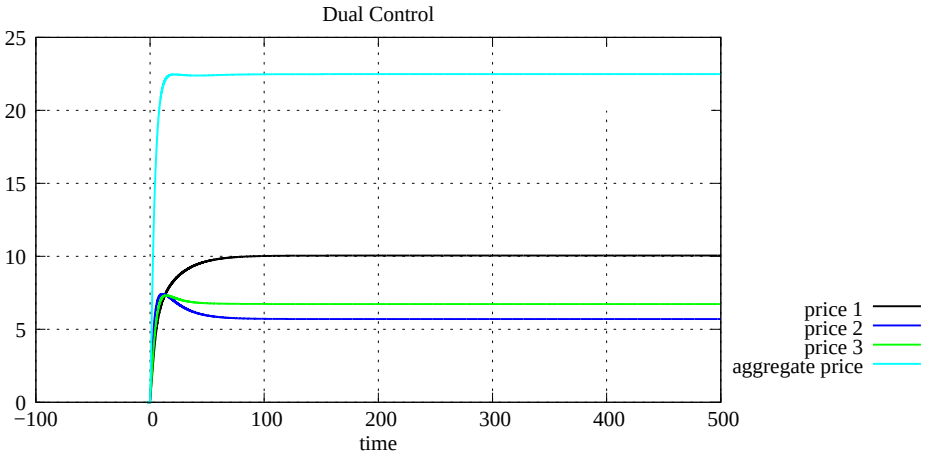


Figure A.25: Link price, case 1 with 3 links and 4 sources (Dual)

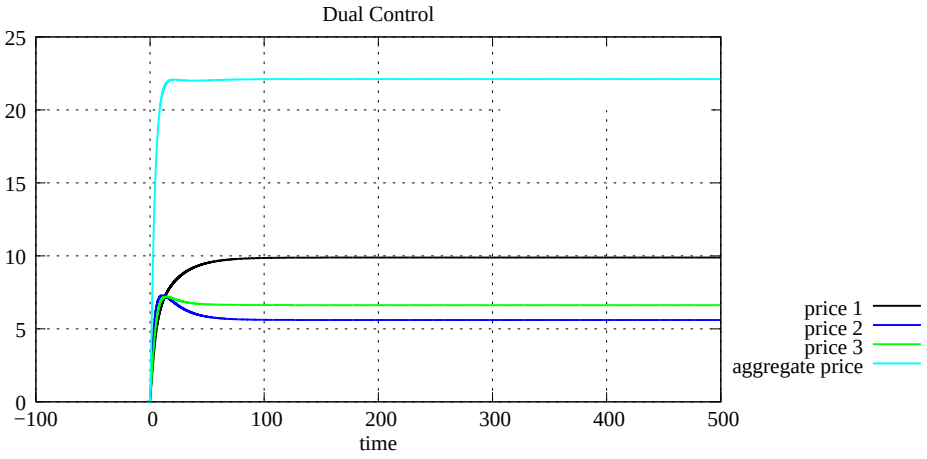


Figure A.26: Link price, case 2 with 3 links and 4 sources (Dual)

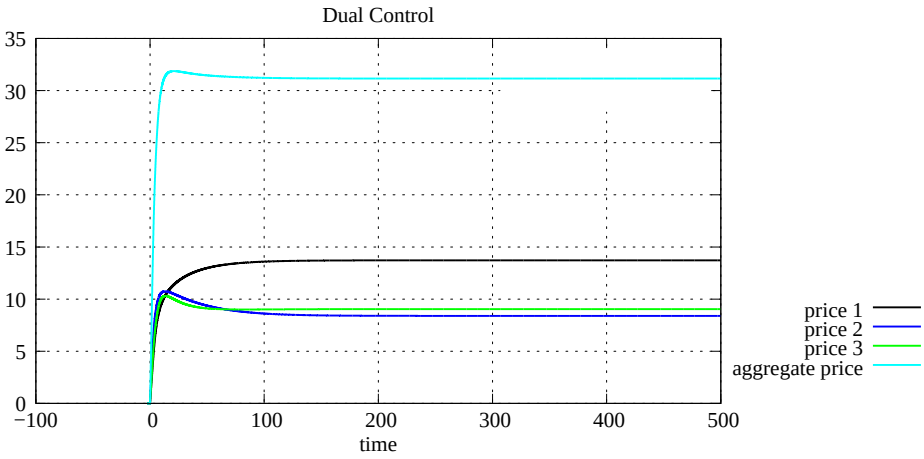


Figure A.27: Link price, case 3 with 3 links and 4 sources (Dual)

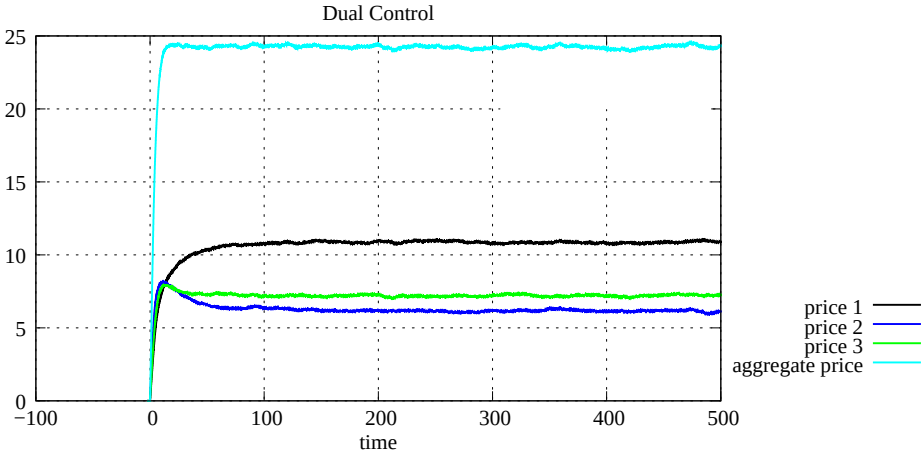


Figure A.28: Link price, case 4 with 3 links and 4 sources (Dual)

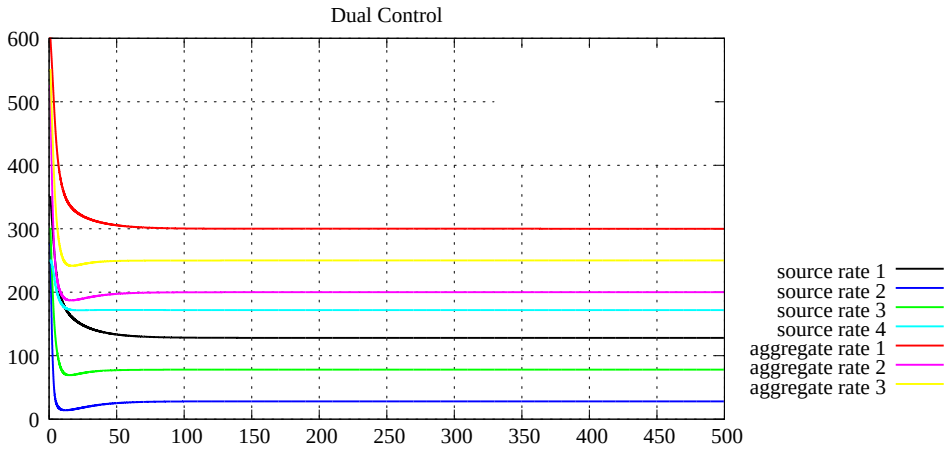


Figure A.29: Source rates, case 1 with 3 links and 4 sources (Dual)

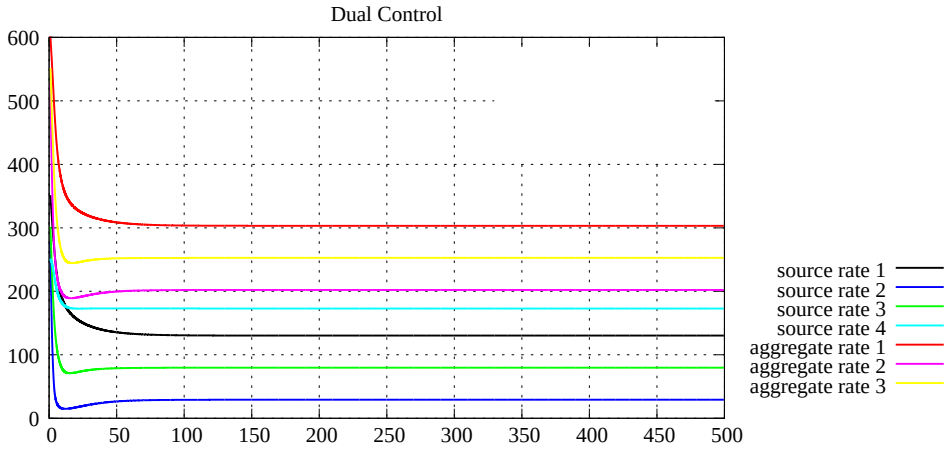


Figure A.30: Source rates, case 2 with 3 links and 4 sources (Dual)

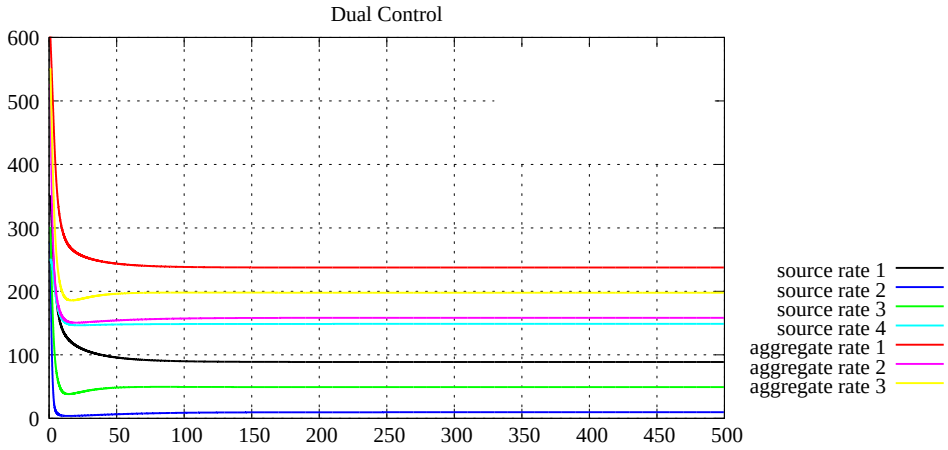


Figure A.31: Source rates, case 3 with 3 links and 4 sources (Dual)

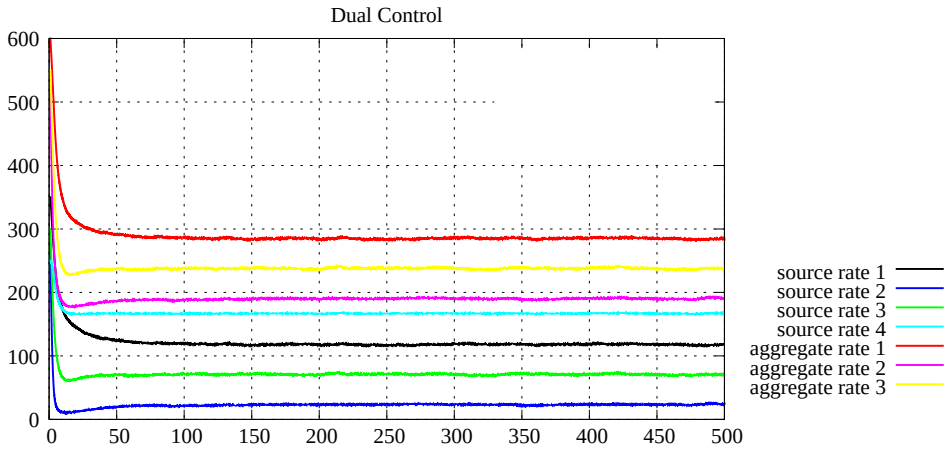


Figure A.32: Source rates, case 4 with 3 links and 4 sources (Dual)

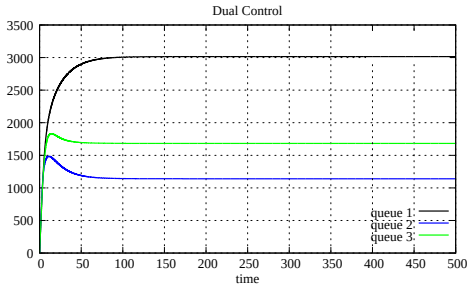


Figure A.33: Queue size, case 1 with 3 links and 4 sources (Dual)

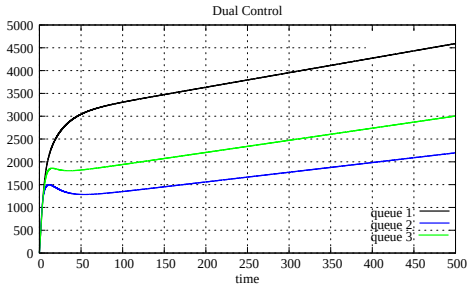


Figure A.34: Queue size, case 2 with 3 links and 4 sources (Dual)

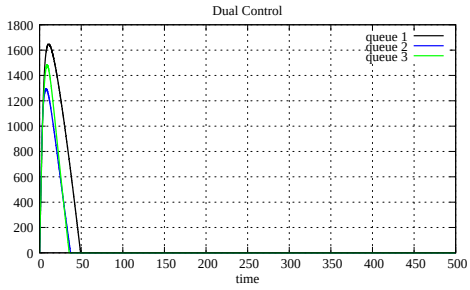


Figure A.35: Queue size, case 3 with 3 links and 4 sources (Dual)

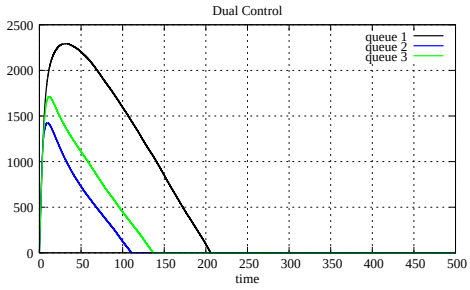


Figure A.36: Queue size, case 4 with 3 links and 4 sources (Dual)

A.2 Primal–Dual Control

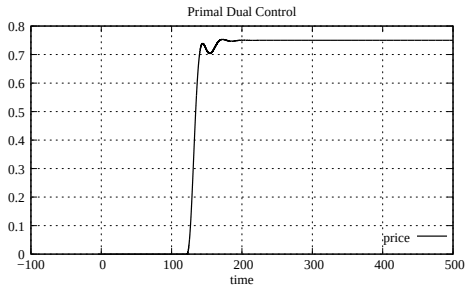


Figure A.37: Link price, case 1 with 1 link and 1 source (P D C)

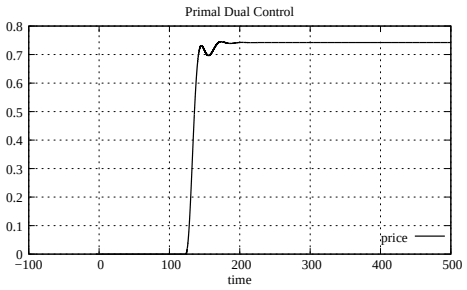


Figure A.38: Link price, case 2 with 1 link and 1 source (P D C)

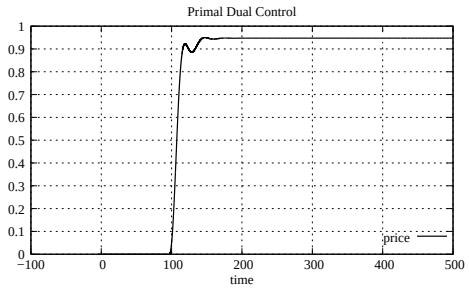


Figure A.39: Link price, case 3 with 1 link and 1 source (P D C)

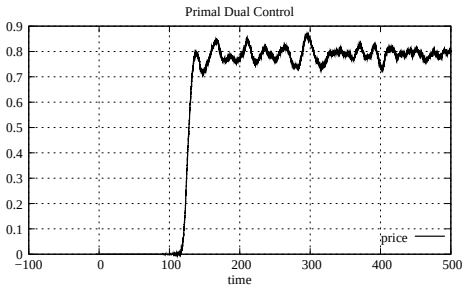


Figure A.40: Link price, case 4 with 1 link and 1 source (P D C)

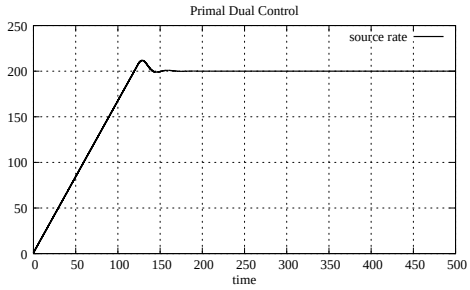


Figure A.41: Source rate, case 1 with 1 link and 1 source (P D C)

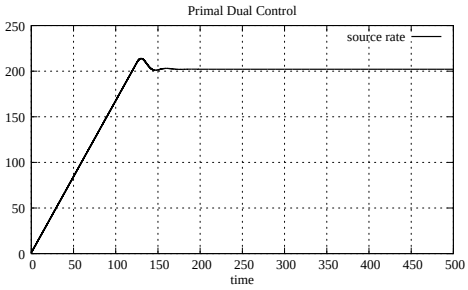


Figure A.42: Source rate, case 2 with 1 link and 1 source (P D C)

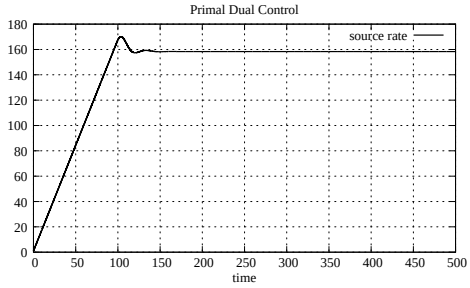


Figure A.43: Source rate, case 3 with 1 link and 1 source (P D C)

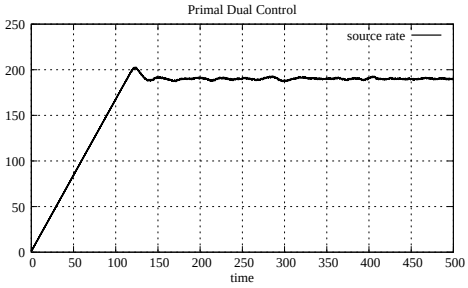


Figure A.44: Source rate, case 4 with 1 link and 1 source (P D C)

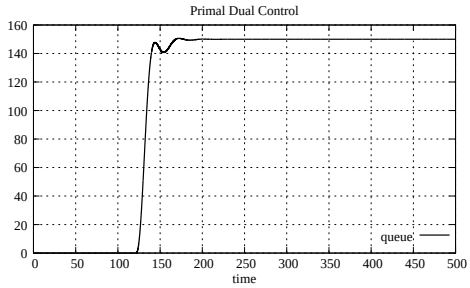


Figure A.45: Queue size, case 1 with 1 link and 1 source (P D C)

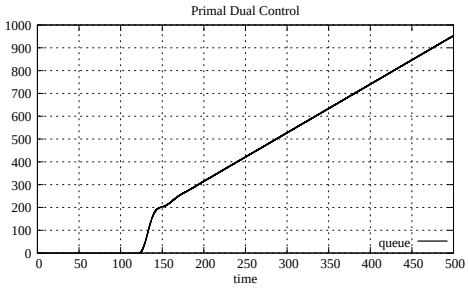


Figure A.46: Queue size, case 2 with 1 link and 1 source (P D C)

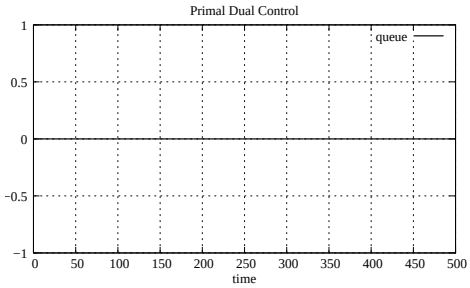


Figure A.47: Queue size, case 3 with 1 link and 1 source (P D C)

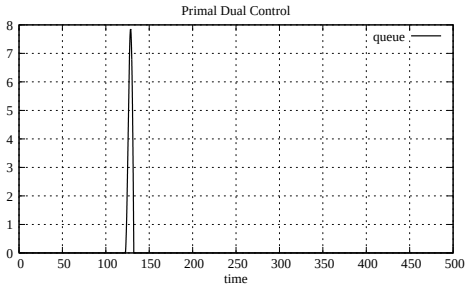


Figure A.48: Queue size, case 4 with 1 link and 1 source (P D C)

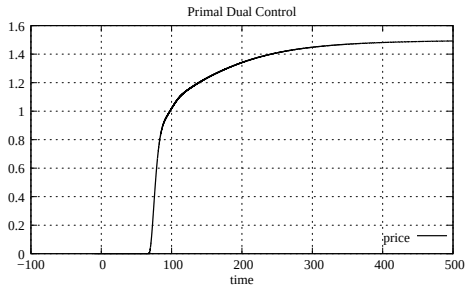


Figure A.49: Link price, case 1 with 1 link and 3 sources (P D C)

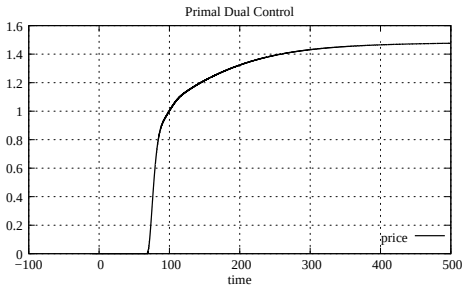


Figure A.50: Link price, case 2 with 1 link and 3 sources (P D C)

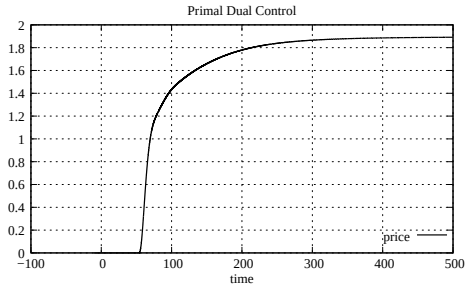


Figure A.51: Link price, case 3 with 1 link and 3 sources (P D C)

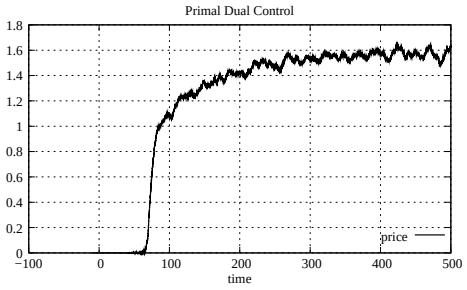


Figure A.52: Link price, case 4 with 1 link and 3 sources (P D C)

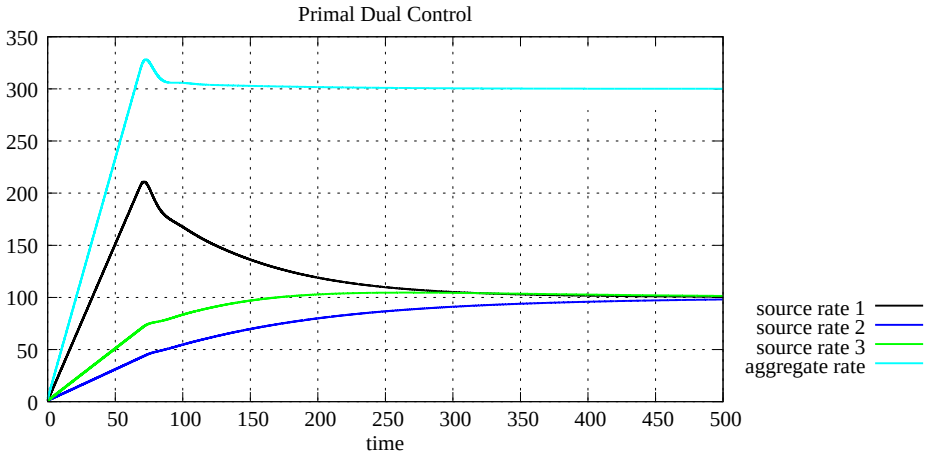


Figure A.53: Source rates, case 1 with 1 link and 3 sources (P D C)

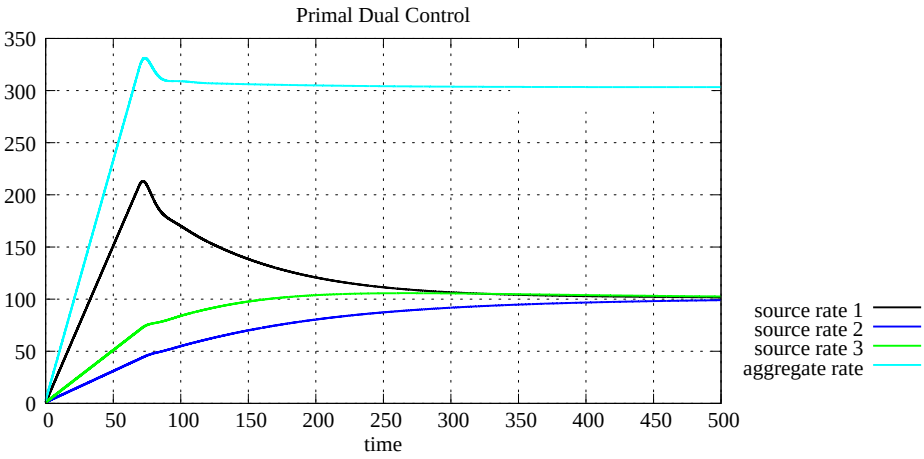


Figure A.54: Source rates, case 2 with 1 link and 3 sources (P D C)

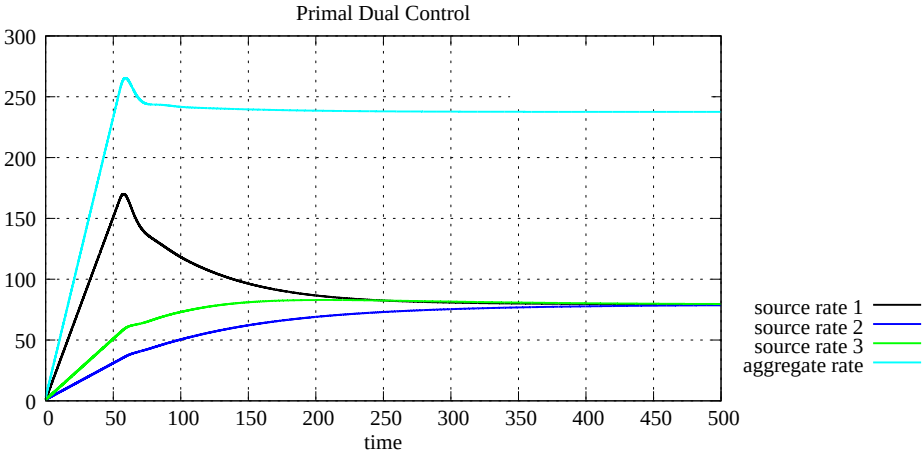


Figure A.55: Source rates, case 3 with 1 link and 3 sources (P D C)

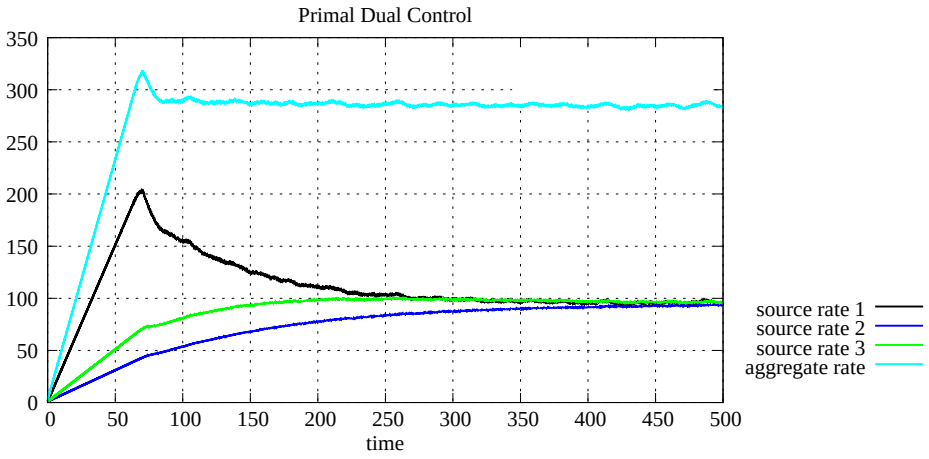


Figure A.56: Source rates, case 4 with 1 link and 3 sources (P D C)

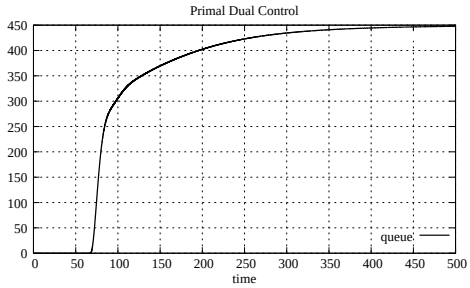


Figure A.57: Queue size, case 1 with 1 link and 3 sources (P D C)

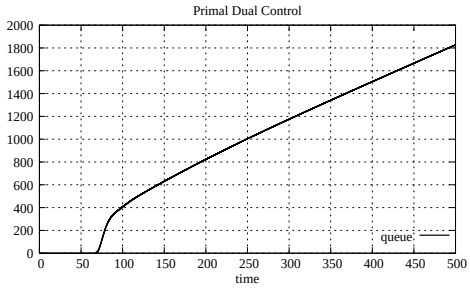


Figure A.58: Queue size, case 2 with 1 link and 3 sources (P D C)

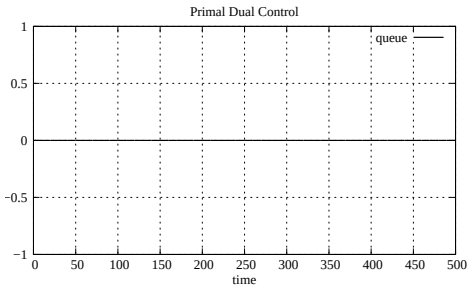


Figure A.59: Queue size, case 3 with 1 link and 3 sources (P D C)

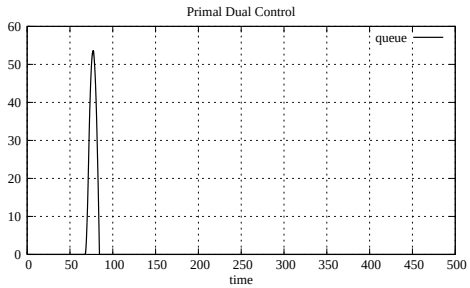


Figure A.60: Queue size, case 4 with 1 link and 3 sources (P D C)

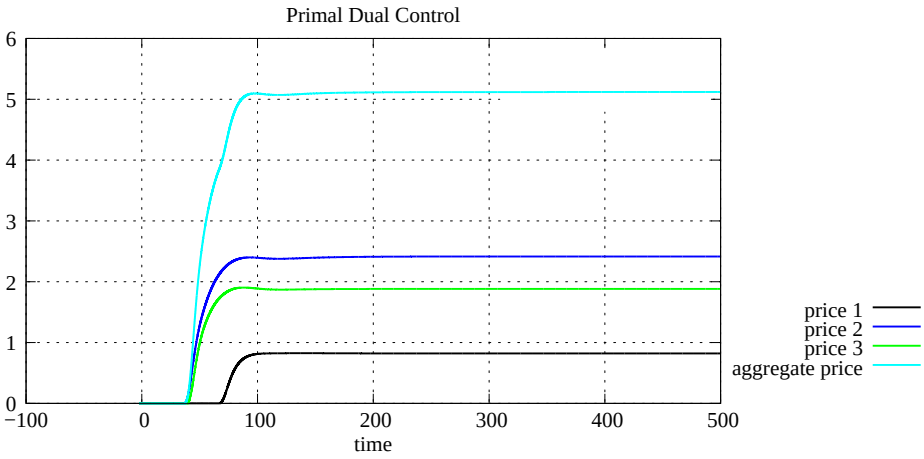


Figure A.61: Link price, case 1 with 3 links and 4 sources (P D C)

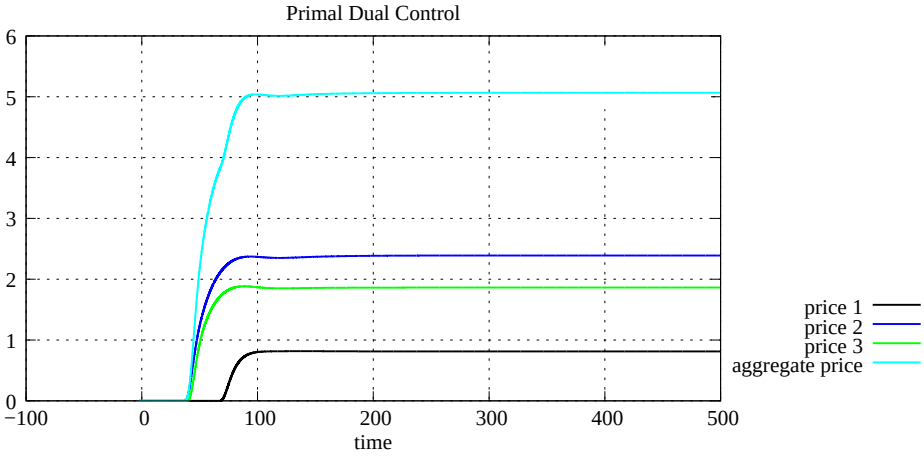


Figure A.62: Link price, case 2 with 3 links and 4 sources (P D C)

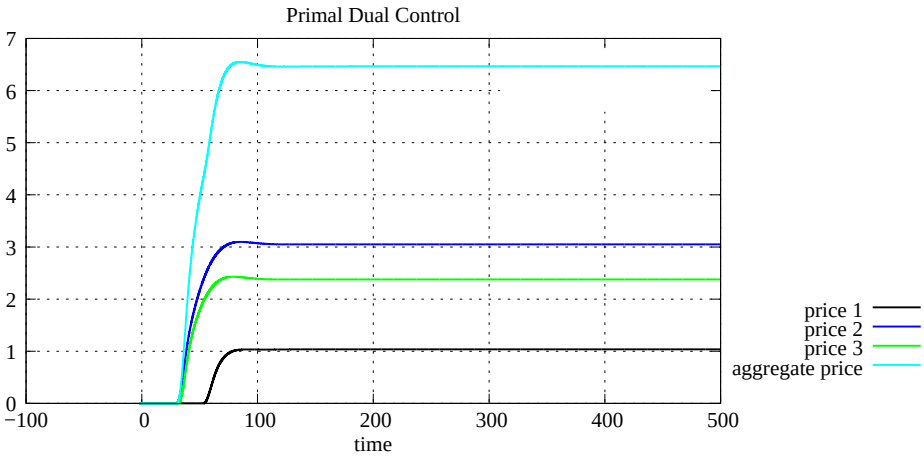


Figure A.63: Link price, case 3 with 3 links and 4 sources (P D C)

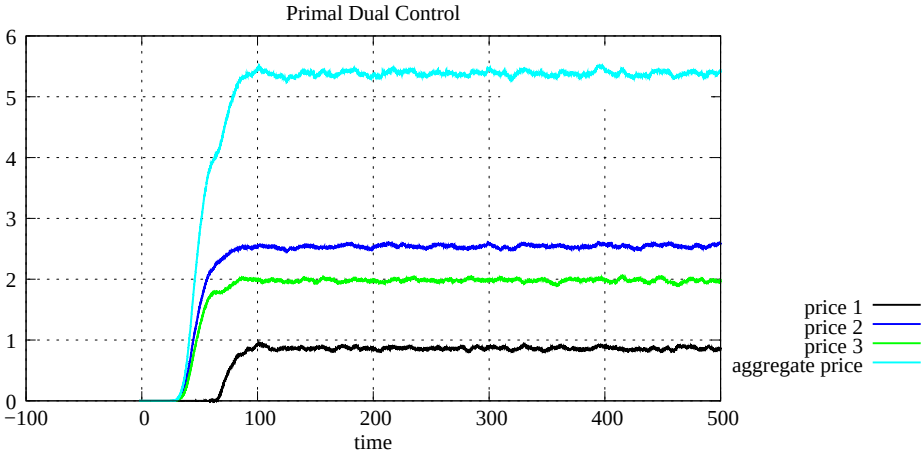


Figure A.64: Link price, case 4 with 3 links and 4 sources (P D C)

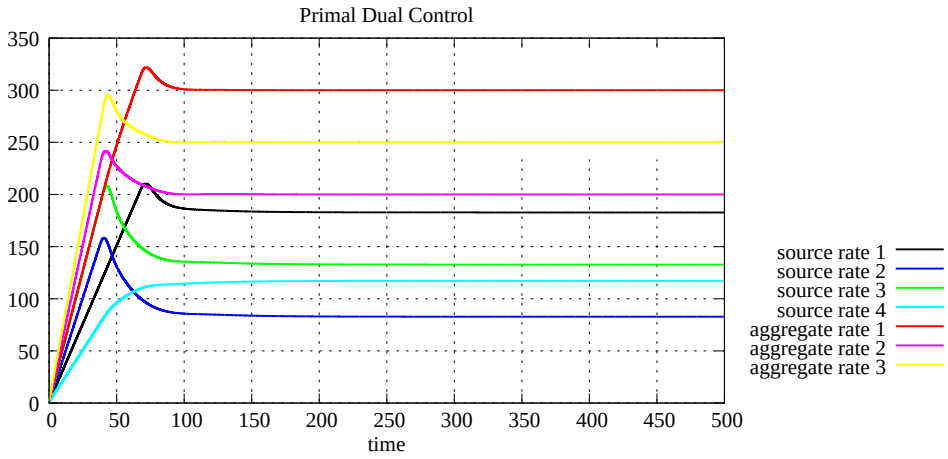


Figure A.65: Source rates, case 1 with 3 links and 4 sources (P D C)

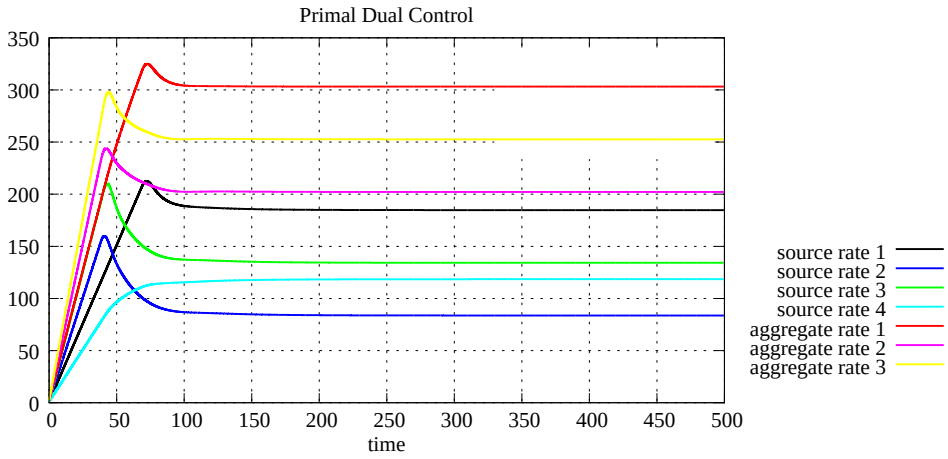


Figure A.66: Source rates, case 2 with 3 links and 4 sources (P D C)

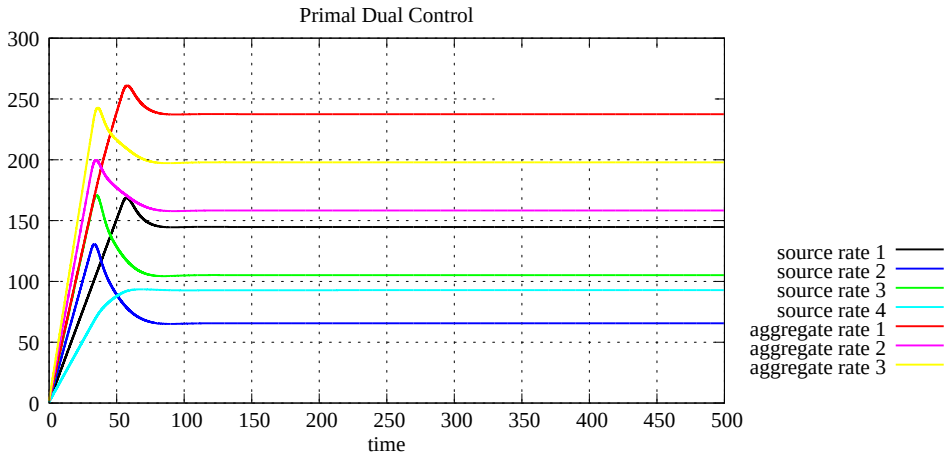


Figure A.67: Source rates, case 3 with 3 links and 4 sources (P D C)

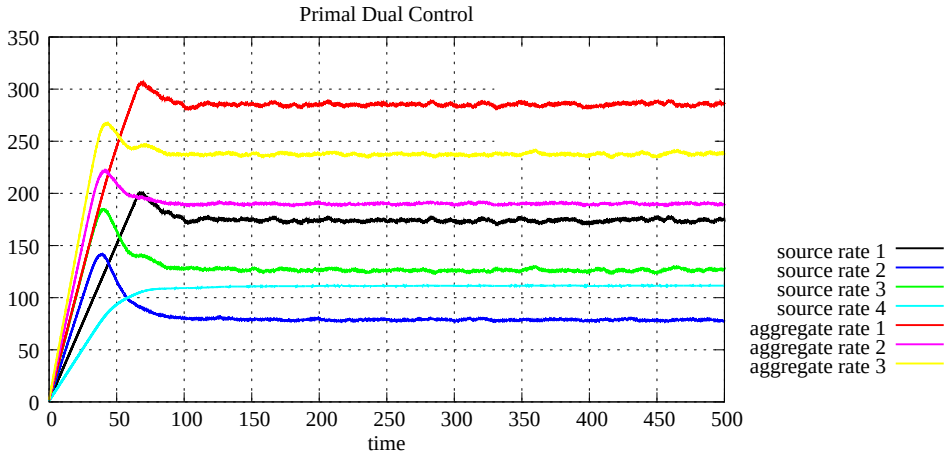


Figure A.68: Source rates, case 4 with 3 links and 4 sources (P D C)

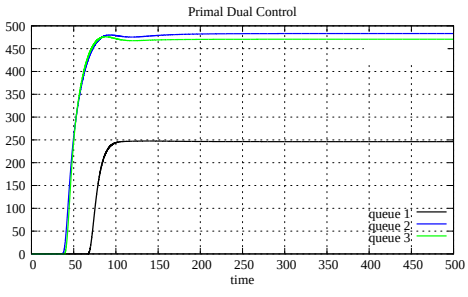


Figure A.69: Queue size, case 1 with 3 links and 4 sources (P D C)

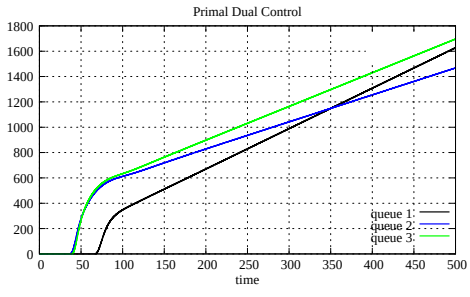


Figure A.70: Queue size, case 2 with 3 links and 4 sources (P D C)

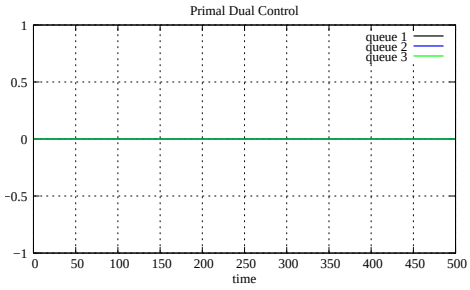


Figure A.71: Queue size, case 3 with 3 links and 4 sources (P D C)

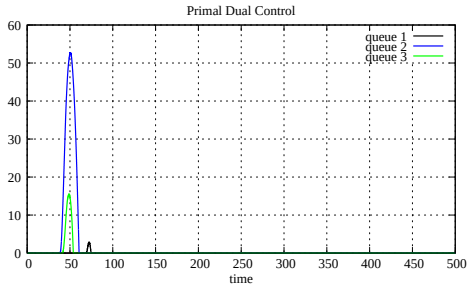


Figure A.72: Queue size, case 4 with 3 links and 4 sources (P D C)

A.3 Control based on Queueing Delay

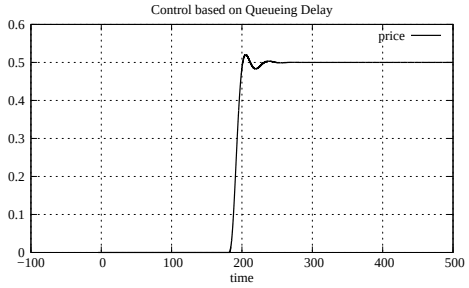


Figure A.73: Link price, case 1 with 1 link and 1 source (C Q D)

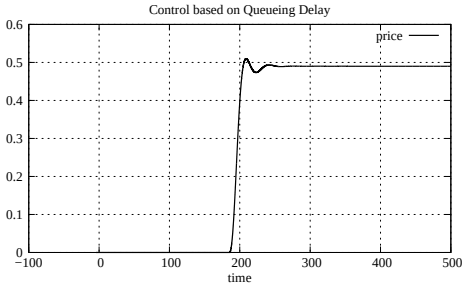


Figure A.74: Link price, case 2 with 1 link and 1 source (C Q D)

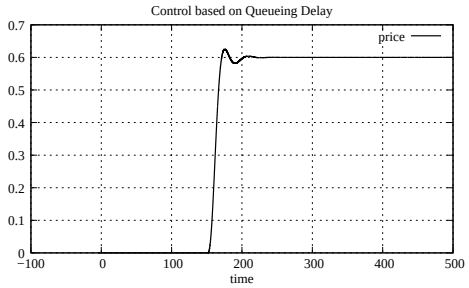


Figure A.75: Link price, case 3 with 1 link and 1 source (C Q D)

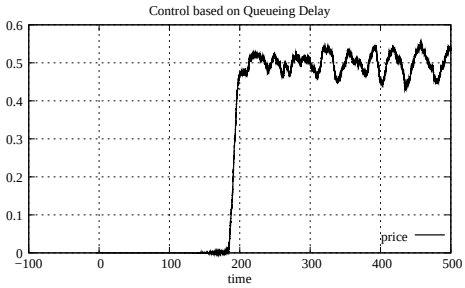


Figure A.76: Link price, case 4 with 1 link and 1 source (C Q D)

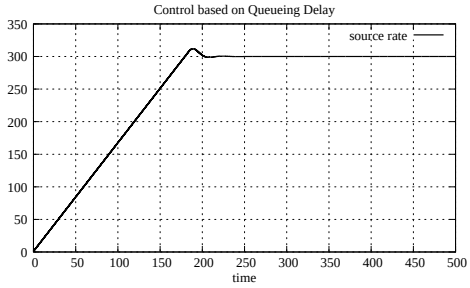


Figure A.77: Source rate, case 1 with 1 link and 1 source (C Q D)

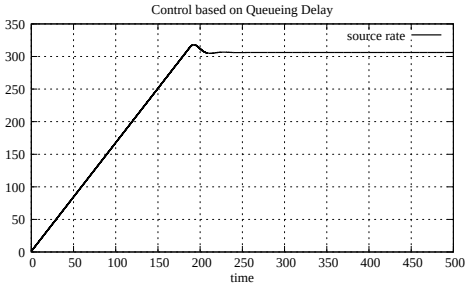


Figure A.78: Source rate, case 2 with 1 link and 1 source (C Q D)

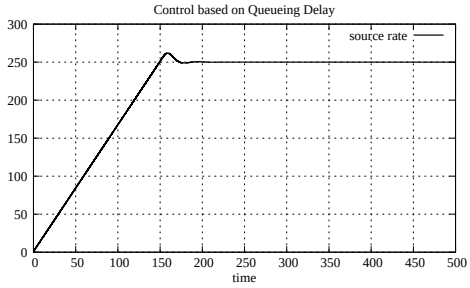


Figure A.79: Source rate, case 3 with 1 link and 1 source (C Q D)

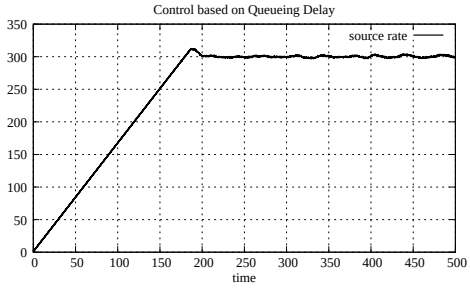


Figure A.80: Source rate, case 4 with 1 link and 1 source (C Q D)

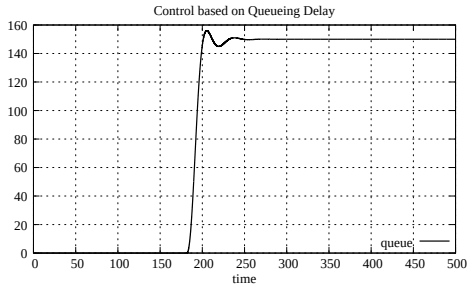


Figure A.81: Queue size, case 1 with 1 link and 1 source (C Q D)

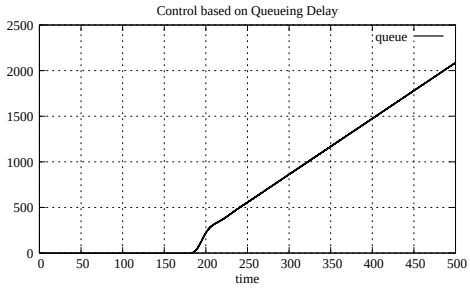


Figure A.82: Queue size, case 2 with 1 link and 1 source (C Q D)

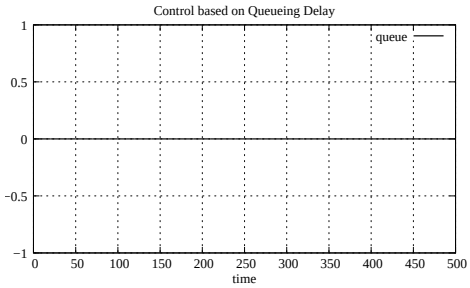


Figure A.83: Queue size, case 3 with 1 link and 1 source (C Q D)

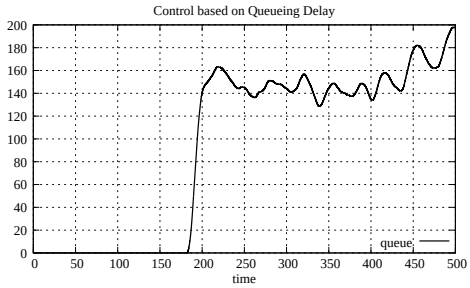


Figure A.84: Queue size, case 4 with 1 link and 1 source (C Q D)

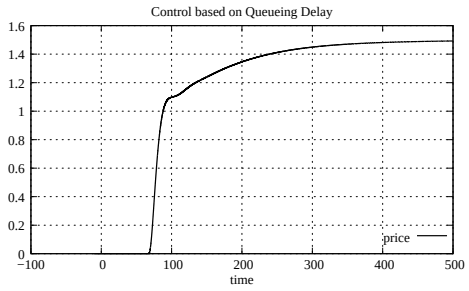


Figure A.85: Link price, case 1 with 1 link and 3 sources (C Q D)

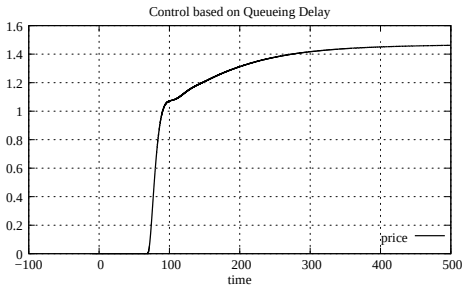


Figure A.86: Link price, case 2 with 1 link and 3 sources (C Q D)

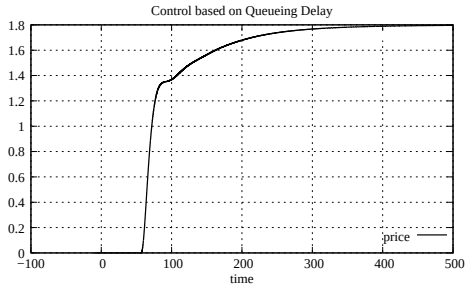


Figure A.87: Link price, case 3 with 1 link and 3 sources (C Q D)

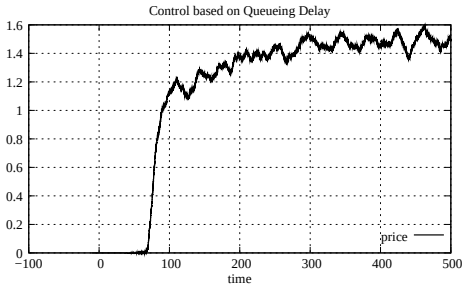


Figure A.88: Link price, case 4 with 1 link and 3 sources (C Q D)

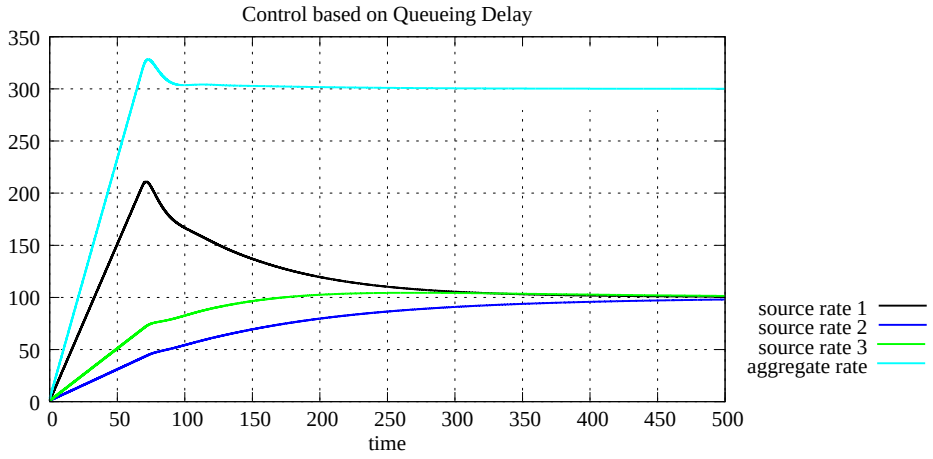


Figure A.89: Source rates, case 1 with 1 link and 3 sources (C Q D)

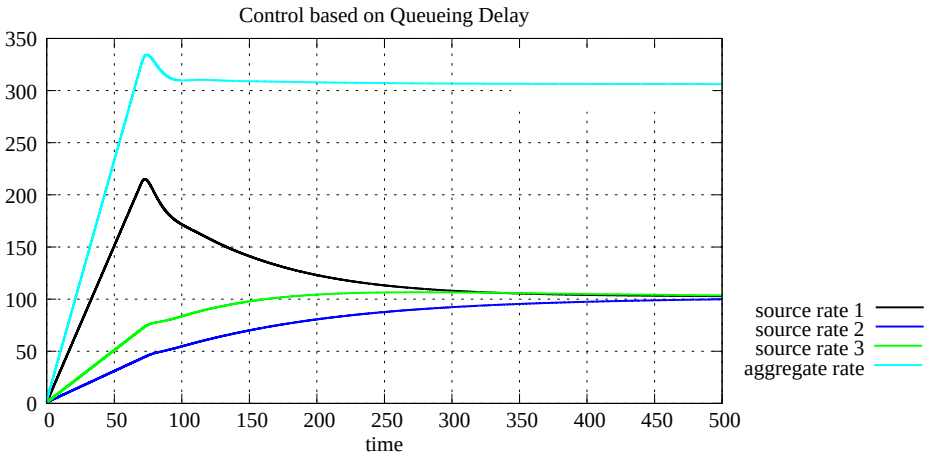


Figure A.90: Source rates, case 2 with 1 link and 3 sources (C Q D)

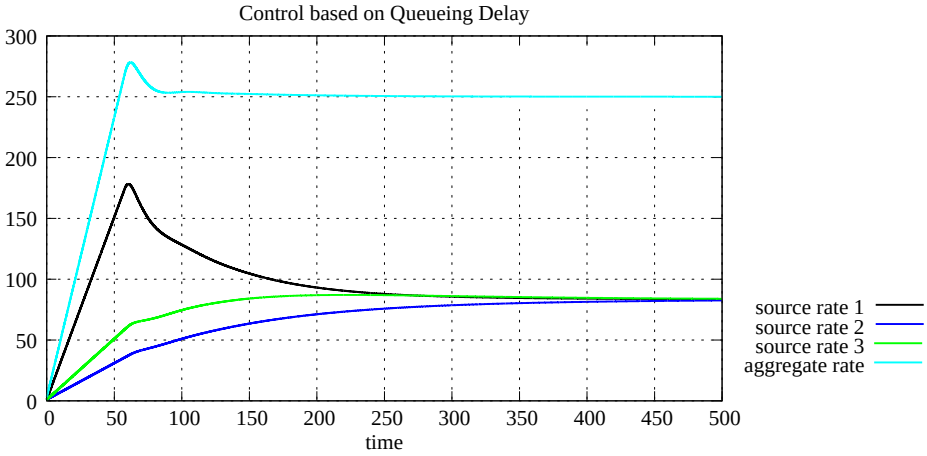


Figure A.91: Source rates, case 3 with 1 link and 3 sources (C Q D)

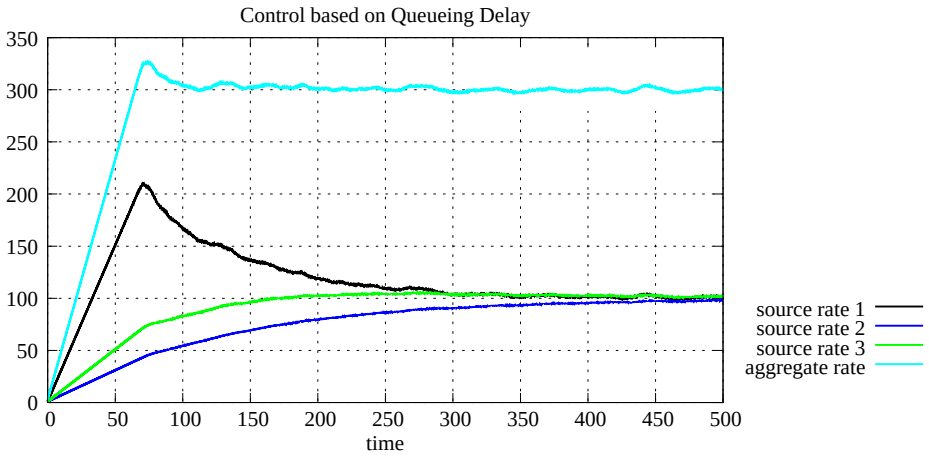


Figure A.92: Source rates, case 4 with 1 link and 3 sources (C Q D)

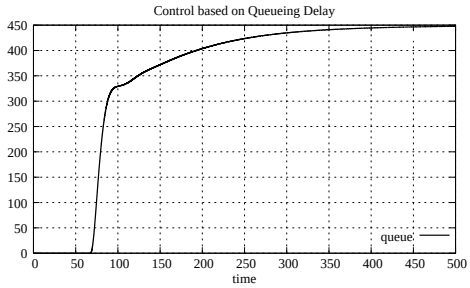


Figure A.93: Queue size, case 1 with 1 link and 3 sources (C Q D)

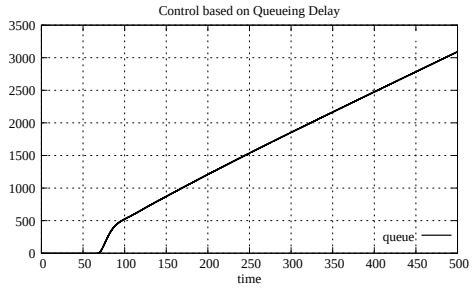


Figure A.94: Queue size, case 2 with 1 link and 3 sources (C Q D)

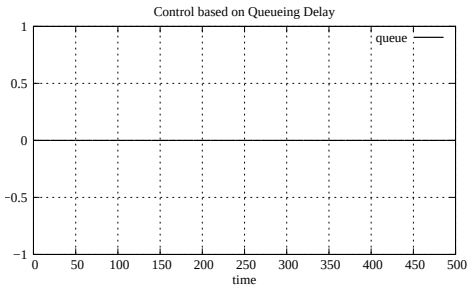


Figure A.95: Queue size, case 3 with 1 link and 3 sources (C Q D)

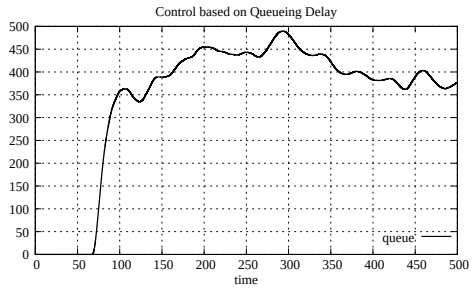


Figure A.96: Queue size, case 4 with 1 link and 3 sources (C Q D)

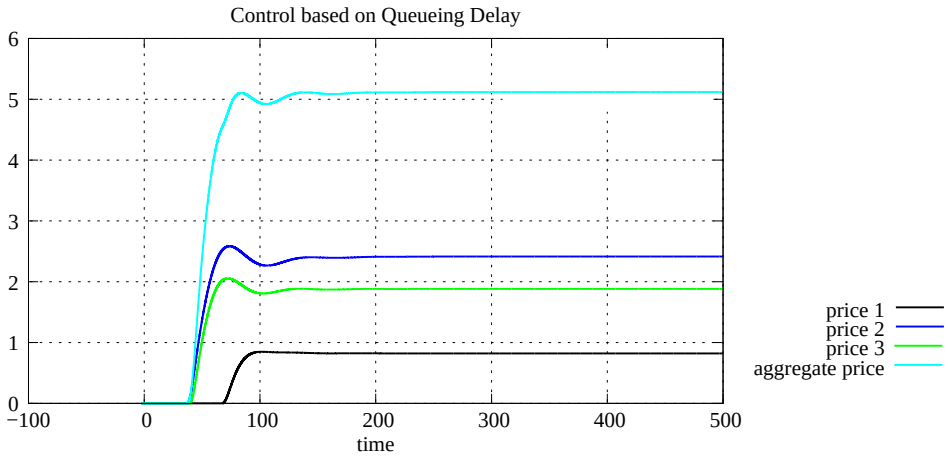


Figure A.97: Link price, case 1 with 3 links and 4 sources (C Q D)

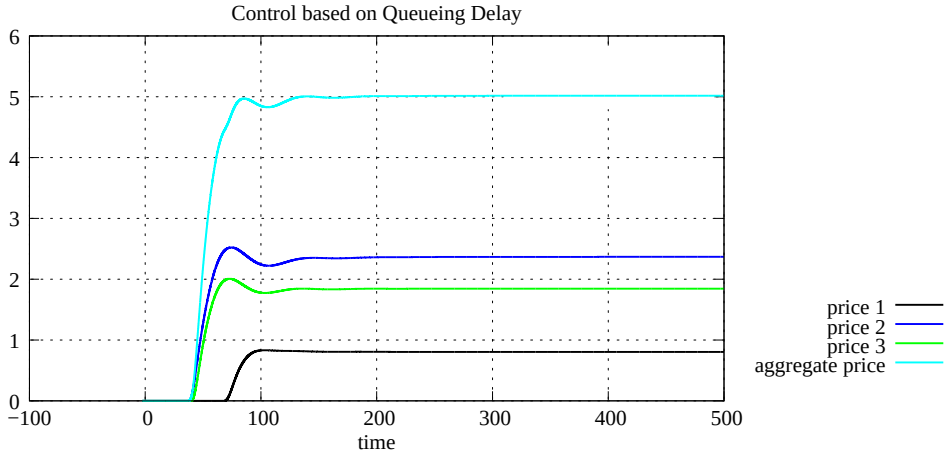


Figure A.98: Link price, case 2 with 3 links and 4 sources (C Q D)

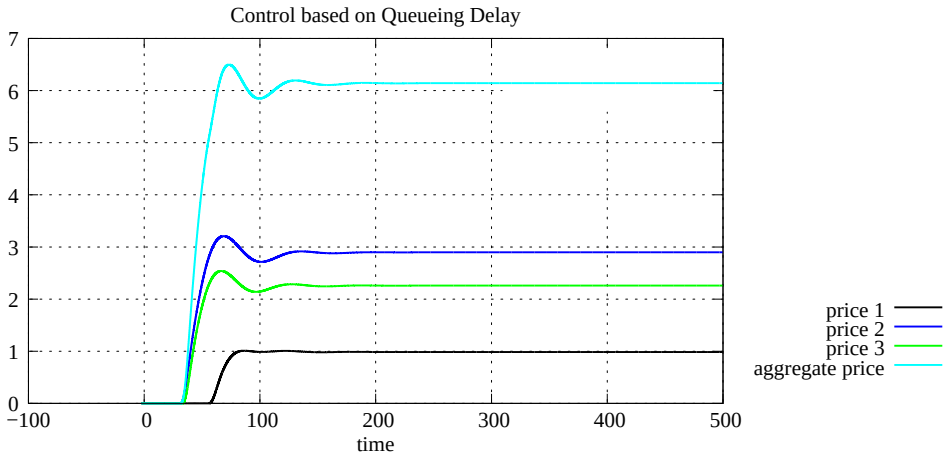


Figure A.99: Link price, case 3 with 3 links and 4 sources (C Q D)

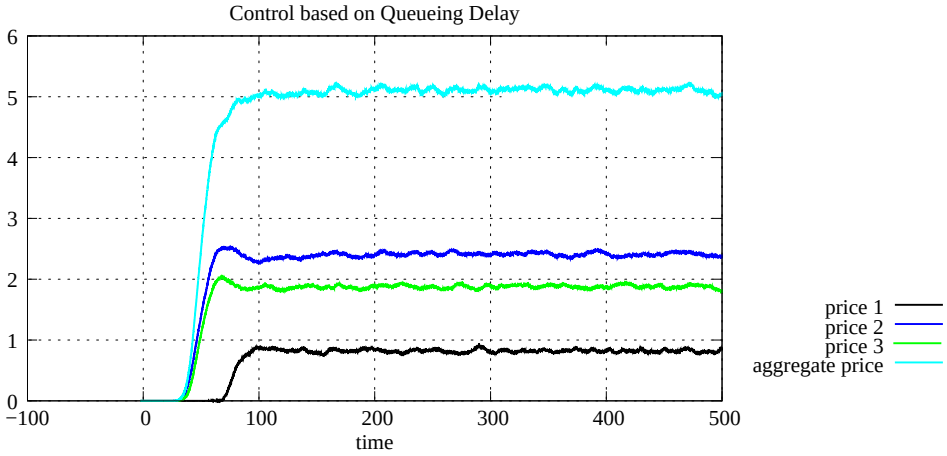


Figure A.100: Link price, case 4 with 3 links and 4 sources (C Q D)

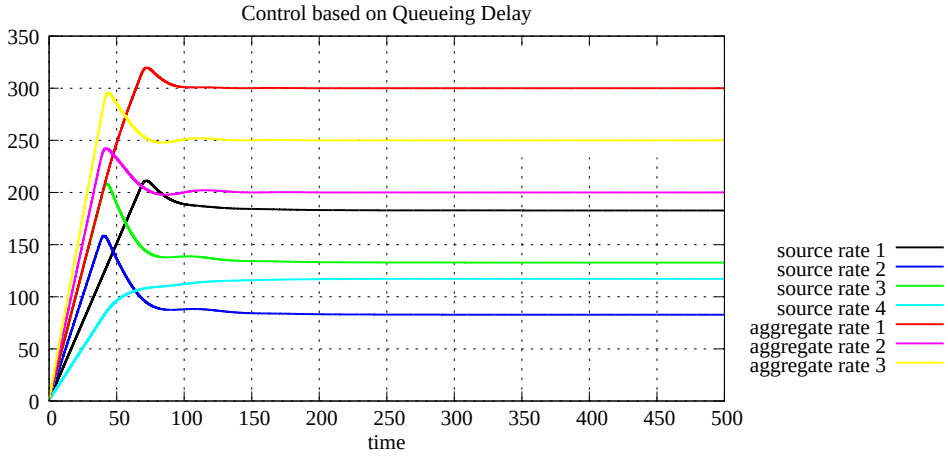


Figure A.101: Source rates, case 1 with 3 links and 4 sources (C Q D)

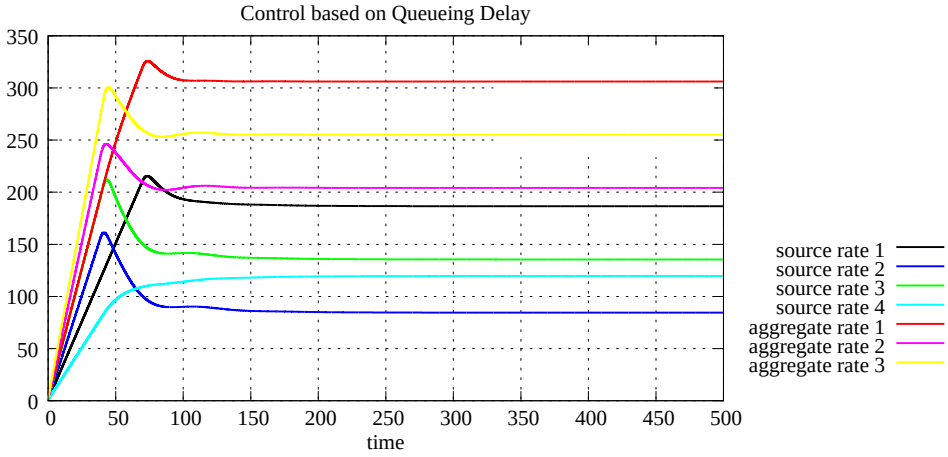


Figure A.102: Source rates, case 2 with 3 links and 4 sources (C Q D)

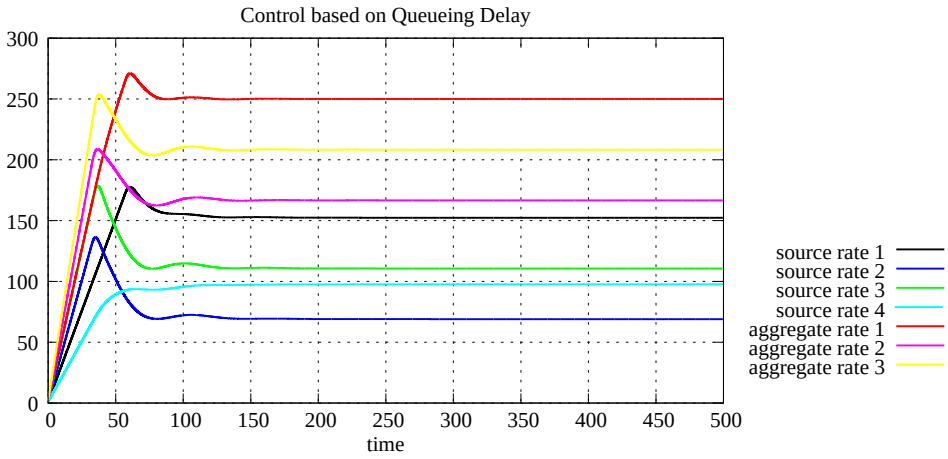


Figure A.103: Source rates, case 3 with 3 links and 4 sources (C Q D)

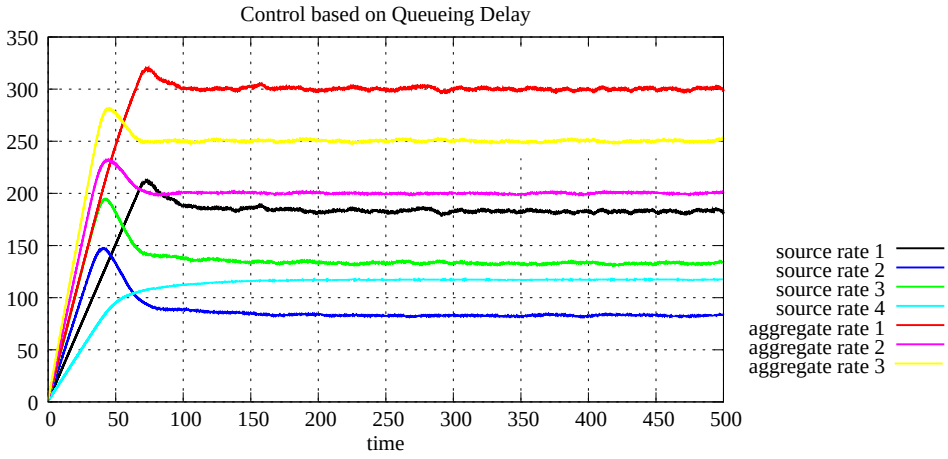


Figure A.104: Source rates, case 4 with 3 links and 4 sources (C Q D)

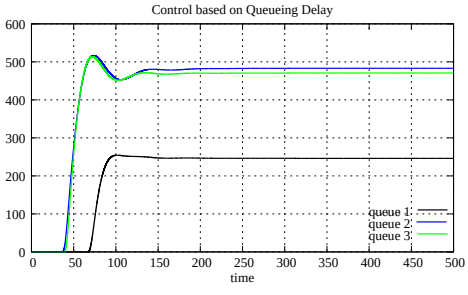


Figure A.105: Queue size, case 1 with 3 links and 4 sources (C Q D)

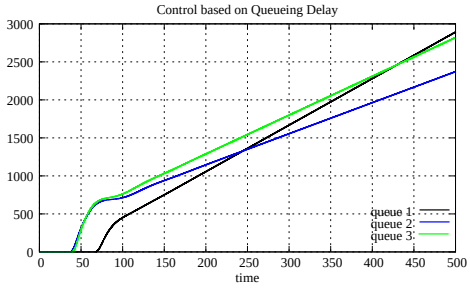


Figure A.106: Queue size, case 2 with 3 links and 4 sources (C Q D)

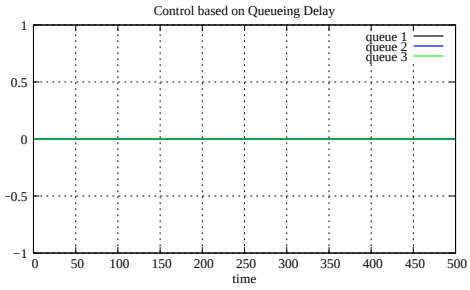


Figure A.107: Queue size, case 3 with 3 links and 4 sources (C Q D)

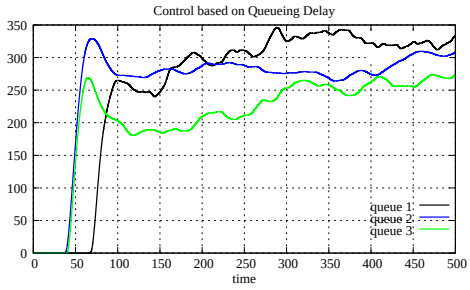


Figure A.108: Queue size, case 4 with 3 links and 4 sources (C Q D)

Bibliography

[1] A. Bellen and M. Zennaro, *Numerical methods for delay differential equations*, Oxford University Press, 2003.

[2] I. Györi, F. Hartung, and J. Turi, *On numerical solutions for a class of nonlinear delay equations with time – and state–dependent delays*, World Congress of Nonlinear Analysts, 1992.

[3] ———, *On numerical approximations for a class of differential equations with time – and state–dependent delays*, Applied Mathematics Letters, 1995.

[4] M. Hassan and H. Sirisena, *Optimal control of queues in computer networks*, Computer Communications (2001).

[5] O. C. Imer and T. Basar, *Dynamic optimization flow control*, (2003).

[6] V. Jacobson and M. J. Karels, *Congestion avoidance and control*, 1988.

[7] L. S. Jennings, M. E. Fisher, K. L. Teo, and C. J. Goh, *MISER 3 optimal control software: Theory and user manual, version 3*, July 2004.

[8] S. H. Low and D. E. Lapsley, *Optimization flow control–I: Basic algorithm and convergence*, IEEE/ACM Transactions on Networking, December 1999.

[9] F. Paganini, J. Doyle, and S. Low, *Scalable laws for stable network congestion control*, 40th IEEE Conference on Decision and Control, 2001.

[10] F. Paganini, Z. Wang, J. C. Doyle, and S. H. Low, *Congestion control for high performance, stability and fairness in general networks*, April 2003.

[11] F. Paganini, Z. Wang, S. H. Low, and J. Doyle, *A new TCP/AQM for stable operation in fast networks*, IEEE Infocom, 2003.

[12] A. Papachristodoulou, *Global stability of a TCP/AQM protocol for arbitrary networks with Delay*, 2004.

[13] A. Papachristodoulou, J. C. Doyle, and S. H. Low, *Analysis of nonlinear delay differential equation models of TCP/AQM protocols using sums of squares*, 2004.

[14] H. Sirisena, M. Hassan, and A. Haider, *Optimal TCP congestion control*, Computer Communications (2002).

[15] Z. Wang and F. Paganini, *Global stability with time–delay in network congestion control*, 42nd Conference on Decision and Control, 2002.

[16] ———, *Global stability with time–delay of a primal–dual congestion control*, 42nd Conference on Decision and Control, December 2003.

[17] ———, *Global stability of nonlinear congestion control with time–delay*, 2004.

[18] K. H. Wong, *An optimal control approach to congestion control*, Private Communication, 2004.