Otto-von-Guericke-Universität Magdeburg

Studienarbeit

A control-based approach to core link provisioning

von

Jonathan Epperlein

2. Mai 2007

Eingereicht an: Otto-von-Guericke-Universität Magdeburg
Fakultät für Elektrotechnik und Informationstechnik
Institut für Automatisierungstechnik
Universitätsplatz 2
Postfach 4120, 39016 Magdeburg
Deutschland

Prüfer: Prof. Dr.-Ing. Jörg Raisch

Betreuer: Prof. Robert Shorten
Hamilton Institute
National University of Ireland, Maynooth

Dr. Chris Kellett
School of Electrical Engineering and Computer Science
University of Newcastle
Australia

Declaration of originality

I hereby declare that this thesis and the work reported herein was composed and originated entirely by myself. Information derived from the published and unpublished work of others is acknowledged in the text and a list of references is given in the bibliography.

Acknowledgements

...not just one - a LOTTA them.

After all the blood, sweat and tears, I can say that I am pretty happy with my work. Nevertheless, it would not have been possible, or nearly as pleasant, if it had not been for several people I want to say "Thanks a million" to.
Firstly, thanks to my main supervisor Prof. Robert Shorten, who gave me the great opportunity to work at the Hamilton Institute, who believed in me and applied some gentle pressure when I needed it, and who in the end was more than a mere supervisor, and to my co-supervisor Dr. Chris Kellett for the valuable comments from the other side of the world.
And of course, deep gratitude to my colleagues, that make the Hamilton Institute a place with a most enjoyable atmosphere. Very special thanks to Rade Stanojević, who patiently shared an office with me for the whole time, for good conversations, philosophy and advice, to Dr. Steven Strachan for the sarcasm and the secretiveness, Diego Oyarzún for the company and Yann Tiersen, Florian Knorn for a piece of good ol' Magdeburg in Maynooth and the pictures, Dr. Peter Clifford for always being in my team and thus tackling the others, Dr. Oliver Mason for having a close look at the theoretical part of the thesis, Dr. Dave Malone, who was always willing to help, and last but certainly not least, to Rosemary Hunt and Kate Moriarty, who are heart and soul of the Institute (or secretaries, as some call it).
Additional thanks to the international and Erasmus crowds of Maynooth for creating a good social environment – and to some of them for much more than that.
Many thanks for generous and uncomplicated financial support to the *Science Foundation Ireland* that supported this work under grant number

04/IN3/I460 and the *Studienstiftung des Deutschen Volkes*, which is supporting me with one of its priceless scholarships.

Vielen Dank an Prof. Jörg Raisch, der den Kontakt nach Irland hergestellt hat und seit Jahren mehr als nur ein Professor ist.
Unersetzlich war und ist Janine Holzmann, die beste Sekretärin der Welt, die mir die Stellung in Magdeburg gehalten hat und noch viel mehr.
Bei Gerd Schmidt, der mit mir die meiste Zeit Büro und Haus geteilt hat, bedanke ich mich für viele gute Tipps und natürlich die vielen "Hammerings" bei Pro Evolution Soccer.
Ein großes "Dankeschön!" auch den Leuten zu Hause, die ich vernachlässigt habe und die mich trotzdem noch gern haben, insbesondere Mogli & Mona, Dana, Chrissi B. und H., Ansgar, Tinchen, Johannes und André.

Und zuletzt will ich denen danken, denen der Dank bei allem was ich tue gebührt: Meinen großartigen Geschwistern Lilith, Jascha und Ruben für 21 bzw. 18 gemeinsame Jahre, und meinen Eltern, Amélie und Andreas, für bald 25 Jahre in denen sie mich immer unterstützt, geliebt und sogar ein bisschen erzogen haben!

Jonathan Epperlein, Maynooth
May 2, 2007

Table of contents

1 Introduction **1**

1.1 Background: Network requirements 1

1.2 Over-provisioning 3

1.3 Network traffic 4

1.4 Outline of the Thesis 5

2 Review of recent work **7**

2.1 Overview 7

2.2 Related work 7

2.3 The model-based approach of van de Meent 8

2.3.1 Link provisioning 8

2.3.2 The models 10

2.3.3 The provisioning rules 12

2.4 Alternatives to over-provisioning 14

2.4.1 DiffServ 14

2.4.2 IntServ 15

2.5 A critique of proposed approaches 15

3 Feedback control and link provisioning **16**

3.1 Structure of the chapter to come 16

3.2 Shortcomings of earlier approaches 16

3.3 Preamble: Adaptive Drop-Tail queueing 17
3.4 Control-based core link provisioning 18
3.4.1 Motivating example 19
3.4.2 Basic idea 20
3.4.3 Discussion of the control-based approach 21
3.5 Control algorithms 23
3.5.1 Linear filter 24
3.5.2 Linear controller 24
3.5.3 Multiplicative Increase Multiplicative Decrease - MIMD 25
3.5.4 Multiplicative Increase Additive Decrease - MIAD . . . 26
3.6 Basic *ns*2 simulation results 26
3.6.1 The motivating example re-revisited 27
3.6.2 Smooth traffic versus bursty traffic 28

4 Theoretical contribution: Stability of discrete time switched linear and Lur'e systems 33

4.1 Overview 33
4.2 Switched systems and Lyapunov functions 34
4.2.1 Lyapunov functions 36
4.2.2 Switched linear systems 38
4.2.3 Switched systems and Lyapunov functions 42
4.3 The Lur'e absolute stability problem 46
4.3.1 The Lur'e problem and switched systems 48
4.4 Stability results for continuous time 49
4.4.1 The circle criterion 49
4.4.2 Spectral conditions 50
4.5 Robust stability results for continuous time 54
4.6 Discrete time versions 58

4.6.1 The Bilinear Transform 58
4.6.2 Results for discrete time 60
4.6.3 Robust result for discrete time 61

5 Connecting the parts: Stability analysis as a Lur'e problem 66
5.1 Overview 66
5.2 Formulation as a Lur'e problem 67
5.3 First order filter 69
5.4 Second order filter 70
5.5 N-th order filter 71

6 Conclusions and future work 74

Appendix 76
A Matlab source codes 76
A.1 Nth_Order.m 76
A.2 Nth_order_plot.m 78
B Network technology basics 79
B.1 A short overview of TCP 79
B.2 One line about UDP 81
B.3 A very short overview of SNMP 81
B.4 Some other terms 82

List of symbols and abbreviations 83

Chapter 1

Introduction

Capacity planning to ensure adequate network performance is one of the most important practical problems in network setup and maintenance. It also remains an open problem as to how best plan network capacity. This chapter should give a short overview of capacity planning, or link provisioning, as it is sometimes called, and what makes it difficult. In Section 1.1, we will provide the background for the problem of link provisioning, defining Quality of Service (QoS) in the process. In Section 1.2, the method of over-provisioning as one possible solution to this problem is introduced and justified. In Section 1.3 some of the difficulties encountered in over-provisioning are described.

1.1 Background: Network requirements

Different users use their computers for very different applications; thus networks are used for completely different purposes. Some use the Internet to chat with people all over the world, some just send and receive e-mails, some are downloading podcasts and some watch streamed videos on youtube, some are playing poker online, some are playing Counter Strike with their friends in a Local Area Network, some are sharing their photographs and some are are sharing songs or movies.
Of course, each of these activities poses completely different requirements on the network they are done on.

An abstraction of the described activities above could be as follows. Every user sitting in front of his computer is running one or more **applications**, which in turn are using (after some steps that are not of interest for this

thesis) a **transport layer protocol** to release their data into the network (again, some steps are omitted, but again, they do not impact the following. For details on the so-called OSI seven layer model, see [Zim80]).
To name a few of these applications:

- ▷ Voice over IP (VoIP) or even video conferencing require an almost delay free connection but bandwidth only when the audio and/or video signal is sent in the respective direction.
- ▷ Streaming applications like webradio or Internet TV require a fixed amount of bandwidth all the time (of course, buffering relaxes this a little), but a certain delay would not be perceived as too annoying.
- ▷ Real time online games like Counter Strike require sufficient bandwidth plus low delay to function properly.
- ▷ Critical applications like remote control of industrial processes (recall that we are not talking about the Internet only) require absolutely guaranteed availability of the communication.
- ▷ "Normal" applications like file sharing, web browsing or webchats do not require anything special, nevertheless there are limits to delays and bandwidth at which the user will call the performance "bad".

The goal of any network operator is, to make every user "happy", that is to meet all requirements of the different applications. The term **Quality of Service (QoS)** has been coined to describe both: The perception of the performance of a network, and technologies and mechanisms, that can be employed to achieve sufficient performance.
Performance basically is perceived in terms of delays, which mainly occur because of buffering, losses, which occur mainly because of full buffers, and jitter, which is a term describing rapid changes of delays.

Our attention in this thesis will be restricted to networks on which computers are communicating via **wired** links and whose topology can be described by the terms **access links** and **core links**, see Figure 1.1. With the term **core link**, we refer to a link in the network where traffic from many routers aggregates. By that definition, the core is not the place where losses should normally occur, and traffic arriving there is an aggregate of aggregates. In addition, we are interested in core links which are serving a rather big number of access links.
So we can assume that losses that occur in the core only affect a small part

of the flows aggregating there. Consequently, as long as the losses stay on a low value, we can assume the traffic arrival process not to depend on the losses. Recall, that the TCP congestion control causes the sender to back off as soon as a loss occurs, but by our assumptions, only a small fraction of the aggregating flows will sense a loss.

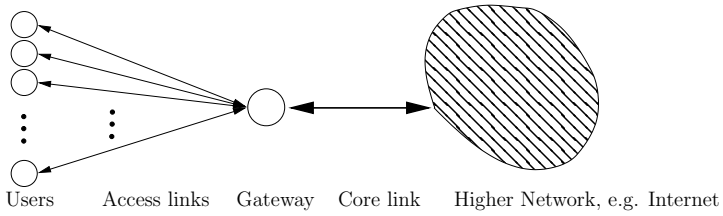


Figure 1.1: Typical network topology with core link and access links

With this additional assumption, we can link losses and delays directly to the available core link bandwidth: A higher bandwidth prevents large queues from building up and by that also prevents losses from happening.

Two principal strategies to ensure QoS can be identified: Applying additional technologies to enable the network to treat data with different requirements differently or to simply meet all requirements for all applications by providing the core link with sufficient bandwidth.

1.2 Over-provisioning

Imagine, the core link had infinite bandwidth. Then, there would be no queueing at the gateway and any access link would get maximum performance. This actually kills two birds with one stone: If the packets are removed fast enough from the gateway queue, there will be no or very short delays, and each access link can send "as fast as possible".

Of course, this can be achieved with a finite bandwidth, as long as the bandwidth is sufficiently large. Because bandwidth costs money, it is a rather unsatisfying solution to simply use the fastest available link. Instead, it is clear, that at a certain bandwidth, the performance will have no perceivable increase anymore and any additional bandwidth could be seen as a waste of resources.
With this in mind, we can define **core link provisioning** as the *process of finding the minimum necessary bandwidth to ensure QoS.*

The benefits of over-provisioning are numerous, among these are

Simplicity Once set up, the link requires almost no supervision and maintenance.

Good performance Losses and delays introduced by the link are minimal.

Redundancy In the case of link failure, the spare capacity of other links can be used to bypass the failed link.

Currently, unsatisfactory rules-of-thumb of the form "Take the average offered traffic plus 50% to cater for fluctuations" are the most commonly used over-provisioning scheme. Of course, this is not very scientific, nor does it yield a guarantee that no resources are wasted (if the link was under-provisioned, it probably would be noticed very quickly).

In recent publications [vdM06, FTD03, Pap03], statistical models are derived from measurements on real core links, which in turn are used to derive formula for the required bandwidth depending on certain performance criteria. We will give a more detailed review of that in Chapter 2, but here we already note, that these approaches are deemed unsatisfying, because they either rely on the applicability of the derived models, what almost certainly cannot be assumed, or on additional measurements, which can only be obtained by introducing additional technology and by that additional complexity and costs.

Our contribution here is to develop a control scheme, first suggested in [LKC+07], which uses the capacity of a buffer in front of the link to maintain a given, very small loss rate at that buffer. The necessary buffer size then can be seen as additional information on the required bandwidth and can be used to trigger an upgrade of the link. The novelty of that approach lies in its closed-loop nature, which makes it independent of traffic modeling and of more than the available, coarse grained measurements and in the fact that, under entirely reasonable assumptions, stability of the closed loop can be demonstrated.

1.3 Network traffic

Figure 1.2 illustrates the main difficulty any link provisioning scheme has to face: the high variance of the traffic arrivals. Here, averages of different resolutions are shown. Measurements via SNMP (see B.3) can only be taken in the order of minutes, so without additional work the dashed line is all

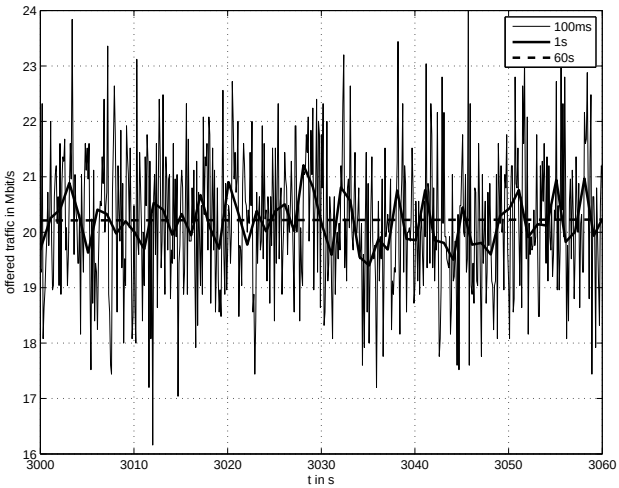


Figure 1.2: Different averages of the same traffic arrivals: 100ms-averages, 1s-averages and 1min-average

the provisioning can be based on. As can be seen in the figure, the highest spike reaches as high as 24 Mbit/s, whereas the 60s-average is just around 20 Mbit/s.

In general, one can say that "typical" network traffic is highly variant, non-stationary, and is a mixture of open loop traffic (caused by connectionless protocols like e.g. UDP, see B.2) and elastic traffic (caused by the congestion avoidance algorithms of protocols like TCP, see B.1).

These properties make it hard to model network traffic, even statistical models are highly questionable because the time variance cannot be captured in them. Moreover, most key parameters like number of active users, running applications and so on are not accessible or change very often. So a link provisioning based on models does not seem wise. As we shall see, our proposal overcomes these difficulties in an elegant and simple manner.

1.4 Outline of the Thesis

We will approach the problem as follows:

- ▷ In Chapter 2, we will review work that has been done on the problem so far. This will be more or less a very detailed summary of the work of van de Meent [vdM06].

- ▷ In Chapter 3, reasoning and implementation of our new approach is described in detail. Some simulation results will also be given.
- ▷ Chapter 4 will contain the theoretical contribution of this Thesis. We will recall some results on the stability of switched linear systems and the Lur'e absolute stability problem, link them together and extend them to discrete time systems.
- ▷ Finally, in Chapter 5, the practical and the theoretical part of the Thesis will be linked together. Our proposal control scheme will be described in the Lur'e framework, so our theoretical results on discrete time Lur'e systems can be used to analyse its stability properties.
- ▷ The Appendix holds some Matlab source codes and some basic facts about network technologies like TCP and SNMP.

Chapter 2

Review of recent work

2.1 Overview

The link provisioning problem has a long history in the communications community. In this chapter we review some of the suggested approaches to this problem, and critically discuss some of the most promising approaches. Recently, a PhD thesis has appeared on this topic. Since this is the most recent substantial work on this area, our review borrows heavily from this work. Consequently, the main part of this chapter will be a summary of the approach and results of [vdM06], because it not only reviews the available work, but also because it is to our best knowledge the only one which claims to solve the problem without requiring involved measurements or additional knowledge.

2.2 Related work

The problem of provisioning link capacity to provide statistical guarantees for network QoS is not new. Much of the recent work on this topic, from both academia and industry researchers, is motivated by the fact that real network traffic cannot be characterised in a simple fashion, a fact that invalidates many provisioning rules based on assumed traffic arrival processes [Kle76]. To the best of our knowledge, the most recent work appears to have been carried out by Fraleigh, Tobagi and Diot [FTD03], by van de Meent and his co-authors [vdMPM$^+$03, MvdM05, BR06] and in the context of the Cisco supported start-up Corvil Networks [Co]. The work presented

by these authors represents to varying degrees a direct extension of the work first presented by Kleinrock. Fraleigh, Tobagi, and Diot [FTD03] use real measurements of backbone traffic to construct statistical models from which link provisioning rules are derived. Similarly, van de Meent uses traffic measurements to validate (or invalidate) rules used by network operators to size backbone links. Typically, network operators assume a relationship between the mean traffic rate and the peak traffic rate across a link to guide link provisioning rules. Van de Meent [vdMPM+03, MvdM05] uses real measurements to argue that the (typical) assumed relationship between mean and peak rate is not supported by evidence from real measurements, and proposes others which he claims are more realistic. We note that to be effective, both of these approaches require stationarity of network traffic, and accurate statistical models, and any deviation from this assumption, or model error, will invalidate the rules being proposed by these authors. The other work by the Cisco supported start-up Corvil Networks is proprietary in nature and thus difficult to quantify. However, it appears that their approach is based upon fine grained (millisecond level) traffic measurements obtained using network probes.

2.3 The model-based approach of van de Meent

In his 2006 PhD-Thesis, van de Meent presented a model-based approach to the link provisioning problem [vdM06]. Very detailed network measurements are used to generate a mathematical model of real network traffic. This model is then utilized to derive an explicit formula for the core link capacity, which not only involves the measured mean traffic level, but also the variance of the assumed Gaussian distribution. Whereas the mean is readily available using network measurements, the variance usually is not. Van de Meents principal contribution is to develop a lemma which links the distribution of the buffer occupancy, which can be estimated using available measurements, to the variance. This relation is then "inverted" to an estimate of the variance involving the empirically calculated distribution of the buffer occupancy.

2.3.1 Link provisioning

Van de Meent states the problem of bandwidth provisioning in the form of an explicit performance criterion: **namely the offered traffic shall not**

exceed the available core link bandwidth for more than a fraction ε of a time interval of length T. In mathematical terms: let T be some timescale (the timescale on which performance is perceived, usually in the order of some tens of milliseconds) and ε a small positive number. Furthermore, let $A(T)$ be the amount of traffic arriving at the core link during an interval of length T, and let C the bandwith of the core link.
Then, the goal is to find a function $f(T, \varepsilon)$ which ensures, that for $C = f(T, \varepsilon)$:

$$P(A(T) \geq CT) \leq \varepsilon \tag{2.1}$$

where P is a probability.

The first step in van de Meent's approach is to model traffic in the network core. Network traces from a variety of points were used to construct this model. In particular, measurements were taken at five different places and at different times, chosen to represent a broad number of "possible conditions". A sample thereby consists of packet data collected during an interval of 15min.
These locations were chosen with with the following characteristics:

- ▷ A location with many access links and a high utilization:
  An uplink of a student residence to the university network. The ca. 2000 access links are 100 Mbit/s each and the core link is just 300 Mbit/s, so a relatively high average load of around 60% is not surprising.

- ▷ A location with not so many access links and very modest utilization:
  An uplink of a research institute to the Dutch academic and research network. Here we have around 200 access links with 100 Mbit/s each, but the core link of 1 Gbit/s sees an average load of only around 1%.

- ▷ A location with many access links and modest utilization:
  An uplink of 1 Gbit/s, where the over 1000 access links with bandwidth 100 Mbit/s of a college are aggregated. The average load is at 10-15%.

- ▷ A location with access links of varying bandwidth and modest utilization:
  Some hundreds of ADSL customers with links of 256 kbit/s to 1 Mbit/s are loading this core link of 1 Gbit/s with ca. 15%.

- ▷ A location, where an access link has a bigger bandwidth than the core:
  A hosting provider is hosting e.g. web-servers for different companies. The access links are at around 100 Mbit/s, the uplink to the internet just 50 Mbit/s. The average load nevertheless is at around 25%.

Full details of the measurements are given in [vdM06]. The technical method of measurement that has been applied is called mirroring, illustrated in Figure 2.1: At the gateway router, an electronic copy of each incoming data packet is taken and sent to a computer which is running the *libpcap/tcpdump*-Tool [Law05a, Law05b] to record the headers (which are containing all relevant information) into a tracefile. Around 500 samples were taken.

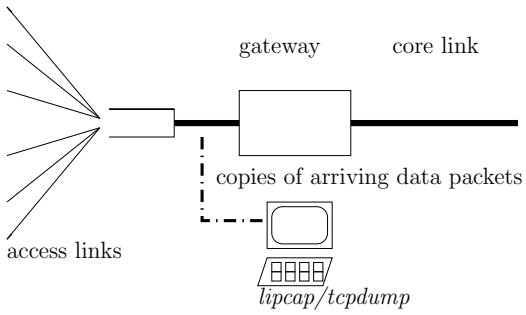


Figure 2.1: Mirroring

2.3.2 The models

Two basic types of traffic models are identified from the given traces:

Flow-based modeling Flow-based models are capturing the characteristics of individual flows. This approach encounters the first difficulty in defining what a flow actually is. There is no common definition in the literature, so possible definitions for a flow could be:

- ▷ Packets belonging to the same TCP or UDP stream
- ▷ Packets, that are exchanged between the same IP addresses (and ports)
- ▷ Packets, that originate from the same source
- ▷ and so on.

After a definition for "flow" is chosen, the model consists of three parameters:

- ▷ A random process, which describes the arrival of flows at the node under consideration.

- ▷ The probability distribution for the time arriving flows stay active.
- ▷ The rate at which flows produce traffic while they are active.

A commonly used model is the so called $M/G/\infty$ input model. That means that the time between the arrival of two flows is a Poisson distributed random variable, the duration for which a flow stays active is Gaussian distributed and the rate at which it produces traffic during that time is constant.

Black box modeling In contrast to the flow-based models, black box models aim at modeling only the aggregate of the arriving flows at the core link. Here, this refers to finding a probability distribution which describes the amount $A(T)$ of arrived traffic during a time interval of length T. Van de Meent uses measurements and the Central Limit Theorem (which, roughly speaking, states, that the sum of many independent random variables approaches a normal distribution as their number goes to infinity) to argue in favor of a Gaussian distribution: $A(T) \sim N(\mu T, \sigma^2(T))$. The mean μ of the offered traffic is easy to estimate with coarse grained measurements but opposed to that, the variance σ^2 is describing the short time behaviour and thus is dependent on the choice of the duration T and not as easy to obtain as the mean.

The fidelity of the proposed model types is then tested with the collected measurements.

For the flow-based modelling it turns out, that the traffic rates, although they are more or less constant for each flow, differ quite a lot between flows. Moreover, the duration and the rate seems not to be independent. These observations are made with different definitions for "flow" and also with additional assumptions like restricting the attention just to the flows that consume much bandwidth ("elephants") or grouping flows according to their rates, but in the end the flow-based modeling seems to be not suitable and is rejected.

For the Gaussian black-box models on the other hand, goodness-of-fit and Kolmogorov-Smirnov tests are applied and they indicate (or, more correctly speaking, they do not indicate the opposite), that the traffic of the collected measurements is more or less Gaussian, if at least some tens of users aggregate their flows in the core link and if the timescale T is not too small. Orders of magnitude between tens and thousands of milliseconds, which is

the interesting range, because as stated earlier performance is believed to be perceived in this orders, for T are "big enough".

Van de Meent found that Gaussian black box models were better because they are applicable to a wide range of conditions, require less knowledge, are far easier to estimate and, as it will turn out, easier to use in the provisioning formulas.
As an advantage of flow-based modeling, however, it is pointed out, that it provides more insight into the traffic distribution, what makes sensitivity analyses, for instance how changes in the flow rates or the number of users would affect the needed core link capacity, possible.

2.3.3 The provisioning rules

Van de Meent uses the so called Markov inequality (which states, that for any nonnegative random variable X and any $a \geq 0$: $P(X \geq a) \leq E(X)/a$) to obtain a "generic" criterion to ensure that the performance criterion (2.1) is met. With the assumption that the offered traffic $A(T)$ is a random variable distributed according to *any stationary* distribution, the following approximation for a lower bound for the core link capacity is derived:

$$C(T,\varepsilon) \;\geq\; \inf_{\vartheta \geq 0} \frac{\log E(e^{\vartheta A(T)}) - \log \varepsilon}{\vartheta\, T} \;\Longrightarrow\; P(A(T) \;\geq\; CT) \;\leq\; \varepsilon \quad (2.2)$$

For details of the derivation, see [vdM06, ch. 4.1]. Clearly, the moment generating function $\mathrm{mgf}(A(T)) := E(e^{\vartheta A(T)})$ of $A(T)$ is needed in order to evaluate (2.2), which is of course normally not known. So van der Meent uses the Gaussian model from 2.3.2 to gain an expression for the mgf.
As the mgf of a Gaussian distribution is well known, it is simply inserted in (2.2), the infimum is determined analytically, and finally, the following formula results:

$$C(T,\varepsilon) = \mu + \frac{1}{T}\sqrt{-2\sigma^2(T)\,\log \varepsilon} \quad (2.3)$$

This formula now includes the performance criterion (2.1) with ε and T, and the model of the offered traffic with μ and $\sigma^2(T)$. The variance $\sigma^2(T)$ is not directly accessible through coarse grained measurements, and so (2.3) is not directly computable. So, van de Meent uses the following constructions to estimate the variance only from coarse grained measurements.

Lemma 2.1. *Let the traffic of n sources be aggregated in a link with buffer capacity L, define l with $L = \frac{l}{n}$ and let nb denote a threshold for the buffer*

occupancy. Furthermore, assume that the traffic of each source is following a Gaussian distribution with mean $\bar{\mu}$ and variance $\overline{\sigma^2}$. Now, if Q is the buffer occupancy, then it holds for $b > 0$ and $l > \bar{\mu}$, that

$$-\lim_{n\to\infty} \frac{1}{n} \log\left(P(Q \geq nb)\right) = \inf_{t>0} \frac{(b + (l - \bar{\mu})t)^2}{2\overline{\sigma^2}(t)} \tag{2.4}$$

Although this lemma looks strange, it is providing an asymptotic relationship between the buffer occupancy *distribution* and the variance of the offered traffic.
As the lemma is not scale free (depends on the n flows from the aggregate traffic), some more approximations are made and finally are yielding the following lemma:

Lemma 2.2. *Let μ be the mean and σ^2 be the variance of the aggregate traffic, L still is the buffer capacity. Then it holds for $L > \mu$ and $B > 0$:*

$$P(Q > B) \approx \exp\left(-\inf_{t>0} \frac{(B + (L - \mu)t)^2}{2\sigma^2(t)}\right) \tag{2.5}$$

This can be inverted to

$$\sigma^2(t) \approx \inf_{B>0} \frac{(B + (L - \mu)t)^2}{-2\log P(Q > B)} \tag{2.6}$$

Equation (2.6) now makes it possible, to use coarse grained samples of the buffer occupancy to gain an empirical distribution which then can be used to approximate the variance. Then it is possible to evaluate (2.3) which yields the necessary bandwidth $C(T, \varepsilon)$.
For a detailed deduction please see [vdM06, Ch. 5].

Validation of these results is given in his thesis empirically. This is done in two steps. First, the variance is estimated using the fine grained measurements, and then is inserted in (2.3) to obtain $C(T, \varepsilon)$. It then is possible to simply "count" the spikes that are exceeding $C(T, \varepsilon)$ and compare this value to the target ε.
Although not applicable in practice, this provides a good way of validating (2.3) and it appears, that the Gaussian model tends to underestimate the frequency of the highest spikes in the offered traffic, nevertheless the formula works quite well.
Also, for every location the parameter α for rules-of-thumb of the type $C \approx (1 + \alpha)\mu$ is calculated, and it turns out, that α varies between close

to 1 and nearly 4, so as a side-effect strong evidence is given, that rules-of-thumb of that type are not very useful.
Second, the variance estimation formula (2.6) is validated by comparing the result to the empirically calculated variance of the sample measurements. The buffer occupancy distribution is thereby estimated using the measurements as input to a virtual queue.
The results suggest, that the derived formula gives a reasonable approximation for the variance.

2.4 Alternatives to over-provisioning

A completely different approach to user satisfaction is to enable network entities to distinguish between packets that are belonging to different applications and thus pose different requirements on the network. The two main implementations of this idea are the reservation based IntServ and the priority based DiffServ.

2.4.1 DiffServ

DiffServ stands for Differentiated Services and yields a relative performance guarantee by assigning each data packet to a class. A reserved byte in the IP-Header of each data packet is used to place each packet into a finite number of classes. There is no standard behaviour for each class, so each operator of a so called DiffServ domain (which refers to a group of routers that implement the same DiffServ policies) is free to choose, how packets belonging to different classes are treated. Nevertheless, in practice some classes are recommended, for instance "Expedited Forwarding" (EF) for traffic which requires low loss and low latency.

So each packet entering a DiffServ aware router is treated according to its class and the policies of the DiffServ domain the router belongs to. There are two major points of critique on this approach. One is, that if a packet passes several networks on its way to its destination, its class can be treated differently in each one of them, which makes end-to-end behaviour unpredictable. Another flaw is, that, as everyone wants his packets to be transmitted, there is a tendency to mark packets with the class of highest priority so in the end one could end up with all packets marked highest priority and thus DiffServ virtually disabled.
For details on DiffServ see e.g. [BBC+98] and the references therein.

2.4.2 IntServ

IntServ stands for Integrated Services. The basic idea is, that flows send reservation requests to the routers on the way to their destinations, and routers either accept or reject these reservations.

This offers an absolute performance guarantee, but the disadvantages are obvious: Each router on the way has to support IntServ. Moreover, IntServ forces the routers to keep a lot of information about reservations, which increases the complexity and introduces scalability problems.
For details on IntServ see [BCS94] and references therein.

2.5 A critique of proposed approaches

As for the model-based approaches, it is generally accepted that no model can cover every condition. In other words models will be wrong under certain circumstances. For example, the mixture of internet traffic changes continuously: Some years ago, bandwidth did not allow for large down- or uploads, whereas novadays, movies and large applications are shared and sold via the internet. At the moment, VoIP and video streaming gain more and more importance, and so any new application or protocol can change the traffic mixture conceivably and thereby invalidate empirically derived models completely.

Additional techniques like DiffServ and IntServ have (in addition to those mentioned in Section 2.4) the intrinsic disadvantage, that additional techniques cause additional complexity, what creates the need for better trained and thus more expensive staff on the one hand and leads to scalability problems on the other hand.

So an approach which ensures QoS without introducing additional complexity and without dependency on the applicability of model assumptions would definitely spark interest in many different areas. We hope to present such an approach in the remainder of this thesis.

Chapter 3

Feedback control and link provisioning

3.1 Structure of the chapter to come

In this chapter, we first recall the shortcomings of earlier approaches to motivate the method proposed here (Section 3.2). The rest of this chapter is structured as follows. We begin in Section 3.3 by reviewing ADT, which can be seen as the "older brother" of our approach. We then introduce a feedback-based approach to the link provisioning problem, and point out the parallels with ADT [SSK06, KSL06] (Section 3.4).
Section 3.5 is dedicated to the technical details of possible control algorithms and finally, in Section 3.6, some simulation examples for the usefulness of the control-based approach are given. These examples illustrate that the queue information can indeed be utilized to obtain fine information concerning traffic at a core link, without the need for finely sampled measurement data.

3.2 Shortcomings of earlier approaches

The approaches in Chapter 2 suffer major shortcomings.

Principally, model based approaches rely crucially on the fidelity of the proposed model. However, all models are only as good as the assumptions that they are built under, and given the nonstationary behaviour of networks, and network traffic, such models are highly questionable in internet environ-

ments. These concerns clearly undermine rules-of-thumb for link provisioning derived from these models.

Conventional measurement-based approaches suffer from the drawback, that the available, coarse grained measurements itself may not contain enough information to base a link provisioning on. Such approaches, to function effectively, must have access not only to mean traffic levels, but also to burst behaviour. Such information is clearly not available without finely sampled measurements [Co].

Our control-based approach however overcomes these concerns. We exploit information in the queue size that not only reflects traffic patterns over short time scales, but also information on traffic bursts.

3.3 Preamble: Adaptive Drop-Tail queueing

The basic idea exploited here has been proposed to solve a different problem. Network buffers are typically provided with capacity according to the BDP (Bandwidth Delay Product) rule. The BDP rule determines the amount of buffering needed to ensure full utilization of a *congested* link and gives the buffer capacity $L = BW \times \overline{RTT}$, where $\overline{RTT}$ is the average round trip time, under the assumption of source synchronisation.

In [AKM04], Appenzeller et. al. proposed an improved formula exploiting the fact that TCP flows backoff in an asynchronous manner. He showed, for highly multiplexed links, that buffers can be sized as $L = \frac{a}{\sqrt{n}} BW \times \overline{RTT}$, where n denotes the number of active flows that are sharing the link under observation and a is a scaling factor which depends on the desired utilization of the link. This utilization can be chosen arbitrarily close to 100% while still keeping a small, for instance corresponds $a = 2$ to a utilization of 99.99997%. Considering, that core links normally carry tens of thousands of flows simultaneously, the reduction of necessary buffer space, and hence the reduction of the costs for memory devices, is huge. The problem in applying this result lies in the number of active flows, which is normally not known. In [Mor00], this problem is handled by parsing every arriving packet to maintain an internal counter. Indeed, this is not possible with standard routers and thus again makes additional technologies necessary.

But the mere observation, that there exists an unique optimal queue capacity for every utilization that only depends on some "environmental" conditions can be used as a starting point for a closed-loop algorithm that seeks to find

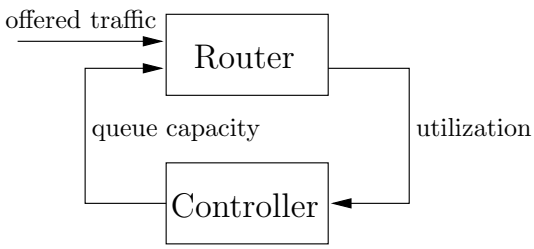


Figure 3.1: The signal flow diagram of ADT

the target buffer for a given target utilization. This idea was introduced in [SSK06]: Adaptive Drop-Tail (ADT). This algorithm applies a control scheme which is trying to find the minimum queue capacity to reach the target utilization. The ADT algorithm consists basically of two steps: In the first step, the controller calculates the average utilization over the last time interval, using the readily available measurements via SNMP. As a second step, the controller decides on the new queue capacity according to some algorithm. This algorithm can be chosen out of a vast range of control algorithms, which of course all have in common that they react to a too low utilization with an increase in queue capacity and vice versa. The new queue capacity is then applied to the router via `ssh`, a functionality which most routers already support.
The implementation of this idea is non-invasive and only requires one additional workstation which is polling the router via SNMP, evaluating the control algorithm and applying the new queue capacity via `ssh` every time interval. Typical time intervals are in the order of minutes, so absolutely no complexity is added. See the signal-flow diagram of the feedback loop in Figure 3.1 and a schematic illustration of the implementation in Figure 3.2.

3.4 Control-based core link provisioning

As already mentioned in Section 3.3, ADT is useful for congested links. A sufficiently over-provisioned core link opposed to that is *not congested*, so very low utilization is intrinsic to it and an ADT would not make much sense in that case.

So instead, one could employ the queue capacity to control another important parameter, namely the loss rate λ. We will show, that using the queue capacity to control λ yields additional insights into the traffic arrival process

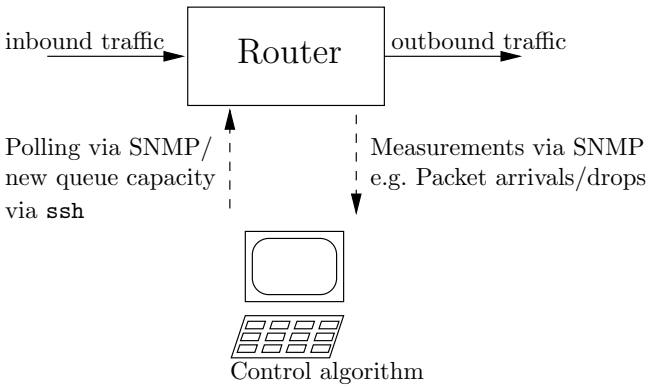


Figure 3.2: Schematic implementation of ADT or control-based link provisioning, respectively

which can be used to trigger link upgrades.

3.4.1 Motivating example

Example 3.1 (On/Off CBR traffic)**.** Imagine a very artificial traffic scenario where the traffic consists of a constant bitrate (CBR) flow with rectangular shaped bitrate, oscillating with amplitude α around the mean μ (see Figure 3.3). For that basic case, it is very easy to derive a formula which connects the queue capacity L and the loss rate explicitly: $\lambda := \frac{\#\{\text{lost bytes}\}}{\#\{\text{arrived bytes}\}}$:
The queue has to cater for the traffic exceeding the Bandwidth β; if the queue is full, the exceeding bytes get dropped. In the phase, where the offered traffic is less than the bandwidth, no bytes are lost, so:

$$\#\{\text{lost bytes}\} = \max\{0, (\mu + \alpha - \beta)T - L\}$$

And, of course $\#\{\text{arrived bytes}\} = (\mu + \alpha)T + (\mu - \alpha)T = 2\mu T$. Thus if we assume that there are losses at all:

$$\lambda = \frac{(\mu + \alpha - \beta)T - L}{2\mu T} \tag{3.1}$$

which is then easily inverted to

$$\alpha T = T(2\lambda - 1)\mu + \beta T + L \tag{3.2}$$

or

$$L = T(\mu(1 - 2\lambda) + \alpha - \beta) \tag{3.3}$$

As L is known and μ and λ are available through coarse grained measurements, we could infer an "effective amplitude" of the offered traffic for any timescale T we choose and, more important, that makes it possible to differ between two scenarios with the same mean.

Simulation results that illustrate the derived relations are given in Example 3.2.

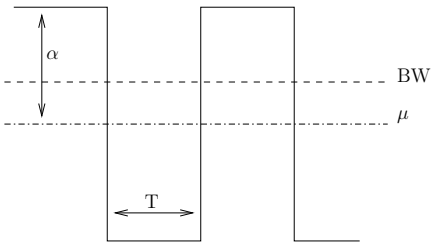


Figure 3.3: Cutout of the offered Traffic in Example 3.1. BW is the bandwidth β of the link, μ is the mean, α the amplitude, and T half the period of the oscillating offered traffic.

3.4.2 Basic idea

The basic idea described here is very simple. Instead of measuring the short time scale behaviour, we employ a control algorithm to find the queue capacity that corresponds to a pre-specified loss rate. Indeed, the target loss rate has to be a very small one, otherwise it would lead to severe performance degradation.
Intuitively, traffic with more spikes that are exceeding the core link capacity should require a bigger queue capacity. So the queue capacity is capable of tracking short time scale behaviour that is not accessible by just observing long term averages. Thus, a rising trend in the queue capacity could be seen as an indication that an upgrade of the core link is advisable.

Another interpretation also is available:
If you have a gateway router to a core link with a fixed (and normally large) buffer capacity, losses only occur when the offered traffic is exceeding the bandwidth in an amount (that means "high" and "long" enough), sufficient to fill the buffer. Then, the number of losses is a secure indicator for how much the link is underprovisioned. So the event of the loss rate exceeding a certain threshhold could be seen as the "too small"-event, the upgrade

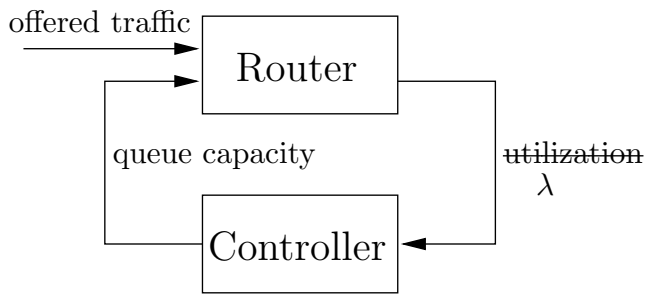


Figure 3.4: The signal-flow diagram of the loss rate control

should be done "at once". Considering that we are talking about core links which normally are laid subsurface, an instantaneous upgrade certainly is not possible.
So the alternative interpretation of our approach could be: by decreasing the queue capacity, we force "too small" events, but we can react to this kind simply by increasing the queuesize. If the event was only valid temporarily, the controller will decrease the capacity again, but a number of "too small"-events at increasing capacities then can be used to trigger the upgrade.

The connection to ADT (Section 3.3) is pointed out in the signal-flow diagram in Figure 3.4. The implementation can be illustrated with Figure 3.2 because it is the same as for ADT.

3.4.3 Discussion of the control-based approach

This approach does not suffer from the weaknesses of other approaches. In particular it is characterised by the following properties.

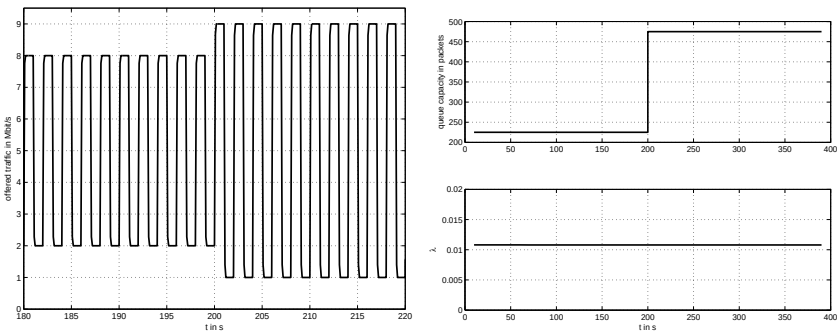
(a) Model-independence: Because our approach is a *closed-loop* approach, it does not depend on a certain model for the traffic arrival process. Model-based approaches on the other hand rely on the applicability of the models they are based on.

(b) It is non-invasive, easily implemented: The controller relies only on coarse grained measurements which are readily available by polling the router via SNMP. The control action of adjusting the queue capacity is only to be executed in the same frequency as the polling and can be done via `ssh`, since most routers novadays are equipped with this functionality. So no additional technologies are needed, just one

workstation in the network to gather the information and adjust the queue capacity. For an illustration of the implementation see Figure 3.2

Of course, there are some drawbacks as well. Opposed to performance criteria like (2.1), our approach is not based upon an explicit performance criterion, so some intuition is still needed to decide which queue capacity is "too much". Further, care must be taken while choosing the target loss rate. Chosen too small, it will make control useless because it might not even correspond to a single dropped packet during a sampling interval, chosen to high it will degrade the utilization because the TCP senders "think" the link is already congested while in fact it is not.

Example 3.2 (The motivating example revisited)**.** We ran a $ns2$ simulation of the scenario of Example 3.1. Our parameters are: $\beta = 7$Mbit/s, $\mu = 5$Mbit/s, $\alpha_1 = 3$Mbit/s, $\alpha_2 = 4$Mbit/s and $T = 1s$. The packet size is hereby fixed to 500bytes, so the calculations can be directly converted from the unit bits to packets. See the results in Figure 3.5.
This should illustrate, that (3.3) does indeed give the right queue capacity, or in other words, that a perfect control would do in that case, what we expect: The queue capacity informs us of the increased peaks in the offered traffic.



(a) A cutout of the offered traffic (b) Queue capacity and lossrate λ

Figure 3.5: $ns2$ results. The slight deviation between λ and the target of 0.01 is to be blamed on the integer nature of the process: The computation of (3.3) requires at least rounding to whole bits, the conversion to packets introduces another rounding error.

3.5 Control algorithms

The basic difficulty in this control problem is, that there is **no model** for our plant. The only assumption, that clearly holds, is that the gain indeed is non-positive: A bigger queue capacity never yields a higher loss rate.

A second problem is caused by the **discrete-event nature** of this problem: Although we are treating loss rate as a continuous quantity, it of course is a discrete one. The highest possible resolution is either a single packet or a bit. Nevertheless, we are ignoring this fact, which seems reasonable, as we are focussing on big core links and large sampling intervals, so a sample contains some tens of thousands up to some tens of milions of packets. The controller maintains the floating point variable $\tilde{q}$ as a state, which is then rounded to an integer value before being sent to the router. Still, some care needs to be taken with the target loss rate, because it should be "reachable", which means it should not be 1.5 packets or 0.2 bytes.

Another difficulty is that of **limited controller output**: Of course, the queue capacity has an upper threshold as well as a lower threshhold. It is known to control engineers, that saturating inputs in connection with integrating controllers can lead to so called Wind-Up phenomena. Roughly speaking this means, that the controller does not notice the saturating input and keeps on demanding higher values. We address this problem by simply limiting the state of the controller $\tilde{q}$. Therefore we define a function SAT : $\mathbb{R} \to [\underline{x}, \overline{x}]$ as follows:

$$\mathrm{SAT}(x) = \begin{cases} \overline{x} & \text{if} \quad x > \overline{x} \\ x & \text{if} \quad \underline{x} \leq x \leq \overline{x} \\ \underline{x} & \text{if} \quad x < \underline{x} \end{cases} \tag{3.4}$$

where $\overline{x} > \underline{x}$ are some upper and lower limits.

Another peculiarity is the **asymmetry** between positive and negative control deviations. A too low loss rate would not trouble anybody, whereas a large peak in the loss rate would cause some problems: Time consuming retransmits and TCP sources backing off would degrade the performance conceivably.

So model-based algorithms like MPC or feedback linearisation are not suitable for this problem. Instead, we focus on simple and robust control algorithms. Therefore we employ a linear filter to smoothen the data, which despite of relatively long sampling intervals still is very noisy. The filter will be detailed in Section 3.5.1.

The filter will then be combined with a simple control algorithm, either a linear integral controller, a MIMD or a MIAD algorithm. Those will be described in Sections 3.5.2, 3.5.3 and 3.5.4.

3.5.1 Linear filter

As we here are dealing with sampled data, the behaviour of the filter can be described either in the z-domain or as a difference equation. Denote the input of the filter, which will be the control deviation, with e, and its output, which will be the input for the control algorithm, with u. The order of the filter N_F gives the length of its memory, i.e. over how many intervals it is averaging.
Now let ρ_i, $i = 1 \dots N_F - 1$, with $\rho_i \geq 0$ and $\sum_{i=0}^{N_F} \rho_i = 1$ denote some weighting factors. Then, the transfer function of such a linear filter is given by

$$F(z) = \frac{\rho_{N_F} z^{N_F}}{z^{N_F} - \sum_{i=0}^{N_F-1} \rho_i z^i} \tag{3.5}$$

The resulting difference equation is

$$u(k + N_F) = \rho_{N_F} e(k + N_F) + \sum_{i=0}^{N_F-1} \rho_i u(k + i) \tag{3.6}$$

or, equivalently, after applying a time shift of N_F time steps

$$u(k) = \rho_{N_F} e(k) + \sum_{i=0}^{N_F-1} \rho_i u(k - N_F + i) \tag{3.7}$$

Note, that for the sake of simplicity, the sampling time T_s is set to 1 here. Otherwise every argument of e and u had to be multiplied by T_s.

3.5.2 Linear controller

As the system is subject to a lot of noise, a differentiating part does not seem wise. The same applies for a proportional part, so the first proposal is a simple integral controller. Note, that the system is indeed a sampled one, so the control law is given by

$$K(z) = \frac{\gamma z}{z - 1} \quad \sim \quad q\left((k+1)T_s\right) = \gamma u\left((k+1)T_s\right) + q(kT_s) \tag{3.8}$$

where T_s denotes the sampling time, γ is the gain of the controller, q is the output of the controller (viz the queue capacity) and u is the input to it. Further, note that, as the gain of the plant is non-positive, at some stage a negative gain needs to be introduced into the loop. We will do this by computing the deviation as $e = \lambda - \lambda^*$.

A possible extension to cater for the asymmetry mentioned in the introduction of a "relative gain" γ_{rel}. This means that for a negative deviation another gain is used than for a positive one.

$$\gamma \rightsquigarrow \begin{cases} \gamma_{rel}\gamma & \text{if} \quad u > 0 \\ \gamma & \text{else} \end{cases} \tag{3.9}$$

Algorithm 1 Linear controller

if now - lasttime $> T_s$ **then**
  $\lambda = \frac{\#\{\text{dropped since lasttime}\}}{\#\{\text{arrived since lasttime}\}}$
  deviation $= \lambda - \lambda^*$
  filtered_deviation $= \rho$ deviation $+ (1 - \rho)$ filtered_deviation_lasttime
  if filtered_deviation > 0 **then**
    $\tilde{q} = \text{SAT}(\tilde{q} + \gamma\ \gamma_{rel}$ filtered_deviation$)$
  else if filtered_deviation < 0 **then**
    $\tilde{q} = \text{SAT}(\tilde{q} + \gamma$ filtered_deviation$)$
  end if
  $q = \text{round}(\tilde{q})$
  lasttime = now
  filtered_deviation_lasttime = filtered_deviation
end if

Note that in algorithm 1, only a first order filter ($N_F = 1$) is described. Higher orders can be realized analogously.

3.5.3 Multiplicative Increase Multiplicative Decrease - MIMD

In [SSK06], a MIMD algorithm is proposed. Of course, a similar algorithm is applicable here.

It is described in algorithm 2. T_s denotes the length of a sampling interval, $I > 1$ and $0 < D < 1$ are the increase and decrease parameters. The asymmetry mentioned in the introduction to this chapter should be accounted

Algorithm 2 MIMD

```
if now - lasttime > T_s then
  λ = #{dropped since lasttime} / #{arrived since lasttime}
  deviation = λ − λ*
  filtered_deviation = ρ deviation + (1 − ρ) filtered_deviation_lasttime
  if filtered_deviation > 0 then
    q̃ = SAT(q̃ × I)
  else if filtered_deviation < 0 then
    q̃ = SAT(q̃ × D)
  end if
  q = round(q̃)
  lasttime = now
  filtered_deviation_lasttime = filtered_deviation
end if
```

for by choosing $I > D^{-1}$, which implies that in the case of a too high loss rate followed by a too low loss rate, the queue remains at a bigger capacity than before.

3.5.4 Multiplicative Increase Additive Decrease - MIAD

As the TCP congestion control is based on a AIMD algorithm, it seems straight forward to base the loss rate control on the counterpart, namely a MIAD algorithm. This algorithm is respecting the asymmetry even more than the MIMD algorithm as the multiplicative increase is responding much more aggressively to too high loss rates than the additive decrease is to too low loss rates.

Again, T_s denotes the length of a sampling interval, I and D are the increase and decrease parameters. This time, D should be a positive integer, $D \in \mathbb{Z}^+$.

3.6 Basic $ns2$ simulation results

In this section we will present some simulation results to illustrate further, how the control-based approach can yield additional information and how this additional information could be interpreted. Therefore we will first revisit the

Algorithm 3 MIAD

```
if now - lasttime > T_s then
    λ = #{dropped since lasttime} / #{arrived since lasttime}
    deviation = λ − λ*
    filtered_deviation = ρ deviation + (1 − ρ) filtered_deviation_lasttime
    if filtered_deviation > 0 then
        q = SAT(q × I)
    else if filtered_deviation < 0 then
        q = SAT(q − D)
    end if
    lasttime = now
    filtered_deviation_lasttime = filtered_deviation
end if
```

motivating Example 3.1 and apply a controller to it to show, that it indeed finds the appropriate queue capacity and by that indicates the difference between the two parts of the experiment.
In a second experiment, we show how the queue can make the difference between a rather smooth arrival process and a rather bursty one, namely a superposition of long-lived TCP-flows and Poisson distributed inter-arrival times, visible.

3.6.1 The motivating example re-revisited

We are now using the same traffic arrival process as in Example 3.2, i.e. rectangular oscillations around mean μ with amplitude α_i and period $2T$. $\beta = 7$Mbit/s, $\mu = 5$Mbit/s, $\alpha_1 = 3$Mbit/s, $\alpha_2 = 4$Mbit/s and $T = 1s$ (also see Figure 3.3). But instead of just calculating the appropriate queue capacity and applying it, we connect the controller as depicted in Figure 3.4. The control algorithm is the integral controller with a first order filter, described in algorithm 1. The parameters for the controller are $\gamma = 600$, $\gamma_{rel} = 2$, $T_s = 60s$, $\rho = 0.7$ and $N_F = 1$.
The results are shown in Figure 3.6 and are quite convincing. Note that, of course, this is no big surprise, as the example is a very artificial one. Nevertheless it is apt to give further motivation for our approach.

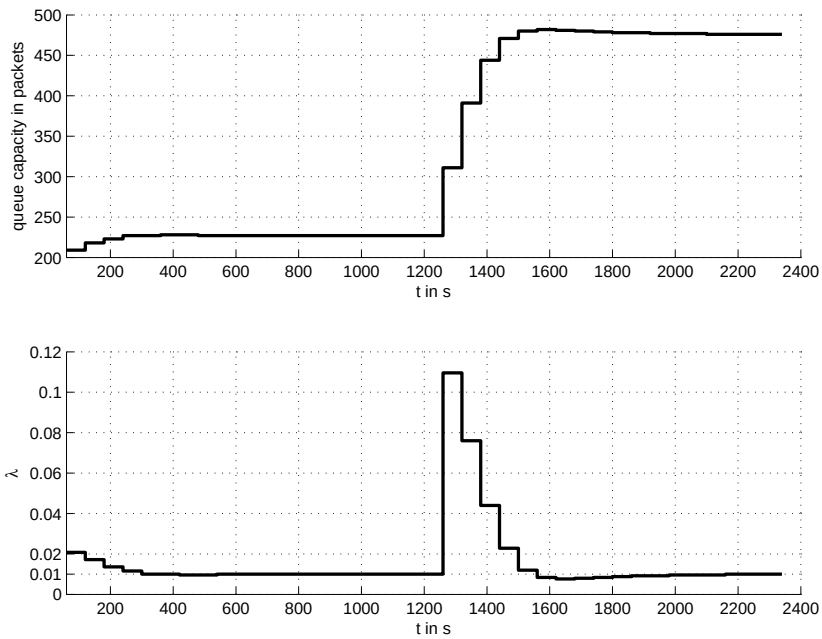


Figure 3.6: Simulation results for Section 3.6.1. The traffic arrivals are the same as in Example 3.2 and are shown in Figure 3.5(a).

3.6.2 Smooth traffic versus bursty traffic

In this section we show how our approach enables us to distinguish between two arrival processes that show quite similar behaviour on the coarse grained timescale of the SNMP measurements but exhibit quite different behaviour on finer timescales, i.e. one has a significantly higher variance than the other.

The first scenario will be Poisson-distributed interarrival times, i.e. the time between two packet arrivals is a Poisson distributed random variable whose mean (along with the size of a single packet) can be used to control the mean arrival rate. This will be the scenario with the higher variance.
The second scenario will include rather well behaved long-lived TCP flows, where we use the number of active flows to control the arrival rate.

The set-ups

Example 3.3 (Poisson arrivals)**.** For this scenario, we set up two links in *ns*2, connected by a drop tail buffer whose capacity is controlled by our algorithm. The bandwidth of the first link has to be only big enough,

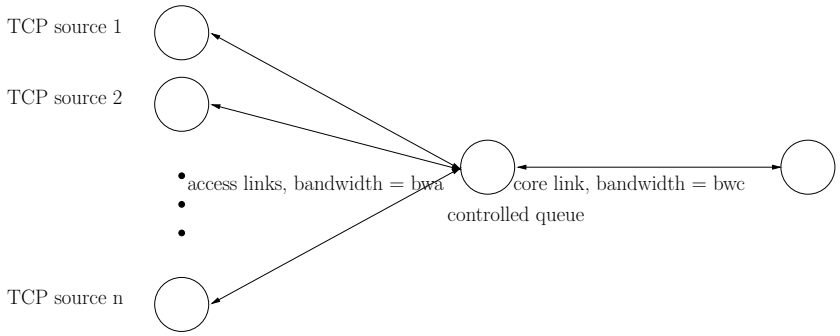


Figure 3.7: Topology of the TCP setup

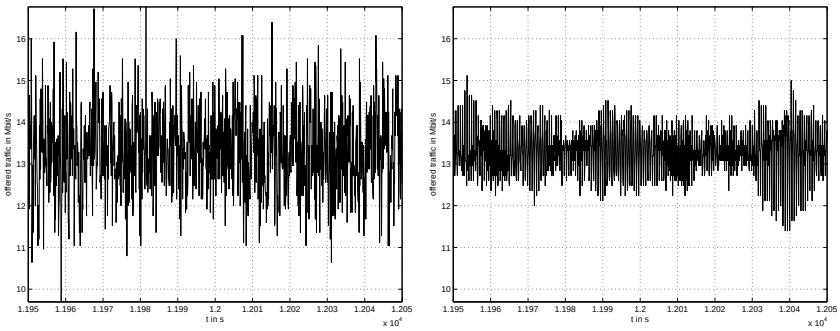
because it will be carrying the Poisson distributed traffic (representing the superposition of all access links) to the core link. The core link has the capacity $bwc = 15$Mbit/s.
We start with an average offered traffic well below the core link bandwidth and increase the intensity every fixed timestep (around 15min) by decreasing the mean interarrival time. The decreases are chosen in a way, that the mean traffic arrival rate increases by 0.1Mbit/s every time.
The integral controller from algorithm 1 with a first order filter is used. The parameters are: $T_s = 60s$, $\gamma = 6000$, $\gamma_{rel} = 5$ and $\underline{q} = 30$. The filter constant is $\rho = 0.7$ and the target loss rate is $\lambda^* = 5 \cdot 10^{-4}$.

Example 3.4 (Long-lived TCP flows)**.** In this experiment, a topology like depicted in Figure 3.7, where there is a clear distinction between access links and core link, was set up in $ns2$.
A certain number N_0 of TCP-flows was started at once and then one was added every fixed time interval to slowly increase the load on the core link. The bandwidth of the access links and the core link, respectively, is $bwa =$ 0.1Mbit/s and $bwc = 15$Mbit/s. This parameters are chosen to make this experiment comparable to the last one, so the aggregate traffic increases in the same steps. The controller parameters are also the same as in Example 3.3.

A sample of the offered traffic for both scenarios is shown in Figure 3.8. The measurements are taken on a much finer timescale than the controller is able to see, namely the plots have a 600 times finer resolution than the data the controller uses.
The different "burstinesses" are easy to recognize. To be able to assign a value to this observation, we took samples of 500 consecutive measurement points

and calculated their variance. This yielded a variance of $\sigma^2_{pois} \approx 1(\text{Mbit/s})^2$ for Figure 3.8(a) and $\sigma^2_{tcp} \approx 0.5(\text{Mbit/s})^2$ for Figure 3.8(b).

Following our earlier reasoning for the control-based approach, the more bursty traffic, namely the Poisson traffic, should require an update earlier than the rather smooth TCP-traffic. Thus, we should see the required queue capacity rising earlier in the first scenario than in the second one.



(a) Poisson case (Example 3.3) (b) TCP case (Example 3.4)

Figure 3.8: Cutouts of the fine grained measurements of the offered traffic. Measurements have been taken every 100ms. The different "burstiness" is obvious.

Results and interpretation

In Figures 3.9 and 3.10, the measurements of the offered traffic the controller has access to (i.e. on the timescale of T_s, which is 60s in this case) and the queue capacity is shown for both scenarios. This is basically the information, a network operator can base his decision on upgrading the link on.
Although not completely identical, the offered traffic looks quite similar, whereas there is a big difference in the queue capacities!

In the Poisson case (Figure 3.9), we see the queue steadily rising from approximately $t_{pois} \approx 16000$s, when the traffic level reached $A_{pois} \approx 13.7$Mbit/s. This already indicates that "something is going on", when the margin between the average offered traffic and the core link bandwith still is $\beta - A_{pois} \approx 1.3$Mbit/s. More precisely, from a rising queue the network operator can tell, that spikes are reaching the core link bandwidth in a sufficient frequency and height to require an increase in the queue capacity.

In contrast to that, in the TCP case (Figure 3.10), the queue is showing no

activity until the link is almost fully utilized. Then it rises very quickly to its maximum.
Indeed, in this scenario, there is no early warning from the queue. But, thinking the other way round, this just shows, that a well-behaved traffic like this can be operated with no problem until a utilization of almost 100%. So in this scenario, the offered traffic could (and should) be used as a trigger instead of the queue.

The simulations demonstrate the effectiveness of our approach as well as the fact, that our approach is rather able to supply decision makers with additional information than to yield explicit criteria one can apply without further knowledge.

This two examples demonstrated, how a clear distinction between a well-behaved traffic scenario and a burstier one can be made *without requiring more than the available, coarse grained measurements* and without introducing any noticeable complexity. The queue indicated the spikes of the bursty traffic early and allowed operation of the core link close to congestion in the less bursty case.

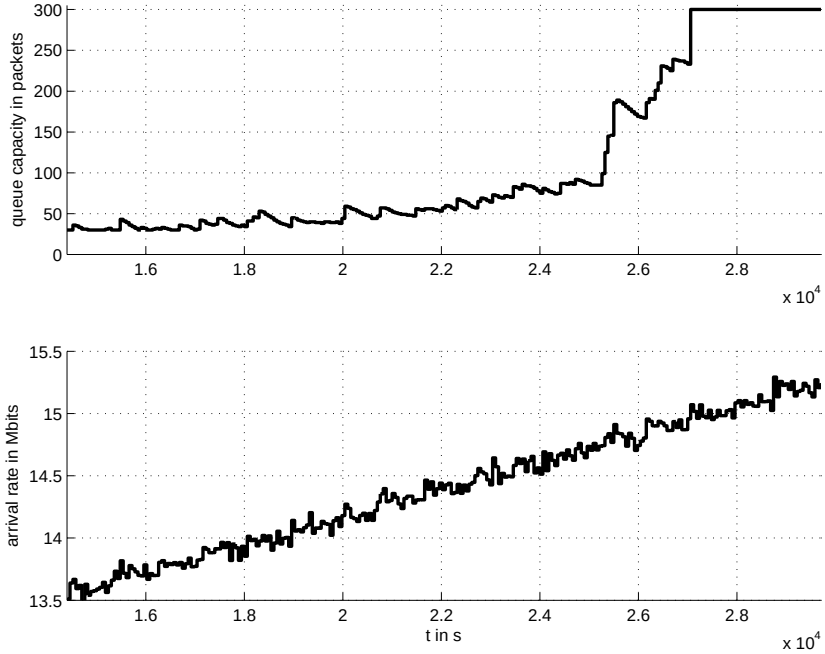


Figure 3.9: $ns2$ simulation results for Example 3.3.

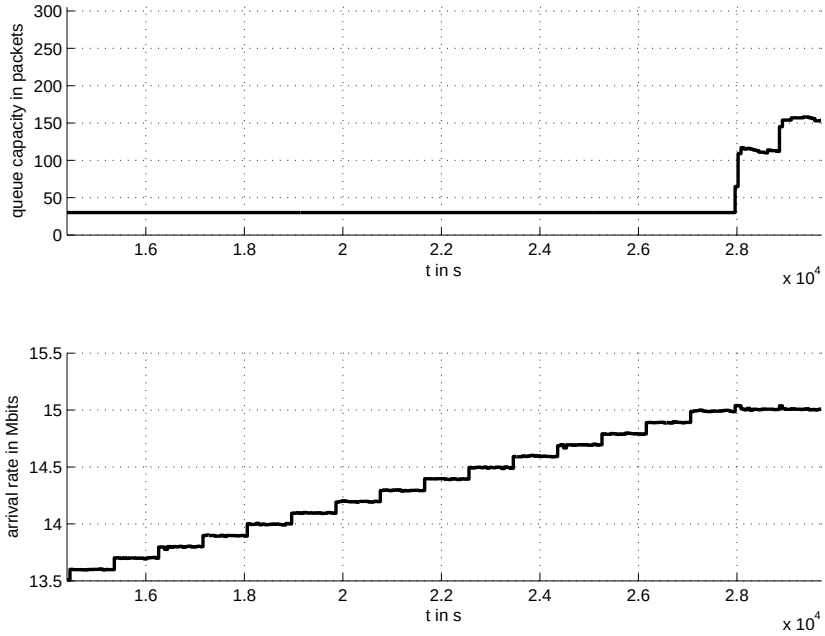


Figure 3.10: $ns2$ simulation results for Example 3.4.

Chapter 4

Theoretical contribution: Stability of discrete time switched linear and Lur'e systems

4.1 Overview

In this chapter, stability results using common quadratic Lyapunov functions on switched linear systems and the Lur'e absolute stability problem will be reviewed and connected. Additionally, the Bilinear Transform will be utilized to translate discrete time problems into continuous time problems.

In Section 4.2, we start with important definitions on systems, switched linear systems and Lyapunov functions and in Theorem 4.4, the first and most important stability result on switched systems will be given.
Section 4.3 will contain the definition of the Lur'e system and connect its stability to switched systems.
The first stability results on continuous time problems will be given in Section 4.4, including the circle criterion and spectral criteria. Robust versions of the spectral criteria will be added in Section 4.5.
In Section 4.6, the translation of discrete time problems to continuous time will be demonstrated and some discrete time versions of earlier results will be given.

4.2 Switched systems and Lyapunov functions

Definition 4.1 (Continuous, discrete-time and continuous-time systems)**.** Let $n \in \mathbb{N}_+$ and f, g: $\mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}_+ \to \mathbb{R}^n$ for the continuous time case and $\mathbb{R}^n \times \mathbb{R}^p \times \mathbb{N} \to \mathbb{R}^n$ for the discrete time case be functions. Then we have the system equations

$$\dot{x}(t) = f(x(t), u(t), t); \qquad x(t_0) = x_0,\ t \geq t_0 \tag{CT}$$
$$y(t) = g(x, u, t) \tag{CT}$$

and

$$x(k+1) = f(x(k), u(k), k); \qquad x(k_0) = x_0,\ k = k_0, k_0 + 1, \ldots \tag{DT}$$
$$y(k) = g(x, u, t) \tag{DT}$$

x is called the **state** of the system, x_0 is called **initial state** and n is the **dimension** of the system. u is called the **input** and y is called the **output** of the system. The functions f and g are called the **state** and **output equations**. We will denote a **solution** of the system according to a certain initial state x_0 and a given input sequence u with $x(t; u, x_0)$ or $x(k; u, x_0)$ respectively.
The systems are:

continuous if the state of the systems evolves in (a continuous subset of) $\mathbb{R}^n$;

autonomous if there are no inputs;

time-variant if f and g directly depend on t or k, respectively and;

time-invariant if f and g do not directly depend on t or k, respectively.

For the sake of simplicity and without loss of generality, we will assume here, that for time-invariant systems $t_0 = 0$, $k_0 = 0$ and that, if equilibria exist, one of them is the origin. Otherwise we can simply shift the state-space: If $x_s \neq 0$ is an equilibrium, then define $\xi := x - x_s$. In the new state ξ, $\xi = 0$ is an equilibrium.

Many of the following definitions and Theorems will have analogous formulations for discrete and continuous time. We will indicate the relation to continuous time by CT and discrete time by DT.

Definition 4.2 (Autonomous LTI systems)**.** Let, for the sake of simplicity, $g(x, u, t) = x$, that is the identity. A system of the form (CT) or (DT) is called a **linear time-invariant (LTI)** system, if f is a linear function and does not depend on the time t or k, respectively, i.e. if there exists a matrix $A \in \mathbb{R}^{n\times n}$ with which the system equations can be written as

$$\dot{x} = Ax; \qquad x(t_0) = x_0 \tag{LCT}$$

and

$$x(k+1) = Ax(k); \qquad x(k_0) = x_0 \tag{LDT}$$

A is called the **system matrix** of the LTI system. As the systems (LCT) and (LDT) are uniquely defined by their system matrices, we sometimes will refer to the "system A". Note, that $x = 0$ always is an equilibrium of (LCT) and (LDT).

Definition 4.3 (State-space representation of a LTI system with inputs)**.** The system descriptions

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{4.7a}$$
$$y(t) = Cx(t) + Du(t) \tag{4.7b}$$

in continuous time and

$$x(k+1) = Ax(k) + Bu(k) \tag{4.8a}$$
$$y(k) = Cx(k) + Du(k) \tag{4.8b}$$

in discrete time are called a **state-space representation** of a system.
We will denote that by $[A, B, C, D]$.
Here, x is again the **state** of the system, u is the **input** and y is the **output**.
If B and C^T are column vectors, that is $x, u \in \mathbb{R}$, the system is called a **single-input single-output (SiSo) system**.

Definition 4.4 (Transfer function and realization of a SiSo LTI system)**.** Let $[A, B, C, D]$ be a state-space representation of a SiSo system, either CT or DT.
We call $G(s) = C(sI - A)^{-1}B + D$ for CT and $G(z) = C(zI - A)^{-1}B + D$ for DT the **transfer function** of the system.
Vice versa, $[A, B, C, D]$ is called a **realization** of $G(s)$ or $G(z)$, respectively.
Note that there are infinitely many realizations for any proper (see the next definition) transfer function.

Definition 4.5 (Proper, strictly proper and biproper)**.** Let $G(w) = \frac{b_0+b_1+\ldots b_m w^m}{a_0+a_1 w+\ldots a_n w^n}$ be a transfer function, where w is either s or z. Then the transfer function is called
proper if $m \leq n$;
strictly proper if $m < n$ (Note that in that case, $D = 0$ in any realization);
biproper if $m = n$ (In that case, $D \neq 0$ in any realization); and
improper if $m > n$. In that case, no realization exists.

4.2.1 Lyapunov functions

The idea behind Lyapunov theory is the following: If the distance between any trajectory and an equilibrium decreases with time to 0, then this equilibrium is asymptotically stable. Now trajectories do not always choose the direct way to the equilibrium, so the Euclidian distance does not decrease necessarily all the time. But if one can find a kind of generalized distance that decreases monotonically, this generalized distance could be used to prove stability by looking at its time derivative. This generalized distance is called a Lyapunov function.

Definition 4.6 (Lyapunov function candidate)**.** A continuous and differentiable function $V\colon \mathbb{R}^n \to \mathbb{R}$ is called a **Lyapunov function candidate**, if

$$V(x) \to \infty \qquad \text{if } \|x\| \to \infty \qquad \text{(radially unbounded)} \tag{4.9a}$$
$$V(x) \geq 0 \qquad \forall x \in \mathbb{R}^n \qquad \text{(positive semidefinite)} \tag{4.9b}$$
$$V(x) = 0 \qquad \Leftrightarrow x = 0 \tag{4.9c}$$

Note, that every vector norm is a Lyapunov function candidate. This supports the interpretation as a generalized distance. Now we will state, for which cases this generalized distance is useful and in Theorem 4.1 the stability criterion will be given.

Definition 4.7 (CT Lyapunov function)**.** A Lyapunov function candidate is a **Lyapunov function (LF)** for the system (CT), if it is strictly decreasing along solutions of (CT), i.e.:

$$\dot{V}(x(t;x_0)) = \frac{\partial V}{\partial x} f(x,t) < 0 \qquad \forall\, t > 0,\ x, x_0 \in \mathbb{R}^n \backslash \{0\} \tag{4.10}$$

Definition 4.8 (DT Lyapunov function)**.** A Lyapunov function candidate is a **Lyapunov function (LF)** for the system (DT), if it is strictly decreasing

along solutions of (DT), i.e.:

$$V\left(x(k+1)\right) - V\left(x(k)\right) = V\left(f(x(k))\right) - V\left(x(k)\right) < 0 \qquad \forall k > 0,\ x, x_0 \in \mathbb{R}^n \backslash \{0\} \tag{4.11}$$

Theorem 4.1 (Global asymptotic stability)**.** *The origin is globally asymptotically stable for the system* (CT) *or* (DT)*, respectively.*
$\Longleftarrow$
There exists a Lyapunov function for the system (CT) *or* (DT)*, respectively.*

So far, no hint on how to actually obtain such a Lyapunov function has been given. For a general system like (CT) or (DT), there is no general answer to this question. Usually this presents no problem since we are only concerned whether a Lyapunov function exists at all. However, for the LTI systems (LCT) and (LDT), there are some very simple results which we will now mention briefly.

Definition 4.9 (QLF)**.** A function of the form $V(x) = x^T P x$ with a symmetric, positive definite $n \times n$ matrix P is called a **quadratic Lyapunov function (QLF)** candidate.
For functions of this type, all conditions of Definition 4.6 are automatically satisfied, so they do not need to be checked anymore.

For this type of Lyapunov function candidate in connection with LTI systems, algebraic criteria for (4.10) and (4.11) can be easily obtained by simply inserting the system Equations (LCT) and (LDT) into Equations (4.10) and (4.11), respectively. These algebraic criteria, known as the Lyapunov and Stein equations (4.12) and (4.13), respectively, and a well known spectral condition for the stability of LTI systems, are given in the following two Theorems.

Theorem 4.2 (Stability of CT LTI systems)**.** *Let A be the system matrix of a CT LTI system. Then the following statements are equivalent:*

1. *The origin is the unique asymptotically stable equilibrium.*
2. *There exist $P > 0$ and $Q > 0$ which satisfy the so called* ***Lyapunov equation***
$$A^T P + PA = -Q \tag{4.12}$$
and $V(x) = x^T P x$ is a QLF.

3. *The matrix A is **Hurwitz**, that is for any eigenvalue λ_i of A:* $\mathrm{Re}(\lambda_i) < 0$.

Theorem 4.3 (Stability of DT LTI systems)**.** *Let A be the system matrix of a DT LTI system. Then the following statements are equivalent:*

1. *The origin is the unique asymptotically stable equilibrium.*
2. *There exist $P > 0$ and $Q > 0$ which satisfy the so called **Stein equation***
$$A^T P A - P = -Q \tag{4.13}$$
and $V(x) = x^T P x$ is a QLF.
3. *The matrix A is **Schur**, that is for any eigenvalue λ_i of A: $|\lambda_i| < 1$.*

4.2.2 Switched linear systems

A switched linear system consists of a finite set $\mathcal{A} = \{A_1, \ldots, A_N\} \subset \mathbb{R}^{n \times n}$ of system matrices and some switching mechanism, that triggers the transition from one system to another. The LTI systems defined by the matrices A_i are called the subsystems of the switched system. All subsystems share the same state vector x, and only the subsystem that is active at a time t is governing the behaviour of the switched system at that time.

Before we give a formal definition for a switched linear system, we list some situations, where it naturally occurs.

Switching process If the process itself is switching between different working modes, then a switched system is the most intuitive way to model it. Examples for this kind of processes are the dynamics of a car, which change when the gear is changed, or a chemical process with a dedicated start-up and shut-down phase.

Switching controller In some situations, the controller itself consists of subcontrollers and is switching between them. One such situation could be, if the control objective changes, for instance from robustness to good disturbance rejection because an external condition changes.

Switching process and controller A symbiosis of a switching system model and a switching controller is the so-called multiple-model adaptive control [NB97], where different models of the plant are used to

represent different working modes and each model is assigned a subcontroller. The parameters of the active subcontroller are tuned, as long as changes remain small, if the parameters change rapidly, instead of tuning the parameters, the controller switches to another subcontroller.

Control-based core link provisioning We will see, that the switched system stability problem is equivalent to the so-called Lur'e absolute stability problem, which provides a framework for our loss rate control.

Before we can finally define a switched linear system, we first have to define the switching mechanism. This will be a function, which simply chooses one system to be active at any time:

Definition 4.10 (Switching signal)**.** Let $\mathcal{J} = \{1, \ldots, N\} \subset \mathbb{N}$ denote an index set. Let σ be a piecewise constant function $\sigma : \mathbb{R}_+ \to \mathcal{J}$ and denote the points of discontinuities with $0 < t_0 < \ldots < t_k < \ldots$. Then, σ is a **switching signal**, if

1. $\exists\, \varepsilon > 0$: $t_{k+1} - t_k > \varepsilon \;\forall\, k \in \mathbb{N}$, which implies that a subsystem remains active for a finite time interval.
2. σ is continuous from the right, i.e. $\lim\limits_{t \to t_k, t > t_k} \sigma(t) = \sigma(t_k) \;\forall\, k \in \mathbb{N}$.

for a discrete time switching signal, we also have to demand that the switching only occurs on multiples of the sampling time T_s, i.e. that σ maps from the natural numbers $\mathbb{N}$ to $\mathcal{J}$, or

3. $\forall\, k \in \mathbb{N}\; \exists\, \nu(k) \in \mathbb{N}$: $t_k = \nu(k) T_s$.

Now we are able to describe a switched linear system by a set of system matrices and a set of admissible switching signals:

Definition 4.11 (Switched linear system)**.** Let $\mathcal{J} = \{1, \ldots, N\}$ be an index set, $\mathcal{A} = \{A_i \mid i \in \mathcal{J}\} \subset \mathbb{R}^{n \times n}$ be a finite set of system matrices and $\mathcal{S}$ be the set of admissible switching signals $\sigma : \mathbb{R}_+ \to \mathcal{J}$. Define further $A : \mathbb{R}_+ \to \mathcal{A}$, $A(t) = A_{\sigma(t)}$. Then we can define the **switched linear system** $\Sigma_{\mathcal{A},\mathcal{S}}$ as

$$\Sigma_{\mathcal{A},\mathcal{S}} : \quad \dot{x} = A(t)x(t) \qquad x(0) = x_0 \quad \forall\, \sigma \in \mathcal{S} \tag{CTSS}$$

for continuous time and

$$\Sigma_{\mathcal{A},\mathcal{S}} : \quad x(k+1) = A(k)x(k) \qquad x(0) = x_0 \quad \forall\, \sigma \in \mathcal{S} \tag{DTSS}$$

for discrete time. Solutions to this equations now of course depend on the initial state x_0 as well as on the switching signal $\sigma \in \mathcal{S}$. We will denote solutions therefore with $x(t; x_0, \sigma)$ and $x(k; x_0, \sigma)$, respectively.

A switched linear system can be seen as a special case of a time-variant linear system, as the system matrix is dependent on the time, e.g. $\dot{x} = A(t)x$, where $A(t) \in \mathcal{A}$. Alternatively, a switched system can be interpreted as a hybrid system, where the active subsystem constitutes the continuous part and the switching events are the discrete-event part. It is worth noting, too, that the theory of switched systems is closely related to the mathematical theory of differential and difference inclusions. A differential or difference inclusion is, roughly speaking, a differential or difference equation where the right hand side is a set of functions. The related differential inclusion to a switched system would be for instance $\dot{x} \in \{A(t)x \mid A(t) \in \mathcal{A}\}$ and the set of solutions of the linear differential inclusion contains the set of solutions of the according switched system.

Depending on the set of admissible switching signals $\mathcal{S}$, different types of switched systems can be identified. Our interest will be focussed on **arbitrary switching signals**, i.e. the set $\mathcal{S}$ will contain all switching signals satisfying definition 4.10.

Stability notions for switched systems

Now that we have defined, what a switched system is, we can define different notions of stability for them. All of them are basically "uniform" extensions of well known stability concepts for ordinary systems. "Uniform", because they are uniform with respect to all switching signals σ in $\mathcal{S}$.

Recall for the following, that the origin is an equilibrium of (CTSS) and (DTSS) and that we denote solutions with $x(t; x_0, \sigma)$ and $x(k; x_0, \sigma)$, respectively. Then we can define

Definition 4.12 (Uniform stability)**.** The origin is called an **uniformly stable equilibrium** of the switched system (CTSS) or (DTSS), if

$$\begin{aligned} &\forall \varepsilon > 0 \quad \exists \delta(\varepsilon): \\ &\qquad \|x_0\| < \delta(\varepsilon) \;\Rightarrow\; \|x(t \text{ or } k; x_0, \sigma)\| < \varepsilon \qquad \forall\, t \text{ or } k \geq 0, \sigma \in \mathcal{S}. \end{aligned} \tag{4.16}$$

Uniform stability does not imply, that the trajectories actually converge to the origin. It only implies, that they will stay in any ε-ball around it if they start close enough to it. Convergence is part of the notion of attractivity:

Definition 4.13 (Uniform attractivity)**.** The origin is called an **uniformly attractive equilibrium** of the switched system (CTSS) or (DTSS), if

$$\exists \delta > 0 : \quad \|x_0\| < \delta \;\Rightarrow\; x(t \text{ or } k; x_0, \sigma) \to 0 \text{ as } t \text{ or } k \to \infty \quad \forall\, \sigma \in \mathcal{S}. \tag{4.17}$$

Combining uniform attractivity and stability yields uniform asymptotic stability:

Definition 4.14 (Uniform asymptotic stability)**.** The origin is called an **uniformly asymptotically stable equilibrium** of the switched system (CTSS) or (DTSS), if and only if it is an uniformly stable and uniformly attractive equilibrium.

An even stronger concept results, if the speed of convergence to the origin is also bounded:

Definition 4.15 (Uniform exponential stability)**.** The origin is called an **uniformly exponentially stable equilibrium** of the CT switched system (CTSS) , if

$$\exists a, b > 0 : \quad \|x(t; x_0, \sigma)\| < a\|x_0\|e^{-bt} \qquad \forall\, t \geq 0, \sigma \in \mathcal{S} \tag{4.18}$$

and of the DT switched system (DTSS), if

$$\exists a > 0, 0 < b < 1 : \quad \|x(k; x_0, \sigma)\| < a\, b^k \|x_0\| \qquad \forall\, k \geq 0, \sigma \in \mathcal{S}. \tag{4.19}$$

The relations between the different notions can be depicted as follows:

Uniform stability & attractivity	$\Leftrightarrow$	Uniform asymptotic stability	$\Leftarrow$	Uniform exponential stability

It is worth noting, that for linear time-varying systems asymptotic stability coincides with *global* asymptotic stability, that is $\delta = \infty$ in definition 4.13. Furthermore, for switched linear systems, the notions of uniform attractivity and of asymptotic and exponential stability are equivalent:

Lemma 4.1 (Stability for switched LTI systems [DM99])**.** *The origin of a switched linear system is uniformly attractive if and only if it is uniformly asymptotically stable if and only if it is uniformly exponentially stable.*

4.2.3 Switched systems and Lyapunov functions

In the last section, we defined several notions of stability for switched systems but did not give any conditions to actually check a given switched system for stability. This will be done now.

Firstly, note that Theorem 4.1 was formulated in terms of the general systems (CT) and (DT), which in particular also allowed time-variant right hand sides. So it still stays valid for the switched system case:

Lemma 4.2 (Stability of switched systems [DM99])**.** *The origin is an uniformly stable equilibrium for the switched system* (CTSS)*, if and only if there exists a Lyapunov function candidate $V(x)$ with*

$$\dot{V}(x(t; x_0, \sigma)) < 0 \qquad \forall\, t \geq 0,\, x_0 \in \mathbb{R}^n \backslash \{0\}, \sigma \in \mathcal{S} \tag{4.20}$$

and for the switched system (DTSS)*, if and only if there exists a Lyapunov function candidate $V(x)$ with*

$$V\left(x(k+1; x_0, \sigma)\right) - V\left(x(k; x_0, \sigma)\right) < 0 \qquad \forall k > 0, x_0 \in \mathbb{R}^n \backslash \{0\}, \sigma \in \mathcal{S}. \tag{4.21}$$

This statement is true for any class of switching signals. Thus also for constant switching signals, so every Lyapunov function for a switched system with arbitrary switching signals certainly also is a Lyapunov function for every subsystem. This leads to the definition of common Lyapunov functions:

Definition 4.16 (CLF)**.** Let $\mathcal{A}$ be a set of system matrices. A Lyapunov function candidate $V(x)$ is a **common Lyapunov function (CLF)** for the family of LTI systems constituted by $\mathcal{A}$, if $V(x)$ is a Lyapunov function for every $A \in \mathcal{A}$.
If $\mathcal{A}$ is the set of system matrices belonging to a switched system (CTSS) or (DTSS), then we call V the CLF for the switched system.

A CLF not only is a Lyapunov function for the the matrices in $\mathcal{A}$, but also for combinations of them. In the case of continuous time systems, this applies to any kind of LF and to linear combinations with nonnegative coefficients:

Lemma 4.3 (CLF and positive linear combinations, CT)**.** *Let $V(x)$ be a CLF for the CT LTI systems constituted by $\mathcal{A} = \{A_i \mid i = 1, \ldots, N\}$. Let $\alpha_i \geq 0$, $i = 1, \ldots, N$ with $\sum_{i=1}^{N} \alpha_i > 0$ be the coefficients of the linear combination*

$$\tilde{A} = \sum_{i=1}^{N} \alpha_i A_i.$$

Then, $V(x)$ is also a Lyapunov function for the CT LTI system constituted by $\tilde{A}$.

Proof. So the system under consideration is $\dot{x} = \tilde{A}x$. Now we simply check condition (4.10) in definition 4.7:

$$\begin{aligned}\frac{\partial V}{\partial x}\dot{x} = \frac{\partial V}{\partial x}\tilde{A}x &= \frac{\partial V}{\partial x}\sum_{i=1}^{N}\alpha_i A_i x \\ &= \sum_{i=1}^{N}\left(\alpha_i \underbrace{\left(\frac{\partial V}{\partial x}A_i x\right)}_{<0}\right) < 0.\end{aligned}$$

This shows, that V is a Lyapunov function for $\tilde{A}$. □

Unfortunately, in the discrete time case, no such general statement could be made so far. But at least we can state a result on quadratic Lyapunov functions, which will be sufficient for our further analysis:

Lemma 4.4 (CQLF and convex combinations, DT)**.** *Given a family of DT LTI systems constituted by $\mathcal{A} = \{A_i \mid i = 1, \dots, N\}$, let $\alpha_i \geq 0$, $i = 1, \dots, N$ with $\sum_{i=1}^{N}\alpha_i = 1$. Then $\tilde{A} = \sum_{i=1}^{N}\alpha_i A_i$ is called a convex combination of the matrices in $\mathcal{A}$.*
Let $V(x) = x^T P x$ be a CQLF for $\mathcal{A}$. Then V is also a QLF for any convex combination $\tilde{A}$ of the matrices in $\mathcal{A}$.

Proof. We will show the following: If $P > 0$ satisfies the Stein inequality (4.13) for every $A_i \in \mathcal{A}$, then it also satisfies the Stein inequality for the convex combination $\tilde{A}$ and then, by Theorem 4.3, $V(x) = x^T P x$ is also LF for $\tilde{A}$.
Let $\mathcal{J} = \{1, \dots, N\}$ be the index set for $\mathcal{A}$, and for simplicity of notation, denote the left hand side of the Stein inequality with S:

$$\begin{aligned} S_{ij} &:= A_i^T P A_j - P && \text{for } i \neq j \\ S_i &:= A_i^T P A_i - P && \text{for } i = j \end{aligned}$$

Note, that $S_{ji} = S_{ij}^T$ and that with this notation, V is a CQLF if and only if $S_i < 0 \ \forall i \in \mathcal{J}$.
Then, the following holds:

$$S_{ij} + S_{ji} = S_i + S_j - (A_i - A_j)^T P (A_i - A_j) < 0.$$

Next, insert $\tilde{A}$ into the Stein inequality:

$$\begin{aligned}
\tilde{A}^T P \tilde{A} - P &= \left(\sum_{i\in\mathcal{J}} \alpha_i A_i^T\right) P \left(\sum_{i\in\mathcal{J}} \alpha_i A_i\right) - \sum_{i\in\mathcal{J}} \alpha_i \sum_{i\in\mathcal{J}} \alpha_i P \\
&= \sum_{i,j\in\mathcal{J}} \alpha_i \alpha_j A_i^T P A_j - \sum_{i,j\in\mathcal{J}} \alpha_i \alpha_j P \\
&= \sum_{i,j\in\mathcal{J}} \alpha_i \alpha_j \left(A_i^T P A_j - P\right) \\
&= \sum_{i\in\mathcal{J}} \alpha_i^2 S_i + \sum_{i,j\in\mathcal{J}, i<j} \alpha_i \alpha_j S_{ij} + \sum_{i,j\in\mathcal{J}, i>j} \alpha_i \alpha_j S_{ij} \\
&= \sum_{i\in\mathcal{J}} \alpha_i^2 S_i + \sum_{i,j\in\mathcal{J}, i<j} \alpha_i \alpha_j S_{ij} + \sum_{i,j\in\mathcal{J}, j>i} \alpha_j \alpha_i S_{ji} \\
&= \sum_{i\in\mathcal{J}} \alpha_i^2 S_i + \sum_{i,j\in\mathcal{J}, i<j} \alpha_i \alpha_j (S_{ij} + S_{ji}) \\
&< 0.
\end{aligned}$$

In the first line, the fact that $\sum_{i\in\mathcal{J}} \alpha_i = 1$ is used. This shows, that P satisfies (4.13) for any convex combination $\tilde{A}$ and thus completes the proof. □

Now consider the following: As a subsystem stays active for at least a finite time interval, at any time t exactly one subsystem is active. If $V(x)$ is a CLF for the switched system, then independent of which subsystem is active, V decreases along the trajectory at any time t. Thus, V satisfies Lemma 4.2. This is detailed in the following Theorem:

Theorem 4.4 (Stability of switched linear systems with arbitrary switching signals [DM99])**.** *The origin is the globally asymptotically stable equilibrium of a switched system $\Sigma_{\mathcal{A},\mathcal{S}}$ like in definition 4.11, where $\mathcal{S}$ consists of arbitrary switching signals.*

$\Longleftrightarrow$

There exists a CLF V for $\Sigma_{\mathcal{A},\mathcal{S}}$.

There are many results on the actual form of the CLF. So it could be shown, that if any CLF exists, then e.g. a common piecewise linear LF and a common piecewise quadratic LF exist. Unfortunately, this does not include common quadratic Lyapunov functions, so unlike in the case of one LTI system treated in Theorems (4.2) and (4.3), in the case of a switched linear system, the existence of a common quadratic Lyapunov function (CQLF) is only sufficient and not necessary for stability.

Theorem 4.5 (Sufficient condition for stability of a CT switched linear system)**.** *Let $\mathcal{A} = \{A_1, \ldots, A_N\} \subset \mathbb{R}^{n\times n}$ and $\Sigma_{\mathcal{A},\mathcal{S}}$ be a continuous time switched linear system under arbitrary switching. Then the origin is globally asymptotically stable, if:*
A CQLF exists: $\exists P \in \mathbb{R}^{n\times n}, P > 0$: $V(x) = x^T P x$ is a CLF for $\Sigma_{\mathcal{A},\mathcal{S}}$ or, which is the same, a common solution P for the Lyapunov equation (4.12) exists:

$$\exists P, Q_i, \ldots, Q_N \in \mathbb{R}^{n\times n}, P, Q_i, \ldots, Q_N > 0 : \\ A_i^T P + P A_i = -Q_i \;\forall\, A_i \in \mathcal{A},\; i = 1, \ldots, N \quad (4.22)$$

Theorem 4.6 (Sufficient condition for stability of a DT switched linear system)**.** *Let $\mathcal{A} = \{A_1, \ldots, A_N\} \subset \mathbb{R}^{n\times n}$ and $\Sigma_{\mathcal{A},\mathcal{S}}$ be a discrete time switched linear system under arbitrary switching. Then the origin is globally asymptotically stable, if:*
A CQLF exists: $\exists P \in \mathbb{R}^{n\times n}, P > 0$: $V(x) = x^T P x$ is a CLF for $\Sigma_{\mathcal{A},\mathcal{S}}$ or, which is the same, a common solution P for the Stein equation (4.13) exists:

$$\exists P, Q_i, \ldots, Q_N \in \mathbb{R}^{n\times n}, P, Q_i, \ldots, Q_N > 0 : \\ A_i^T P A_i - P = -Q_i \;\forall\, A_i \in \mathcal{A},\; i = 1, \ldots, N \quad (4.23)$$

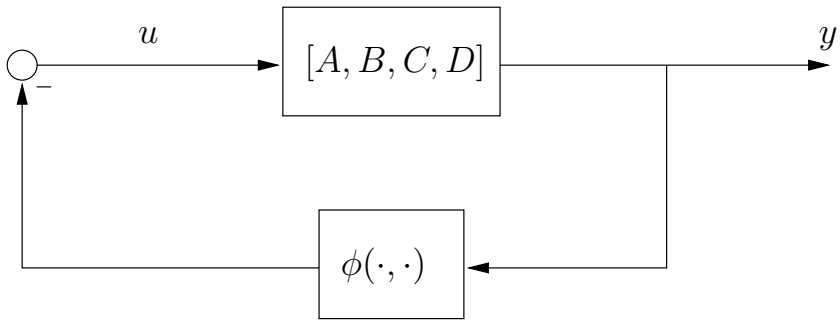


Figure 4.1: Block diagram of the Lur'e system

4.3 The Lur'e absolute stability problem

A Lur'e system is a nonlinear system, which consists of a linear dynamical system and a memoryless nonlinearity in the the feedback path. The block structure is shown in Figure 4.3, and the constituting equations are:

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad \text{(CT Lure i)}$$
$$y(t) = Cx(t) + Du(t) \qquad \text{(CT Lure ii)}$$
$$u(t) = -\phi(t, y(t)) \qquad \text{(CT Lure iii)}$$

for the continuous time case and

$$x(k+1) = Ax(k) + Bu(k) \qquad \text{(DT Lure i)}$$
$$y(k) = Cx(k) + Du(k) \qquad \text{(DT Lure ii)}$$
$$u(k) = -\phi(k, y(k)) \qquad \text{(DT Lure iii)}$$

for the discrete time case.

In every case, we are focussing on the single-input single-output (SiSo) Lur'e System, so $u, y, D \in \mathbb{R}$, $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{n\times n}$ and $B, C^T \in \mathbb{R}^{n\times 1}$. n is called the order of the system.
The feedback nonlinearity $\phi(\cdot,\cdot)$ is an arbitrary function $\mathbb{R}_+ \times \mathbb{R} \to \mathbb{R}$ which satisfies the so-called sector condition

$$k_1 y(t)^2 \leq \phi(t, y(t))y(t) \leq k_2 y(t)^2. \tag{4.26}$$

A geometric interpretation is, that the graph of $\phi(t, y)$ at any time entirely lies between the two lines $k_1 y$ and $k_2 y$. We then say, that ϕ **belongs to the sector** $[k_1, k_2]$.

Now we can define the notion of absolute stability:

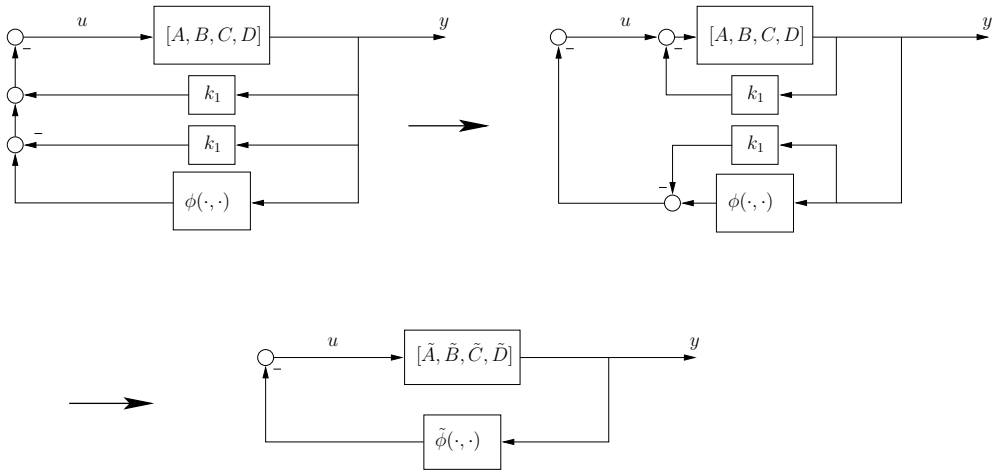


Figure 4.2: Loop transformation

Definition 4.17 (Absolute stability)**.** Given the Lur'e system defined by the linear system (CT Lure i) and (CT Lure ii) or (DT Lure i) and (DT Lure ii) and the two bounds $k_1 < k_2$, the origin is **absolutely stable**, if it is globally asymptotically stable for any feedback nonlinearity ϕ: $\mathbb{R}_+ \times \mathbb{R} \to \mathbb{R}$ belonging to the sector $[k_1, k_2]$.

Remark 4.1 (Loop transformation)**.** A Lur'e system with nonlinearity belonging to the sector $[k_1, k_2]$ can always be transformed to a Lur'e system with nonlinearity belonging to the sector $[0, k]$, where $k = k_2 - k_1 > 0$. Consider the loop transformation in Figure 4.3 with

$$\begin{aligned}
\tilde{A} &= A - Bk_1(1 + Dk_1)^{-1}C \\
\tilde{B} &= B(1 + Dk_1)^{-1} \\
\tilde{C} &= (1 + Dk_1)^{-1}C \\
\tilde{D} &= (1 + Dk_1)^{-1}D \\
\tilde{\phi}(\cdot, y) &= \phi(\cdot, y) - k_1 y.
\end{aligned}$$

Then, the Lur'e system defined by the linear system $[A, B, C, D]$ (CT or DT) and a nonlinearity belonging to $[k_1, k_2]$ is absolutely stable if and only if the Lur'e system defined by the linear systen $[\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}]$ and a nonlinearity belonging to the sector $[0, k_2 - k_1]$ is absolutely stable.

4.3.1 The Lur'e problem and switched systems

In this section we will show the connection between the absolute stability of a Lur'e system in discrete or continuous time and the uniform asymptotic stability of a corresponding switched system. Remark 4.1 allows us to assume a nonlinearity belonging to $[0, k]$ without loss of generality.

Theorem 4.7 (DT: Quadratic stability of switched linear system implies absolute stability)**.** *The DT Lur'e system consisting of the linear system $[A, B, C, D]$ and a nonlinearity belonging to $[0, \gamma]$ is absolutely stable.*
$\Longleftarrow$
The DT switched system $\Sigma_{\mathcal{A},\mathcal{S}}$ as in (DTSS) with

$$\mathcal{A} = \{A, A - B\gamma(1 - D\gamma)^{-1}C\}$$

has a common quadratic Lyapunov function.

Proof. We will show, that the CQLF $V(x)$ also is a Lyapunov function for the Lur'e system with any possible nonlinearity.

Define ψ: $\mathbb{N}_+ \to \mathbb{R}$ with

$$\psi(k) = \begin{cases} 0 & \forall\, k: \quad y(k) = 0 \\ \frac{\phi(k,y(k))}{y(k)} & \text{else.} \end{cases}$$

Then: $\psi(k) \in [0, \gamma]$ and at any time k, the Lur'e system is equivalent to a closed loop system with feedback gain $\psi(k)$:

$$x(k+1) = (A - B\psi(k)(1 - \psi(k)D)^{-1}C)x(k)$$

On the other hand, let $V(x)$ be the CQLF. Then, by Lemma 4.4, V also is a LF for every matrix out of the set

$$M := \{\alpha A + (1 - \alpha)(A - B\gamma(1 - D\gamma)^{-1}C) \,|\, 0 \leq \alpha \leq 1\}$$

that is V is decreasing along solutions of any system $x(k+1) = M_i x(k)$ with $M_i \in M$.

Now choose $\alpha_\psi = 1 - \frac{1+\gamma D}{\gamma}\frac{\psi}{1+\psi D}$. Then, the following holds:

1. $\alpha_\psi \in [0, 1]$ for any $\psi \in [0, \gamma]$.
2. $\alpha_\psi A + (1 - \alpha_\psi)(A - B\gamma(1 - D\gamma)^{-1}C) = A - B\psi(1 - \psi D)^{-1}C$

Thus, V is also decreasing along any solution of the Lur'e system. Thus it is a Lyapunov function and by Theorem 4.1 absolute stability follows. □

Remark 4.2 (Necessary condition for absolute stability)**.** Of course it is necessary for the absolute stability of the Lur'e system, that the corresponding switched linear system is uniformly stable. Otherwise a switching signal σ^* exists, for which the switched system is not stable. Then choose ϕ such that ψ is γ, where subsystem $A - B\gamma(1 - D\gamma)^{-1}C$ is active and 0, where subsystem A is active according to σ^*. For this ϕ, the Lur'e system is not stable, thus it is not absolutely stable.

In the case of a continuous time Lur'e system, even a strict equivalence holds:

Theorem 4.8 (CT: Equivalence of absolute and uniform asymptotic stability [Wul05])**.** *The CT Lur'e system consisting of the linear system* $[A, B, C, D]$ *and a nonlinearity belonging to* $[0, k]$ *is absolutely stable.*
$\Longleftrightarrow$
The CT switched system $\Sigma_{\mathcal{A},\mathcal{S}}$ *like in* (CTSS) *with*

$$\mathcal{A} = \{A, A - Bk(1 - Dk)^{-1}C\}$$

is uniformly asymptotically stable under arbitrary switching.

The proof is analogous to the proof of Theorem 4.7 but unlike in the discrete time case, Lemma 4.3 extends to *any* type of Lyapunov function, therefore strict equivalence holds in continuous time. For the proof, see [Wul05].

4.4 Stability results for continuous time

4.4.1 The circle criterion

Theorem 4.9 (SISO $[0, k]$ circle criterion [NG64, GGS01])**.** *Let* $[A, B, C, D]$ *be a stable linear system and let* $G(s) = C(sI - A)^{-1}B + D$ *be its transfer function. There exists a CQLF for the Lur'e system consisting of* $[A, B, C, D]$ *and any nonlinearity belonging to* $[0, k]$*, which ensures absolute stability, if and only if*

$$\frac{1}{k} + \mathrm{Re}\,(G(j\omega)) > 0 \qquad \forall\, \omega \in \mathbb{R} \tag{4.27}$$

Remark 4.3 ($[0, \infty)$ circle criterion)**.** The circle criterion also holds for $k \to \infty$.

4.4.2 Spectral conditions

Before we start giving conditions in terms of the spectrum of a matrix product, we first define a special type of matrices, which will be important later:

Definition 4.18 (Companion form [HJ85])**.** A matrix in companion form is completely defined by its last row. So we can define:

$$\begin{bmatrix} a_0 & a_1 & \dots & a_{n-1} \end{bmatrix}^\flat = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix}.$$

The characteristic polynomial $p_A(\lambda)$ of a companion matrix is given by

$$p_{\begin{bmatrix} a_0 & a_1 & \dots & a_{n-1} \end{bmatrix}^\flat}(\lambda) = \lambda^n + a_{n-1}\lambda^{n-1} + \ldots + a_1\lambda + a_0$$

Theorem 4.10 (CQLF for rank-1-difference [Mas04, Th.5.3.3])**.** *Let A_1, A_2 be Hurwitz $n \times n$-matrices with* $\operatorname{rank}(A_2 - A_1) = 1$*. Then:*
The LTI systems constituted by A_1 and A_2 have a CQLF
$\Longleftrightarrow$
The matrix product $A_2\,A_1$ has no negative real eigenvalue.

Note, that this Theorem gives a necessary and sufficient condition for the existence of a common *quadratic* Lyapunov function. As we know by Theorem 4.5, the existence of a CQLF is just a *sufficient* condition for absolute stability. Hence, the application of this result to the Lur'e system via Theorem 4.8 only yields a sufficient condition:

Theorem 4.11 (Spectral condition for absolute stability)**.** *The SiSo CT Lur'e system consisting of the linear system $[A, B, C, D]$ and a nonlinearity belonging to $[0, k]$ is absolutely stable*
$\Longleftarrow$
The matrix product $A\left(A - \frac{k}{1+kD}BC\right)$ has no negative real eigenvalue.

Proof. By Theorem 4.8, the Lur'e system is absolutely stable if and only if the switched system $\Sigma_{\mathcal{A},\mathcal{S}}$ with $\mathcal{A} = \{A, A - \frac{k}{1+kD}BC\}$ is uniformly asymptotically stable. By Theorem 4.5, CQLF existence is sufficient for that. Now note that $\operatorname{rank}\left(A - (A - \frac{k}{1+kD}BC)\right) = \operatorname{rank}\left(\frac{k}{1+kD}BC\right) = 1$ because BC is the dyadic product of a column and a row vector. Application of Theorem 4.10 then completes the proof. □

Remark 4.4 ($[0,\infty)$-version for $D \neq 0$)**.** Note, that the proof of Theorem 4.5 stays valid for $k \to \infty$. Note further, that $\frac{k}{1+kD} \to \frac{1}{D}$ as $k \to \infty$. Then, the matrix product under consideration is $A(A - \frac{1}{D}BC)$ and we can state:
The SiSo CT Lur'e system consisting of the linear system $[A, B, C, D]$ with $D \neq 0$ and a nonlinearity which belongs to the sector $[0,\infty)$ is absolutely stable
$\Longleftarrow$
The matrix product $A(A - \frac{1}{D}BC)$ has no negative real eigenvalue.

There are two results from [SK04] and [ZS06] which can be used alongside with the circle criterion 4.9 to derive a criterion for the case $k \to \infty$ and $D = 0$:

Theorem 4.12 (Strict positive realness of a strictly proper transfer function [SK04, Th.3.2])**.** *Let A be Hurwitz and $[A, B, C, 0]$ be a strictly proper CT SiSo LTI system. Let $G(s) = C(sI - A)^{-1}B$. Then,* $\mathrm{Re}(G(j\omega)) > 0 \; \forall \omega \in \mathbb{R}$
$\Longleftrightarrow$

1. *$CAB < 0$;*
2. *$CA^{-1}B < 0$; and*
3. *$A\left(I - \frac{1}{CAB}ABC\right)A$ has no negative real eigenvalue.*

In [ZS06], another criterion was derived, which reduces the order of the problem by 1, so instead of the $n \times n$-matrix of 3. in Theorem 4.12, only a $(n-1) \times (n-1)$-matrix, which is admittedly tricky to construct, needs to be checked.

Theorem 4.13 (Strict positive realness of a strictly proper transfer function [ZS06, Th.1])**.** *Let $[A, B, C, 0]$ be a CT SiSo-system. Further denote its transfer function with $G(s) = C(sI - A)^{-1}B = \frac{\sum_{i=0}^{n-1} b_i s^i}{s^n + \sum_{i=0}^{n-1} a_i s^i}$.*
Construct the matrices $A_1 \in \mathbb{R}^{(n-1)\times(n-1)}$, $D_1 \in \mathbb{R}$ and $C_1, B_1 \in \mathbb{R}^{(n-1)\times 1}$:

$$A_1 = \begin{bmatrix} \frac{b_0}{b_{n-1}} & \cdots & \frac{b_{n-2}}{b_{n-1}} \end{bmatrix}^\flat \tag{4.28a}$$
$$B_1 = \begin{bmatrix} b'_1 & \cdots & b'_{n-1} \end{bmatrix} \tag{4.28b}$$
$$C_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \tag{4.28c}$$
$$D_1 = a_{n-1} - \frac{b_{n-2}}{b_{n-1}} \tag{4.28d}$$

where

$$b_1' = a_{n-2} - \frac{b_{n-3}}{b_{n-1}} - \frac{b_{n-2}}{b_{n-1}} d_1 \tag{4.28e}$$

$$b_2' = a_{n-3} - \frac{b_{n-4}}{b_{n-1}} - \frac{b_{n-2}}{b_{n-1}} b_1' - \frac{b_{n-3}}{b_{n-1}} d_1 \tag{4.28f}$$

$$\vdots$$

$$b_{n-2}' = a_1 - \frac{b_0}{b_{n-1}} - \frac{b_{n-2}}{b_{n-1}} b_{n-3}' \dots - \frac{b_2}{b_{n-1}} b_1' - \frac{b_1}{b_{n-1}} d_1 \tag{4.28g}$$

$$b_{n-1}' = a_0 - \frac{b_{n-2}}{b_{n-1}} b_{n-2}' - \frac{b_{n-3}}{b_{n-1}} b_{n-3}' \dots - \frac{b_1}{b_{n-1}} b_1' - \frac{b_0}{b_{n-1}} d_1. \tag{4.28h}$$

Then: $\mathrm{Re}\,(G(j\omega)) > 0$
$\Longleftrightarrow$

1. $b_{n-1} > 0$;
2. A_1 *is Hurwitz; and*
3. *the matrix product* $(A_1 - B_1 C_1^T / D_1) A_1$ *has no real nonpositive eigenvalue.*

Now the circle criterion 4.9 can be used to link each one of them to the CQLF existence problem for $D = 0$ and $k \to \infty$.

Theorem 4.14 ($[0, \infty)$-version for $D = 0$)**.** *Let A be Hurwitz and $[A, B, C, 0]$ be a LTI system with transfer function $G(s) = C(sI - A)^{-1}B = \frac{\sum_{i=0}^{n-1} b_i s^i}{s^n + \sum_{i=0}^{n-1} a_i s^i}$. Further, define A_1, b_1, c_1 and d_1 like in (4.28). Then, all of the following equivalent conditions are sufficient for absolute stability of the Lur'e system with linear part $[A, B, C, 0]$ and nonlinearity belonging to $[0, \infty)$:*

1. *A and $A - kBC$ have a CQLF for any $k \in \mathbb{R}_+$.*
2. $\mathrm{Re}(G(j\omega)) > 0$ *for all $\omega \in \mathbb{R}_+$.*
3. *1.–3. of Theorem 4.12 hold.*
4. *1.–3. of Theorem 4.13 hold.*

Furthermore it follows from 1. through 4., that there exists a $P \geq 0$ which simultaneously satisfies the non-strict Lyapunov equations

$$A^T P + PA \leq 0$$
$$(-BC)^T P + P(-BC) \leq 0$$

We then call $V(x) = x^T Px$ a "weak CQLF" for A and $-BC$.

Proof. All conditions are equivalent:
1.⇔2. is exactly the circle criterion 4.9 for $k \to \infty$.
2.⇔3. is exactly Theorem 4.12.
2.⇔4. is exactly Theorem 4.13.

For **1.⇒weak CQLF existence** consider the following:
From e.g.[Mas04], it is known that CQLF existence corresponds to the intersection of two convex cones in the space of positive definite matrices. Because of 1., the cones for A and $\frac{1}{k}A - BC$ have a non-empty intersection for any finite k. If $k \to \infty$, $\frac{1}{k}A - BC \to -BC$. As $\text{rank}(BC) = 1$, a CQLF cannot exist anymore for $k = \infty$, but it certainly has existed for any finite k, so it follows from continuity arguments, that the two cones must touch each other, which yields a weak CQLF.

Sufficiency of 1. for absolute stability follows from Theorems 4.8 and 4.5. □

Note that this Theorem also yields a sufficient condition for weak CQLF existence for a Hurwitz matrix A and a rank-1-matrix BC.

4.5 Robust stability results for continuous time

In practical applications, parameters are often not known exactly but can be said to lie in a certain interval. By means of the so called Kharitonov Theorem 4.15, which reduces checking the stability of infinitely many polynomials in an infinite set of polynomials to checking only four polynomials, the results of Section 4.4 can be extended to the case with uncertain matrix entries.

Theorem 4.15 (Kharitonov's Theorem)**.** *Let* $\mathcal{P} = \left\{\sum_{i=1}^{n} p_i s^i \,|\, p_i \in [\underline{p}_i, \overline{p}_i]\ \forall i = 1 \ldots n\right\}$ *be a set of polynomials of order* n *and uncertain coefficients* p_i *in* s*. Then:*
The zeros of any $p(s) \in \mathcal{P}$ *have negative real parts*
$\Longleftrightarrow$
The zeros of the four polynomials

$$\begin{aligned}
k_1(s) &= \underline{p}_0 + \underline{p}_1 s + \overline{p}_2 s^2 + \overline{p}_3 s^3 + \underline{p}_4 s^4 + \ldots \\
k_2(s) &= \underline{p}_0 + \overline{p}_1 s + \overline{p}_2 s^2 + \underline{p}_3 s^3 + \underline{p}_4 s^4 + \ldots \\
k_3(s) &= \overline{p}_0 + \underline{p}_1 s + \underline{p}_2 s^2 + \overline{p}_3 s^3 + \overline{p}_4 s^4 + \ldots \\
k_4(s) &= \overline{p}_0 + \overline{p}_1 s + \underline{p}_2 s^2 + \underline{p}_3 s^3 + \overline{p}_4 s^4 + \ldots
\end{aligned}$$

have all negative real parts.

Corollary 4.1 (Kharitonov's Theorem for companion matrices)**.** *Let* $\mathcal{A} = \left\{\begin{bmatrix} a_0 & a_1 & \ldots & a_{n-1} \end{bmatrix}^\flat \,|\, a_i \in [\underline{a}_i, \overline{a}_i]\ \forall i = 1 \ldots n-1\right\}$ *be a set of companion matrices with interval uncertainties. Then:*
Any $A \in \mathcal{A}$ *is Hurwitz*
$\Longleftrightarrow$
The four matrices

$$\begin{aligned}
A_1 &= \begin{bmatrix} \underline{a}_0 & \underline{a}_1 & \overline{a}_2 & \overline{a}_3 & \underline{a}_4 & \ldots \end{bmatrix}^\flat \\
A_2 &= \begin{bmatrix} \underline{a}_0 & \overline{a}_1 & \overline{a}_2 & \underline{a}_3 & \underline{a}_4 & \ldots \end{bmatrix}^\flat \\
A_3 &= \begin{bmatrix} \overline{a}_0 & \underline{a}_1 & \underline{a}_2 & \overline{a}_3 & \overline{a}_4 & \ldots \end{bmatrix}^\flat \\
A_4 &= \begin{bmatrix} \overline{a}_0 & \overline{a}_1 & \underline{a}_2 & \underline{a}_3 & \overline{a}_4 & \ldots \end{bmatrix}^\flat
\end{aligned}$$

are Hurwitz.

Proof. $\begin{bmatrix} a_0 & a_1 & \ldots & a_{n-1} \end{bmatrix}^\flat$ is Hurwitz $\Leftrightarrow p(A) = \lambda^n + a_{n-1}\lambda^{n-1} + \ldots + a_1\lambda + a_0$ has all zeros in the open left complex halfplane $\Leftrightarrow$ k_1 through k_4 like in Theorem 4.15 have their zeros there $\Leftrightarrow$ A_1 through A_4 are Hurwitz. □

In [ZMSS05], an extension of that theorem to rational functions with uncertain parameters in [CDB91] is utilized to derive robust versions of Theorem 4.10 for companion matrices (which always differ by rank-1) and for general rank-1-difference. As the result restricted to matrices A and $A - \frac{k}{1+kD}BC$ in companion form is much simpler than that for general rank-1-difference, and shall turn out sufficient for our analysis of the Lur'e problem, we restrict our attention to that.

Theorem 4.16 (Theorem 4.10 for companion matrices with uncertainties [ZMSS05])**.** *Let*

$$\mathcal{A} = \left\{ \begin{bmatrix} a_0 & a_1 & \dots & a_{n-1} \end{bmatrix}^\flat \mid a_i \in [\underline{a}_i, \overline{a}_i] \; \forall i = 0 \dots n-1 \right\}$$

and $\mathcal{B}$ in a similar manner denote two sets of companion form matrices with uncertain entries.
Define further

$$\begin{aligned} A_1 &= \begin{bmatrix} \underline{a}_0 & \underline{a}_1 & \overline{a}_2 & \overline{a}_3 & \underline{a}_4 & \dots \end{bmatrix}^\flat \\ A_2 &= \begin{bmatrix} \underline{a}_0 & \overline{a}_1 & \overline{a}_2 & \underline{a}_3 & \underline{a}_4 & \dots \end{bmatrix}^\flat \\ A_3 &= \begin{bmatrix} \overline{a}_0 & \underline{a}_1 & \underline{a}_2 & \overline{a}_3 & \overline{a}_4 & \dots \end{bmatrix}^\flat \\ A_4 &= \begin{bmatrix} \overline{a}_0 & \overline{a}_1 & \underline{a}_2 & \underline{a}_3 & \overline{a}_4 & \dots \end{bmatrix}^\flat \end{aligned}$$

and $B_{1,\dots,4}$ accordingly. Then, any two matrices $A \in \mathcal{A}$ and $B \in \mathcal{B}$ have a CQLF
$\Longleftrightarrow$

1. *Every $A \in \mathcal{A}$ and $B \in \mathcal{B}$ are Hurwitz and,*

2. *None of the eight matrix products*

$$\begin{aligned} &A_1B_2,\ A_1B_3,\ A_2B_1,\ A_2B_4, \\ &A_3B_1,\ A_3B_4,\ A_4B_2,\ A_4B_3 \end{aligned}$$

   has a negative real eigenvalue.

The application of this Theorem to the Lur'e problem is straightforward, if the linear part $[A, B, C, D]$ is in a form, that A and $A - \frac{k}{1+kD}BC$ are both in companion form. Of course, this may not be assumed in general. However, any Input-Output behaviour $G(s)$ can be represented by a state space realization which leaves the both aforementioned matrices in companion form.

Definition 4.19 (Control Canonical Form)**.** Let $G(s) = \frac{b_n s^n + b_{n-1} s^{n-1} + \ldots + b_0}{s^n + a_{n-1} s^{n-1} + \ldots + a_0}$ be a transfer function model for the Input-Output-behaviour of a system. The quadruple $[A, B, C, D]$ is a **realization** of the I/O-behaviour of $G(s)$, if $D + C(sI - A)^{-1}B = G(s)$.
The realization $[A_c, B_c, C_c, D_c]$ with

$$\begin{aligned} A_c &= \begin{bmatrix} a_0 & a_1 & \ldots & a_{n-1} \end{bmatrix}^\flat \\ B_c &= \begin{bmatrix} 0 & \ldots & 0 & 1 \end{bmatrix}^T \\ C_c &= \begin{bmatrix} b_0 - a_0 b_n & b_1 - a_1 b_n & \ldots & b_{n-1} - a_{n-1} b_n \end{bmatrix} \\ D_c &= b_n \end{aligned}$$

is called the **Control Canonical Form**.

Lemma 4.5 (Control Canonical Form always exists [Kai80])**.** *For any realization $[A, B, C, D]$ there exists a Control Canonical Form $[A_c, B_c, C_c, D]$ that realizes the same I/O behaviour as $[A, B, C, D]$.*

Now we can formulate a robust stability condition for a Lur'e system in terms of the I/O behaviour.

Theorem 4.17 (Robust stability of a Lur'e system)**.** *Let*

$$\mathcal{G} = \left\{ \frac{\sum_{i=0}^{n-1} b_i s^i}{s^n + \sum_{i=0}^{n-1} a_i s^i} \mid a_i \in [\underline{a}_i, \overline{a}_i],\ b_i \in [\underline{b}_i, \overline{b}_i]\ \forall i = 0 \ldots n-1 \right\}$$

be a set of nth-order, strictly proper transfer functions with uncertain coefficients.
Define further

$$\begin{aligned} A_1 &= \begin{bmatrix} \underline{a}_0 & \underline{a}_1 & \overline{a}_2 & \overline{a}_3 & \underline{a}_4 & \ldots \end{bmatrix}^\flat \\ A_2 &= \begin{bmatrix} \underline{a}_0 & \overline{a}_1 & \overline{a}_2 & \underline{a}_3 & \underline{a}_4 & \ldots \end{bmatrix}^\flat \\ A_3 &= \begin{bmatrix} \overline{a}_0 & \underline{a}_1 & \underline{a}_2 & \overline{a}_3 & \overline{a}_4 & \ldots \end{bmatrix}^\flat \\ A_4 &= \begin{bmatrix} \overline{a}_0 & \overline{a}_1 & \underline{a}_2 & \underline{a}_3 & \overline{a}_4 & \ldots \end{bmatrix}^\flat \end{aligned}$$

and

$$\begin{aligned} K_1 &= \begin{bmatrix} \underline{a}_0 - k\underline{b}_0 & \underline{a}_1 - k\underline{b}_1 & \overline{a}_2 - k\overline{b}_2 & \overline{a}_3 - k\overline{b}_3 & \underline{a}_4 - k\underline{b}_4 & \ldots \end{bmatrix}^\flat \\ K_2 &= \begin{bmatrix} \underline{a}_0 - k\underline{b}_0 & \overline{a}_1 - k\overline{b}_1 & \overline{a}_2 - k\overline{b}_2 & \underline{a}_3 - k\underline{b}_3 & \underline{a}_4 - k\underline{b}_4 & \ldots \end{bmatrix}^\flat \\ K_3 &= \begin{bmatrix} \overline{a}_0 - k\overline{b}_0 & \underline{a}_1 - k\underline{b}_1 & \underline{a}_2 - k\underline{b}_2 & \overline{a}_3 - k\overline{b}_3 & \overline{a}_4 - k\overline{b}_4 & \ldots \end{bmatrix}^\flat \\ K_4 &= \begin{bmatrix} \overline{a}_0 - k\overline{b}_0 & \overline{a}_1 - k\overline{b}_1 & \underline{a}_2 - k\underline{b}_2 & \underline{a}_3 - k\underline{b}_3 & \overline{a}_4 - k\overline{b}_4 & \ldots \end{bmatrix}^\flat \end{aligned}$$

Then, the Lur'e system with linear part $G(s) \in \mathcal{G}$ and a feedback nonlinearity belonging to $[0, k]$ is absolutely stable
$\Longleftarrow$

1. *All K_i and A_i, $i = 0, \ldots, 4$, are Hurwitz.*
2. *None of the eight matrix products*

$$A_1K_2,\ A_1K_3,\ A_2K_1,\ A_2K_4,$$
$$A_3K_1,\ A_3K_4,\ A_4K_2,\ A_4K_3$$

has a negative real eigenvalue.

Proof. Any $G(s) \in \mathcal{G}$ can be realized by $[A, B, C, 0]$ with

$$\begin{aligned} A &= \begin{bmatrix} a_0 & a_1 & \ldots & a_{n-1} \end{bmatrix}^\flat \\ B &= \begin{bmatrix} 0 & 0 & \ldots & 0 & 1 \end{bmatrix}^T \\ C &= \begin{bmatrix} b_0 & b_1 & \ldots & b_{n-1} \end{bmatrix} \end{aligned}$$

From the definition of $\mathcal{G}$, one can directly derive the set $\mathcal{A}$ of A-matrices and the set $\mathcal{C}$ of C-vectors:

$$\begin{aligned} \mathcal{A} &= \left\{ \begin{bmatrix} a_0 & a_1 & \ldots & a_{n-1} \end{bmatrix}^\flat \mid a_i \in [\underline{a}_i, \overline{a}_i]\ \forall 0 = 1 \ldots n-1 \right\} \\ \mathcal{C} &= \left\{ \begin{bmatrix} b_0 & b_1 & \ldots & b_{n-1} \end{bmatrix} \mid b_i \in [\underline{b}_i, \overline{b}_i]\ \forall 0 = 1 \ldots n-1 \right\} \end{aligned}$$

The problem thus can be restated as: Is the Lur'e problem stable for any $[A, B, C, 0]$ with $A \in \mathcal{A}$ and $C \in \mathcal{C}$?
By Theorem 4.8 this is equivalent to asking, whether the switched system $\dot{x} = F(t)x(t)$ with $F(t) \in \{A, A - kBC\}$ is uniformly asymptotically stable for any $A \in \mathcal{A}$ and $C \in \mathcal{C}$.
A sufficient condition for that is by Theorem 4.5, that A and $A - kBC$ share a CQLF for any $A \in \mathcal{A}$ and $C \in \mathcal{C}$.
Now denote the set of all $A - kBC$ by $\mathcal{K}$. Then direct calculation yields:

$$\mathcal{K} = \left\{ \begin{bmatrix} \alpha_0 & \alpha_1 & \ldots & \alpha_{n-1} \end{bmatrix}^\flat \mid \alpha_i \in [\underline{a}_i - k\underline{b}_i, \overline{a}_i - k\overline{b}_i]\ \forall i = 0 \ldots n-1 \right\}$$

And the question of CQLF existence can be treated by Theorem 4.16. The matrix product conditions directly translate to each other, condition 1 is a direct translation of condition 1 of Theorem 4.16 via Corollary 4.1. □

4.6 Discrete time versions

In this section we will use the so-called Bilinear Transform to link continuous-time with discrete-time and extend results to DT by translating conditions on the DT systems to conditions on corresponding CT systems.

4.6.1 Linking discrete-time and continuous time results: The Bilinear Transform

There is a close relationship between discrete and continuous time CQLF existence problems. Every discrete time LTI system can be assigned a continuous time equivalent (and vice versa) and conditions on the one can be directly translated to conditions on the other.

Definition 4.20 (The Bilinear Transform)**.** Let $\mathbb{R}_H^{n\times n}$ and $\mathbb{R}_S^{n\times n}$ be the set of Hurwitz and Schur $n \times n$-matrices and I denote the identity matrix in appropriate dimension.
The mapping

$$\mathrm{C}:\ \mathbb{R}_S^{n\times n} \to \mathbb{R}^{n\times n}\ \ \mathrm{C}(A) = (A-I)(A+I)^{-1}$$

then is called the **Bilinear Transform**, the mapping

$$\mathrm{C}^{-1}:\ \mathbb{R}_H^{n\times n} \to \mathbb{R}^{n\times n}\ \ \mathrm{C}^{-1}(A) = (I+A)(I-A)^{-1}$$

the **inverse Bilinear Transform**.

Remark 4.5 (Bilinear transform is a bijective mapping from $\mathbb{R}_S^{n\times n}$ to $\mathbb{R}_H^{n\times n}$)**.** Note, that C maps the set of Schur stable matrices to the set of Hurwitz stable matrices and C^{-1} vice versa. In other words, if the eigenvalues of A lie in the unit circle, then the eigenvalues of $\mathrm{C}(A)$ lie in the open left half plane. To see that, consider the characteristic polynomial of $\mathrm{C}(A)$:

$$\begin{aligned}
& \det(\lambda I - (A-I)(A+I)^{-1}) = 0 \\
\Leftrightarrow\quad & \det(\lambda(A+I) - (A-I)) = \det((\lambda-1)A + (\lambda+1)I) = 0 \\
\Leftrightarrow\quad & \det\left(\frac{1+\lambda}{1-\lambda}I - A\right) = 0
\end{aligned}$$

Because A is Schur, any eigenvalue has magnitude smaller than one, so for

any eigenvalue λ of $\mathrm{C}(A)$ it holds:

$$\begin{aligned}
\left|\frac{1+\lambda}{1-\lambda}\right| < 1 \;&\Leftrightarrow\; \left|\frac{1+\lambda}{1-\lambda}\right|^2 < 1 \\
&\Leftrightarrow\; \frac{(1+\lambda)\overline{(1+\lambda)}}{(1-\lambda)\overline{(1-\lambda)}} = \frac{(1+\mathrm{Re}(\lambda))^2+\mathrm{Im}(\lambda)^2}{(1-\mathrm{Re}(\lambda))^2+\mathrm{Im}(\lambda)^2} < 1 \\
&\Leftrightarrow\; \mathrm{Re}(\lambda) < 0
\end{aligned}$$

In other words, the set of eigenvalues of $\mathrm{C}(A)$ is given by $\left\{\lambda \in \mathbb{C} \mid \left|\frac{1+\lambda}{1-\lambda}\right| < 1\right\} = \{\lambda \in \mathbb{C} \mid \mathrm{Re}(\lambda) < 0\}$. A similar calculation yields the corresponding result for C^{-1}.
So the Bilinear Transform exactly maps the stability regions into each other and is invertible, so it is a bijective mapping.

Theorem 4.18 (QLF is preserved under Bilinear Transform [MNMK06])**.** *$V = x^T P x$ is a QLF for the DT system $x(k+1) = Ax(k)$*
$\Longleftrightarrow$
$V = x^T P x$ is a QLF for the associated CT system $\dot{x} = \mathrm{C}(A)x$.

Lemma 4.6 (Multiplication with nonsingular matrices does not change rank, e.g. [HJ85])**.** *Let $A \in \mathbb{R}^{n\times n}$ with $\mathrm{rank}(A) = n$, $B \in \mathbb{R}^{m\times m}$ with $\mathrm{rank}(B) = m$ and $X \in \mathbb{R}^{n\times m}$ with $\mathrm{rank}(X) = r$. Then:*

$$\mathrm{rank}(AXB) = \mathrm{rank}(X) = r.$$

Theorem 4.19 (Bilinear Transform preserves rank of differences)**.** *Let $A, B \in \mathbb{R}_H^{n\times n}$ with $D = A - B$ and $\Delta = \mathrm{C}^{-1}(A) - \mathrm{C}^{-1}(B)$*
Then $\mathrm{rank}(D) = p \Longleftrightarrow \mathrm{rank}(\Delta) = p$.
In other words: If two matrices have a rank-p-difference, their Bilinear Transforms also have a rank-p-difference.

Proof.

$$\begin{aligned}
\Delta &= \mathrm{C}^{-1}(A) - \mathrm{C}^{-1}(B) \\
&= (I+A)(I-A)^{-1} - (I+B)(I-B)^{-1} \\
&= (I+A)(I-A)^{-1} - (I+A-D)(I-A+D)^{-1} \quad \mid \cdot (I-A+D) \\
\Delta(I-A+D) &= (I+A)(I-A)^{-1}(I-A+D) - (I+A) + D \\
&= (I+A) + (I+A)(I-A)^{-1}D - (I+A) + D \\
&= \left[I + (I+A)(I-A)^{-1}\right] D \qquad \mid \cdot (I-A+D)^{-1} \\
\Delta &= \left[I + (I+A)(I-A)^{-1}\right] D(I-A+D)^{-1} \\
&= \left[\mathrm{C}^{-1}(A) + I\right] D(I-B)^{-1}
\end{aligned}$$

Now observe, that $\det\left(\lambda I - \left[\mathrm{C}^{-1}(A) + I\right]\right) = \det\left((\lambda - 1)I - \mathrm{C}^{-1}(A)\right)$, so if $\left[\mathrm{C}^{-1}(A) + I\right]$ has an eigenvalue 0, then $\mathrm{C}^{-1}(A)$ has an eigenvalue -1, which is not possible, because $\mathrm{C}^{-1}(A)$ is Schur. So: $\operatorname{rank}\left(\left[\mathrm{C}^{-1}(A) + I\right]\right) = n$. Of course, $\operatorname{rank}\left((I - B)^{-1}\right) = n$, too, so by Lemma 4.6: $\operatorname{rank}(\Delta) = \operatorname{rank}(D)$. □

The two Theorems from this Section now enable us to translate every CQLF existence problem in discrete time to a equivalent CQLF existence and thereby make use of the rank of the differences.

4.6.2 Results for discrete time

We are now going to demonstrate, how the results for continuous time can be translated to results for discrete time. First of all, we will give Theorem 4.10 in discrete time and then derive a sufficient condition for the Lur'e system in discrete time.

Theorem 4.20 (CQLF rank-1-difference)**.** *Let A_1, A_2 be Schur $n \times n$-matrices with* $\operatorname{rank}(A_2 - A_1) = 1$. *Then:*
The LTI systems constituted by A_1 and A_2 have a CQLF
$\Longleftrightarrow$
The matrix product $\mathrm{C}(A_2)\ \mathrm{C}(A_1)$ *has no negative real eigenvalue.*

Proof. From Theorem 4.18 it follows, that A_1 and A_2 have a CQLF $\Leftrightarrow$ $\mathrm{C}(A_1)$ and $\mathrm{C}(A_2)$ have a CQLF. With Theorem 4.19 it follows, that $\operatorname{rank}\left(\mathrm{C}(A_1) - \mathrm{C}(A_2)\right) = 1$, so Theorem 4.10 can be applied which completes the proof. □

Now we are applying Theorem 4.7 to translate Theorem 4.20 in a sufficient condition for absolute stability of a discrete time Lur'e system:

Theorem 4.21 (Spectral condition for absolute stability)**.** *The SiSo DT Lur'e system consisting of the linear system $[A, B, C, D]$ and a nonlinearity belonging to $[0, k]$ is absolutely stable*
$\Longleftarrow$
The matrix product $\mathrm{C}(A)\ \mathrm{C}\left(A - \frac{k}{1+kD}BC\right)$ *has no negative real eigenvalue.*

Proof. By Theorem 4.7, a sufficient condition for absolute stability is, that A and $A - \frac{k}{1+kD}BC$ have a CQLF. Theorem 4.20 completes the proof. □

Remark 4.6 ($[0, \infty)$-version for $D \neq 0$)**.** This again holds for $k \to \infty$ if $D \neq 0$.

Remark 4.7 ($[0, \infty)$-version for $D = 0$)**.** No strictly proper discrete time system (i.e. $D = 0$) can be stable with any gain, so a condition like that **cannot exist**. To see that, consider the root-locus of a strictly proper transfer function: At every pole a branch originates, and in every zero one branch terminates. As the system is strictly proper, at least one branch does not have a zero to terminate in, so at least one branch goes to infinity and thus leaves the unit circle.

4.6.3 Robust result for discrete time

Theorem 4.16 can also be extended to discrete time, but there is much more work involved and the result is not straightforward to apply. Thus the results of this Section are clearly to be called preliminary.

We will nevertheless show, that the question, whether a DT CQLF exists for any two matrices out of two families of Schur companion matrices with interval uncertainties can be translated into the question, whether a CT CQLF exists for two families of Hurwitz companion matrices, which in turn can be answered with Theorem 4.16.

To do that, we need the following Lemma:

Lemma 4.7 (Bilinear transforms of companion matrices are simultaneously similar to companion matrices)**.** *Let* $A = \begin{bmatrix} a_0 & a_1 & \dots & a_{n-1} \end{bmatrix}^\flat$ *be a Schur stable matrix. Then there exists a nonsingular matrix* $T \in \mathbb{R}^{n \times n}$ *that only depends on the dimension* n *of* A*, such that* $T\,\mathrm{C}(A)T^{-1}$ *also is in companion form.*

One such matrix is given by

$$T = \begin{bmatrix} m^T \\ m^T R \\ \vdots \\ m^T R^{n-1} \end{bmatrix}, \tag{4.29}$$

where

$$m^T = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} \beta & R\beta & \cdots & R^{n-1}\beta \end{bmatrix}^{-1}, \tag{4.30}$$

$$R = \begin{bmatrix} -1 & 2 & -2 & 2 & \cdots & 2(-1)^n \\ 0 & -1 & 2 & -2 & \cdots & 2(-1)^{n+1} \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & \cdots & & & & -1 \end{bmatrix}, \tag{4.31}$$

$$\left(r_{ij} = \begin{cases} 0 & \text{if } i > j \\ -1 & \text{if } i = j \\ 2(-1)^{i+j+1} & \text{if } i < j \end{cases} \right)$$

$$\beta = 2 \begin{bmatrix} (-1)^n & (-1)^{n+1} & \cdots & 1 & -1 \end{bmatrix}^T. \tag{4.32}$$

We will derive this matrix in the course of the proof, which is a bit more involved and makes use of the follwing two Lemmas, which are well-known in linear algebra and control theory, respectively:

Lemma 4.8 (Sherman-Morrison-Woodbury formula)**.** *Let $A \in \mathbb{R}^{n\times n}$ be invertible and $b^T, c^T \in \mathbb{R}^{m\times n}$. If $\lambda := -I_n - c^T A^{-1} b$ is invertible:*

$$(A + bc^T)^{-1} = A^{-1} + A^{-1} b \lambda^{-1} c^T A^{-1} \tag{4.33}$$

Lemma 4.9 (Controllability)**.** *Let $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times 1}$ be two matrices. The pair (A, B) is **controllable** if and only if one of the following conditions is satisfied:*

1. *Kalman-Criterion:* $\operatorname{rank} \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix} = n$
2. *Popov-Belevitch-Hautus-Criterion:* $\operatorname{rank} \begin{bmatrix} \lambda I - A & B \end{bmatrix} = n \ \forall \lambda \in \mathbb{C}$. *Note that it is sufficient to check the rank for all eigenvalues of A because otherwise the first block $\lambda I - A$ already has full rank.*

If the pair is controllable there exists a nonsingular matrix $T \in \mathbb{R}^{n\times n}$, for which TAT^{-1} is in companion form and $TB = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T$.

Proof of Lemma 4.7. Let $c^T = \begin{bmatrix} -a_0 & -a_1 & \cdots & -a_{n-1} \end{bmatrix}$, $b = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T$ and $A_0 = \begin{bmatrix} 0 & 0 & \cdots & 0 \end{bmatrix}^\flat$. Then $A = A_0 + bc^T$. First we will show, that $\mathrm{C}(A) = \mathrm{C}(A_0) + \beta\gamma^T(A)$, where β is a constant column vector and $\gamma^T(A)$ is a row vector which depends on $a_0, \ldots, a_{n-1}$. If the pair $(\mathrm{C}(A_0), \beta)$ is controllable, let T be the matrix from Lemma 4.9. Then

$$T\left(\mathrm{C}(A_0) + \beta\gamma^T(A)\right)T^{-1} = \underbrace{T\,\mathrm{C}(A_0)T^{-1}}_{\text{companion matrix}} + \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T \underbrace{(\gamma^T(A)T^{-1})}_{\text{row vector}}$$

which again is a companion matrix, because the dyadic product results in a matrix with non-zero entries only in the last row. So at this point, we only will need to show that $(\mathrm{C}(A_0), \beta)$ is controllable, which is a trivial task.

Let $A_1 = A_0 - I$, $A_2 = A_0 + I$. Then:

$$\begin{aligned}
\mathrm{C}(A) &= \mathrm{C}(A_0 + bc^T) = (A_1 + bc^T)(A_2 + bc^T)^{-1} \\
&\overset{(4.33)}{=} (A_1 + bc^T)(A_2^{-1} + \frac{1}{\lambda} A_2^{-1} bc^T A_2^{-1}) \\
&= A_1 A_2^{-1} + \frac{1}{\lambda}\left(\lambda bc^T A_2^{-1} + A_1 A_2^{-1} bc^T A_2^{-1} + b \underbrace{c^T A_2^{-1} b}_{\text{scalar!}} c^T A_2^{-1}\right) \\
&= A_1 A_2^{-1} + \frac{1}{\lambda}\left(\lambda I + A_1 A_2^{-1} + c^T A_2^{-1} b I\right) bc^T A_2^{-1} \\
&= A_1 A_2^{-1} + \frac{1}{\lambda}\left(I(-1 - c^T A_2^{-1} b + c^T A_2^{-1} b) + A_1 A_2^{-1}\right) bc^T A_2^{-1} \\
&= A_1 A_2^{-1} + \frac{1}{\lambda}(A_1 A_2^{-1} - I) = \mathrm{C}(A_0) + ((\mathrm{C}(A_0) - I)b)\,\frac{c^T A_2^{-1}}{\lambda} \\
&=: R + \beta\gamma^T(A)
\end{aligned}$$

where R and β are exactly like in Equations (4.31) and (4.32). As stated in Lemma 4.8, we have to require that $\lambda = -1 - c^T(A_0 + I)^{-1} b \neq 0$. This is shown relatively easily given the following: As $\det(A_2) = 1$, $A_2^{-1} = \mathrm{adj}(A_2)$.

$$A_2^{-1} = \mathrm{adj}\begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & & \cdots & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 & \cdots & (-1)^{n+1} \\ 0 & 1 & -1 & & (-1)^n \\ \vdots & & \ddots & & \vdots \\ 0 & & \cdots & & 1 \end{bmatrix}$$

Given the definitions of c^T and b, it follows that

$$\begin{aligned}
-\lambda &= 1 + c^T \underbrace{A_2^{-1}\begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T}_{\text{last row of } A_2^{-1}} = 1 + \begin{bmatrix} -a_0 & -a_1 & \cdots & -a_{n-1} \end{bmatrix}\begin{bmatrix} (-1)^{n+1} \\ (-1)^n \\ \vdots \\ -1 \\ 1 \end{bmatrix} \\
&= 1 + \sum_{i=0}^{n-1} (-1)^{n+i} a_i = (-1)^n p_A(-1)
\end{aligned}$$

where $p_A(s) = \det(sI - A)$ denotes the characteristic polynomial of A. As A was assumed to be Schur stable, $p_A(-1) \neq 0$.

Now it only remains to show that (R, β) is controllable for every n, then we can start to construct the transformation matrix T. Therefore we employ the Popov-Belevitch-Hautus-Criterion. As all eigenvalues of R are -1, this leads to

$$\begin{aligned}\operatorname{rank}\begin{bmatrix}-I-R & \beta\end{bmatrix} &= \operatorname{rank}\left[\begin{array}{ccccc|c} 0 & 2 & -2 & \cdots & 2(-1)^{n+1} & 2(-1)^n \\ 0 & 0 & 2 & \cdots & 2(-1)^n & 2(-1)^{n+1} \\ \vdots & & \ddots & \ddots & \vdots & \vdots \\ & & & & 2 & \\ 0 & & \cdots & & 0 & -2 \end{array}\right] \\ &\overset{!}{=} n\end{aligned}$$

which obviously is satisfied for every n.

Let $M = \begin{bmatrix}\beta & R\beta & \cdots & R^{n-1}\beta\end{bmatrix}$ and let m^T be the last row of M^{-1}. In e.g. [Lue79], it is shown that for the matrix $T = \begin{bmatrix} m^T \\ m^T R \\ \vdots \\ m^T R^{n-1}\end{bmatrix}$ it then holds that TRT^{-1} is in companion form and $TB = \begin{bmatrix}0 & \cdots & 0 & 1\end{bmatrix}^T$. As shown in the beginning of the proof, the matrix T thus is the one we are looking for. This completes the proof. □

Theorem 4.22 (Robust stability for companion matrices with interval uncertainties)**.** *Let*

$$\mathcal{A} = \left\{ \begin{bmatrix}a_0 & a_1 & \dots & a_{n-1}\end{bmatrix}^\flat \mid a_i \in [\underline{a}_i, \overline{a}_i]\ \forall i = 0 \dots n-1 \right\}$$

and $\mathcal{B}$ in a similar matter denote two sets of Schur stable companion matrices with uncertain entries.
Then there exist two sets $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{B}}$ of Hurwitz stable companion matrices with interval uncertainties such that:
There exists a CQLF for any $A \in \mathcal{A}$ and any $B \in \mathcal{B}$ $\Leftrightarrow$ There exists a CQLF for any $A \in \tilde{\mathcal{A}}$ and any $B \in \tilde{\mathcal{B}}$.

Proof. A CQLF for any A and B exists $\Leftrightarrow$ A CQLF for any $F \in \{\mathrm{C}(A) \mid A \in \mathcal{A}\}$ and any $G \in \{\mathrm{C}(B) \mid B \in \mathcal{B}\}$ exists $\Leftrightarrow$ For any nonsingular matrix T, there exists a CQLF for any TFT^{-1} and TGT^{-1}. Now choose T to be the matrix from Lemma 4.7 and apply it to all $F \in \{\mathrm{C}(A) \mid A \in \mathcal{A}\}$ and all $G \in \{\mathrm{C}(B) \mid B \in \mathcal{B}\}$. That yields two sets $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{B}}$ of companion matrices. As similarity transforms preserve the eigenvalues and the Bilinear Transform

maps Schur matrices to Hurwitz matrices, all matrices in $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{B}}$ are Hurwitz. So it only remains to show, that they are sets of matrices with interval uncertainties.
Therefore consider how $\tilde{\mathcal{A}}$ is obtained: The Bilinear Transform involves the inverse of $(I - A)$, which is exists for any $A \in \mathcal{A}$, otherwise one A would have an eigenvalue 1, which contradicts the assumption, that all A in $\mathcal{A}$ are Schur. Thus every entry of $\mathrm{C}(A)$ consists of some polynomial in the entries of A divided by $\det(I - A)$, which is nonzero. The following similarity transforms just "mixes things up" a little, but in the end every entry of any $F \in \tilde{A}$ consists of the same rational function in the entries of "its" A. As the denominators are still $\det(I - A) \neq 0$, the functions are continuous on the closed convex set $[\underline{a}_0, \overline{a}_0] \times [\underline{a}_1, \overline{a}_1] \times \cdots \times [\underline{a}_{n-1}, \overline{a}_{n-1}]$ and thus have a global minimum and maximum and can take every value between them. In other words: their values are constrained to intervals. The same applies of course for $\tilde{\mathcal{B}}$. That completes the proof. □

This Theorem is preliminary in that it leaves the question, how to decide whether $\mathcal{A}$ and $\mathcal{B}$ consist only of Schur matrices, open. If that is clear, the following algorithm can be applied:

1. Compute the Bilinear Transform for a general companion matrix of dimension n
2. Compute the transformation T
3. Minimize and maximize the entries over the sets $[\underline{a}_0, \overline{a}_0] \times [\underline{a}_1, \overline{a}_1] \times \cdots \times [\underline{a}_{n-1}, \overline{a}_{n-1}]$ and $[\underline{b}_0, \overline{b}_0] \times [\underline{b}_1, \overline{b}_1] \times \cdots \times [\underline{b}_{n-1}, \overline{b}_{n-1}]$. This gives you the new upper and lower bounds $\underline{\alpha}_i$, $\overline{\alpha}_i$, $\underline{\beta}_i$ and $\overline{\beta}_i$, respectively, for the matrices in $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{B}}$.
4. Apply condition 2 of Theorem 4.16 to $\tilde{\mathcal{A}}$ and $\tilde{\mathcal{B}}$.

Chapter 5

Connecting the parts: Stability analysis as a Lur'e problem

5.1 Overview

As our proposal includes the closing of a feedback loop, a natural question is, whether this feedback loop could introduce instability. In this Chapter, we will therefore investigate the stability properties of the loss rate control and thereby make use of the results of the last Chapter.

In the next Section, we will show, how the control loop can be treated in the theoretical framework of Lur'e systems. Therefore we will assume a nearly static relation between queue capacity and loss rate in an uncongested link and will identify this relation with the nonlinearity in the feedback path. As there can be no assumptions made but the negative gain, the sector will be $(0, \infty)$. As the Lur'e framework is only useful in connection with linear systems, this only includes the linear controller of Section 3.5.2, the MIMD and MIAD algorithms of Sections 3.5.3 and 3.5.4 will not be treated in this Chapter.

In the following Sections, we then will apply our theoretical results to filters of varying degrees. We will give bounds for the filter coefficients ρ_i to ensure stability with arbitrary gain for a first (Section 5.3), second (Section 5.4) and a special Nth (Section 5.5) order filter.

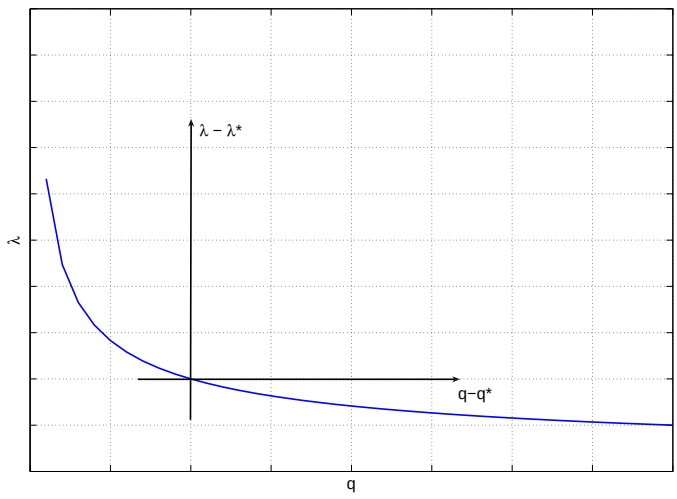


Figure 5.1: An example for the relation between queue capacity and loss rate

5.2 Formulation as a Lur'e problem

We have the linear filter $F(z)$ and the linear controller $K(z)$. They will constitute the linear system in the forward path. For the nonlinearity in the feedback path, some more work is needed.

In [AKM04] a static relation between queue capacity and utilization in a congested link was shown, which in [SSK06] was used to propose a closed loop algorithm which adapts the queue capacity to the external conditions. See Section 3.3 for details. We now assume, that there is a similar relationship between queue capacity and loss rate in an uncongested link. Since the only further assumption we can make is, that a bigger queue capacity can not result in a bigger loss rate, the line shown in Figure 5.1 is just an example.

As in the Lur'e framework only sector nonlinearities can be treated, we now have to transform the line into one. Therefore we assume, that there exists an equilibrium queue capacity q^* to each target loss rate λ^* and "move" the origin in Figure 5.1 by considering the new variables $q - q^*$ and $\lambda - \lambda^*$. Then we define g with

$$\lambda - \lambda^* = g(q - q^*, t) \tag{5.1}$$

and observe:$-g(\cdot)$ is a sector nonlinearity belonging to the sector $(0, \infty)$.

As q^* is an exogenous input, it does not influence stability, so we completely

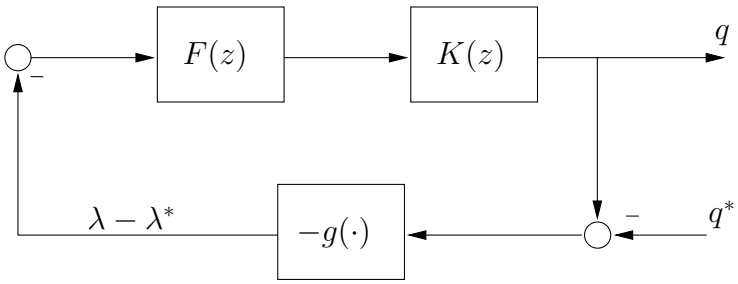


Figure 5.2: Loss rate control as a Lur'e system

fitted the proposal in the Lur'e framework like depicted in Figure 5.2.

Recall from Section 3.5, that the controller is given by $K(z) = \frac{\gamma z}{z-1}$ like in (3.8) and the filter of n-th order has the transfer function

$$F(z) = \frac{\rho_{N_F} z^{N_F}}{z^{N_F} - \sum_{i=0}^{N_F-1} \rho_i z^i} \tag{5.2}$$

As the sector under consideration is $(0, \infty)$ and we are only interested in stability, we can move the gains ρ_{N_F} and γ into the nonlinearity without changing the result.
So let $G(z)$ be the "gain free" linear part

$$\begin{aligned} G(z) &= \frac{z^{N_F}}{z^{N_F} - \sum_{i=0}^{N_F-1} \rho_i z^i} \frac{z}{z-1} \\ &= \frac{z^{N_F+1}}{z^{N_F+1} - (1+\rho_{N_F-1}) z^{N_F} + \sum_{i=1}^{N_F-1} (\rho_i - \rho_{i-1}) z^i + \rho_0} \end{aligned} \tag{5.3}$$

Then we can pose the question under consideration:

> Consider the Lur'e system with linear part $G(z)$ and feedback nonlinearity belonging to $(0, \infty)$. For which choices of N_F and ρ_i is the system absolutely stable?

Remark 5.1 (Marginally stable open loop)**.** Note, that $G(z)$ has one pole $z_i = 1$ and thus is no stable transfer function. In other words, the system matrix in any realization of $G(z)$ has an eigenvalue 1 and therefore is not Schur stable, so the Bilinear Transform does not exist.
On the other hand, closing the loop with a gain of at least $\varepsilon > 0$ moves that eigenvalue inside the unit circle. For this reason we consider the sector $(0, \infty)$ instead of $[0, \infty)$.

5.3 First order filter

In this section we consider $N_F = 1$, thus $G(z) = \frac{z^2}{z^2-(1+\rho)z+\rho}$. A realization $[A, B, C, D]$ of G is given by

$$A = \begin{bmatrix} 0 & 1 \\ -\rho & \rho+1 \end{bmatrix} \tag{5.4a}$$

$$B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{5.4b}$$

$$C = \begin{bmatrix} -\rho & \rho+1 \end{bmatrix} \tag{5.4c}$$

$$D = 1. \tag{5.4d}$$

Now consider the sector $[\varepsilon, \infty)$ with $\varepsilon > 0$ arbitrarily small (we later will send it to zero anyway). By remark 4.1, we can look at the sector $[0, \infty)$ and the linear system $[\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}]$ with

$$\tilde{A} = \begin{bmatrix} 0 & 1 \\ \frac{-\rho}{1+\varepsilon} & \frac{\rho+1}{1+\varepsilon} \end{bmatrix} \tag{5.5a}$$

$$\tilde{B} = \begin{bmatrix} 0 \\ \frac{1}{1+\varepsilon} \end{bmatrix} \tag{5.5b}$$

$$\tilde{C} = \begin{bmatrix} \frac{-\rho}{1+\varepsilon} & \frac{\rho+1}{1+\varepsilon} \end{bmatrix} \tag{5.5c}$$

$$\tilde{D} = \frac{1}{1+\varepsilon} \tag{5.5d}$$

instead .

By the remark to Theorem 4.21, we now have to check the matrix product $\mathrm{C}(\tilde{A})\,\mathrm{C}\left(\tilde{A} - (1+\varepsilon)\tilde{B}\tilde{C}\right) =: W_\varepsilon$ for negative, real eigenvalues to gain a sufficient condition for absolute stability.

$$\mathrm{C}(\tilde{A}) = \begin{bmatrix} \frac{\varepsilon-2}{2\rho+2+\varepsilon} & \frac{2+2\varepsilon}{2\rho+2+\varepsilon} \\ \frac{-2\rho}{2\rho+2+\varepsilon} & \frac{2\rho-\varepsilon}{2\rho+2+\varepsilon} \end{bmatrix} \tag{5.6}$$

$$\mathrm{C}\left(\tilde{A} - (1+\varepsilon)\tilde{B}\tilde{C}\right) = \begin{bmatrix} -1 & 2 \\ 0 & -1 \end{bmatrix} \tag{5.7}$$

$$W_\varepsilon = \begin{bmatrix} \frac{\varepsilon+2}{2\rho+2+\varepsilon} & -2\frac{3+2\varepsilon}{2\rho+2+\varepsilon} \\ \frac{2\rho}{2\rho+2+\varepsilon} & \frac{-6\rho+\varepsilon}{2\rho+2+\varepsilon} \end{bmatrix} \tag{5.8}$$

Now we get rid of ε by sending it back to zero and obtain an expression for the eigenvalues of W_0. One of them is always zero (which corresponds to the fixed eigenvalue 1 in the discrete time system), the other one is $\lambda(\rho) := \frac{1-3\rho}{\rho+1}$. Obviously, $\lambda(\rho) \in \mathbb{R}\ \forall \rho \in [0,1]$, so we need to require $\lambda(\rho) > 0$ to satisfy the conditions of Theorem 4.21. A simple calculation yields

$$\lambda(\rho) > 0 \qquad \Longleftrightarrow \qquad \rho < \frac{1}{3} \tag{5.9}$$

Recall that $\sum \rho_i = 1$ for any filter, then the system remains stable with the filter $F(z) = \frac{(1-\rho)z}{z-\rho}$ and the controller $K(z) = \frac{\gamma z}{z-1}$ for any gain $\gamma > 0$ as long as $0 < \rho < \frac{1}{3}$.

5.4 Second order filter

Now let $N_F = 2$. Then $G(z) = \frac{z^3}{z^3-(1+\rho_1)z^2+(\rho_1-\rho_0)z+\rho_0}$ and a realization $[A, B, C, D]$ is

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\rho_0 & \rho_0-\rho_1 & \rho+1 \end{bmatrix} \tag{5.10a}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{5.10b}$$

$$C = \begin{bmatrix} -\rho_0 & \rho_0-\rho_1 & \rho+1 \end{bmatrix} \tag{5.10c}$$

$$D = 1. \tag{5.10d}$$

We again assume a sector $[\varepsilon, \infty)$, use remark 4.1, calculate the matrix product W_ε and send ε back to zero. Then we arrive at the matrix W_0 with

$$W_0 = \begin{bmatrix} \frac{1+\rho_1}{1+\rho_1-\rho_0} & -\frac{4+3\,\rho_1}{1+\rho_1-\rho_0} & \frac{7+4\,\rho_1}{1+\rho_1-\rho_0} \\ -\frac{\rho_0}{1+\rho_1-\rho_0} & \frac{3\,\rho_0+1}{1+\rho_1-\rho_0} & -\frac{4\,\rho_0+3}{1+\rho_1-\rho_0} \\ \frac{\rho_0}{1+\rho_1-\rho_0} & \frac{-4\,\rho_0+\rho_1}{1+\rho_1-\rho_0} & -\frac{-7\,\rho_0+3\,\rho_1}{1+\rho_1-\rho_0} \end{bmatrix} \tag{5.11}$$

which again has one fixed eigenvalue at 0, the other two are given by

$$\lambda_{+/-}(\rho_0, \rho_1) = \frac{5\,\rho_0 + 1 - \rho_1 \pm 2\sqrt{{\rho_1}^2 - 2\,\rho_1\,\rho_0 + 5\,{\rho_0}^2 + 4\,\rho_0}}{1+\rho_1-\rho_0}. \tag{5.12}$$

Again, $\lambda_{+/-}(\rho_0, \rho_1) \in \mathbb{R}$ and $\lambda_- < \lambda_+$, so we have to require

$$\lambda_-(\rho_0, \rho_1) = \frac{5\,\rho_0 + 1 - \rho_1 - 2\sqrt{{\rho_1}^2 - 2\,\rho_1\,\rho_0 + 5\,{\rho_0}^2 + 4\,\rho_0}}{1 + \rho_1 - \rho_0} > 0 \tag{5.13}$$

which holds for $\rho_0 < \frac{1}{5} - \frac{3}{5}\rho_1$. Still $\sum \rho_i = 1$, so an algorithm of finding suitable ρ_i can be to first choose a pair (ρ_0, ρ_1), for which the inequality holds, and then $\rho_2 = 1 - \rho_0 - \rho_1$. Observe, that $\rho_0 + \rho_1 < \frac{1}{3}$, so for any $\rho_2 > \frac{2}{3}$, a suitable pair (ρ_0, ρ_1) can be found.

5.5 N-th order filter

As could be seen in the last sections, checking stability for a given filter order N_F involves calculating the eigenvalues for a $(N_F + 1) \times (N_F + 1)$-matrix. Although this is still possible for $N_F = 3$, the conditions for the filter coefficients ρ_i can be expected to be quite unwieldy. Additionally, from a practical point of view, it is desirable to have not too many parameters to choose. Therefore we propose here a choice of the ρ_i which allows high filter orders and easily checkable stability conditions.

The two parameters to choose are the filter order N_F and a decaying rate $q \in (0, 1)$ which is the ratio of two consecutive ρ_i: $\rho_i/\rho_{i+1} = q$. Then define:

$$\rho_{N_F} = \frac{1 - q}{1 - q^{N_F+1}} \tag{5.14}$$

$$\rho_{N_F - i} = \rho_{N_F} q^i \qquad i = 1, \ldots, N_F. \tag{5.15}$$

If one recognizes the geometric series, it is easily checked, that

$$\sum_{i=0}^{N_F} \rho_i = \rho_{N_F} \sum_{i=0}^{N_F} q^i = \frac{1 - q}{1 - q^{N_F+1}} \frac{1 - q^{N_F+1}}{1 - q} = 1. \tag{5.16}$$

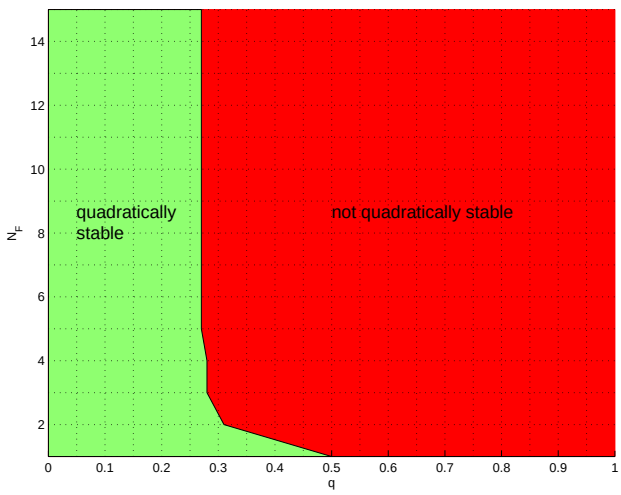


Figure 5.3: Regions for stable pairs of N_F and q

Inserting the ρ_i in (5.3) and computing a realization yields

$$A = \begin{bmatrix} 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ -\rho_{N_F} q^{N_F} & \rho_{N_F}(q-1)q^{N_F-1} & \cdots & \rho_{N_F}(q-1)q & 1+\rho_{N_F} q \end{bmatrix} \in \mathbb{R}^{(N_F+1)\times(N_F+1)} \tag{5.17}$$

$$B = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix}^T \in \mathbb{R}^{(N_F+1)\times 1} \tag{5.18}$$

$$C = \begin{bmatrix} -\rho_{N_F} q^{N_F} & \rho_{N_F}(q-1)q^{N_F-1} & \cdots & \rho_{N_F}(q-1)q & 1+\rho_{N_F} q \end{bmatrix} \in \mathbb{R}^{1\times(N_F+1)} \tag{5.19}$$

$$D = 1. \tag{5.20}$$

Note, that ρ_{N_F} also depends only on q and N_F.

Now it is possible to calculate the matrix products W_0 like in the preceding sections for every N_F. They only depend on q, so after choosing a finite set $Q \subset (0,1)$ of values for q, the eigenvalues can be checked numerically and pairs (N_F, q) for stable loops can be identified. The results for $N_F = 1, \ldots, 15$ are shown in Figure 5.3.

Of course that is just one way of choosing filter coefficients which ensure stability; many more could be developed. A reduction of the parameters however seems advisable in any case.

We also stress, that our stability analysis is probably very conservative, be-

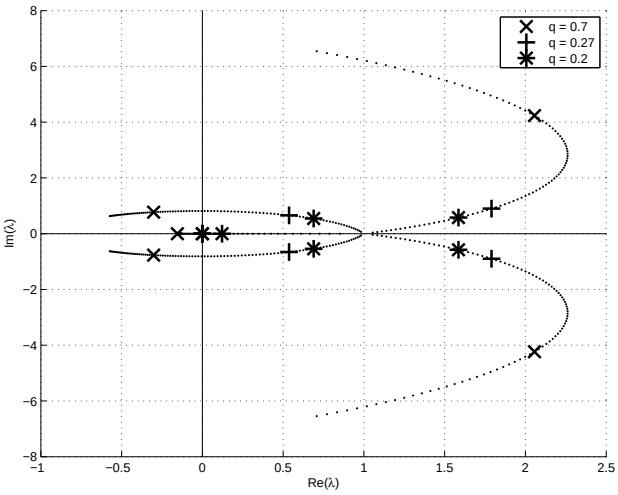


Figure 5.4: A root locus for $N_F = 6$ and $q \in (0, 1)$

cause the maximum feedback gain can be expected to be well below ∞. However, here we derived conditions for guaranteed stability under minimal assumptions.

Chapter 6

Conclusions and future work

In the course of this thesis, we identified over-provisioning as a way to ensure a certain performance level. We introduced the term Quality of Service (QoS), which has been coined to describe this performance level and **defined core link provisioning** as the process of finding the minimum necessary bandwidth to ensure Quality of Service (Chapter 1).

In Chapter 2, representatively for all model-based approaches, the **recent work** of van de Meent [vdM06] **was reviewed**. Additionally, we gave a short outline of DiffServ and IntServ, which are technologies to ensure QoS by treating different flows with different requirements differently. The review also included **shortcomings of the existing solution**s: Model-based link provisioning heavily relies on the applicability of the models, which in general can not be assumed due to the time invariant behaviour of network traffic. DiffServ and IntServ suffer from scalability and complexity issues.

We then **presented our proposal** for a practicable way of link provisioning with minimal additional effort in Chapter 3. A **control loop** is introduced: The capacity q of the buffer in front of the link in question is used to maintain a certain low loss rate λ^*. We then argued, that the consideration of the buffer capacity q yields additional information on the fluctuations and variance of the traffic arrival process. Some basic simulation results to further illustrate the proposal are also given.

In Chapter 4, **theoretical tools** to assess the stability of the control loop are summarized and developed further. Namely, we give conditions on the existence of a common quadratic Lyapunov function (CQLF) for two linear time invariant systems in continuous as well as in discrete time. This results are linked to the absolute **stability of Lur'e systems**. The main contri-

bution in terms of the later application to our problem is Theorem 4.21, we restate it here for completeness:

Theorem 4.21 (Spectral condition for absolute stability)**.** *The SiSo DT Lur'e system consisting of the linear system* $[A, B, C, D]$ *and a nonlinearity belonging to* $[0, k]$ *is absolutely stable*
$\Longleftarrow$
The matrix product $\mathrm{C}(A)\ \mathrm{C}\left(A - \frac{k}{1+kD}BC\right)$ *has no negative real eigenvalue.*

The **application** of the theoretical results to our proposal is then carried out in Chapter 5. The control loop is transformed into a discrete time Lur'e problem and with the help of Theorem 4.21, **constraints on the control parameters** which guarantee stability are derived.

Clearly, **future work** is needed: Good reasoning and a sound stability theoretical basis for our novel approach is provided here. The next step should be to evaluate the approach with further simulations and experiments on real networks. At the same time, guidelines on the actual tuning of the control parameters to reach good controller performance need to be developed. We also still owe an experimental evaluation of the assumption in Section 5.2.

Appendix

A Matlab source codes

A.1 Nth_Order.m

This file takes the order N_F of the filter as an argument and puts out the eigenvalues of the matrix product W_0 for different values of q as well as the values of q, for which real negative eigenvalues occur.

```
function [PLT,S] = Nth_Order(N)

%%%
% This function takes one argument - the order N of a filter.
% It then calculates the matrix product W and puts out:
% PLT : each row contains the value of q in the first and the
% eigenvalues of W in the remaining columns
% S : Contains the values of q for which W has a real negative eigenvalue

if (round(N)≠N)
    disp('Please... the order has to be a whole number');
    return;
end

syms q e

% this is rho_NF
r = (1-q)/(1-q^(N+1));

% setting up the system matrix A
H = [ zeros(N,1) eye(N)];
h = sym(zeros(1,N+1));
h(1) = -r*q^N;
```

```
h(N+1) = 1+r*q;

for i = 2:N
    h(i) = r*(q - 1)*q^(N+1-i);
end

A = [H;h];
B = sym([zeros(N,1);1]);
C = h;
D = 1;

% closing the feedback loop over a gain eps
AE = A - e/(1+e)*B*C;
BE = B/(1+e);
CE = C/(1+e);
DE = D/(1+e);

% bilinear transform
CA = cayley(AE);
CAC = cayley(AE-BE*CE/DE);

% matrix product
WE = CA*CAC;

% sending eps -> 0
W0 = simplify(subs(WE,e,0));

PLT = [];
for qq = 0.02:0.01:0.98
    % inserting values for q
    WQ = subs(W0,q,qq);
    EV = eig(WQ);
    PLT = [PLT;qq EV'];
end

% tolerance for negative eigenvalues
% (recall that one of them is always 0. numerical errors can make
% make that one appear negative.)
eps = 1e-12;

% finding the indices of negative real eigenvalues and
% storing the according q's in S.
ind1 = find( (imag(PLT)==0)&(real(PLT)<-eps));
[rs,cs] = ind2sub(size(PLT),ind1);
S = PLT(rs,1);
```

A.2 Nth_order_plot.m

This script calls `Nth_order.m` with various values for N_F and creates the plot shown in Figure 5.3.

```
% Filter orders to check
N = [1:15];

figure;
axis([0 1 N(1) N(end)]);
hold on;

PLTotal = {};
SS = {};
P = zeros(1,length(N));

% calling Nth_Order with the values for N
for i=N
    [PLT,S] = Nth_Order(i);
    % saving the minimum q for every N
    P(i-N(1)+1) = min(S)
    % plotting the "forbidden" values for q
    plot(S,i*ones(size(S)),'-v')

    PLTotal{end+1} = PLT;
    SS{end+1} = S;
end
grid
hold off;

figure;

% plotting the area of guaranteed stability
patch([0 0 P],[N(end) N(1) N],10);
set(gca,'XLim',[0 1],'Color','r')

% plotting the grid - otherwise it's behind the patch
for i=N
    line([0 1],[i i],'Color','k','LineStyle',':');
end

for i=0.05:0.05:0.95
    line([i i],[N(1) N(end)],'Color','k','LineStyle',':');
end
```

B Network technology basics

B.1 A short overview of TCP

Protocol

TCP is an abbreviation for Transmission Control Protocol and is a

- ▷ transport layer
- ▷ connection-oriented
- ▷ reliable delivery

Protocol. For a networks expert this might say it all, for all the others we will now give a short explanation, which is *not* meant to be comprehensive, will not even cover all basic facts, but will just consist of the facts about TCP that are interesting in some way concerning this thesis.
For comprehensive information about TCP, see e.g. [Pos81].

Transport layer This description originates from the so-called OSI Seven-Layer-Model [Zim80] and means, that TCP is passed packets of data by the top layer and has to make sure, that these packets arrive at their destinations. So TCP passes them to the layer below it (the so-called Network layer) which then is responsible for the transport to the destination.
The important fact for us is: By doing so, the TCP implementation of the sending machine is responsible for the rate at which the packets are released into the network.

Connection-oriented This states that, before any real data packets are sent, a connection has to be established between the source and the destination. After successful transmission, the connection will be closed.

Reliable delivery Unlike TCP, UDP (User Datagram Protocol B.2) is a connectionless transport-layer-protocol and thus does not enable the source to notice, if a packet gets lost on its way to the destination.
A TCP source on the other hand expects an acknowledgement from the destination for each packet it has sent. If there is no acknowledgement during a certain time interval, the TCP source sends it once more. This is called a retransmit and guarantees, that each packet arrives at the

destination.
Important here: Using TCP enables the source to sense data losses and to react accordingly.

Working modes

Each TCP source maintains a transmission window. The size of this window specifies, how many packages may be sent without having received the acknowledgement for the first one.
Furthermore, each TCP source is trying to exploit the available bandwidth. To do so the transmission window is resized: As long as there are "no problems", i.e. no packets are lost, the window is enlarged. When the first packet loss is noticed, it's been "too much" and the window size is decreased.
How this de- and increase is done, will be described now. A very good and much more detailed description can be found in the excellent paper [Jac88].

Slow Start When a TCP connection is established, the source starts off with a very small window. But it seems wise, to decrease it very fast in the beginning to get a rough idea of the network conditions quickly. So the window size is doubled each time a full window of data packets gets acknowledged.
When the first packetloss is noticed, the source backs off by halving its transmission window. In other words it is reset to the last known working configuration.
This kind of algorithm is called MIMD (Multiplicative Increase, Multiplicative Decrease) algorithm. When the Slow Start has provoked the first packetloss and by that probed the available bandwidth, the source goes into the so-called Congestion Avoidance mode.

Congestion Avoidance Now the source starts increasing its window more carefully by just *adding* a fixed number α of packets to its transmission window each time a full window of data packets gets acknowledged. The standard is $\alpha = 1$.
When, again, a packetloss occurs, the window is again decreased, this time by a factor β. Here, a common choice is $\beta = 0.5$.
Nevertheless, each TCP implementation can have other values α and β.
This kind of algorithm is called AIMD (Additive Increase, Multiplicative Decrease) algorithm.

So each TCP source steadily increases the number of packets in the network, then backs off if a packet gets lost by decreasing this fixed number and starts to increase it again.
The details of this "increasing" and "backing off" can be different for each TCP source and are impossible to know from a network operator point-of-view.

TCP data transfer constitutes the major part of common network traffic [Nat].

B.2 One line about UDP

UDP stands for User Datagram Protocol and is a very simple transport layer protocol, in the sense that it just sends data and expects no acknowledgements. So it is not able to sense lost packets or to control the sending rate. For details see [Pos80].

B.3 A very short overview of SNMP

SNMP stands for Simple Network Management Protocol. Like for TCP, only information that is interesting regarding this thesis will be provided, for comprehensive information see [CFS90].

SNMP enables a network manager to gather information about each network element where SNMP is implemented. It is novadays available on most routers, switches, printers and so on.
It is an application layer protocol (which is the top layer of the transport layer) and of course, each time the monitoring application on the manager machine calls for information, additional network traffic is caused. To be exact, SNMP uses UDP.

Important for us: From modern routers, information about traffic conditions like queuesize, number of packet arrivals, number of lost packets etc. can be gathered by a central machine via SNMP. Because each measurement causes network traffic itself, the frequency of measurements has to be coarse relative to the other traffic. A common choice is to take one measurement every 5 minutes.

B.4 Some other terms

The time which passes when a packet is sent until the acknowledgement reaches the sender is called **Round Trip Time** (RTT). Each flow has an average or mean RTT which is used, for instance, to calculate the time-out of a TCP implementation.

A **DropTail Queue** is a queue in a network entity, e.g. a router, where data packets line up waiting to be sent. If the queue is full, all new arriving packets are dropped.

The **Lossrate** is the ratio of the dropped packets to the arrived packets in a certain time interval. Thus $\lambda = \frac{\#_{dropped}}{\#_{arrived}}$.

List of symbols and abbreviations

QoS	Quality of Service
IP	Internet Protocol - A network layer protocol
TCP	Transmission Control Protocol – See Section B.1
UDP	User Datagram Protocl – See Section B.1
SNMP	Simple Network Management Protocol – See Section B.1
Mbit/s	Megabits per second
CBR	Constant BitRate
$P(X > x)$	The Probability, that the random variable X is greater than x
$E(X)$	The Expectation of the random variable X
σ^2	The Variance of a probability distribution
mgf	The Moment Generating Function
CT, DT, LTI	Continuous Time, Discrete Time, Linear Time Invariant
LF, QLF, CLF, CQLF	Lyapunov Function, Quadratic Lyapunov Function, Common Lyapunov Function, Common Quadratic Lyapunov Function
I/O, SISO	Input-Output, Single Input Single Output
$[A, B, C, D]$	State space model: $\dot{x} = Ax + Bu$, $y = Cx + Du$ in continuous time and $x(k+1) = Ax(k) + Bu(k)$, $y(k) = Cx(k) + Du(k)$ in discrete time

$\mathbb{N}$, $\mathbb{N}_+$, $\mathbb{R}$, $\mathbb{R}_+$ and $\mathbb{C}$	The sets of the natural, positive natural, real, non-negative real and complex numbers
$\mathbb{R}^n$	The set of n-dimensional vectors with real entries
$\mathbb{R}^{m\times n}$	The set of the $m \times n$ matrices with real entries
$\mathbb{R}_H^{n\times n}$, $\mathbb{R}_S^{n\times n}$	The sets of Hurwitz and Schur stable $n \times n$ matrices with real entries
$P > 0$ ($P \geq 0$)	The quadratic matrix P is positive (semi-)definite
$\|\cdot\|$	Some norm
$\dot{x}$	The derivative of x with respect to the time t
$\frac{\partial V}{\partial x}$	If $x, V(\cdot) \in \mathbb{R}$: The partial derivative of V with respect to x If $V(\cdot) \in \mathbb{R}$, $x \in \mathbb{R}^n$: The gradient of V If $V(\cdot), x \in \mathbb{R}^n$: The Jacobian of V
A^T	The transpose of the matrix A
$\Leftarrow$, $\Rightarrow$, $\Leftrightarrow$	"if", "only if", "if and only if"

Bibliography

[AKM04] Guido Appenzeller, Isaac Keslassy, and Nick McKeown. Sizing router buffers. In *SIGCOMM '04*, Portland, Oregon, USA, 2004.

[BBC+98] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wanf, and W. Weiss. RFC 2475: An Architecture for Differentiated Services, 1998. http://tools.ietf.org/html/rfc2475.

[BCS94] L. Bradsen, D. Clark, and S. Shenker. RFC 1633: Integrated Services in the Internet Architecture: An Overview., 1994. http://tools.ietf.org/html/rfc1633.

[BR06] S. Bessler and P. Reichl. A network provisioning scheme based on decentralized bandwidth auctions. In TELECOMMUNICATIONS PLANNING: Innovations in Pricing, Network Design and Management, Springer, Operations Research/Computer Science Interfaces Series, ISBN 0-387-29222-5, 2006.

[CDB91] H. Chapellat, M. Dahleh, and S. Bhattacharyya. On robust nonlinear stability of interval control systems. *IEEE Transactions on Automatic Control*, 36(1):59–67, 1991.

[CFS90] Jeffrey D. Case, M. Fedor, and M. Schoffstall. RFC 1157: A Simple Network Management Protocol, 1990. http://tools.ietf.org/html/rfc1157.

[Co] Corvil Networks. http://www.corvil.com.

[DM99] Wijesuriya Dayawansa and C.F. Martin. A converse Lyapunov theorem for a class of dynamical systems which undergo switching. *IEEE Transactions on Automatic Control*, 4(44):751–760, 1999.

[FTD03] Chuck Fraleigh, Fouad Tobagi, and Christophe Diot. Provisioning IP backbone networks to support latency sensitive traffic. In Proc. IEEE INFOCOM, 2003.

[GGS01] Graham Goodwin, Stefan Graebe, and Mario Salgado. *Control System Design*. Prentice Hall, 2001.

[HJ85] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[Jac88] Van Jacobsen. Congestion avoidance and control. *Proceedings of SIGCOMM '88*, 1988.

[Kai80] Thomas Kailath. *Linear Systems*. Prentice Hall, 1980.

[Kle76] C. Kleinrock. Queueing systems: Volume 2. John Wiley and Sons, 1976.

[KSL06] Chris Kellett, Robert Shorten, and Douglas Leith. Sizing internet router buffers, active queue management, and the Lur'e problem. In *IEEE Conference on Decision and Control*, San Diego, 2006.

[Law05a] libpcap: Packet Capture Library, 2005. http://www.tcpdump.org.

[Law05b] TCP-dump: The protocol packet capture and dumper program, 2005. http://www.tcpdump.org.

[LKC+07] Douglas Leith, Chris Kellett, Peter Clifford, Jonathan Epperlein, and Robert Shorten. A non-invasive method for capacity planning using coarse grained network measurements. To appear in IEEE Communications Letters, 2007.

[Lue79] David Luenberger. *Introduction to Dynamic Systems: Theory, Models, and Applications*. John Wiley and Sons, Inc. New York, 1979.

[Mas04] Oliver Mason. *Switched systems, convex cones and common Lyapunov functions*. PhD dissertation, National University of Ireland, Maynooth, 2004.

[MNMK06] T. Mori, T.V. Nguyen, Y. Mori, and H. Kokame. Preservation of Lyapunov functions under Bilinear]Mpping. 42:1055–11058, 2006.

[Mor00] Robert Morris. Scalable TCP congestion control. In *INFOCOM '00*, number 26, Tel Aviv, Israel, 2000.

[MvdM05] Michel Mandjes and Remco van de Meent. Inferring traffic burstiness by sampling the buffer occupancy. In Proceedings of NETWORKING 2005: 4th International IFIP-TC6 Networking Conference, 2005.

[Nat] National Laboratory for Applied Network Research. Lots of traffic traces are collected here. In a major part of them, TCP-traffic constitutes a major part of the total traffic. http://pma.nlanr.net/Special/.

[NB97] Kumpati Narendra and J. Balakrishnan. Adaptive control using multiple models. *IEEE Transactions on Automatic Control*, 3(42):171–187, 1997.

[NG64] Kumpati Narendra and R Goldwyn. A geometrical criterion for the stability of certain nonlinear nonautonomous systems. *IEEE Transactions on circuit theory*, 1964.

[Pap03] Konstantina Papagiannaki. *Provisioning IP Backbone Networks Based on Measurements*. PhD dissertation, University College London, 2003.

[Pos80] Jon Postel. RFC 768: User Datagram Protocol, 1980. http://tools.ietf.org/html/rfc768.

[Pos81] Jon Postel. RFC 793: Transmission Control Protocol, 1981. http://tools.ietf.org/html/rfc793.

[SK04] Robert Shorten and Chris King. Spectral conditions for positive realness of single-input-single-output systems. *IEEE Transactions on Automatic Control*, 49(10):1875–1877, October 2004.

[SSK06] Rade Stanojević, Robert Shorten, and Chris Kellett. Adaptive tuning of drop-tail buffers for reducing queueing delays. *IEEE Communications Letters*, 10(7), July 2006.

[vdM06] Remco van de Meent. *Network link dimensioning - A measurement & modelling based approach*. PhD dissertation, University of Twente, 2006. http://wwwhome.cs.utwente.nl/~meentr.

[vdMPM+03] Remco van de Meent, A. Pras, M. Mandjes, H. van den Berg, and L. Nieuwenhuis. Traffic measurements for link dimensioning. In Proceedings of the 14th IFIP/IEEE Workshop on Distributed Systems: Operations and Management, 2003.

[Wul05] Kai Wulff. *Quadratic and non-quadratic stability criteria for switched linear systems*. PhD dissertation, National University of Ireland, Maynooth, 2005.

[Zim80] Hubert Zimmermann. OSI reference model – the ISO model of architecture for Open Systems Interconnection. *IEEE Transactions on Communication*, 28(4):425–432, April 1980.

[ZMSS05] Ezra Zeheb, Oliver Mason, Selim Solmaz, and Robert Shorten. On the quadratic stability of switched interval systems: Preliminary results. In *IEEE International Symposium on Intelligent Control*, Limassol, Cyprus, June 2005.

[ZS06] Ezra Zeheb and Robert Shorten. A note on spectral conditions for positive realness of single-input-single-output systems with strictly proper transfer functions. *IEEE Transactions on Automatic Control*, 51(5):897–900, May 2006.