Non-Uniform Sampling in Statistical Signal Processing

Frida Eng

Thesis Cover — Application Snapshots

Human heart
`http://en.wikipedia.org/`
`  wiki/Image:Humhrt2.jpg`
Copyright released by
owner

Supernova 1987A in the
Large Magellanic Cloud, as
seen by the Hubble telescope.
Courtesy NASA and STScI
`http://hubblesite.org/`
In the public domain

Death valley, as seen from the
Space Shuttle's synthetic aper-
ture radar instrument.
Courtesy NASA/JPL–Caltech
`http://photojournal.jpl.nasa.`
`  gov/catalog/PIA01349`
In the public domain

Ultra sound
Copyright Frida Eng

Glowing sun
From LiU's picture data base

Oscilloscope
Thanks to Peter Johansson,
ES, ISY, LiU
Copyright Frida Eng

Tire
Copyright Frida Eng

Scania R470 flat nose truck.
`http://en.wikipedia.org/`
`  wiki/Image:Scania_R470`
`  _topline.JPG`
Licensed under Creative
Commons Attribution Share
Alike 1.0 License

Part of *Full Internet map, 22*
Nov 2003, with colors separa-
ting routers on different cont-
inents.
`http://www.opte.org/maps/`
Licensed under a Creative
Commons License

To Maja, who gives me perspective

Abstract

Non-uniform sampling comes natural in many applications, due to for example imperfect sensors, mismatched clocks or event-triggered phenomena. Examples can be found in automotive industry and data communication as well as medicine and astronomy. Yet, the literature on statistical signal processing to a large extent focuses on algorithms and analysis for uniformly, or regularly, sampled data. This work focuses on Fourier analysis, system identification and decimation of non-uniformly sampled data.

In non-uniform sampling (NUS), signal amplitude and time stamps are delivered in pairs. Several methods to compute an approximate Fourier transform (AFT) have appeared in literature, and their posterior properties in terms of alias suppression and leakage have been addressed. In this thesis, the sampling times are assumed to be generated by a stochastic process, and the main idea is to use information about the stochastic sampling process to calculate a priori properties of approximate frequency transforms. These results are also used to give insight in frequency domain system identification and help with analysis of down-sampling algorithms.

The main result gives the prior distribution of several AFTs expressed in terms of the true Fourier transform and variants of the characteristic function of the sampling time distribution. The result extends leakage and alias suppression with bias and variance terms due to NUS. Based on this, decimation of non-uniformly sampled signals, using continuous-time anti-alias filters, is analyzed. The decimation is based on interpolation in different domains, and interpolation in the convolution integral proves particularly useful. The same idea is also used to investigate how stochastic unmeasurable sampling jitter noise affects the result of system identification. The result is a modification of known approaches to mitigate the bias and variance increase caused by the sampling jitter noise.

The bottom line is that, when non-uniform sampling is present, the approximate frequency transform, identified transfer function and anti-alias filter are all biased to what is expected from classical theory on uniform sampling. This work gives tools to analyze and correct for this bias.

Preface

Is it ever easy to decide on the direction for a research project, that should span over several years, give enough challenges for the student, fully utilize the competence of the research group and give results to fill a PhD thesis? Probably quite a few professors can answer that question. I cannot.

This thesis is the result of a project that quite dramatically changed direction somewhere in the middle. Going from a study of a particular application, control of packet data networks, that lead to a theoretical analysis of issues with non-uniform sampling.

The mix is still quite nice, with a profound motivation for the deep theoretical work, although some might say that a closed loop is missing, since the theories are not applied to the original application. To those I say, you are very welcome to do this, but my focus is still more on the theoretical side, since the results are intriguing.

Whatever your concerns, I hope you enjoy the time spent with this work, otherwise it was not time well spent. One of my wishes is to, if only a tiny bit, express the joy that working with this kind of mathematics can be.

Frida Eng
February 2007
Linköping

Acknowledgments

I will here mention names that partly explains why this thesis is what it is. Maybe not the whole truth, but I hope you all know why you are mentioned. Otherwise, just go ahead and ask.[1]

I was born and raised by mom and dad, and I guess that is part of the explanation, although I am not sure how. Anyway, I love you both, and hope you can keep up for a few more lines. Thanks also to Lasse and Christine for taking such good care of my dear parents.

Although I hate to say it, Uncle Johan was a major source of inspiration for moving to Linköping, starting at Y and also aiming for a PhD diploma. Now, when I am grown up (!) I can even enjoy your company.

My professor Fredrik Gustafsson helped me to start the PhD journey, a long time ago: As has been the case for all our "common projects", the beginning was a bit shaky, but the final result is pretty good. Thanks for helping me find the way.

I don't know what this group would be without Lennart Ljung leading it. I agree with everything that has already been said in previous acknowledgments, but also: Thanks for knowing about pregnancy and family life and for not kicking me out when I broke your rib. And Ulla Salaneck, I think you have heard it all before, but I honestly would have been lost without your help.

The working environment would be quite awful without all the other colleagues, you never know what the coffee breaks or junkmail will bring. Memorable things are (not limited to): KRAV with Ragnar; sound with Johan S; history with Torkel; anything with Anna; Kent; coffee machines, gender issues and local news with almost everyone; and, of course, private girl talk with Henrik O.

A little more namedropping is required:
My co-supervisor Fredrik Gunnarsson always has time for questions, when he is here . . . Thanks for being supportive, especially for a confused first-year PhD student. Martin Enqvist has been a good friend ever since I moved into "his" town. I enjoy having you around although giving you a hard time about all the crosses in the margin. Gustaf Hendeby sighs as heavily as I do every time I have to knock on his door. I believe we make an excellent LaTeX/BibTeX-team by now. I only hope I can return any favor some day.

Except from the previous three, a couple of other people also read and improved parts of the thesis: Ola, Jonas J, Jonas G and David T were involved the last time I wrote a book, and Henrik O have helped this time.

Johanna and Linnéa: The fikas (and more) have been great, so thanks for starting our network. I hope I can stay in touch with . . . us. My out-of-the-office friends: I hope you are still with me, after this awfully long period of silence.

Finally, Anders, Thanks for trying to read between the lines. I love you!

---

[1]For those who don't know me: I occasionally use jokes and irony, and the people in question will understand. Don't worry, I don't offend my loved ones, if I don't have to.

Contents

All articles are removed due to copyright restrictions.

1

Introduction

In statistical signal processing, the sampling times are most often taken to be equally spaced. However, several applications indicate that non-uniform sampling is important. The major work performed on non-uniform sampling is for when the sampling times can be specified, and the signal processing community lacks tools to deal with standard issues like identification and decimation for signals sampled at non-uniform times. With a stochastic view, this thesis aims to fill this gap, and it provides tools to deal with errors induced by non-uniform sampling. Much is gained by studying a priori properties of frequency transforms and estimates, and the tools can be used for several signal processing problems.

1.1 Thesis Outline

For the reader not interested in sequential, but optimized, reading, hopefully this brief description of the different chapters and papers in the thesis can help.

To clearly motivate the need for theories on non-uniform sampling, several applications are described in Chapter 2. Both the reason for the sampling being non-uniform, and what kind of problems that need to be solved, are pointed at. Much has been done in the area of non-uniform sampling, and Chapter 3 sets a common notation and also sorts different research efforts in to categories, based on the type of problem that is solved. Here, also references to the coming contributions in the thesis are made.

Earlier work by the author, on modeling and control of packet data traffic, clearly showed the need for different solutions, given non-uniform sampling, and in Chapter 4 this work is presented together with pointers to the open problems that this resulted in. The main contributions of the thesis are described in Chapter 5, with chosen figures and equations to describe the results in the following

papers. Here, guidelines for future work are also given.

Each of the included papers investigate different aspects of non-uniform signal processing. Paper A contains theoretical analysis of a priori properties in the frequency domain, such as mean value and covariance of the frequency transform. Also, asymptotic analysis is possible when the number of measurements increases. The results separate the effect of the signal, the non-uniform sampling, and the finite number of samples in a nice way.

A slightly different problem is considered in Paper B, where the sampling times are corrupted by unknown noise, and the underlying model is desired. It is shown that, inclusion of the knowledge of the sampling procedure removes the bias in the identification.

Paper C considers resampling from a set of fast non-uniform samples, given by the application, to a slower uniformly sampled set, chosen by the user. This is done together with low-pass filtering, and the results from Paper A are used to perform analysis of the down-sampling procedure.

1.2   Contributions and Relevant Publications

The main contributions of the thesis are stated here, together with the relevant publications where the author of the thesis is the main contributor. Note that Mrs. Eng was Miss Gunnarsson before May 2004. As mentioned before, Section 5.1 further explains the contributions.

- Investigation of performance measurements in networks, overview and model-based improvement of a control problem in packet data networks. This has previously been published in:

    Gunnarsson, Frida; Gunnarsson, Fredrik; and Gustafsson, Fredrik (2001). TCP performance based on queue occupation. In *Towards 4G and Future Mobile Internet, Proceedings for Nordic Radio Symposium 2001*. Nynäshamn, Sweden.

    Gunnarsson, Frida; Gunnarsson, Fredrik; and Gustafsson, Fredrik (2002). Issues on performance measurements of TCP. In *Radiovetenskap och Kommunikation*. Stockholm, Sweden.

    Gunnarsson, Frida; Gunnarsson, Fredrik; and Gustafsson, Fredrik (2003). Controlling Internet queue dynamics using recursively identified models. In *IEEE Conference on Decision and Control*. Maui, Hawaii, USA.

    In the thesis, the control problem for packet data networks is described in **Chapter 4**.

- Stochastic analysis of frequency transforms based on signals with non-uniform sampling times. Derivation of first and second order moments, as well as asymptotic analysis is performed. The publications concerning this contribution are:

Gunnarsson, Frida; Gustafsson, Fredrik; and Gunnarsson, Fredrik (2004). Frequency analysis using nonuniform sampling with applications to adaptive queue management. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Montreal, Canada.

Eng, Frida and Gustafsson, Fredrik (2005a). Frequency transforms based on nonuniform sampling — basic stochastic properties. In *Radiovetenskap och Kommunikation*. Linköping, Sweden.

Eng, Frida; Gustafsson, Fredrik; and Gunnarsson, Fredrik (2007). Frequency domain analysis of signals with stochastic sampling times. In *IEEE Transactions on Signal Processing*. Submitted.

Paper A covers this analysis in the thesis.

- Identification in the frequency domain, with unknown sampling time jitter. The bias is removed by including knowledge about the sampling time jitter when performing frequency domain identification. The relevant publications are:

  Eng, Frida and Gustafsson, Fredrik (2005b). System identification using measurements subject to stochastic time jitter. In *IFAC World Congress*. Prague, Czech Republic.

  Eng, Frida and Gustafsson, Fredrik (2006). Bias compensated least squares estimation of continuous time output error models in the case of stochastic sampling time jitter. In *IFAC Symposium on System Identification (SYSID)*. Newcastle, Australia.

  Eng, Frida and Gustafsson, Fredrik (2007c). Identification with stochastic sampling time jitter. In *Automatica*. Provisionally accepted as regular paper.

  In the thesis, the identification approach is presented in **Paper B**.

- Analysis of down-sampling when sampling times are non-uniform, given the results in Paper A. The impact of the sampling times on the low-pass filter is derived and the estimate is proved to be asymptotically unbiased. Publications about this contribution are:

  Eng, Frida and Gustafsson, Fredrik (2007a). Algorithms for down-sampling non-uniformly sampled data. In *European Signal Processing Conference (EUSIPCO)*. Poznań, Poland. Submitted.

  Eng, Frida and Gustafsson, Fredrik (2007b). Down-sampling non-uniformly sampled data. In *EURASIP Journal of Advances in Signal Processing*. Submitted.

  Paper C is devoted to down-sampling in this thesis.

Part I

Background Theory and Application Overview

2

Applications

With relevant applications, theoretical work is easy to motivate. This chapter gives an overview of applications where non-uniform sampling occurs, and also states some of the problems that need to be solved in the respective cases. Section 2.1 is an overview of the applications and their problems, related to the work presented in this thesis. Then, Sections 2.2–2.7 describe the applications in more detail. These sections are aimed for readers with a particular interest in a certain area. Much of the presentation here is taken from previous work, see the given citations.

2.1   Introduction

A number of applications use non-uniform sampling inherently or by choice. They include:

- *Packet data traffic*, where calculations are done at the arrival of a data packet. This application is described in Section 2.2, and Chapter 4 addresses some of its problems.

- *Automotive applications*, where rotating toothed wheels are used to obtain measurements of angular velocity. These types of applications are described in Section 2.3.

- *Medical applications*, which often use human activities to decide on measurement, such as peaks in the electrocardiogram, ECG, or lung volume, see more in Section 2.4.

- *Radar*, where frequency estimation is used to detect movements by Doppler effects. The non-uniform sampling is introduced to increase performance and to avoid jamming, see Section 2.5 for more details.

Table 2.1: *Summary of applications with non-uniform sampling*

Application	freqest	ident	control	downsamp
packet data	X	X	X	X
automotive	X	X		X
medical	X	X		X
radar	X			
astronomy	X	X		

- *Astronomical time series*, which are collected during long time spans with weather conditions and equipment failures causing non-uniform spacing of the data. Section 2.6 describes the problem using a collection of datasets.

Different hardware can also induce non-uniform sampling, for example, due to clock imperfections, a few cases are described in Section 2.7.

Several signal processing actions can be of interest for systems or signals with non-uniform sampling. For example,

- Fourier analysis aims to find the main frequency content of a signal.

- Identification aims to find a mathematical model to describe the dynamics of a signal.

- Automatic control aims to vary an input to enforce a desired behavior of a signal.

- Downsampling is used, when the non-uniform samples are much closer than needed, to obtain the wanted information from a signal.

The relation between different applications and actions are given in Table 2.1, indicating potentially interesting areas in non-uniform signal processing for the respective application.

In this section, we have seen several examples of where we find non-uniform signal processing, also what kind of actions that can be of interest. The main focus of this thesis is to further analyze the different actions, without focusing on any particular application. The following sections include a more thorough description of the different applications, for the interested reader.

2.2   Packet Data Networks

Internet is one of the largest man-made systems in the world, and it is continuously growing. The algorithms controlling it are both situated at end-nodes (e.g., home PCs) and within the network (e.g., core routers). In routers, some of the algorithms controlling the data flows are based on packet data arrivals, and are at these instants assumed to collect measurements, calculate decision variables, and perform a suitable action. It is clear that data packets will arrive totally at random, and therefore several of the network calculations have to deal with non-uniform

Figure 2.1: *Tire with* 100 % *(left) and* 70 % *(right) inflation. Thanks to Peter Lindskog and Urban Forssell at Nira Dynamics AB.*

sampling. This is mainly done in a heuristic fashion, but many research efforts aim at improving both models and control and also try to measure at different (slower) time scales. In Chapter 4, these problems are exemplified and discussed further. Also, a thorough background to Internet and packet data traffic is given there.

2.3 Automotive Applications

In the vehicles of today, numerous sensors are included and massive amounts of data are analyzed for different purposes. For example, toothed wheels are inserted in the wheels with sensors registering when teeth pass the sensor, giving a non-uniformly sampled signal, affected by the wheel speed. The measurements were originally only recorded for the ABS system, and are now finding more uses. The sensor values are times t_m, that can be used for estimation of the angular velocity,

$$\hat{\omega}_m = \frac{2\pi}{L(t_m - t_{m-1})} \tag{2.1}$$

when there are L identical teeth. This angular velocity estimate can be used for analysis of, for example, tire pressure in cars, and non-round wheels in trucks.

2.3.1 Tire Pressure Monitoring

A Tire Pressure Monitor System, TPMS, was presented in Persson (2002b) based on the wheel speed sensor. This presentation is entirely based on the work in that publication.

Using the correct tire pressure is important, both for safety, economical and environmental reasons. In Figure 2.1, two tires with 100 % and 70 % inflation are shown. Lately, there is also laws in U.S. making it mandatory with TPMSs in new cars due to an increased number of fatal accidents with under-pressurized tires. Two types of TPMSs are available, direct and indirect.
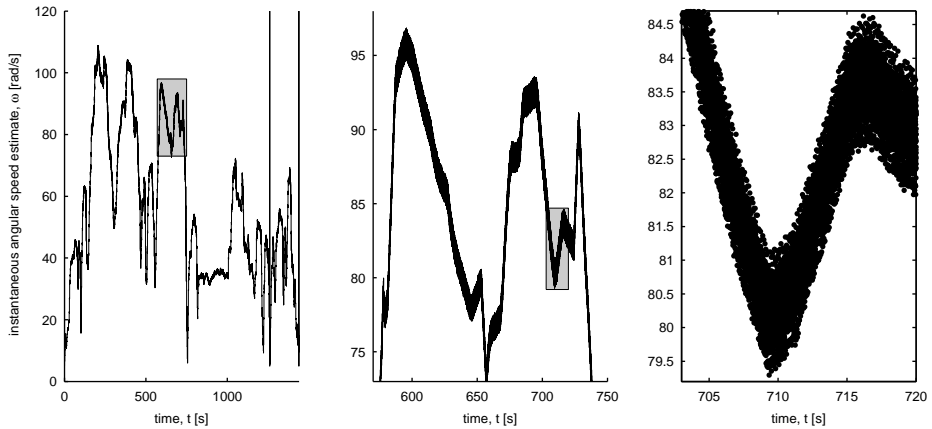
Figure 2.2: *Estimate of the angular velocity (2.1), given measurements from one wheel speed sensor of a car. Data is gradually zoomed in as indicated by the gray-shaded area.*

The direct system uses a pressure sensor mounted in the wheel, communicating with the central system of the car. This is expensive and requires new hardware in every car.

The indirect system utilizes the available sensors and estimates the tire pressure from them. One way is to perform wheel radius analysis, since a decrease in the tire pressure decreases the effective wheel radius.

Another way to perform indirect tire pressure monitoring is by vibrational analysis, since the rubber in the tire is excited from road roughnesses. The resonance frequency is highly dependent on the tire pressure, and can thus be used to detect changes in the pressure. This method is suggested in Persson (2002b). The algorithm corrects for sensor errors, i.e., unideal toothed wheels giving errors in (2.1), and converts the data from the event sampled sequence to a uniformly sampled one by interpolation, and then down-samples the data including low-pass filtering. It is argued that a combination of the two indirect TPMSs increases performance. Persson (2002b) also discusses interesting future estensions, which includes estimation directly from the non-uniform data, more efficient down-sampling, and spectral analysis of non-uniformly sampled data.

An example of measurements from one of the wheel speed sensors in a car is shown in Figure 2.2. In this particular case, the number of teeth are $L = 48$, and the average sampling interval is 2.3 ms over the whole interval.

2.3.2   Non-Round Wheels

The Master's thesis Nilsson (2007) studies detection of non-round wheels in trucks, with the purpose of reducing the oscillations in the cabin. Vibrations in truck cabins can be amplified when tires are non-round. At speeds around 90 km/h, the frequency from the non-round wheels coincide with the resonance

Figure 2.3: An example of an ECG curve. From `http://en.wikipedia.org/
wiki/Image:EKG2.png`. *Licensed under GNU Free Documentation License.*

frequency of the cabin, and this is experienced as very annoying. Garages have difficulties in measuring the roundness of wheels, and, since trucks can have about 10 wheels, changing and testing them one by one is a long process.

The non-round wheel is modeled as rotating around an offset, δ, from the hub, which gives a sinusoidal varying angular speed, ω, during one revolution of the wheel. Using several revolutions, the variations can be detected as a function of the angle of the wheel, θ, and the offset can be estimated, both by Fourier analysis and AR-modeling of the estimated radius, $r(\theta) = v/\omega(\theta)$, where v is the speed of the truck.

In this context, problems with imperfect sensors are also present, and conversion from non-uniform samples to a uniform grid is of interest.

2.4   Medical Applications

There are numerous applications with medical connections, and here we study the case of the electrocardiogram, ECG, which is used to monitor the electrical activity in a person's heart. An example of an ECG is given in Figure 2.3. There are times when the whole ECG curve is measured, both invasive and non-invasive, see for example, Wallin (2005) and Wikström (2005) for analysis of artifacts in the invasive case. It is also common to record the times of the highest peaks in the ECG, and use them for estimation (van Steenis and Tulen, 1991). Let t_m be the time of the peak of the mth heartbeat, the instantaneous heart rate is then

$$r_m = \frac{1}{t_m - t_{m-1}}, \tag{2.2}$$

which gives a non-uniform sampling of the heart rate, $r(t)$. This is very similar to the automotive case, cf. (2.1).

Healthy persons experience periodic fluctuations in the instantaneous heart rate, known as respiratory sinus arrhythmia, RSA. A decrease in the magnitude of the fluctuations normally indicates some kind of heart problem, and one of the main reasons for studying the heart rate variability is to predict the ability to recover after for example a heart attack.

In order to detect heart rate variability, the heart rate spectrum can be investigated, in particular its low frequency content. An accurate heart rate spectrum

can also be used to study the heart condition after a transplant, e.g., rejection probability (Sands *et al.*, 1989), or investigate patients with a problem known as the Guillain-Barre syndrome (Flachenecker *et al.*, 1997).

The heart rate is also used in order to study fetal health during both pregnancy and labour. For example, the spectrum can be used to assess the maturation of the Autonomous Nervous System (Romano *et al.*, 2003).

2.5   Radar

A radar is an electronic system used for detection, location and classification of objects. It uses remote sensing by emitting signals of a certain wavelength and detecting the echo signal. Many radar applications result in non-uniform data on a two-dimensional grid, see for example, Grönwall (2006) and Belmont *et al.* (2003), where the latter describes problems with switching to a uniform grid in order to perform spectral estimation for vessel movements.

One-dimensional radar measurements, e.g., long range radars or pulse radars use, frequency estimation to detect movements by Doppler effects. Usually, the wanted frequency is much larger than the possible Nyquist frequency, and therefore different non-uniform sampling techniques are of interest to increase performance (Alavi and Fadaei, 1994). Also, for military applications, detection and jamming of the radar is better avoided by the use of non-uniform sampling (Pribić, 2004).

2.6   Astronomy

In stellar physics, the luminosity of variable stars are recorded to describe their frequency contents, see Roques and Thiebaut (2003) and van der Ouderaa and Renneboog (1988). Measurements are done over long time spans, usually several years, and the measurement series are corrupted by both weather conditions and telescope failures. This gives long periods with missing data.

In Hertz and Feigelson (1997), a total of 13 datasets are described, giving an idea of the problems in astronomical time series. For example, detection of periodicities in both the number of sunspots and observed properties of other stars is an important task for astronomers. A glowing sun with its spots is shown on the cover of the thesis. The data is often non-uniformly sampled as well as sparse and noisy. It is believed that a majority of the stars are in what is known as binary systems, where two stars orbit each other, and this is one of the properties that have to be detected with the mentioned type of data. Measurements of the radial velocity of two different stars are shown in Figure 2.4, and it is obvious that the sampling is quite non-uniform.

An extremely important example in astronomy is the detection of neutrinos from the supernova explosion observed in 1987, SN1987A. One picture taken after the supernova explosion, by the Hubble telescope, is on the cover of this thesis. In this type of measurement the arrival times of the neutrinos, or other individual
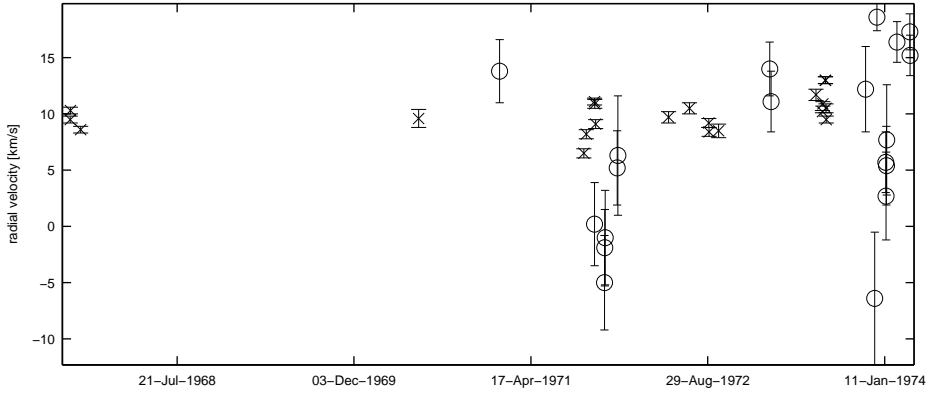
Figure 2.4: *Radial velocities of two different stars. Data is taken from Hertz and Feigelson (1997). Error bounds for the measurements are shown with vertical bars.*

particles, are recorded, and the result is non-uniformly spaced data, which often is relatively sparse. For example, only 20 neutrinos were recorded from the supernova explosion, SN1987A. Figure 2.5 shows the non-uniform arrival times of the neutrinos recorded by two sensors.

2.7   Hardware

Here are a few examples of where non-uniform sampling may appear due to hardware, to further stress the importance of the area.

2.7.1   A/D Converters

The thesis Elbornsson (2003) thoroughly describes different errors arising in interleaved A/D converters, A/D-Cs. Interleaved A/D-Cs are necessary in order to increase sampling rates. If N A/D-Cs are used to sample a signal, the nth A/D-C samples at time

$$t_{k,n} = (kN + n)T,$$

to get an overall inter-sampling time T. The results from the individual A/D-Cs are multiplexed to get the sampled sequence $y(kT)$. Individual A/D-Cs suffer from time errors, δ_n, due to the internal clocks, and the sampling clock also suffers from noise, which gives rise to jitter, $\tau_{k,n}$. The actual sampling time for A/D-C n is therefore,

$$t_{k,n} = (kN + n)T + \delta_n + \tau_{k,n}.$$

Here δ_n is constant for individual A/D-Cs and $\tau_{k,n}$ is random. The resulting sequence $t_{1,1}, t_{1,2}, \ldots, t_{1,N}, t_{2,1}, \ldots, t_{2,N}, \ldots$ is clearly non-uniform. The thesis states
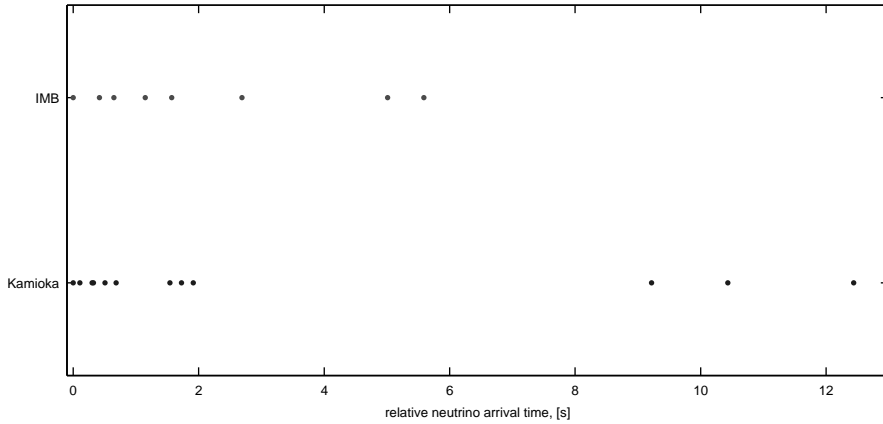
Figure 2.5: *Arrival times of the neutrinos after the supernova explosion, SN1987A. A total of* 20 *arrivals were measured by two units. Data is taken from Hertz and Feigelson (1997).*

methods to correct for the time errors, δ_n, and later work by Janik and Bloyet (2004) investigates the jitter errors, $\tau_{k,n}$, further.

2.7.2 Oscilloscope

In Verbeyst *et al.* (2006), a 50 GHz sampling oscilloscope is studied and its time base jitter is identified to have standard deviation 0.965 ps, which corresponds to approximately 80% of the sampling interval. The model is that the actual measurement time is corrupted by zero-mean Gaussian noise, $t_k = kT + \tau_k$, and the second order moment of τ_k is identified in experiments. This identification is important when other effects, such as amplitude distortion, of the oscilloscope is studied. For example, the paper shows that a bias of more than 10% is introduced on additive noise estimates, if the time base distortion is not compensated for.

3

Non-Uniform Signal Processing in Literature

Non-uniform, irregular, uneven, staggered and non-equidistant sampling in time are all names that have been given to the type of sampling this thesis deals with. This chapter presents the research problems and suggested solutions that exist in the area. The preceding Chapter 2 on applications gave an idea about what the different problems are, but here the research efforts are categorized, without any connection to specific applications. We start with two sections to define the notation that is used in this thesis: for the signal model in Section 3.1, and for non-uniform sampling in Section 3.2. The sampling is done at times t_m to get sample values y_m from a continuous-time signal $s(t)$, see Figure 3.1:

$$y_m = s(t_m). \tag{3.1}$$

The publications are then sorted in the following topics:

Frequency analysis described in Section 3.3 focuses on the following problem.

Problem 3.1. Given measurements y_m at times t_m, how do we best characterize the frequency content in the original signal, $s(t)$?

Reconstruction and estimation discussed in Section 3.4, can be stated as solving the following problem.

Problem 3.2. Given measurements y_m at times t_m, how do we find the best approximation of the original continuous-time signal, $s(t)$?

Optimal sampling investigated in Section 3.5, where focus is on solving the following problem.

Problem 3.3. Given a signal $s(t)$, how do we optimally place the sampling instants t_m, and what is optimality in this case?
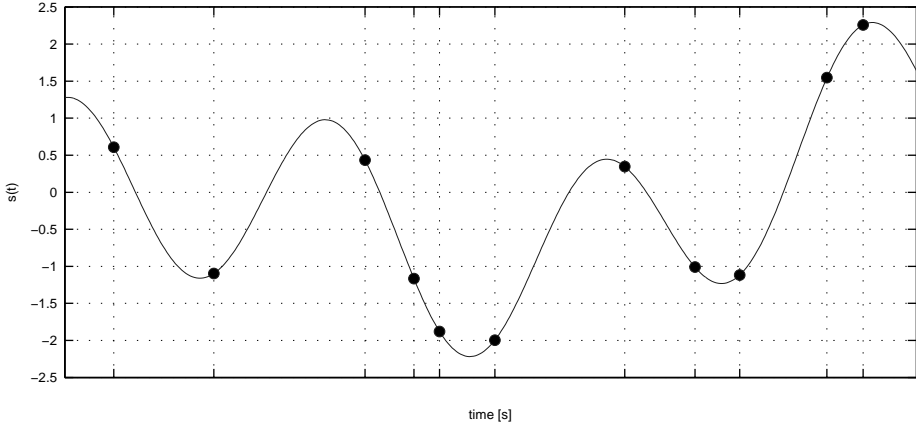
Figure 3.1: *A continuous-time signal, that is non-uniformly sampled.*

There is also work on fast implementations, Section 3.3.1, mainly for the Fourier transform.

Overviews of non-uniform sampling can be found in, for example, Papoulis (1977), Bilinskis and Mikelsons (1992) and Marvasti (2001), and these collections also cover parts of the ideas presented here.

3.1   Signal Model

We consider a deterministic continuous-time signal, $s(t)$, with Fourier transform $S(f)$,

$$s(t) = \int S(f)e^{i2\pi ft}\, df, \tag{3.2}$$

which is non-uniformly sampled M times. The sample times, t_m, are stochastic variables with probability density functions (pdfs),

$$p_m(t) = \begin{cases} \frac{d}{dt}P(t_m \leq t) & \text{continuous,} \\ P(t_m = t) & \text{discrete,} \end{cases} \tag{3.3}$$

for $m = 1, \ldots, M$, and the number of samples, M, is deterministic. From the continuous-time signal (or function) $s(t)$ we get a stochastic observation, y_m, of the corresponding deterministic sample value:

$$y_m = s(t_m). \tag{3.4}$$

One example of non-uniform sampling of a signal $s(t)$ is given in Figure 3.1.

The sample value, y_m, is a function of the stochastic variable t_m and its stochastic properties can be investigated accordingly, for example,

$$\mathrm{E}[y_m] = \mathrm{E}[s(t_m)] = \int s(t)p_m(t)\,dt,$$

$$\mathrm{Var}(y_m) = \mathrm{E}[s(t_m)^2] - \mathrm{E}[s(t_m)]^2 = \int s(t)^2 p_m(t)\,dt - \left(\int s(t)p_m(t)\,dt\right)^2.$$

Any transform of these stochastic function values or measurements, y_m, can be investigated to study its a priori properties, for example, for parameter estimation or frequency analysis. In this work, we consider the case of frequency analysis of a continuous-time signal, $s(t)$, sampled non-uniformly and then transformed to approximate the true Fourier transform, $S(f)$.

Sampling of stochastic processes will not be considered here, but all sensors give corrupt measurements, therefore

$$y_m = s(t_m) + e_m \tag{3.5}$$

is considered, where e_m is stochastic measurement noise, usually independent and identically distributed with zero mean. In some of the analyses, the additive measurement noise is not important, and the effects can be included afterwards. For example, by adding zero-mean measurement noise, we get $\mathrm{E}[s(t_m) + e_m] = \mathrm{E}[s(t_m)]$ and $\mathrm{Var}(s(t_m) + e_m) = \mathrm{Var}(s(t_m)) + \mathrm{Var}(e_m)$.

3.2   Non-Uniform Sampling

Non-uniform sampling can occur in different forms and this section lists the most common descriptions. The probability distribution for the sampling times, t_m, was given as $p_m(t)$ in (3.3). Depending on the type of sampling, $p_m(t)$ can be deduced from the pdf, $p_\tau(t)$ of the *sampling noise*, τ_m. In this work we use the sampling noise, τ_m, to construct the non-uniform sampling instants, t_m, based on different sampling models.

For *additive random sampling*, ARS, the sampling times are constructed by adding the sampling noise to the previous sampling time,

$$t_m = t_{m-1} + \tau_m = \sum_{k=1}^{m} \tau_k, \qquad\qquad t_0 = 0 \tag{3.6}$$

where $\tau_m \in (0, \infty)$ and $\mathrm{E}[\tau_m] = T$. This means that $\mathrm{E}[t_m] = mT$, while the variance increases with m. The pdf is given as a convolution of the sampling noise pdf m times,

$$p_m(t) = \underbrace{p_\tau \star \ldots \star p_\tau}_{m \text{ times}}(t) \triangleq p_\tau^{(m)}(t). \tag{3.7}$$

Table 3.1: *Summary of different sampling models, additive random sampling (ARS), stochastic jitter sampling (SJS) and missing data (MD).*

Type	Update	$E[t_m]$	$\tau_m \in$	$p_m(t)$
ARS	$t_m = t_{m-1} + \tau_m$	mT	$(0, \infty)$	$p_\tau^{(m)}(t)$
SJS	$t_m = mT + \tau_m$	mT	$(-T/2, T/2)$	$p_\tau(t - mT)$
MD	$t_m = t_{m-1} + \tau_m$	$> mT$	$\{T, 2T, \ldots\}$	—

For example, the exponential distribution, $p_\tau(t) = T^{-1}e^{-t/T}$, gives a Poisson sampling process. The central limit theorem gives that $p_m(t)$ will approach a Gaussian distribution when m goes to infinity, since it is the pdf of a sum of m independent identically distributed variables, c.f., (3.6). Additive random sampling is one of the considered sampling types in Paper A, while Paper C only uses this model to describe the sampling procedure.

For *stochastic jitter sampling*, SJS, the sampling noise is added to the expected sampling time,

$$t_m = mT + \tau_m, \tag{3.8}$$

with $\tau_m \in (-T/2, T/2)$ and $E[\tau_m] = 0$. In this case the variance is constant over time and the pdf is given directly by the pdf for τ_m,

$$p_m(t) = p_\tau(t - mT). \tag{3.9}$$

One natural distribution is the rectangular distribution, $p_\tau(t) = 1/T, \quad -T/2 < t < T/2$, but it is also possible to imagine a truncated Gaussian distribution or other bounded distributions. The sampling noise can both be known and unknown. Paper B considers the case of unknown jitter noise, while Paper A discusses known jitter noise as one of the sampling procedure models.

Another case is the problem with *missing data*, MD, where the underlying sampling procedure is uniform but sometimes samples are missed. This can, for example, be described with a discrete sampling noise,

$$t_m = t_{m-1} + \tau_m, \tag{3.10}$$

and $\tau_m \in \{T, 2T, \ldots\}$. This can be seen as a special case of ARS, with a discrete pdf for the sampling noise, for example, $p_\tau(nT) = P(\tau_m = nT) = p(1-p)^{n-1}$ gives a First success distribution. The expected value $E[t_m] > mT$ whenever $E[\tau_m] > T$, which is the case for every nontrivial pdf. Some notes on the missing data problem are given in Section 5.2. The missing data problem is also discussed in Ljung (1999, Ch. 14) with further references.

In Table 3.1, these sampling types are summarized with mean value of the sampling times, support of the sampling noise, the update equation for the sampling times, and the pdf.

3.3   Frequency Analysis

In Chapter 2 we saw that in many applications, such as radar, image processing, astronomy and cardiology, the frequency content in the sampled signal is of interest. Problem 3.1 is considered in the following literature.

Wojtkiewicz and Tuszyński (1992) have chosen to start from the z-transform,

$$Y[z] = \sum_{m=0}^{\infty} y_m z^{-m},$$ (3.11)

for sampled signals and construct the Dirichlet transform,

$$Y(x) = \sum_{m=1}^{M} y_m e^{-x t_m},$$ (3.12)

with $x = \sigma + i\omega$. This transform is argued to be better suited for analysis of non-uniformly sampled signals, since it preserves information about the time instants. The sampling is considered deterministic and the inverse transform is also derived. Only the case of jitter sampling is discussed in the analysis.

Lomb (1976) and Scargle (1982) use

$$y_m = a\sin(2\pi f(t_m - \tau)) + b\cos(2\pi f(t_m - \tau)) + v_m$$ (3.13)

as a model and from this uses least squares fitting to find a and b. The time shift τ is chosen such that

$$\tan(4\pi f \tau) = \frac{\sum_{m=1}^{M} \sin(4\pi f t_m)}{\sum_{m=1}^{M} \cos(4\pi f t_m)},$$ (3.14)

for easier computations, since it ensures that the cross-term

$$\sum_{m=1}^{M} \cos(2\pi f(t_m - \tau))\sin(2\pi f(t_m - \tau)) = 0, \qquad \text{for all } f.$$ (3.15)

This gives the periodogram

$$P_Y(f) = \frac{1}{M}\left[\left(\sum_m y_m \sin(2\pi f(t_m - \tau))\right)^2 + \left(\sum_m y_m \cos(2\pi f(t_m - \tau))\right)^2\right],$$ (3.16)

$$\approx (a^2 + b^2)$$

Comparing the Dirichlet transform (3.12), with $x = i2\pi f$, and the Lomb-Scargle periodogram, we get $P_Y(f) = |Y(i2\pi f)|^2/M$. Lomb and Scargle also perform probability calculations and correlation analysis between frequencies when the true signal is sinusoidal and the measurement noise is Gaussian.

In Qi *et al.* (2002) the same signal model is used but extended to a sum over several frequencies,

$$y_m = a_0 + v_m + \sum_k a_k \sin(2\pi f_k t_m) + b_k \cos(2\pi f_k t_m).$$ (3.17)

The coefficients a_k and b_k are then considered varying and are estimated recursively using the Kalman filter, for an a priori chosen set of frequencies f_k. This gives a useful algorithm when the frequency content varies over time, but there is no closed form expression like (3.16).

Estimation of the spectrum when y_m is given as samples from a stochastic process is given attention in three papers by Masry and co-authors. Masry and Lui (1976); Masry (1978b); and Masry *et al.* (1978) consider Poisson sampling, i.e., the sampling times are given from $t_m = t_{m-1} + \tau_m$ and τ_m is taken from an exponential distribution with mean β. First the spectrum is estimated using

$$P_Y(f) = \frac{1}{\pi \beta M} \sum_{k=1}^{\log M} \sum_{m=1}^{M-k} y_m y_{m+k} \cos(2\pi f(t_{m+k} - t_m)), \qquad (3.18)$$

and the estimate is shown to be asymptotically unbiased, when $M \to \infty$, for any value on β,

$$\mathrm{E}[P_Y(f)] \to \int_0^\infty \mathrm{Cov}(y(t), y(t + \tau)) \cos(2\pi f \tau) \, d\tau \qquad M \to \infty, \qquad (3.19)$$

when the sample values y_m are taken from a Gaussian process, $y(t)$. In the second paper the estimate is generalized with the inclusion of a window function so that

$$P_Y(f) = \frac{1}{\pi \beta M} \sum_{k=1}^{\log M} \sum_{m=1}^{M-k} y_m y_{m+k} w_M(t_{m+k} - t_m) \cos(2\pi f(t_{m+k} - t_m)), \qquad (3.20a)$$

$$w_M(t) = \int H(f) e^{i 2\pi f / \log M} \, df, \qquad (3.20b)$$

and $H(f)$ is *any* symmetric real-valued function scaled so that $w_M(0) = 1$. Also here, the bias and variance of $P_Y(f)$ are studied. Finally, the third paper compares estimation of the spectrum from uniform sampling to Poisson sampling, and in particular the case for finite sample sizes, as opposed to the asymptotic analysis performed in the first two papers. As mentioned the analysis is based on Poisson sampling, so the usefulness is limited when sampling times are given, see also Section 3.5.

Paper A studies the frequency analysis problem in this thesis, with the major contribution being a priori analysis of existing transforms. The Dirichlet transform seen above will then be thoroughly examined.

3.3.1 Fast Algorithms for the Fourier Transform

This thesis is not concerned with the speed of the implementation, although it is of course always of importance when algorithms are designed for real-life. For the uniform sampling case, the FFT is instrumental because of its good implementation performance. For non-uniform time sampling, new considerations have to be made. We mention two contributions to this area.

Algorithm 3.1 Fast Implementation of (3.21) (Potts *et al.*, 2001, Alg. 12.1)

Given: M, $\alpha > 1$, f_k, y_m, and $\phi(f)$.

1. Precompute

$$\phi(f_k - \frac{m}{\alpha M}) \qquad\qquad\qquad m = 1, \ldots, \alpha M,$$

$$c_k = \int \phi(f)e^{i2\pi f(k-1)} \, df, \qquad\qquad k = 1, \ldots, M,$$

2. Form

$$\hat{a}_k = \begin{cases} y_k/c_k, & k = 1, \ldots, M, \\ 0, & k = M+1, \ldots, \alpha M \end{cases}$$

3. Use the FFT to get

$$a_m = \frac{1}{\alpha M} \sum_{k=1}^{\alpha M} \hat{a}_k e^{-i2\pi km/\alpha M}, \qquad\qquad m = 1, \ldots, \alpha M,$$

4. Finally

$$\hat{Y}(f_k) = \sum_{m=1}^{\alpha M} a_m \phi(f_k - \frac{m}{\alpha M}), \qquad\qquad k = 1, \ldots, M,$$

is the approximation of the transform.

Potts *et al.* (2001) show a fast algorithm for computing an approximation of

$$Y(f_k) = \sum_{m=1}^{M} y_m e^{-i2\pi f_k m/M}, \qquad\qquad k = 1, \ldots, M, \qquad\qquad (3.21)$$

at frequencies f_k below the Nyquist rate. This is done by approximation of the transform as well as zero-padding. An oversampling factor $\alpha > 1$ is introduced and $Y(f_k)$ is approximated with

$$\hat{Y}(f_k) = \sum_{m=1}^{\alpha M} a_m \phi(f_k - \frac{m}{\alpha M}), \qquad\qquad (3.22)$$

where $\phi(f)$ is some function with the same period as $Y(f)$. One of the suggested algorithms is summarized in Algorithm 3.1, and the paper also shows how to use this when the non-uniform grid is instead in time domain. This is also extended to include non-uniform sampling in both time and frequency. It is shown that Algorithm 3.1 is considerably faster than computing (3.21) directly.

The characteristics of the computation error are investigated for several choices of the approximation kernel $\phi(f)$.

Another approach is taken by Mednieks (1999). In this approach the sampling times are assigned based on a fine grid and on average the *Nth* point is used,

$$t_m = t_{m-1} + (N + \tau_m)\delta, \tag{3.23}$$

called additive pseudorandom sampling. The randomness is given by the integer variable τ_m, $E[\tau_m] = 0$, and δ is the granularity of the time grid, which gives $E[t_m] = mN\delta$. This enables the use of the FFT algorithm on the extended sequence

$$\tilde{y}_k = \begin{cases} y_m, & k = t_m/\delta \\ 0, & \text{otherwise} \end{cases}, \qquad k = 1, \dots, MN, \tag{3.24}$$

to find the Fourier transform in a fast way. This method is improved for this special case of sampling in order to reduce alias effects and increase implementation speed further. The resulting transform is exactly the same as using the Dirichlet transform on the sequence y_m when t_m is given by (3.23).

3.4   Reconstruction and Estimation

Reconstruction of a signal is particularly useful for certain image applications, and for A/D and D/A conversions. Recovery of the sampled signal at specific time points are important for a wide range of applications, for example, when analyses are performed on a slower time scale than the original sampling. Here, Problem 3.2 is considered. The estimate or reconstruction of $s(t)$ is denoted $\hat{s}(t)$ in this presentation.

3.4.1   Basis Expansions

More than 40 years ago, Beutler (1966) promised that for signals $s(t)$ with a Fourier transform (even more general in fact), there exist functions $h_m(t)$ such that

$$s(t) = \lim_{M \to \infty} \sum_{m=1}^{M} s(t_m) h_m(t) \tag{3.25}$$

converges uniformly for a fixed t, when the number of measurements on a fixed interval increases. It remains to find the basis functions $h_m(t)$ and hope that the error when using a finite M is small enough. Several contributions follow in this direction.

Yao and Thomas (1967) discuss the existence of Lagrange expansions, where a signal can be represented as

$$\hat{s}(t) = \sum_{m=1}^{M} y_m h_m(t), \tag{3.26a}$$

$$h_m(t) = \frac{G(t)}{(t - t_m)G'(t_m)}, \tag{3.26b}$$

$$G(t) \propto \prod_m (1 - t/t_m). \tag{3.26c}$$

The requirements on the sampling sequence for this expansion to exist are given in the often cited non-uniform sampling theorem recapitulated in Theorem 3.1.

Theorem 3.1 (Non-Uniform Sampling Theorem (Yao and Thomas, 1967, Thm. 2))
All functions s(t), with frequency content strictly below 1/2T Hz, possess a sampling expansion given by

$$s(t) = \sum_{m=-\infty}^{\infty} s(t_m) h_m(t)$$

for each sampling sequence t_m fulfilling

$$|t_m - mT| < L < \infty,$$
$$|t_m - t_k| > \delta > 0, \qquad\qquad m \neq k,$$
$$m, k = 0, \pm 1, \pm 2, \ldots.$$

The expansion is given by (3.26).

When $L = 0$ the well-known Shannon expansion with sinc functions can be recovered from (3.26). Russell (2002) uses the same idea and develops a recovery algorithm to calculate $h_m(t)$ and, ultimately, $\hat{s}(t)$. Extensive analysis is performed to enable easy real-time implementation.

Benedetto (1992) uses frame theory[1] to derive reconstruction formulas for band-limited signals and specific assumptions on the sampling sequence, we refer to the article for details. A function $s(t)$, which is band-limited on $[-\omega, \omega]$, can be reconstructed with a basis function, $h_m(t) = h(t - t_m)$, which is band-limited on a wider band and has a frequency transform equal to 1 on $[-\omega, \omega]$, as follows

$$\hat{s}(t) = \sum_m a_m h(t - t_m), \tag{3.27}$$

where the parameters $a_m \approx K y_m$. The constant K is defined by the sampling times t_m, and an exact expression for a_m is also given in the article. This is a special way of defining the basis function $h_m(t)$.

Eldar (2003) considers a more abstract sampling procedure using inner product constructions and frame theory. For a special choice of sampling functions, $x_m(t)$,

---

[1]We refer to Benedetto (1992); Eldar (2003) and references therein for an introduction to frames.

this approach gives an alternative way of computing the coefficients, a_m, in (3.27). First, assume that the samples are given by an inner product with the original signal, and define X^* such that

$$y_m = \langle x_m, s \rangle, \qquad \Longleftrightarrow \qquad y \triangleq X^* s. \qquad (3.28)$$

Then, also define W so that

$$\hat{s}(t) = \sum_m a_m w_m(t) \triangleq aW. \qquad (3.29)$$

The coefficients a_m collected in the vector a are then found using a pseudo inverse,

$$a = (X^* W)^\dagger y. \qquad (3.30)$$

This can be seen as an orthogonal projection of $s(t)$ on the space spanned by the measurements. Choosing $x_m(t) = \delta(t - t_m)$ and $\langle x_m, s \rangle = \int s(t) x_m(t)\, dt$ gives a perfect match with our usual understanding of sampling, i.e., $y_m = s(t_m)$. The only thing left is to interpret X^* and W accordingly, which might be non-trivial.

3.4.2   Iterative Solutions

Marvasti *et al.* (1991) consider stable sampling sets, t_m, and band-limited signals, $s(t)$. Stable sampling sets uniquely define the sampled signal (see for example Marvasti, 1987). The reconstruction of $s(t)$ is done recursively, by defining the sample operator, S,

$$\mathrm{S}\, x(t) = \sum_m x(t_m) \delta(t - t_m), \qquad (3.31)$$

giving a train of impulses. The kth recursion, $s^{(k)}(t)$, is then

$$s^{(k+1)}(t) = \lambda\, \mathrm{PS}\, s(t) + (\mathrm{P} - \lambda\, \mathrm{PS}) s^{(k)}(t), \qquad (3.32)$$
$$= \mathrm{P}\left(s^{(k)}(t) + \lambda\, \mathrm{S}\left(s(t) - s^{(k)}(t) \right) \right) \qquad (3.33)$$

Here P is the ideal band-limiting operator, so that $s(t)$ and $s^{(k)}(t)$ have the same bandwidth for all k. Marvasti (1996) studies this reconstruction further, for the special case of $t_m = mT + \tau_m$ with bounded τ_m.

   Feichtinger and Gröchenig (1994) give explicit iterative algorithms for reconstruction of signals band-limited to $[-\omega, \omega]$, and also state an upper bound on the reconstruction error. One example is the Adaptive Weights Method, which requires oversampling. This method is investigated further in Feichtinger *et al.* (1995), and here a specialized implementation algorithm is given for polynomials of degree R and period 1,

$$s(t) = \sum_{k=-R}^{R} a_k e^{i2\pi kt}. \qquad (3.34)$$

Algorithm 3.2 presents a condensed version of the algorithm, with $\langle a, b \rangle$ being the inner product of two vectors. With some effort, it might be possible to extract a fast algorithm for calculation of the Dirichlet transform from this as well. The paper also includes a numerical comparison of several reconstruction algorithms.

Algorithm 3.2 Fast Reconstruction (Feichtinger *et al.*, 1995, Thm. 1)

Let R be the polynomial degree (3.34) and let $0 \leq t_1 < \ldots < t_M < 1$ be an arbitrary sequence of sampling points with $M \geq 2R + 1$. Set $t_0 = t_M - 1$, $t_{M+1} = t_1 + 1$ and $w_m = (t_{m+1} - t_{m-1})/2$ and compute

$$\gamma_k = \sum_{m=1}^{M} w_m e^{-i2\pi k t_m}, \qquad\qquad k = 0, 1, \ldots, 2R, \qquad (3.35a)$$

$$b_k = \sum_{m=1}^{M} s(t_m) w_m e^{-i2\pi k t_m}, \qquad\qquad |k| \leq R. \qquad (3.35b)$$

Let $b = (b_{-R}, \ldots, b_R)$ and T be the matrix with elements $\{T\}_{l,k} = \gamma_{l-k}$, for $|l|, |k| \leq R$. Initialize $r^{(0)} = q^{(0)} = b$ and $a^{(0)} = 0$. From $n \geq 1$, iteratively compute

$$a^{(n)} = a^{(n-1)} + \frac{\langle r^{(n-1)}, q^{(n-1)} \rangle}{\langle Tq^{(n-1)}, q^{(n-1)} \rangle} q^{(n-1)}, \qquad (3.36a)$$

$$r^{(n)} = r^{(n-1)} - \frac{\langle r^{(n-1)}, q^{(n-1)} \rangle}{\langle Tq^{(n-1)}, q^{(n-1)} \rangle} Tq^{(n-1)}, \qquad (3.36b)$$

$$q^{(n)} = r^{(n)} - \frac{\langle r^{(n-1)}, Tq^{(n-1)} \rangle}{\langle Tq^{(n-1)}, q^{(n-1)} \rangle} q^{(n-1)}. \qquad (3.36c)$$

After $N \leq 2R + 1$ steps $Ta^{(N)} = b$ is solved, and

$$\hat{s}(t) = \sum_{k=-R}^{R} a_k^{(N)} e^{i2\pi k t} \qquad (3.37)$$

gives the reconstructed function.

3.4.3   Other Reconstruction Cases

Masry and Cambanis (1981) focus on estimation of a signal, $s(t)$, that is corrupted by a static nonlinearity, $f(x)$, for example, the sign-function, and show convergence when the sampling frequency increases. The reconstruction is done by adding noise to the samples,

$$y_m = f(s(mT) + e_m). \qquad (3.38)$$

It is assumed that the signal is limited by a known constant $|s(t)| \leq b$ and that the distribution of e_m is chosen wisely, for example a rectangular distribution $e_m \in Re[-b, b]$. Masry and Cambanis describe some cases when a static function $g(x)$ can be used to find $s(t)$ from an estimate of the mean

$$m(t) = E[f(s(t) + e)]. \tag{3.39}$$

It is shown that $g(\hat{m}(t)) \rightarrow s(t)$ when $T \rightarrow 0$, and the convergence properties are also discussed.

In Souders *et al.* (1990), the main goal is to study effects of timing jitter, and $s(t + \tau)$ is studied when $p_\tau(t)$ is given. It is shown that the mean is a biased estimator of $s(t)$. This is further discussed in Paper B, where also the effects of a finite number of samples are considered.

In the publication pair Kybic *et al.* (2002a,b), generalized sampling in higher domains is discussed. For this problem, the theory boils down to a cubic spline minimizing the second derivative of the reconstructed function. The result is known from previous spline literature, and both publications give extensive references to other work. The resulting reconstruction is given by defining

$$\hat{s}(t) = a_0 + a_1 t + \sum_{m=1}^{M} \lambda_m |t - t_m|^3, \tag{3.40}$$

and then solving the linear equation system

$$\hat{s}(t_m) = y_m, \tag{3.41a}$$

$$\sum_m \lambda_m = 0, \tag{3.41b}$$

$$\sum_m \lambda_m t_m = 0, \tag{3.41c}$$

for a_0, a_1 and λ_m.

A special case of reconstruction is resampling where only certain values of the underlying function, $s(t)$, are sought. Paper C discusses this when the task is down-sampling and low-pass filtering of the sampled sequence. The algorithms in this section are then too complicated or have too hard requirements on the sampling sequence, t_m, to be of use. Therefore, a more practical approach is taken.

3.5   Optimal Sampling

Optimal sampling is possible when the sampling points can be chosen before hand. It can be of importance, for example, for anti-jamming in radars, suppression of alias frequencies in the frequency transform, and placement of sensors for spatial sampling. Examples of this have already been given in previous sections, yet some more specialized ones exist. Here, Problem 3.3 is considered.

This aspect of non-uniform signal processing is outside the scope of this thesis, and the research is therefore only briefly presented. The most common optimality

criterion is suppression of alias frequencies, also known as alias-free sampling found in Section 3.5.2, but other contributions concerning the benefits of non-uniform sampling exist as well.

3.5.1   Benefits from Non-Uniform Sampling

Bilinskis and Mikelsons (1992) present several aspects on randomization in the sampling procedure, for example, that correlation between inter-sampling times can be beneficial. Work similar to Lemma A.1 given later in Paper A are also presented.

Jacod (1993) investigates an estimation problem for a stochastic process and shows an optimal sampling procedure in the sense of maximizing the Fisher information for all the parameters. The interest is in asymptotic properties as the number of sampling instances, M, tends to infinity.

Bland and Tarczynski (1997) empirically motivate non-uniform sampling and give some user guidelines on sampling time placements.

In Papenfuss *et al.* (2003), an algorithm for hardware implementation of jittered sampling ($t_m = mT + \tau_m$, where τ_m is a random process) promises 40 times the bandwidth of the corresponding uniform sampling process, $\tau_m = 0$.

3.5.2   Alias-Free Signal Processing

Digital alias-free signal processing (DASP) and alias-free sampling (AFS) are investigated in numerous works. The original mentioning of DASP methods can be found in Shapiro and Silverman (1960) and the methods have been enhanced after that. The main idea is the ability to choose placement of sampling points in order to reduce or remove aliasing. Shapiro and Silverman (1960) give a condition for when the sampling is alias free. Given additive random sampling,

$$t_m = t_{m-1} + \tau_m, \tag{3.42}$$

the requirement is that the characteristic function $\varphi_\tau(f) = \mathrm{E}[e^{-i2\pi f\tau}]$, for the sampling noise, τ, should be one-to-one on the real axis.

Beutler (1970) discusses AFS for a specific class, S, of spectra, and gives a similar, but more general definition than Shapiro and Silverman. The sampling sequence, t_m, is alias-free with respect to the class S if no two random processes with different spectra yield the same correlation sequence

$$r[n] = \mathrm{E}[y_{m+n}y_m]. \tag{3.43}$$

This will also correspond to demands on $\varphi_\tau(f)$, namely: The sampling sequence is alias free if

$$\int_{-\infty}^{\infty} \varphi_\tau(f)^n h(f)\,df = 0, \text{ for } n = 1, 2, \ldots, \qquad \Rightarrow \qquad \int h(f)\,df = 0. \tag{3.44}$$

Note that the correlation sequence is formed independently of the actual sampling times, t_m. Here, several examples of alias-free sampling time distributions are given, for example, it is shown that a uniform sequence where each sample is missing with probability $q < 1$, is alias-free if the sampled signal $s(t)$ is band-limited to the Nyquist frequency (then the uniform sequence, mT, is also alias-free). It is also noted, that a formula for finding the spectra is still missing, or was at the time, see Section 3.3.

The definition from Shapiro and Silverman is explored further in Masry (1978a) together with a new definition of AFS, which is ensured when

$$g_\infty(t) = \sum_{m=1}^{\infty} p_m(|t|) > 0, \text{ almost everywhere,} \tag{3.45}$$

where $p_m(t)$ is the probability density function for the sampling time t_m. The definitions concern the ability to separate the covariance sequence of the processes with different spectra, when sampled using a certain sampling sequence. The sequence t_m, is assumed to be stationary, as well as the sampling intervals, $t_m - t_{m-1}$. Only the sampling intervals were assumed stationary in the previous definitions. It is also shown here that the two definitions by Shapiro and Silverman and Masry do not coincide, and Masry argues that the second one is more practical for reconstruction purposes.

Bland and Tarczynski (1997) find the optimum conditions on sampling time placements for maximum alias frequency suppression.

Vandewalle *et al.* (2004) investigate the advantages of aliasing at image reconstruction. Here, two uniformly sampled sets are used, first $t_m = mT$ and second $t_m = mT + t_1$ where t_1 is a small shift. The resulting two sequences are then combined to enhance image quality.

Tarczynski and Allay (2004) study two ways of incorporating a window, $w(t)$, in the Dirichlet transform. First, the sampling times are uniformly distributed over $[0, t_M]$,

$$p_m(t) = \frac{1}{t_M}, \qquad t \in [0, t_M], \tag{3.46}$$

and

$$Y(f) = \frac{t_M}{M} \sum_{m=1}^{M} y_m w(t_m) e^{-i2\pi f t_m} \tag{3.47}$$

giving an unbiased estimate of $\int s(t) w(t) e^{-i2\pi f t} \, dt$. Second, the window is used in the placement of the sampling times,

$$p_m(t) = \frac{w(t)}{A t_M}, \qquad t \in [0, t_M], \tag{3.48}$$

where the constant A ensures $\int p_m(t) \, dt = 1$. The transform

$$Y(f) = A \frac{t_M}{M} \sum_{m=1}^{M} y_m e^{-i2\pi f t_m} \tag{3.49}$$

is also an unbiased estimate of $\int s(t)w(t)e^{-i2\pi ft} \, dt$. The authors also discuss the purpose of DASP without formal definitions of AFS. Note that the probability distribution is the same for all sampling times ($p_m(t) = p_\tau(t)$) and they are independent of each other, as opposed to the case with ARS described previously.

4

Packet Data Traffic − A Motivating Example

Packet data traffic on the Internet was discussed in Section 2.2 as one application that could benefit from non-uniform signal processing. This chapter will explore this further by a performance study, and a test of improved controllers, for the active queue management, AQM, problem.

This presentation is aimed for the reader interested in this particular application or in more bridges between applications and the theoretical work given in later papers. This chapter does not present solutions to the non-uniform sampling problems in the application, but exemplifies and discusses them. The presentation is based on Gunnarsson *et al.* (2002), Gunnarsson *et al.* (2003) and Gunnarsson (2003). It also includes a summary of the problems in non-uniform sampling that arise for this application, together with references to the relevant papers in the thesis.

By controlling network flows or network queue lengths, using AQM, the performance of a network can be improved. This work focuses on the flow level, and is based on current solutions, but the ideas easily carry over to independent settings. The approach is system theoretic, which is fairly new for network research. Several contributions have emerged that study network problems including queue management from a system theoretic viewpoint, e.g., Altman *et al.* (2000); Park *et al.* (2003); Hollot *et al.* (2001); Low *et al.* (2002); Kunniyur and Srikant (2004); Jacobsson and Hjalmarsson (2006).

The rest of the chapter is organized as follows. Section 4.1 gives relevant background information on the complex Internet system. Section 4.2 starts by defining performance for a network and discusses performance assessments based on network queue lengths, together with an overview of performance measures used by other researchers. Section 4.3 develops models based on filtered measurements of the queue length. The models are recursively identified to adopt to varying network settings. In Section 4.4, various controllers are described and evaluated.

The discussions are accompanied by explicit examples on a dataset from the net-work simulator, *ns-2*(NS, 2003, ver. 2.1b8a). Finally, Section 4.6 summarizes the non-uniform signal processing problems and discusses how a solution could be found.

4.1   Background on Packet Data Networks

Usually when studying Internet issues, the focus is on a certain level in the system or on a particular problem. To understand the complexity of controlling the traffic, an understanding of the whole picture is needed. Different levels interact and counteract in their demands on the communication.

Design of new algorithms require evaluation of the performance over the full network. Simulations can be done with different open source simulators, such as *ns-2* (Fall and Varadhan) and REAL (Keshav, 1997), or using proprietary environments, for example, ULC-1000AN developed by OPNET (OPNET, 2007). In order to make a good design, on the other hand, models of different components and how they affect the performance of the specific task are needed.

Due to the complexity of the network, simplifications at different levels are made to facilitate easier analysis. When designing for a certain layer, it is common to model lower layers as a delay, possibly varying, and a queue. In most cases, this gives a sufficient accuracy, but the behavior of the variation can be difficult to model and is strongly affected by the specific underlying protocols. The design parameter for the queue is the maximum length and the service rate which mostly is affected by the underlying topology.

Standards and proposals in the network architecture are described in Requests For Comments, RFCs. Anyone can write an RFC and they have different categories such as standards track, informational, draft and experimental. As the name suggests, anyone can have opinions on the content, and originally the aim was to make Internet a product of consensus. More information about available RFCs can be found at `http://rfc-editor.org`, where also information about how to publish new ones are available.[1]

4.1.1   Network Preliminaries

To understand the structure of the network application, some background knowl-edge is necessary. First, a short overview of some of the key events during the first years of computer network history is given. Then, the building blocks of the Internet communication system, the layers and their functionality, are discussed.

Internet History

To give an overview of the time horizon for the Internet evolution, some of the key events are presented here in chronological order. A continuously developing

---

[1]Note that there are also a number of April Fool's Day RFCs that can be entertaining to read

time line can be found in Zakon (2003), where also more references to and copies of the original papers are listed.

In *1961*, the first paper on packet switching, which was written by Leonard Kleinrock, was published and later on there was also a book on the subject by the same author. In *1962*, a series of memos from J.C.R. Licklider were published, envisioning a globally interconnected set of computers, where everyone could access data from any site. In *1965*, the first wide-area network connecting two computers was built by Lawrence Roberts. This experiment showed that packet switching was needed and circuit switching was totally inadequate. In *1966*, several papers on packet networks were presented at a conference and, in *1968*, the design of the first packet switches started at a small firm called BBN. Besides the team at BBN lead by Frank Heart, Robert Kahn was involved in the overall network architectural design, network topology and economics were designed and optimized by Roberts, and the network measurement system was prepared by Kleinrock's team at UCLA.

The first node in the network was installed at UCLA in *1969*. By the end of the year four hosts,at the universities in Stanford, Santa Barbara and Utah, were connected together. In *1970*, the first host-to-host protocol, Network Control Protocol (NCP), was designed and after the final implementation, the users of the network could start with application implementations. In *1972*, the network was demonstrated to the public by Roberts and electronic mail was introduced. In *1973*, the design of what was to become the protocol suite TCP/IP (Transmission Control Protocol/Internet Protocol) was started by Robert Kahn and Vinton Cerf. Much more information including references to the mentioned publications can be found also in Leiner *et al.* (2000, Sec. Origins of the Internet).

Layer Structure

To manage the large system of interacting computers, the information flow on the Internet is structured into layers. Each layer is responsible for a certain task and does not know anything about how layers above or below carry out their tasks. The actual data traverses the layer stack top down at the sender but the layer structure is transparent and information is seemed to be carried between the corresponding layers at the sender and the receiver using headers attached to the data. Every packet contains a header with information about the packet and data, which is delivered. Every layer adds a new header with information relevant for the corresponding layer at the receiver. The task of each layer is carried out using different *protocols*. Protocols are rules on how to handle files and packets at different layers. Different layers can use different protocols, depending on the type of transmission.

─── **Example 4.1: Data Transfer** ──────────────────────────────────────

When the goal is to deliver a file from sender A to receiver B, different things happen at different layers. Both horizontal and vertical communication take place. The structure is described in Figure 4.1 together with names of some of the protocols for each layer.
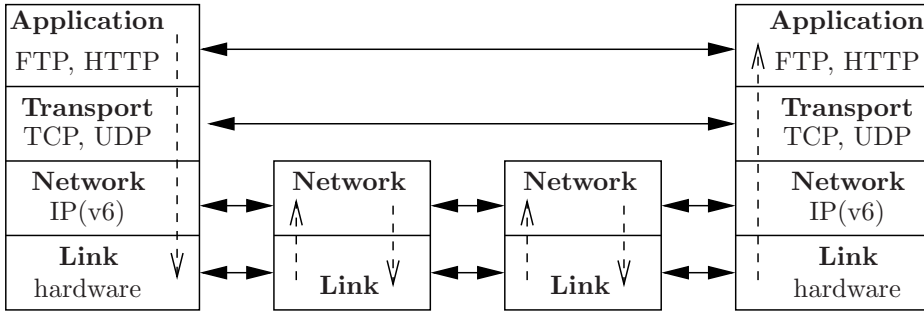
Figure 4.1: *The explicit data transfer (dashed) and the communication (solid) in the layered network structure. Each bar represents a physical position in the network, such as routers and servers.*

Application A file is sent from A to B. When the whole file is successfully delivered a notification is sent to the sending application. The application delivers the IP address and port number to be used to the layer below.

Transport The file is separated into packets, each with a header stating the size of the packet and the port to which it should be delivered. After each packet is successfully delivered an acknowledgment is received, otherwise it is retransmitted. The IP address is forwarded downwards.

Network The packet is provided a header with the source and destination address, the IP addresses of A and B respectively, producing an IP frame. After each frame is sent, the next one is prepared. The address of the next computer in the network is determined and delivered to the next layer.

Link The frame is given a header with the local address of the next computer and delivered on the physical link. The physical link can be both a wire and a radio link. In the latter case, a number of internal layers are used to enhance the communication over the radio link.

In this case the delivery of the file is always successful since the transport layer makes sure that lost packets are retransmitted. For other application types, for example, real-time transmissions, this might not be a desired behavior and the correction can instead be made in the application, e.g., by using redundant data, coding or retransmissions.

In Figure 4.2, the principal components of a network are shown. The senders are at the ends and the packets traverses links and queues to the receiver. The receipts or acknowledgments, ACKs, go the other way. The setup can be compared to Figure 4.1 where the link and network layer exists in all nodes but the higher layers are only present in the sources and destinations.
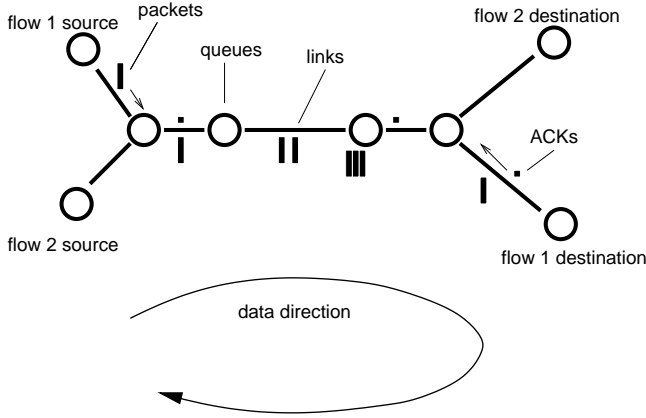
Figure 4.2: *A simple network with packets and receipts, queues and links. Each end node contains all the layers and the core nodes contains the two lower layers, see Figure 4.1.*

A generic layer provides a service to the upper layer and uses a service from the lower layer. Decisions are made on whether to forward additional data downward based on quality measures. The lower layer is thus fed with a varying data rate and the lower layer needs to try to provide its service fast enough.

This four layer model belongs to the TCP/IP protocol suite, which is a collection of protocols for all the different layers. In Stevens (1994, Ch. 1), a more formal description of the layers is given. The different parts are described further in the following sections.

4.1.2   Flow Control

TCP includes mechanisms for control of the send rate in order to interact with all traffic and prevent overloading or congestion of the network. As shown previously in Figure 4.2, the flow control performed by TCP resides in the end nodes, senders and receivers. The sender transmits data in packets and the receiver transmits ACKs in returning packets. Both the sender and the receiver use TCP mechanisms. At the packet arrival times, t_k, control variables are calculated. TCP keeps an internal window, the congestion window c_k, which is adapted as a measure of the capacity of the network, in bytes or packets. TCP also keeps track of the amount of data that is sent but not yet acknowledged, the outstanding data or flight size, $f(t)$. The received advertised window, a_k, which is delivered with the arriving packet, is the capacity of the receiver. These three, c_k, $f(t)$ and a_k, are used to limit the outgoing send rate, $r_{TCP}(t)$,

$$r_{TCP}(t) = \min(c_k - f(t), a_k), \text{ for } t \geq t_k \qquad (4.1)$$

The congestion control algorithm is specified in RFC 2581 (1999). It states the updating mechanism for c_k based on acknowledged data. A threshold, s, should be used to separate different parts of the update, slow start and congestion avoidance. During slow start, $c_k < s$, c is increased with the size of one packet at the arrival of a new ACK. When $c_k \geq s$, the congestion avoidance phase is entered and the update is done with approximately one packet for each round trip time period, T_{RT}, RTT. The round trip time is a measure of the time it takes from a packet is sent until it is acknowledged. The flight size, $f(t)$, was defined as the sent but not yet acknowledged data and can therefore be calculated as the sent packets during the last round trip time

$$f(t) = \int_{t-T_{RT}}^{t} r_{TCP}(\tau)\, d\tau. \tag{4.2}$$

Representing c and s in packets, rather than bytes, we get

$$c_k = \begin{cases} c_{k-1} + 1 & c_{k-1} \leq s \\ c_{k-1} + \frac{1}{c_{k-1}} & c_{k-1} > s \end{cases}. \tag{4.3}$$

There are two ways of noticing a failure of delivery for TCP, timeout and dupacks.

Time out Time out occurs when one packet's individual RTT is higher than a constant times the mean RTT. This indicates a traffic jam and several lost packets and therefore c is reinitialized. How to calculate and measure the RTT was described in Gunnarsson (2003, App. 2.A).

Dupacks The ACKs inform the sender about expected deliveries. Therefore, subsequent ACKs with the same number indicate a loss. TCP interprets three duplicate ACKs, dupacks, as a single packet loss.

The threshold, s, is also updated when a data loss is detected in one of the ways described above. The updates of c and s after a congestion is detected are given by

$$c_k = \begin{cases} 1 & \text{time out,} \\ s & \text{3 dupacks,} \end{cases}$$
$$s = \max(f(t)/2, 2). \tag{4.4}$$

The TCP algorithm is summarized in Algorithm 4.1.

The specification of TCP has evolved over time and the one described here is the most used, called TCP Reno. Later version have more refined the mechanisms that trigger loss recoveries and also the updating of c_k during congestion avoidance. A short overview of these updates were given in Gunnarsson (2003, App. 2.A).

Algorithm 4.1 TCP Flow Control

At each arrival time, t_k, the current congestion window, c_k is calculated as

$$c_k = \begin{cases} 1 & \text{time out,} \\ s & \text{3 dupacks,} \\ c_{k-1} + 1 & c_{k-1} \leq s, \\ c_{k-1} + \frac{1}{c_{k-1}} & c_{k-1} > s. \end{cases}$$

The slow-start threshold, s, is updated on congestion according to

$$s = \max(f(t_k)/2, 2).$$

The flight size, $f(t)$, changes continuously as

$$f(t) = \int_{t-T_{RT}}^{t} r_{TCP}(\tau) \, d\tau.$$

The send rate, r_{TCP}, is given by

$$r_{TCP}(t) = \min(c_k - f(t), a_k), \text{ for } t \geq t_k,$$

where a_k denotes the received advertised window.

4.1.3 Active Queue Management

The network layer protocols are present in all nodes in the network, for example in the ones shown in Figure 4.2. Active queue management (AQM) is a set of suggested additions to the network layer. The protocols use dropping and marking to inform TCP (explicitly or implicitly) about congestion dangers. Since TCP lowers its send rate, r_{TCP}, when packets are lost, early dropping could be used to gently lower the send rate instead of causing a time out. This is supposed to attenuate oscillations in the queue length, prevent overflow and minimize empty queue time.

Random Early Detection (RED) is one AQM suggestion, that uses a simple dropping profile based on the current queue length to decide which packets should be dropped. The longer the queue, the higher the probability of dropping an arriving packet. This technique was introduced in Floyd and Jacobson (1993) and has been widely adopted for further research. More specifically, when a packet arrives at the router at time, t_k, a filtered value, y_k, of the queue length, q_k, is calculated as

$$y_k = (1 - \lambda)^m y_{k-1} + \lambda q_k. \tag{4.5}$$

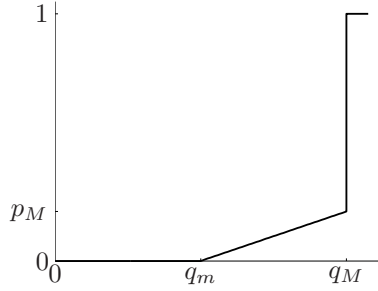Here, $m = 1$ if $q_k > 0$ and otherwise it is used to account for the time the queue

Figure 4.3: *The probability of dropping a packet as a function of the filtered queue length. The parameters q_m, q_M and p_M are design variables.*

was zero. If r is the service rate of the queue,

$$m = \begin{cases} 1, & q_k > 0, \\ (t_k - t_e)r, & q_k = 0. \end{cases}$$

The time t_e is used to denote the time when the queue became empty, i.e., the smallest t_e such that $q(t) = 0, t_e \leq t \leq t_k$. The probability, p, of dropping the new packet is then calculated as

$$p_k = \begin{cases} 0, & y_k \leq q_m, \\ \frac{p_M}{q_M - q_m}(y_k - q_m), & q_m < y_k \leq q_M, \\ 1, & y_k > q_M, \end{cases} \tag{4.6}$$

i.e., according to a linear function between the constants q_m and q_M. The constant $0 < p_M < 1$ is suggested to be in the order of 0.1 for good performance. An example of p_k as a function of y_k is given in Figure 4.3 and Algorithm 4.2 summarizes the actions.

Explicit Congestion Notification (ECN) is described in RFC 3168 (2001). It is a technique that enables marking of packets instead of dropping, by using the TCP header. This requires changes in the TCP implementation as opposed to only dropping the packets, and the issue becomes how TCP should respond to these marks.

Several proposals for improvements on the dropping/marking decision mechanisms have been made. Most of them are based on the same structure as is proposed for RED and a few of them are described in Gunnarsson (2003). A nice comparison of a few algorithms can be found in Zhu *et al.* (2002) and another overview is given in Asfand-E-Yar *et al.* (2004). Basically, (4.6) is replaced in the different proposals, for some of them (4.5) as well.

There are also techniques that use RED functionality to achieve service differentiation. RFC 2474 (1998) describes differentiated services for IP traffic, **DiffServ**, where the RED parameters in (4.6) are tuned differently for each traffic class. DiffServ is also described in Kilkki (1999).

Algorithm 4.2 Random Early Detection (RED)

At each arrival time, t_k, the queue length q_k is measured. A filtered estimate, y_k, is calculated as

$$y_k = (1 - \lambda)^m y_{k-1} + \lambda q_k,$$

with the scaling factor, m, being

$$m = \begin{cases} 1 & q_k > 0, \\ (t_k - t_e)r & q_k = 0, \end{cases}$$

where r is the service rate of the queue and

$$t_e = \min_{\tau}\{\tau : q(t) = 0, \tau \leq t \leq t_k\}.$$

The probability of drop, p, is then

$$p_k = \begin{cases} 0, & y_k \leq q_m, \\ \frac{p_M}{q_M - q_m}(y_k - q_m), & q_m < y_k \leq q_M, \\ 1, & y_k > q_M. \end{cases}$$

4.1.4   Control Structures and Interactions

The standards described previously imply some control structure for each layer. These descriptions are here translated into block diagrams and cross effects between layers will be identified. As shown in Figure 4.2, packets on a network traverse queues and links. The queues add a queuing delay depending on their length and service rate and the links add a transmission delay. Therefore it is common to model the effects of a network on sent packets as a queue and a delay.

Using block diagram representation, a pure delay,

$$y(t) = x(t - T), \tag{4.7}$$

is written using the Fourier transform e^{-sT}. The queue is fed with a certain arrival rate, r_a, and sends packets with a service rate, r_s. The difference between these two will be the increase in queue length. The total queue length at time t will be the accumulated difference from the initial time, i.e., the integral ($\frac{1}{s}$) of $r_a - r_s$,

$$q(t) = \int_0^t (r_a(\tau) - r_s(\tau)) \, d\tau. \tag{4.8}$$

The block diagram representations of the delay and of the queue are shown in Figure 4.4.
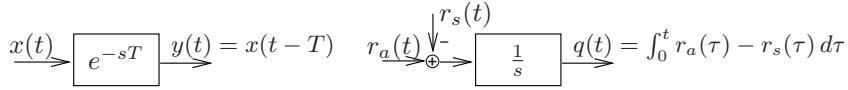
Figure 4.4: *A delay (left) and a queue (right) shown in block diagram representation. For the delay, x is the input and y is the output. For the queue, r_a is the arrival rate and r_s is the service rate.*
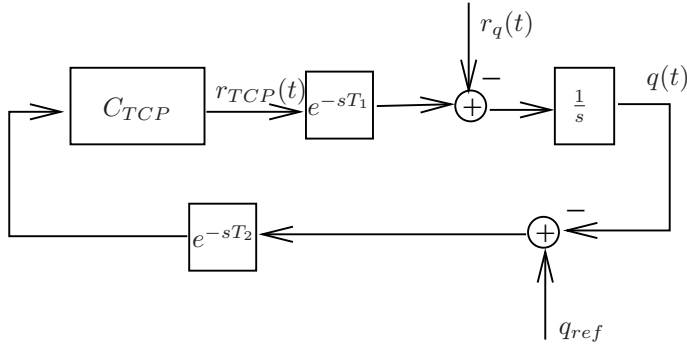
Figure 4.5: *The block diagram for a TCP connection. $d(t)$ is the available capacity for that flow in the queue and can be considered as a disturbance. T_1 and T_2 are transport delays for the connection and T_q is the queuing time, i.e., $T_{RT} = T_1 + T_2 + T_q$.*

Transmission Control Protocol

The control strategy in the transport layer is provided by TCP and was described in Section 4.1.1. The controller, C_{TCP}, acts based on notifications of deliveries and drops ($q > q_{max} = q_{ref}$) and it controls the send rate, r_{TCP}, into the network. Figure 4.5 shows the network as delays and a queue, which is how TCP sees it. The queue increases with the delivery rate, $r_{TCP}(t - T_1)$ and decreases with the service rate $r_q(t)$, i.e., $\dot{q}(t) = r_{TCP}(t - T_1) - r_q(t)$. The total round trip time of the network, T_{RT}, will depend on both the transmission delays, T_1 and T_2, and the queuing time, T_q. The queuing time can be found from solving the equation

$$\int_{t}^{t+T_q} r_q(\tau)\, d\tau = q(t), \tag{4.9}$$

i.e., the time it would take to empty a queue of length $q(t)$ with the specified send rate, r_q. Since TCP Reno only acts based on explicit drops, the block diagram could be extended with a switch. The control signal back to C_{TCP} is 1 if $q > q_{ref}$ and 0 otherwise.

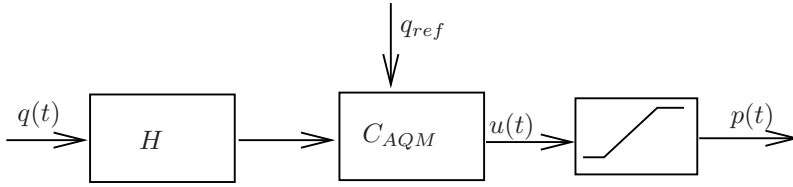**Figure 4.6:** *The block diagram for a AQM queue. The blocks can be added to TCP's block diagram, Figure 4.5, between the queue, q, and the return delay, T_2.*

Active Queue Management

The additional functionality provided by the AQM mechanisms adds to the control loops. In Figure 4.6, a block diagram for an AQM scheme is shown. The queue length, q, is filtered through H. A control decision, u, is calculated in C_{AQM} based on q_{ref} and limited to produce the dropping probability, p. The dropping probability was before either 0 or 1, either there was room in the queue or there was not. Incorporating the AQM control with the TCP representation shows that AQM wants to change the reference for TCP, q_{ref}, from being the maximum queue length to something lower. The blocks should be included between the queue, q, and the return delay, T_2, in Figure 4.5.

The AQM block diagram shows that only drops are used for the control in TCP. This part is hidden in C_{TCP} in Figure 4.5. Several TCP connections will add to the queue length in the AQM queue and each drop affects one of them. If DiffServ, Section 4.1.3, is deployed, parallel AQM links with different controllers and different reference values will divide the TCP flows among them according to traffic class.

4.1.5 In-house Work

A number of Master's theses have studied problems with packet data traffic in different contexts. Gunnarsson (2000) studied problems with TCP congestion control and laid the foundation for the work presented here on packet data.

TCP in other contexts was studied by Persson (2002a), focusing on ad hoc networks, and Brunberg (2002) in a Digital Radio Broadcast framework. Tronarp (2003) focused on evaluation of Quality of Service, QoS, in ad hoc networks.

Björsson (2003), Knutsson (2004), and Zhao and Wu (2005) included 3rd generation radio network capabilities in *ns-2* (NS, 2003, ver. 2.1b9a), and also performed simulation studies. Studies of performance in 3G radio nets have been done by Adolfsson (2003), Fredholm and Nilsson (2003), and Eriksson (2004).

4.2 Performance Issues in Queue Management

Good network performance means different things depending on who is asked. The network provider might have a different measure than the user. To find a good performance measure, the different issues that can be identified will be discussed here.

For each user, transmission time and reliability are important. When TCP is used, the reliability is assured and the only consideration is the transmission time. Thus, one issue in network performance is the time to transmit a certain number of bytes, measured on a per flow (or per user) basis.

The network provider wants to fully utilize the available capacity and divide it among the users. This gives the requirement that there should always be packets processed by the slowest node. The issue of dividing the resources in an efficient way is not easy to resolve. Efficiency can be measured from both the overall network and from the single user, and optimizing the efficiency of the network might not be fair among the users and might cause users to change network provider. The two extreme cases are "total fairness", which means that every user gets exactly the same share of the resources, and "best effort", where no guarantees are given about delay or bandwidth share, and the matter has been discussed in various scheduling contexts (see for example Kelly, 2003). We will not discuss the matter of resource sharing more here.

Flow control is the main way to of vary the network load. As was described in the history review of Internet development in Section 4.1.1, the original flow control was performed at the end nodes, by a predecessor to the TCP protocol. After an immense increase in both size and number of available applications, the Internet traffic does not behave as it did initially. In this section, the previously identified problems caused by TCP are described, together with the research efforts that have been made based on these problems. Current research about performance measurements based on the network queues is also discussed.

4.2.1 Problems with TCP

For a long time, the transport protocol TCP has been seen as the main troublemaker, when performance is concerned. A number of problems have been identified and corresponding solutions have been proposed. Here, the research efforts have been grouped according to the problems they address. A more thorough description of each problem can be found in the corresponding references. The problems are also described in correspondence with the previous block diagram representation, see Figure 4.5.

1. TCP provokes packet losses to get an idea about the state of the network. We get implicit feedback only when something goes wrong.

   - This problem has been addressed with the inclusion of explicit feedback by Mascolo (1999).
   - This problem arises because the controller, C_{TCP}, in Figure 4.5, only bases its decisions on packet drops.

2. The use of the available bandwidth oscillates. Thus the load of the network varies and the network becomes slower than it has to be.

   - This problem has been addressed using adaptive queue management, AQM, starting with RED and extensions such as ECN (Floyd and Jacobson, 1993; Floyd, 1994).
   - Many research efforts aim to improve performance of the initial AQM algorithms, e.g., Kunniyur and Srikant (2004); Deng *et al.* (2002)
   - In the block diagram, Figure 4.5, this is an effect of the design of the controller dynamics, i.e., the algorithm for calculating r_{TCP} based on measurements (drops).

3. There has been a shift in the Internet traffic mix. (Measurements can be found in Balakrishnan *et al.* (1998); Padhye *et al.* (1998).)

   (a) Due to short transmissions, TCP is "always" (85%) in slow start.

   (b) Originally less than 1% of the packets were lost, now it is as much as $5 - 7\%$, which means a lot of retransmissions, (13% of the transmissions), most of which are due to timeouts.

      - The newer versions of TCP, such as Vegas, address this problem (Brakmo and Peterson, 1995)
      - The problem with changing traffic characteristics is also discussed in Jacobsson *et al.* (2006).
      - The original design of C_{TCP} was done based on other assumptions about the rest of the system. A simple way is to say that the characteristics of the noise, or available capacity, r_s, has changed.

4. No difference is made between different traffic classes, such as FTP files, video streams or e-mail.

   - This falls under the area of quality of service, QoS, and DiffServ has been developed to solve this problem (Kilkki, 1999).
   - This is the responsibility of the network, and the problem lies in the construction of the controller C_{AQM}, Figure 4.6. The same parameter settings are used for different flows or types of flows.

5. TCP always assumes that packet losses are due to congestion.

   - This is a faulty assumption in wireless networks where packet losses can be caused by bad links or complete link failures. Some solutions have been proposed for ad hoc networks, where all nodes are both routers and potential receivers (Chandran *et al.*, 2001; Liu and Singh, 2001).
   - Some of the proposed changes use this feature to improve performance, especially in adaptive queue management (Floyd and Jacobson, 1993; Kunniyur and Srikant, 2004).

- This is due to the design of the controller, C_{TCP}, and how it interprets drops.

Many proposals solve a problem by treating some of the other problems as features of TCP and include them in the knowledge of the system. In this way, queue management is moved from the end nodes into the queues, i.e., the routers. As mentioned before, this family of techniques is called adaptive queue management, AQM. The idea is that it is easier to include changes in some of the routers and improve performance locally instead of having to include changes in most of the TCP implementations to improve somewhere. Since any TCP implementation can be used over any router it seems logical to place the improvements in the router of interest.

4.2.2   Measurements Used in Current Research

With all the research efforts put into the problem of improving TCP, performance evaluations become more and more important. Some of the measures that are used in different contexts are briefly described below.

Sachs and Meyer (2001); Meyer (1999); Peisa and Meyer (2001)
  use the time to transmit a certain number of bytes or the packet bit rate to evaluate mobile Internet access, to analyze TCP and file transfers in 3rd generation mobile network.

Misra *et al.* (1999)
  develop a model of the distribution of the queue length and the congestion window and use these to investigate TCP's performance.

Brakmo and Peterson (1995); Mascolo *et al.* (2001)
  consider throughput (bytes/s) and Jain's fairness index (Jain, 1991), as performance measures to analyze the improvements of TCP when using different TCP versions, such as Reno, Vegas and Westwood.

Hollot *et al.* (2001)
  models RED queue behavior using a probabilistic viewpoint and use the model to calculate RED parameters that achieve stability and also fulfill other control criteria.

Park *et al.* (2003)
  study the oscillations of the queue length and try to attenuate them.

Jacobsson *et al.* (2004); Altman *et al.* (2000)
  use the overall throughput and the number of packet losses as performance measures.

After the discussion above and previously in the section, a personal view is presented for network performance measuring. The emphasize is put on the network queue since the earlier discussions showed that throughput, oscillations in the queue and other queue mechanisms are important in the performance discussion.

4.2.3   Using the Bottleneck Queue

We will now take the perspective of a network provider who wants satisfied customers. The focus will be on a specific part of the network, namely the bottleneck queue.

Definition 4.1 (The bottleneck queue). The *bottleneck queue* of a flow is the queue that is the slowest one for that flow and, consequently, a queue where that flow has more than one packet at a time.

There can of course be many bottlenecks in one network, but only one per flow and time instant. When investigating the network performance from the bottleneck queue perspective, we can measure

The time the queue is unused.
   This indicates how efficiently the network is used. An idle bottleneck queue indicates a waste of capacity.

The fluctuations of the queue length.
   This illustrates the ability to find a steady state of the network. The more stable the queue length is, the more stable is also the round trip time experienced by each connection. A stable queue will also make it easier for new connections if the total network is experienced as being in steady state, and thus scalability is improved.

The fraction of dropped packets.
   This fraction represents how fast changes of the network are noticed. A larger number of dropped packets indicates a longer response time to over-flows.

The throughput.
   The throughput, also, signals if the queue often is empty. Of course, a higher throughput is better. It is also interesting for the owner of this bottleneck queue if the income is based on delivered bytes or packets.

The average queue length
   A long average queue implies large round trip delays and a high probability to drop packets, while a short average queue means high probability for an unused queue.

Control objective: The conclusion is that we want to control the tails of the queue length distribution to keep them small. If this is done, drops and empty queues are avoided. We also want to make the distribution narrow around a, not too high and not too low, mean value, which assures small variations in the queue.

This, once again, points at the importance of having information about the actual queue, in particular its length.

4.3   Modeling and Identification of the Queue Dynamics

In order to achieve the control objective, the block diagram description, introduced in Section 4.1.4, will be used. Figure 4.7 is essentially the same as Figure 4.6 with P describing the dynamical relationship between the drop decision and the arrival rate to the queue, r_{nw}. This is the considered system and it is a connection of TCP links, similar to the one seen in Figure 4.5. From the queue perspective this system is unknown: the number of flows, their respective round trip times and the type of TCP controller they obey. It is only known that, after some delay, the queue length decreases when packets are dropped and it decreases more rapidly if a larger number of packets are dropped at the same time. The queue send rate, r_s, is given and the AQM controller, C_{AQM}, bases the dropping decision on the filtered queue length, y.
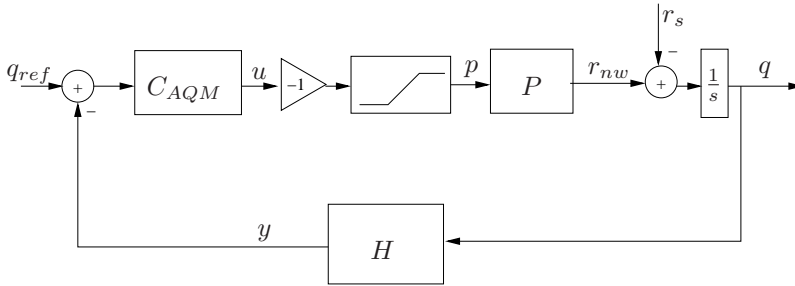

Figure 4.7: *Block diagram of the queue system. The AQM part is the same as in Figure 4.6. The filtered queue length is denoted y. The system P describes the network effects on the queue length after a certain drop is done.*

Several modeling efforts that focus on the bottleneck queue have been made. For example, Asfand-E-Yar *et al.* (2004) find a closed form model of the RED mechanisms in order to analyze AQM performance; Jacobsson *et al.* (2006) develop a model from the acknowledgment count to the bottleneck queue length and hereby study the impact of different traffic types; Deb and Srikant (2006) model a single bottleneck queue with the aim to predict steady state properties of the arrival process. A nice overview of the estimation problems in congestion control is given in Jacobsson and Hjalmarsson (2004).

The focus in this work is on a simple model of the queue length dynamics without any knowledge of the rest of the system P. The qualitative knowledge about P will be used in simple controllers in later sections. To exemplify and motivate the modeling choices throughout this section, a typical data set is used.

┌── **Example 4.2: Dataset** ───┐

Using *ns-2* (NS, 2003, ver. 2.1b8a) realistic network data can be produced. We measure the queue length, q, at the bottleneck node in a network. The network
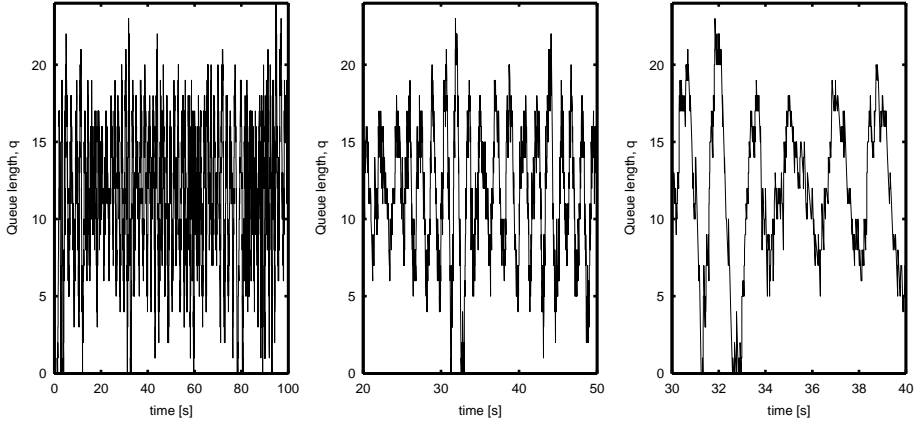
Figure 4.8: *The motivation data example from a bottleneck queue with the number of active transmissions varying from 0 to 100. Both the full time span and shorter segments are shown. The data is taken from ns-2 (NS, 2003).*

has between 0 and 100 sources sending TCP traffic over the bottleneck queue at the same time. The sending periods for the sources are selected randomly. Figure 4.8 shows the measured queue length over time. Inspection of the plots shows that the basic variation, for this network setup, is in the order of 0.6 Hz.

4.3.1 Frequency Analysis and Filtering

The Fourier transform, $X_c(f)$, can be used to describe the frequency content of a signal, $x(t)$,

$$X_c(f) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft}\, dt. \tag{4.10}$$

This demands that $\int |x|^2 < \infty$. If the signal is sampled, $x(kT)\ k = 1, \ldots, N$, the DFT, X_d, is used

$$X_d(f) = \frac{1}{\sqrt{NT}} \sum_{k=1}^{N} x(kT)e^{-i2\pi fkT}. \tag{4.11}$$

$X_d(f)$ is what will be used in this work.

┌── **Example 4.3: Frequency Content** ──────────────────────────

Visual inspection of primarily the right part of Figure 4.8 shows that the signal is oscillating rapidly around the basic variation. Frequency analysis of the queue
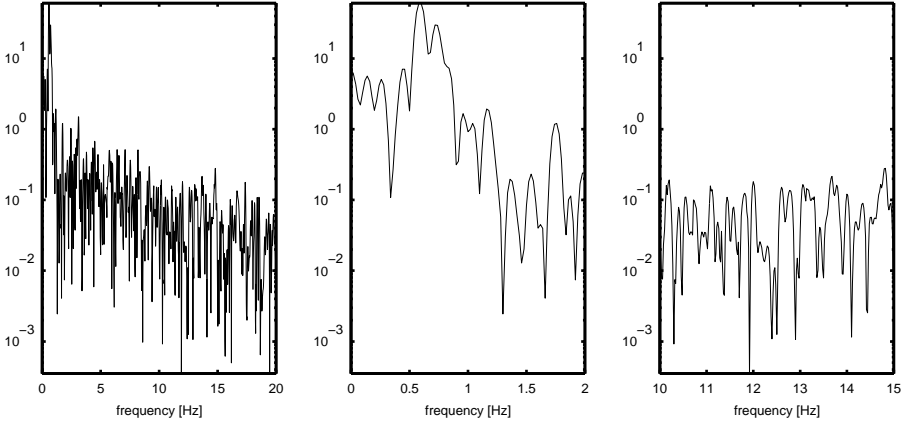
Figure 4.9: *The frequency content of the dataset described in Example 4.2. $|X_s(f)|^2$ is shown for the shortest time interval. Focusing on different frequency ranges shows the peak around 0.6. The high frequency variations are spread over several frequencies.*

length shows, see Figure 4.9, that a majority of the frequency content is around 0.6 Hz. The faster variations that were visible in the time plot cannot be seen in a single frequency peak.

Filtering of the signal is performed to reduce the fast oscillations and the noise. The control can then be based on the long term variations (0.6 Hz). The noise will be reduced by a simple first order low pass filter,

$$H(s) = \frac{a}{s + a}, \tag{4.12}$$

which attenuates frequencies higher than $\frac{a}{2\pi}$ Hz more than 3 dB.

Example 4.4: Filtering

The main frequency content in the shorter time segment is below $f = 1$ Hz. With $a = 6.5$ the low pass filter effectively attenuates frequencies above 1.03 Hz more than 3 dB. The result is clearly visible in a comparison of the two signals, see Figure 4.10. The phase delay introduced by the filter is inevitable in the real-time implementation, and the filter choice will be a trade-off between smoothing and delay.
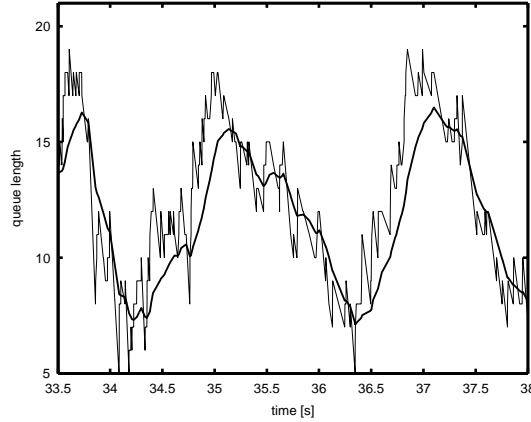
Figure 4.10: *Comparing the filtered signal (thick line) and the original one (thin line) for a short time.*

4.3.2   AR Modeling of Nonzero-Mean Signals

Auto regressive (AR) models are models of a signal based only on the signal itself, see for example Ljung (1999) for a thorough discussion on AR models. The basic model equation contains the signal and time shifts of it. Here, we study variations around a fixed unknown level b, and get

$$
\begin{aligned}
y(t) &= \varphi^T(t)\theta + v(t) \\
\varphi(t) &= \left(\begin{array}{cccc} -y(t-T) & \ldots & -y(t-nT) & 1 \end{array} \right)^T \\
\theta &= \left(\begin{array}{cccc} a_1 & \ldots & a_n & b \end{array} \right)^T
\end{aligned}
\tag{4.13}
$$

where $\varphi(t)$ contains the known signal and θ are the parameters to be estimated. This model structure can capture the basic frequencies of a signal. The frequencies are given by the angles of the complex roots, x_0, to the characteristic polynomial

$$
x^n + a_1 x^{n-1} + \ldots + a_n = 0.
\tag{4.14}
$$

The frequencies are thus

$$
f = \frac{\arg(x_0)}{\pi} \frac{1}{2T},
\tag{4.15}
$$

where T is the sampling interval as before.

Recursive identification is performed using recursive least squares (RLS), see for example Ljung (1999). The algorithm is summarized in Algorithm 4.3. The AR model structure is particularly simple to handle in this type of identification.

After an estimate of the current model parameters is obtained, future values of the signal, y, can be predicted. For a model with sample time T, one step ahead

---

Algorithm 4.3 Recursive Least Squares (RLS) (Ljung, 1999, Sec. 11.2)

Calculate a new estimate of θ at time t according to

$$\varepsilon(t) = y(t) - \varphi^T(t)\hat{\theta}(t - T)$$

$$P(t) = \frac{1}{\lambda}\left(P(t - T) - \frac{P(t - T)\varphi(t)\varphi^T(t)P(t - T)}{\lambda + \varphi^T(t)P(t - T)\varphi(t)}\right)$$

$$K(t) = P(t)\varphi(t)$$

$$\hat{\theta}(t) = \hat{\theta}(t - T) + K(t)\varepsilon(t)$$

Here, λ is the forgetting factor chosen as $\lambda < 1$.

---

predictions are calculated as

$$\hat{y}(t + T) = \varphi^T(t + T)\hat{\theta}(t). \tag{4.16}$$

Studying (4.13) we see that $\varphi(t + T)$, in the expression above, contains values of y from time t and older. The estimated parameters $\hat{\theta}(t)$ are the most recent estimate. Predictions further away into the future can also be calculated using (4.16) recursively, but of course with larger uncertainty, both because of noise and of model changes. In the case of an AR model, when future values of y are needed in φ the predicted values are used instead. The accuracy of the prediction is thus effected by the accuracy of earlier predictions as well as of the accuracy of the model estimation.

Example 4.5: AR Modeling

An AR model has been estimated from the data set described in Example 4.2. It turned out that a fifth order AR model gave a good fit. A segment of the filtered queue length, y, and the estimated model output, $\hat{y}$, is shown in Figure 4.11, together with the two estimated frequencies. For this example, RLS with $\lambda = 0.999$ was used. The basic frequency is around 0.6 Hz, while the second one varies over time, cf., the frequency plots in Figure 4.9 where no distinct second peak was visible.

We compare the one-step ahead predictor (4.16) for the AR model with the trivial predictor $\hat{y}(t + T) = y(t)$. To measure the performance of the estimates the relative distance between the measurements and the estimates is calculated,

$$R = 1 - \sqrt{\frac{\sum|y - \hat{y}|^2}{\sum|y|^2}}.$$

A value close to 1 indicates a good fit. For the AR model $R = 0.83$ and for the trivial predictor $R = 0.75$. From this we see that the knowledge about the data is increased by the identified AR model.
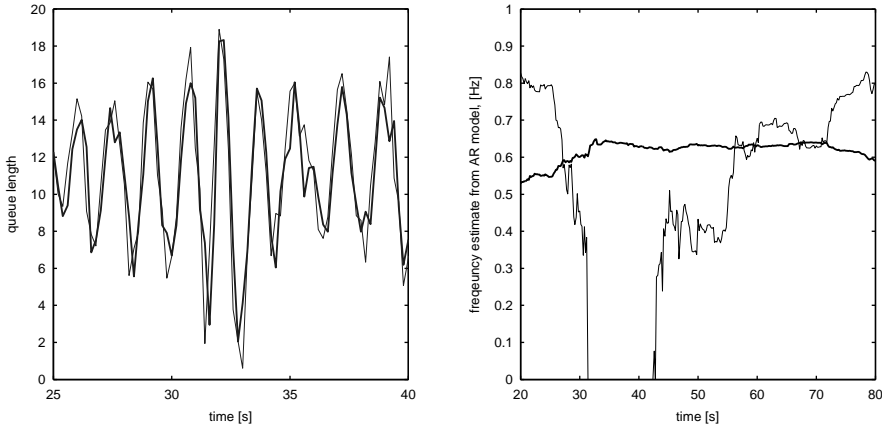
Figure 4.11: *To the left is a comparison of the filtered queue length, $y(t)$ (thin) and model output, $\hat{y}(t) = \varphi^T(t)\hat{\theta}(t)$ (thick), from Example 4.5. To the right is the corresponding frequency estimates, given by (4.15).*

4.4   Control of the Queue Length

In Section 4.2.1, a number of problems with TCP were discussed. The control of the queue dynamics that will be discussed here addresses Problems 2 (bandwidth oscillation) and 3b (increase of packet losses). The control technique utilizes the identification procedure discussed earlier and considers Problem 5 (packet losses always lowers the send rate) as system knowledge that can be used in the control design.

As discussed previously, the goals in queue management are to attenuate oscillations and to keep the queue from emptying and overflowing. The previous section indicated that the queue dynamics is well described by an AR model. To reduce the oscillations, derivative action is needed and a few simple versions will be described here. The control is primarily based on the ideas in the existing RED algorithm.

In standard RED, the packet dropping probability is first calculated and then the packet is dropped or accepted based on this probability. A natural extension is to drop packets only when the queue is increasing, which is a very simple inclusion of derivative action into the controller. Derivative action changes the control decision based on the derivative of the signal, and a large derivative typically yields a lower control signal.

The idea with the adaptively estimated AR model is to predict the evolution of the queue length in the near future and to use this prediction as a base for control decisions.

Combining prediction and this type of derivative action with the basic RED algorithm produces four alternatives, which are described below. For reference,

the saturation operator is first defined:

$$\text{sat}(x) = \begin{cases} 0, & x \leq 0 \\ x, & 0 < x < 1 \\ 1, & x \geq 1 \end{cases}. \tag{4.17}$$

The saturation can also be seen in the block diagram Figure 4.7, when the control signal, u, is transformed into the dropping probability, p.

1. Proportional control (P)

Ordinary RED essentially calculates the drop probability, p, as an affine transformation of the filtered queue length, y,

$$u(t) = k(y(t) - r)$$
$$p(t) = \text{sat}(u(t)).$$

2. Proportional control with prediction (P pred)

The RED-functionality is applied to the predicted value, $\hat{y}$,

$$u(t) = k(\hat{y}(t + T) - r)$$
$$p(t) = \text{sat}(u(t)).$$

3. Proportional and derivative control (PD)

A positive value of the dropping probability is used only if the derivative is positive, i.e., the queue length increases,

$$u(t) = \begin{cases} k(y(t) - r), & \dot{y}(t) > 0 \\ 0 & \dot{y}(t) \leq 0 \end{cases}$$
$$p(t) = \text{sat}(u(t)).$$

4. PD with prediction (PD pred)

As above but based on the predicted value instead of the measured,

$$u(t) = \begin{cases} k(\hat{y}(t + T) - r), & \dot{\hat{y}}(t + T) > 0 \\ 0 & \hat{y}(t + T) \leq 0 \end{cases}$$
$$p(t) = \text{sat}(u(t))$$

For all cases, the constant r can be seen as the reference value towards which the queue length is controlled. Since only the bottleneck queue is considered, the queue length is increased, by increasing TCP send rates, until the control kicks in, $y(t) > r$. The difference between the first two approaches can be seen as the difference between using the shifted model or the identified AR model in Example 4.5. To implement the derivative action, an estimate of the derivative is needed.

The derivative action can be included in other ways, the more typical setup gives a fifth alternative.

5. Explicit D action

An extension is to use derivative action more explicitly

$$u(t) = k_p(y(t) - r) + k_d \dot{y}(t)$$
$$p(t) = \text{sat}(u(t))$$

in which case it becomes important to have reliable derivative estimates. This alternative can of course also be used with predictions.

Here we restrict the investigation to methods 1—4, and use a simple backward difference approximation of the derivative. A discussion about the derivative estimation for this case was done in Gunnarsson (2003, Sec. 3.3).

4.4.1   Simulation Results

Here, the performance of the described control strategies is analyzed using simulations in *ns-2* (NS, 2003, ver. 2.1b8a). A special version of the RED queue module has been implemented, which allows testing of both identification and control performance.

The setup is a network with varying number of flows and only one queue is studied. For comparative studies, identical realizations of the network are used, i.e., the start and stop times for the flows are fixed in each simulation to get comparable and reproducible conditions for all controllers.

Since it is not likely that one of the update times t_i matches $t - kT, k = 1, 2, \ldots, n$, simple linear interpolation between the two surrounding measurement points is used whenever an old value of y is needed. Here, more advanced methods for down-sampling of *non-uniformly sampled signals* could be beneficial.

Because of the uncertainty in the derivative calculation, controller number 5 has not been implemented. All the other control versions can easily be implemented using the generic Algorithm 4.4, where a simple backward difference estimate is used for the derivative. During the simulations, $M_p = 0.2$, M_q is set to the actual total queue length and $m_q = M_q/2$.

The question arises what the suitable sampling time, T, is. Using the estimated model, the oscillating frequency, f, can be found and T should then, as a rule of thumb, be chosen as a tenth of the period, i.e., $T \approx \frac{1}{f} \frac{1}{10}$. Spectral analysis may also be used to find the main frequency. Here, methods for spectral analysis of *non-uniformly sampled signals* would be appropriate.

Simulations indicate that it is the send rate of the queue that is the main cause of the varying frequency, see Figure 4.12. Here, the scenarios have been adapted to work for $T = 0.2\,\text{s} \approx \frac{1}{10} \frac{1}{0.6}$, the results are valid no matter what value we choose but the choice is still a design parameter. To improve performance, automatic detection of an accurate time constant should be considered.

A few performance measures are summarized in Table 4.1. These numbers are calculated for one single run with exactly the same setup for all four controllers. In Figure 4.13 the resulting queue length is shown for the four controllers.

The following observations can be made from Table 4.1 and Figure 4.13:

Algorithm 4.4 Generic AQM Implementation

$$y_d = \frac{y_1 - y_2}{T_d},$$

$$u = \begin{cases} M_p \frac{y_1 - m_q}{M_q - mq}, & y_d > 0, \\ 0, & \text{otherwise,} \end{cases}$$

$$p = \text{sat}(u).$$

The variables y_1, y_2 and T_d are used to get the different versions from p. 52,

1. P: $y_1 = y(t)$, $y_2 = 0$ and $T_d = y_1 - y_2$,

2. P pred: $y_1 = \hat{y}(t + T)$, $y_2 = y(t)$ and $T_d = y_1 - y_2$,

3. PD: $y_1 = y(t)$, $y_2 = y(t - T)$ and $T_d = T$,

4. PD pred: $y_1 = \hat{y}(t + T)$, $y_2 = \hat{y}(t)$ and $T_d = T$.
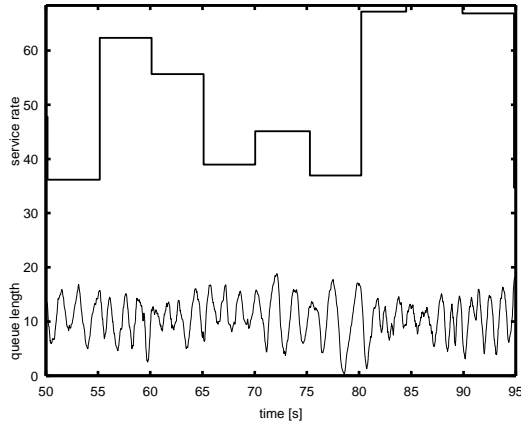

Figure 4.12: *Comparing different send rates (top thick line) with the corresponding queue length (lower thin line). A lower send rate in the queue seems to slow down the oscillations of the queue length.*
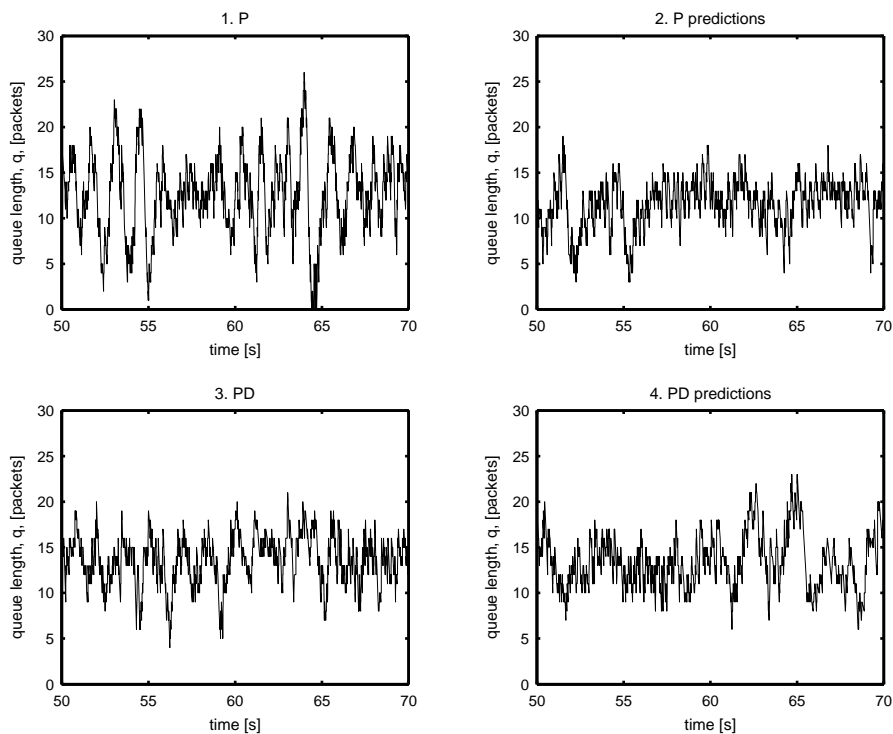
Figure 4.13: *A segment of the queue length, q, when using the four different controllers. Both differences in the height and the freqeuncy of the variations can be seen.*

Table 4.1: *Comparison of the different control schemes, see p. 52 for a description. The number of drops and the total throughput is studied.*

Type	Drops early+hard = tot	Sent unique packets
1. P	55 + 250 = 305	3260
2. P pred	66 + 247 = 313	3569
3. PD	82 + 207 = 289	3126
4. PD pred	62 + 235 = 297	3342

- An increase in throughput is accompanied with an increase in the total number of drops.

- Model predictions can improve the throughput (7–10%), both in basic RED (compare 1 to 2) and in RED with D action (compare 3 to 4).

- The simple derivative action used does not increase the throughput.

- In both cases the derivative action reduces the number of hard drops, compare 1 to 3 and 2 to 4, which also was part of the stated control objective.

- The derivative action lowers the span of the main variations in the queue length, which explains the reduced number of hard drops.

- Using predictions reduces the number of highest peaks in the variations and increases the frequency. Reducing the highest peaks should also indicate a lower number of hard drops, while an increased basic frequency indicates that using the predictions removes the frequencies captured in the model.

Ordering the algorithms becomes hard, since the aim is to both reduce the number of hard drops and to increase the throughput. It is apparent that predictions should be used and, actually the fourth algorithm (PD pred) performs second best for both cases. The evaluation is promising for the model based control, and further investigations of prediction and improved derivative action would be interesting.

Closed-loop Identification

The system performs in closed loop and the output affects later inputs. This can be a problem when system identification is used. In this case, the model identification is based on finding variations and basic frequencies and the control goal is to attenuate the oscillations. This means that a perfect controller will make it hard to estimate the model and a bad model will make the controller worse. These properties will make it impossible to find a perfect controller for this system. However, the delay between a packet loss and a reaction in the queue length is so large that it becomes impossible to control the system perfectly because of this. Therefore, it is worth the effort to use model based control for the network

system. Closed-loop identification and validation problems for congestion control are discussed in Jacobsson and Hjalmarsson (2006).

4.5   Summary

Network performance can be studied from a queue perspective. The control objectives for the queue is to attenuate oscillations, prevent overflow and avoid an empty queue.

Queue management can improve network performance by intelligently dropping packets. The standard algorithm, RED, for AQM can be improved by using prediction. To attenuate queue length oscillations derivative action has been tested. The throughput was not improved using the simple derivative action, but the oscillations were attenuated. Explicit derivative action was not included in the evaluation runs.

The experiments indicate that model based estimation of the derivative can give improvements. In the simulation run, only a difference approximation was used to find the derivative. The improved controller was based on accurate modelling of the queue length dynamics. It seems that a biased AR model is well suited for this purpose.

4.6   Open Problems

This chapter was given as a motivating example to the work with non-uniform sampling. Here we summarize the issues that could improve the work presented here, and give references to the text. It is clear that, in this application, the non-uniform sampling is given by events (packet arrivals), and the interesting measurement, the queue length, is completely described from this sampling process.

The problems that arise are summarized below.

Problem 1 In order to find a good value of the sampling time T, the low frequency content in $y(t)$ is needed, see Example 4.3 and the comments on p. 53.

> **Theory** In this thesis, further work in this area is done in Paper A.
>
> **Application** The theories can be applied quite easily to the network data in order to perform off-line frequency estimation, as well as other off-line investigations of the spectrum.

Problem 2 Low pass filtering and down-sampling of q would give y only in the interesting time points, see Example 4.4, Section 4.3.2 and the notes on p. 53.

> **Theory** The problem of down-sampling from non-uniform sampling times is addressed in Paper C.
>
> **Application** It is an easy task to implement down-sampling for the network queue, given the results. The problem is probably to satisfy the real-time demands. A solution to this problem might be to perform the

model identification and control actions on a slower time scale than the packet arrival times.

Problem 3 Identification directly from non-uniform samples is another issue that gives improvements. This could give continuous-time models without the dependence on the choice of T.

> **Theory** Paper B discusses this matter for a specific case, but recursive identification from non-uniform samples is still an open question.

> **Application** Off-line identification could be used to get a nominal model, that could be used in real-time implementations, while the recursive identification still would follow the current procedure.

Problem 4 This applications have high demands on speed, which makes the implementation performance important.

> **Theory** Some thoughts on this are given in Paper C, while it is a major problem in Paper B, and more research needs to be done.

5

Summary of the Thesis

This chapter gives a little foretaste of what has been done, and can be investigated further in the included papers, and also makes some suggestions on what could be done in the future, based on these results.

5.1 Main Contributions

Here, the main results of the included papers are given, in a very compact way, with selected equations and figures from the papers. The notation in Chapter 3 is still valid.

5.1.1 Frequency Domain Analysis

In Paper A, two types of transforms and two sampling models are studied. We exemplify with the Dirichlet transform,

$$Y(f) = \sum_{m=1}^{M} y_m e^{-i2\pi f t_m}, \tag{5.1}$$

and additive random sampling, ARS,

$$t_m = t_{m-1} + \tau_m, \qquad m = 1, \ldots, M. \tag{5.2}$$

The results give analytical expressions for mean value and covariance of the transform $Y(f)$, with the effects from the original signal, $s(t)$, the non-uniform sample times, t_m, and the limited number of samples, M, clearly separated. Here, the characteristic function (CF),

$$\varphi_\tau(t) = \mathrm{E}[e^{-i2\pi f \tau}], \tag{5.3}$$

of the sampling noise, τ_m, plays a crucial role. For example, in uniform sampling the discrete-time spectrum is given by a convolution of the spectrum of $s(t)$ with the *normalized Dirchlet kernel*,

$$d_M(f) = \frac{1 - e^{-i2\pi fMT}}{1 - e^{-i2\pi fT}}. \tag{5.4}$$

This convolution gives the aliasing effect due to a finite number of samples. For ARS, this corresponds to convolution with

$$d_M^{\text{ARS}}(f) = \frac{1 - \varphi_\tau(f)^M}{1 - \varphi_\tau(f)} = \frac{1 - \mathrm{E}[e^{-i2\pi f\tau}]^M}{1 - \mathrm{E}[e^{-i2\pi f\tau}]}. \tag{5.5}$$

The results for the transform $Y(f)$ lead us to results for the periodogram $P_Y(f) = |Y(f)|^2$. The mean value of $P_Y(f)$ is straightforward to find, and given some asymptotic analysis ($M \to \infty$), the variance of $P_Y(f)$ can also be calculated.

As an example, we study the signal

$$s(t) = 1.2\sin(2\pi t) + 0.8\sin(4\pi t), \tag{5.6a}$$

$$y_m = s(t_m), \tag{5.6b}$$

$$t_m = t_{m-1} + \tau_m, \tag{5.6c}$$

using ARS with τ_m taken from a rectangular distribution between 0.1 and 1.3,

$$p_\tau(t) = \frac{1}{1.2}, \qquad t \in [0.1, 1.3]. \tag{5.7}$$

The periodogram and its first two moments for this signal are shown in Figure 5.1. The mean value clearly shows the two frequency peaks, but the sampling noise contributes to several other peaks for the single realization.

5.1.2 Frequency Domain Identification

For the study of identification of non-uniform sampled systems, we consider the special case of stochastic jitter sampling with unknown jitter noise, i.e.,

$$s(t) = (g_\theta \star u)(t), \tag{5.8a}$$

$$y_m = s(mT + \tau_m) + e_m, \tag{5.8b}$$

$$\tau_m \in p_\theta(\tau) \tag{5.8c}$$

and the values of τ_m are unknown. The input $u(t)$ is assumed known, or at least its Fourier transform, $U(f)$, and also the measurement noise, e_m, has known properties. Both the system $g_\theta(t)$ and the jitter sampling noise pdf $p_\theta(\tau)$ are parameterized with θ. We use the discrete time Fourier transform (DTFT) of this sequence,

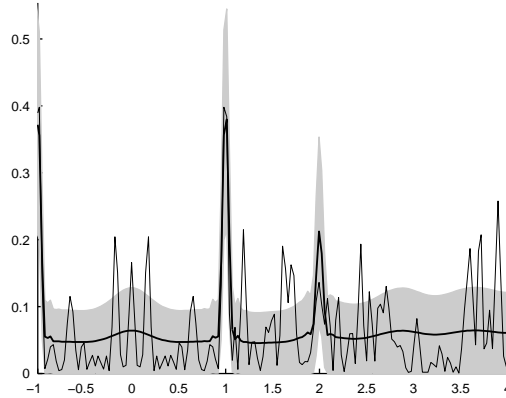$$Y(f) = \sum_{m=1}^{M} y_m e^{-i2\pi fmT}, \tag{5.9}$$

Figure 5.1: Replica of Figure A.3b. *Mean value (thick line), confidence band (gray shade) and one realization (thin line) of the periodogram, when the signal is a multi-sine, (5.6), and the sampling is ARS with rectangular noise between* 0.1 *and* 1.3.

and derive mean value and variance for $Y(f)$. These moments are once again functions of the characteristic function of the sampling noise, τ_m, and of the original signal. This enables us to calculate optimal parameters for both the unknown system, $g_\theta(t)$, and unknown sampling noise pdf, $p_\theta(\tau)$, based on minimization in the frequency domain. For example, a least squares setting is

$$\hat{\theta} = \arg\min_\theta \sum_k \left| Y(f_k) - \mu_Y(f_k; \theta) \right|^2, \qquad (5.10)$$

where μ_Y can be varied to give different criteria.

LS A least squares criterion is given with

$$\mu_Y(f_k; \theta) = \sum_m s(mT; \theta)e^{-i2\pi f_k mT},$$

i.e., simply the DTFT of the sequence $s(mT)$.

BCLS Bias compensation is achieved when μ_Y is given by the mean value of $Y(f)$,

$$\mu_Y(f_k; \theta) = \sum_m \mathrm{E}[s(mT + \tau_m); \theta]e^{-i2\pi f_k mT},$$

which is given explicitly in the paper.

AsML The criterion is also extended in the paper to give an asymptotic Maximum Likelihood estimate, using also the covariance of $Y(f)$.

Table 5.1: Replica of Table B.2. *Results for identification of a second order system with known sampling pdf. Mean value and standard deviation of $\hat{\theta} = (\hat{k}_0 \ \hat{z} \ \hat{p}_r \ \hat{p}_i)^T$ for the three estimates LS, BCLS and AsML, as well as the true value, θ^0, are shown.*

	k_0	p_r	p_j
θ^0	6.25	-1.5	2
$E[\hat{\theta}^{LS}]$	5.8468	-1.5876	1.9508
$Stdev[\hat{\theta}^{LS}]$	0.1723	0.0448	0.0045
$E[\hat{\theta}^{BCLS}]$	6.2536	-1.4981	2.0019
$Stdev[\hat{\theta}^{BCLS}]$	0.0479	0.0385	0.0415
$E[\hat{\theta}^{AsML}]$	6.2499	-1.5008	2.0000
$Stdev[\hat{\theta}^{AsML}]$	0.0480	0.0343	0.0401

The results for identification of a second order system, $g_\theta(t) = \mathcal{L}^{-1}(G_\theta(s))$, parameterized as

$$G_\theta(s) = \frac{k_0}{(s - (p_r + ip_i))(s - (p_r - ip_i))}, \tag{5.11a}$$

$$\theta = (\ k_0 \quad p_r \quad p_i \ )^T, \tag{5.11b}$$

are given in Table 5.1. The sampling is done with $T = 2$ and τ_m is taken from a zero-mean rectangular distribution covering the whole sampling interval. The main advantage is that a bias error is removed when estimating based on the mean value of $Y(f)$ instead of the DTFT of $s(mT)$.

Figure 5.2 shows the true system $G_{\theta^0}(i2\pi f)$ together with $G_\theta(i2\pi f)$ using $\theta = E[\hat{\theta}^{LS}]$ from the estimation without bias compensation. Using $\theta = E[\hat{\theta}^{BCLS}]$ or $\theta = E[\hat{\theta}^{AsML}]$ does not give a visible difference from the true system.

5.1.3 Analysis of Down-Sampling

When down-sampling is studied, we consider ARS,

$$t_m = t_{m-1} + \tau_m, \tag{5.12}$$

$$y_m = s(t_m) + e_m \tag{5.13}$$

and the aim is to uniformly resample and filter the signal, $s(t)$, at a lower rate, $T \gg E[\tau_m]$, given a low-pass filter $h(t)$. Optimally, this would give

$$z(nT) = \int h(nT - t)s(t)\,dt, \tag{5.14}$$

and we analyze three different ways of realizing this equation when y_m is given. It is shown that interpolation of the convolution integral is an appropriate procedure, both in terms of performance and complexity. In the analysis, results from
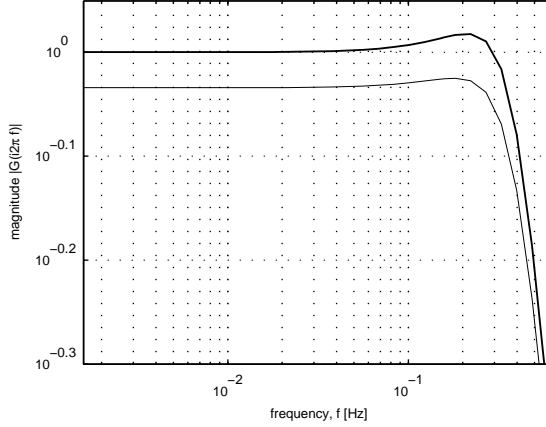
Figure 5.2: *(Replica of Figure B.5). A comparison of $|G_\theta(i2\pi f)|$ (thin line), given by (5.11) and $\theta = \mathrm{E}[\hat\theta^{LS}]$ from Table 5.1, with the true $|G_{\theta^0}(i2\pi f0)|$ (thick line). When using $\hat\theta^{BCLS}$ or $\hat\theta^{AsML}$ the difference from the true system is not visible.*

Paper A show that this implementation corresponds to computing

$$\hat z(nT) = \int \tilde h(nT - t)s(t)\,dt, \tag{5.15}$$

where $\tilde h(t) = \mathcal{F}^{-1}(\tilde H(f))$ is given by the original filter $h(t)$ and the non-uniform sampling properties, such that

$$\tilde H(f) = \sum_{m=1}^{M} \tau_m h(t_m)e^{-i2\pi f t_m}, \tag{5.16}$$

recognized as the Riemann transform of $h(t_m)$. For all cases, this results in a bias, $\hat z(nT) \neq z(nT)$, since $\mathrm{E}[\tilde H(f)] \neq H(f)$. However, for smooth filters, $h(t)$, with compact support

$$\mathrm{E}[\tilde H(f)] \to H(f), \qquad M \to \infty. \tag{5.17}$$

For four different rectangular distributions on τ_m, the resulting $\mathrm{E}[\tilde H(f)]$ is shown in Figure 5.3, where $H(f)$ is a second order Butterworth filter.

5.2   Future Work

There is never enough time before a deadline. Therefore, here are some more or less elaborated ideas that this work could continue with.
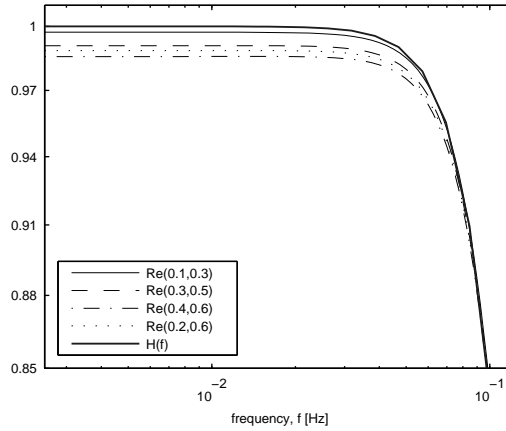
Figure 5.3: Replica of Figure C.5. *The true $H(f)$ (thick line) compared to $|\mathrm{E}[\tilde{H}(f)]|$ (5.16), when $H(f)$ is a second order butterworth filter, for four different rectangular sampling noise distributions, and $M = 250$.*

5.2.1 Frequency Domain Analysis

The last part of Paper A partly depends on the likely statement in (A.37), where it is argued that the transforms, for example in (5.1), have asymptotic complex Gaussian distribution. This statement would benefit from a rigorous proof using an existing or new central limit theorem.

The setup is as follows: If

$$Y(f) = \sum_{m=1}^{M} s(t_m)e^{-i2\pi f t_m} \tag{5.18}$$

where $t_m = \sum_{k=1}^{m} \tau_k$ and τ_m are independent identically distributed random variables, what can then be said about the probability distribution of $Y(f)$ when M approaches infinity?

Discussions with Professor Timo Koski, Mathematical statistics, Linköping University, made us believe that existing theorems need to be refined for this particular case, and probably will result in requirements on the underlying function $s(t)$. It is likely that James Davison: *Stochastic Limit Theory. Advanced Texts in Econometrics.* Oxford University Press 1994, ISBN 0-19-877402-8, can be of some help here.

5.2.2 Identification

Missing data identification is the case when a uniform sampling sequence is distorted with missing samples. Modeling this as an ARS sequence

$$t_m = t_{m-1} + \tau_m, \tag{5.19}$$

with $\tau_m \in [T, 2T, \ldots]$ being discrete sampling noise, can enable us to use the same setup as in Paper B for frequency domain identification. This would give identification of the continuous-time system, when only the pdf of the missing samples is known and not their exact location.

The frequency domain identification requires the whole measurement sequence to be recorded before identification is started. An analysis of possibilities to implement this recursively and also of the error that this would give, would be beneficial. Implementing a recursive update of the Dirichlet transform could be done as

$$Y_m(f) = \sum_{k=1}^{m} y_k e^{-i2\pi f t_k} = Y_{m-1}(f) + y_m e^{-i2\pi f t_m}, \qquad (5.20)$$

which is quite straightforward. Using $Y_m(f)$ for identification in each step then requires calculation of $E[Y_m(f)]$ for different parameter settings and also a recursive update of the parameter estimate. The main problem here is probably computation of this mean value in each time step, since this turned out to be fairly complex in Paper B.

In all the different identification cases, it is interesting to see how good the estimate can get at best. Therefore, calculating the Cramér-Rao Lower Bound, for the identified parameters, would be interesting.

5.2.3   Other Aspects

Both for recursive implementation, but also for normal frequency analysis, efficient implementation of algorithms is crucial. Ideas for fast implementation of the sum

$$\sum_m g_m e^{-i2\pi f t_m} \qquad (5.21)$$

exist in literature, and could be useful for the results in both Paper A and Paper C. For the identification, the problems probably need to be re-formulated to fit in an optimization framework, and then solved using existing solvers. The main obstacle is the mean value and covariance, since they are constructed using integrals. Approximations using numerical analysis may be needed to speed up this part. In that case, analysis of error propagation to the identified parameters is also needed.

Given the large amount of applications where non-uniform sampling occurs, it is of course of great interest to test and modify theories for specific cases. Also, this would most probably give more ideas for future research.

Bibliography

Adolfsson, Klas (2003). *TCP performance in an EGPRS system*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3163.
URL *http://www.ep.liu.se*

Alavi, Hossein and Fadaei, Mehdi (1994). Frequency estimation of low rate non-uniformly sampled signals using neural networks. In *IEEE Singapore International Conference on Communication Systems ICCS*, vol. 1, (pp. 229–233). Singapore.

Altman, Eitan; Başar, Tamer; and Hovakimian, Naira (2000). Worst-case rate-based flow control with an ARMA model of the available bandwidth. In *Annals of Dynamic Games*, vol. 6:pp. 3–29.

Asfand-E-Yar; Awan, Irfan; and Woodward, Michael E (2004). Stochastic analysis of Internet traffic congestion control using active queue management (AQM) scheme. In *Performance Modelling and Evaluation of Heterogenous Networks, HET-NETs*. West Yorkshire, England.
URL *http://www.comp.brad.ac.uk/het-net/HET-NETs04/*

Balakrishnan, Hari; Padmanabhan, Venkata N; Seshan, Srinivasan; Stemm, M; and Katz, Randy H (1998). TCP behavior of a busy Internet server: analysis and improvements. In *Joint Conference of the IEEE Computer and Communications Societies, INFOCOM*, vol. 1, (pp. 256–262). San Fransisco, California, USA.

Belmont, Michael R; Horwood, J Mike K; and Thurley, Richard WF (2003). Non-uniform sampling issues arising in shallow angle wave profiling LIDAR. In *IEEE/OES Seventh Working Conference on Current Measurement Technology*, (pp. 135–139). San Diego, California, USA.

Benedetto, John J (1992). Irregular sampling and frames. In Chui, Charles K (Ed.), *Wavelets: A Tutorial in Theory and Practice*. Academic Press Inc.

Benedetto, John J and Ferreira, Paulo JSG (Eds.) (2001). *Modern Sampling Theory*. Birkhäuser Verlag.

Beutler, Frederick J (1966). Error-free recovery of signals from irregularly spaced samples. In *SIAM Review*, vol. 8(3):pp. 328–335.

Beutler, Frederick J (1970). Alias-free randomly timed sampling of stochastic processes. In *IEEE Transactions on Information Theory*, vol. 16(2):pp. 147–152. ISSN 0018-9448.

Bilinskis, Ivar and Mikelsons, Arnolds (1992). *Randomized Signal Processing*. Prentice Hall.

Björsson, Anders (2003). *Simulation analysis of RLC/MAC for UMTS in Network Simulator version 2*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3399.
URL *http://www.ep.liu.se*

Bland, Denise M and Tarczynski, Andrzej (1997). Optimum nonuniform sampling sequence for alias frequency suppression. In *IEEE International Symposium on Circuits and Systems*. Hong Kong.

Brakmo, Lawrence S and Peterson, Larry L (1995). TCP Vegas: end to end congestion avoidance on a global Internet. In *IEEE Journal on Selected Areas in Communications*, vol. 13:pp. 1465–1480.

Brunberg, Henrik (2002). *Distributing digital radio over an IP backbone*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3317.
URL *http://www.ep.liu.se*

Chandran, Kartik; Raghunathan, Sundarshan; Venkatesan, Subbarayan; and Prakash, Ravi (2001). A feedback-based scheme for improving TCP performance in ad hoc wireless networks. In *IEEE Personal Communications*, vol. 8(1):pp. 34–39. ISSN 1070-9916.

Deb, Supratim and Srikant, R (2006). Rate-based versus queue-based models of congestion control. In *IEEE Transactions on Automatic Control*, vol. 51(4):pp. 606–619. ISSN 0018-9286.

Deng, Xidong; Yi, Sungwon; Kesidis, George; and Das, Chita R (2002). Stabilized virtual buffer (SVB) - an active queue management scheme for Internet quality-of-service. In *IEEE Global Telecommunications Conference, Globecom*, vol. 2, (pp. 1628–1632).

Elbornsson, Jonas (2003). *Analysis, Estimation and Compensation of Mismatch Effects in A/D Converters*. Ph.D. thesis, Linköpings universitet, Sweden. Thesis No. 811.
URL *http://www.control.isy.liu.se/*

Eldar, Yonina C (2003). Sampling with arbitrary sampling and reconstruction spaces and oblique dual frame vectors. In *Journal of Fourier Analysis and Applications*, vol. 9(1):pp. 77–96.

Eng, Frida and Gustafsson, Fredrik (2005a). Frequency transforms based on nonuniform sampling — basic stochastic properties. In *Radiovetenskap och Kommunikation*. Linköping, Sweden.

Eng, Frida and Gustafsson, Fredrik (2005b). System identification using measurements subject to stochastic time jitter. In *IFAC World Congress*. Prague, Czech Republic.

Eng, Frida and Gustafsson, Fredrik (2006). Bias compensated least squares estimation of continuous time output error models in the case of stochastic sampling time jitter. In *IFAC Symposium on System Identification (SYSID)*. Newcastle, Australia.

Eng, Frida and Gustafsson, Fredrik (2007a). Algorithms for down-sampling non-uniformly sampled data. In *European Signal Processing Conference (EU-SIPCO)*. Poznań, Poland. Submitted.

Eng, Frida and Gustafsson, Fredrik (2007b). Down-sampling non-uniformly sampled data. In *EURASIP Journal of Advances in Signal Processing*. Submitted.

Eng, Frida and Gustafsson, Fredrik (2007c). Identification with stochastic sampling time jitter. In *Automatica*. Provisionally accepted as regular paper.

Eng, Frida; Gustafsson, Fredrik; and Gunnarsson, Fredrik (2007). Frequency domain analysis of signals with stochastic sampling times. In *IEEE Transactions on Signal Processing*. Submitted.

Eriksson, Jonas (2004). *Providing Quality of Service for Streaming Applications in Evolved 3G Networks*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3493.
URL *http://www.ep.liu.se*

Fall, Kevin and Varadhan, Kannan (Eds.) (). The *ns* manual.
URL *http://nsnam.isi.edu/nsnam/*

Feichtinger, Hans G and Gröchenig, Karlheinz (1994). Theory and practice of irregular sampling. In Benedetto, John J and Frazier, Michael W (Eds.), *Wavelets: Mathematics and Applications*. CRC Press.

Feichtinger, Hans G; Gröchenig, Karlheinz; and Strohmer, Thomas (1995). Efficient numerical methods in non-uniform sampling theory. In *Numerische Matematik*, vol. 69(4):pp. 423–440.

Flachenecker, Peter; Hartung, Hans-Peter; and Reiners, Karlheinz (1997). Power spectrum analysis of heart rate variability in Guillain-Barre syndrome. a longitudinal study. In *Brain*, vol. 120(10):pp. 1885—1894.

Floyd, Sally (1994). TCP and explicit congestion notification. In *ACM Computer Communication Review*, vol. 24(5):pp. 10–23.

Floyd, Sally and Jacobson, Van (1993). Random early detection gateways for congestion avoidance. In *IEEE/ACM Transactions on Networking*, vol. 1(4):pp. 397–413.

Fredholm, Kenth and Nilsson, Kristian (2003). *Implementing an application for communication and quality measurements over UMTS network*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3369.
URL *http://www.ep.liu.se*

Grönwall, Christina (2006). *Ground Object Recognition using Laser Radar Data - Geometric Fitting, Performance Analysis, and Applications*. Ph.D. thesis, Linköpings universitet, Sweden. Thesis No. 1055.
URL *http://www.control.isy.liu.se/*

Gunnarsson, Frida (2000). *Problems and Improvements of Internet Traffic Congestion Control*. Master's thesis, Department of Electrical Engineering Linköpings universitet, Sweden. LiTH-ISY-EX-3098.

Gunnarsson, Frida (2003). On modeling and control of network queue dynamics. Licentiate Thesis No. 1048, Department of Electrical Engineering, Linköpings universitet, Sweden.

Gunnarsson, Frida; Gunnarsson, Fredrik; and Gustafsson, Fredrik (2001). TCP performance based on queue occupation. In *Towards 4G and Future Mobile Internet, Proceedings for Nordic Radio Symposium 2001*. Nynäshamn, Sweden.

Gunnarsson, Frida; Gunnarsson, Fredrik; and Gustafsson, Fredrik (2002). Issues on performance measurements of TCP. In *Radiovetenskap och Kommunikation*. Stockholm, Sweden.

Gunnarsson, Frida; Gunnarsson, Fredrik; and Gustafsson, Fredrik (2003). Controlling Internet queue dynamics using recursively identified models. In *IEEE Conference on Decision and Control*. Maui, Hawaii, USA.

Gunnarsson, Frida; Gustafsson, Fredrik; and Gunnarsson, Fredrik (2004). Frequency analysis using nonuniform sampling with applications to adaptive queue management. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Montreal, Canada.

Hertz, Paul L. and Feigelson, Eric D. (1997). A sample of astronomical time series. In Subba Rao, T.; Priestley, M. B.; and Lessi, O. (Eds.), *Applications of Time Series Analysis in Astronomy and Meteorology*. Chapman & Hall. ISBN 0-412-63800-2.
URL *http://xweb.nrl.navy.mil/timeseries/timeseries.html*

Hollot, C. V.; Misra, Vishal; Towsley, Don; and Gong, Wei-Bo (2001). A control theoretic analysis of RED. In *Joint Conference of the IEEE Computer and Communications Societies, INFOCOM*. Anchorage, Alaska, USA.

Jacobsson, Krister and Hjalmarsson, Håkan (2004). Estimation problems in congestion control. In *On the State of the Art in Controlled Bandwidth Sharing*. Euro-NGI.
URL *http://eurongi.enst.fr/archive/127/jra21_deliverable_final.pdf*

Jacobsson, Krister and Hjalmarsson, Håkan (2006). Closed loop aspects of fluid flow model identification in congestion control. In *IFAC Symposium on System Identification (SYSID)*. Newcastle, Australia.

Jacobsson, Krister; Hjalmarsson, Håkan; and Möller, Niels (2006). ACK-clock dynamics in network congestion control – an inner feedback loop with implications on inelastic flow impact. In *IEEE Conference on Decision and Control*. San Diego, San Fransisco, USA.

Jacobsson, Krister; Möller, Niels; Johansson, Karl-Henrik; and Hjalmarsson, Håkan (2004). Some modeling and estimation issues in control of heterogeneous networks. In *International Symposium on Mathematical Theory of Networks and Systems, MTNS*. Leuven, Belgium.

Jacod, Jean (1993). Random sampling in estimation problems for continuous Gaussian processes with independent increments. In *Stochastic Processes and their Applications*, vol. 44(2):pp. 181–204. ISSN 0304-4149.

Jain, Raj (1991). The art of computer systems performance analysis: Techniques for experimental design, measurement, simulation and modeling. In *ACM SIGMETRICS Performance Evaluation Review*, vol. 19(2):pp. 5–11.

Janik, Jean-Marie and Bloyet, Daniel (2004). Timing uncertainties of A/D converters: theoretical study and experiments. In *IEEE Transactions on Instrumentation and Measurement*, vol. 53(2):pp. 561–565. ISSN 0018-9456.

Kelly, Frank (2003). Fairness and stability of end-to-end congestion control. In *European Journal of Control*, vol. 9:pp. 159–176.

Keshav, Srinivasan (1997). *An Engineering Approach to Computer Networking*. Addison Wesley. Description of the REAL simulator.
URL *http://www.cs.cornell.edu/skeshav/real/*

Kilkki, Kalevi (1999). *Differentiated Services for the Internet*. Macmillan Technical Publishing. ISBN 1-57870-132-5.

Knutsson, Björn (2004). *Simulation of radio resource management for UMTS*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3414.
URL *http://www.ep.liu.se*

Kunniyur, Srisankar and Srikant, R (2004). Analysis and design of an adaptive virtual queue algorithm for active queue management. In *IEEE/ACM Transactions on Networking*, vol. 12(2):pp. 286–299.

Kybic, Jan; Blu, Thierry; and Unser, Michael (2002a). Generalized sampling: a variational approach. I. Theory. In *IEEE Transactions on Signal Processing*, vol. 50(8):pp. 1965–1976. ISSN 1053-587X.

Kybic, Jan; Blu, Thierry; and Unser, Michael (2002b). Generalized sampling: a variational approach. II. Applications. In *IEEE Transactions on Signal Processing*, vol. 50(8):pp. 1977–1985.

Leiner, Barry M; Cerf, Vinton G; Clark, David D; Kahn, Robert E; Kleinrock, Leonard; Lynch, Daniel C; Postel, Jon; Roberts, Larry G; and Wolff, Stephen (2000). A brief history of the Internet.
URL *www.isoc.org/internet/history/brief.shtml*

Liu, Jian and Singh, Suresh (2001). ATCP: TCP for mobile ad hoc networks. In *IEEE Journal on Selected Areas in Communications*, vol. 19(7):pp. 1300–1315. ISSN 0733-8716.

Ljung, Lennart (1999). *System Identification – Theory for the User*. Prentice Hall, 2 ed.

Lomb, N R (1976). Least-squares frequency analysis of unequally spaced data. In *Astrophysics and Space Science*, vol. 39:pp. 447–462.

Low, Steven H; Paganini, Fernando; and Doyle, John C (2002). Internet congestion control. In *IEEE Control Systems Magazine*, vol. 22(1):pp. 28–43. ISSN 0272-1708.

Marvasti, Farokh (1987). *Zero-Crossings and Nonuniform Sampling of Single and Multidimensional Signals and Systems*. Nonuniform Publications.

Marvasti, Farokh (1996). Nonuniform sampling theorem for bandpass signals at or below the Nyquist density. In *IEEE Transactions on Signal Processing*, vol. 44(3):pp. 572–576. ISSN 1053-587X.

Marvasti, Farokh (Ed.) (2001). *Nonuniform Sampling: Theory and Practice*. Kluwer Academic Publishers.

Marvasti, Faroukh; Analoui, Mostafa; and Gamshadzahi, Mohsen (1991). Recovery of signals from nonuniform samples using iterative methods. In *IEEE Transactions on Signal Processing*, vol. 39(4):pp. 872–878.

Mascolo, Saverio (1999). Congestion control in high-speed communication networks using the Smith principle. In *Automatica*, vol. 35(12).

Mascolo, Saverio; Casetti, Claudio; Gerla, Mario; Sanadidi, M. Y.; and Wang, Ren (2001). TCP westwood: Bandwidth estimation for enhanced transport over wireless links. In *International Conference on Mobile Computing and Networking*, (pp. 287 – 297). Rome, Italy.

Masry, Elias (1978a). Alias-free sampling: An alternative conceptualization and its applications. In *IEEE Transactions on Information Theory*, vol. 24(3):pp. 317–324.

Masry, Elias (1978b). Poisson sampling and spectral estimation of continuous-time processes. In *IEEE Transactions on Information Theory*, vol. 24(2):pp. 173–183.

Masry, Elias and Cambanis, Stamatis (1981). Consistent estimation of continuous-time signals from nonlinear transformations of noisy samples. In *IEEE Transactions on Information Theory*, vol. 27(1):pp. 84–96. ISSN 0018-9448.

Masry, Elias; Klamer, Dale; and Mirabile, Charles (1978). Spectral estimation of continuous-time processes: Performance comparison between periodic and Poisson sampling schemes. In *IEEE Transactions on Automatic Control*, vol. 23(4):pp. 679–685. ISSN 0018-9286.

Masry, Elias and Lui, Ming-Chuan C (1976). Discrete-time spectral estimation of continuous-parameter processes – a new consistent estimate. In *IEEE Transactions on Information Theory*, vol. 22(3):pp. 298–312.

Mednieks, Ints (1999). Methods for spectral analysis of nonuniformly sampled signals. In *International Workshop on Sampling Theory and Application*. Loen, Norway.

Meyer, Michael (1999). TCP performance over GPRS. In *IEEE Wireless Communications and Networking Conference*. New Orleans, Louisiana, USA.

Misra, Archan; Baras, John S; and Ott, Tennis (1999). The window distribution of multiple TCP's with random loss queues. In *IEEE Global Telecommunications Conference, Globecom*, vol. 3, (pp. 1714–1726). Rio de Janeiro, Brazil.

Nilsson, Martin (2007). *Estimation of Radial Runout*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX–07/3949–SE.
URL *http://www.ep.liu.se*

NS (2003). Ucb/lbnl/vint network simulator, *ns-2*.
URL *http://nsnam.isi.edu/nsnam/*

OPNET (2007). OPNET Technologies Co., Ltd.
URL *http://www.opnet.com*

van der Ouderaa, Edwin and Renneboog, Jean (1988). Some formulas and applications of nonuniform sampling of bandwidth-limited signals. In *IEEE Transactions on Instrumentation and Measurement*, vol. 37(3):pp. 353–357. ISSN 0018-9456.

Padhye, Jitendra; Firoiu, Victor; Towsley, Don; and Kurose, Jim (1998). Modeling TCP throughput: A simple model and its empirical validation. In *Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, ACM SIGCOMM*. Vancouver, Canada.

Papenfuss, Frank; Artyukh, Yuri; Boole, E; and Timmermann, Dirk (2003). Optimal sampling functions in nonuniform sampling driver designs to overcome the Nyquist limit. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Hong Kong.

Papoulis, Athanasios (1977). *Signal Analysis*. McGraw-Hill.

Park, Kyung-Joon; Lim, Hyuk; Başar, Tamer; and ho Choi, Chong (2003). Anti-windup compensator for active queue management in TCP networks. In *Control Engineering Practice*, vol. 11(10):pp. 1127–1142.

Peisa, Janne and Meyer, Michael (2001). Analytical model for TCP file transfers over UMTS. In *IEEE Third Generation Wireless Communications Conference (3G'Wireless)*. San Francisco, California, USA.

Persson, Katarina (2002a). *TCP/IP in tactical ad hoc networks*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3206.
URL *http://www.ep.liu.se*

Persson, Niclas (2002b). Event Based Sampling with Application to Spectral Estimation. Licentiate Thesis No. 981, Department of Electrical Engineering, Linköpings universitet, Sweden.
URL *http://www.control.isy.liu.se/*

Potts, Daniel; Steidl, Gabriele; and Tasche, Manfred (2001). Fast Fourier transform for nonequispaced data: A tutorial. In Benedetto and Ferreira (2001), chap. 12.

Pribić, Radmila (2004). Radar irregular sampling. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Montreal, Canada.

Qi, Yuan; Minka, Thomas; and Picard, Rosalind (2002). Bayesian spectrum estimation of unevenly sampled nonstationary data. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Orlando, Florida, USA.

RFC 2474 (1998). Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers. Proposed standard. K. Nichols, S. Blake, F. Baker and D. Black.

RFC 2581 (1999). TCP Congestion Control. Proposed standard. M. Allman, V. Paxson and W. Stevens.

RFC 3168 (2001). The Addition of Explicit Congestion Notification (ECN) to IP. Proposed standard. K. Ramakrishnan, S. Floyd and D. Black.

Romano, Maria; Cesarelli, Mario; Bifulco, Paolo; Sansone, Mario; and Bracale, Marcello (2003). Study of fetal autonomous nervous system's response by means of FHRV frequency analysis. In *International IEEE EMBS Conference on Neural Engineering*, (pp. 399–402). Capri Island, Italy.

Roques, Sylvie and Thiebaut, Carole (2003). Spectral contents of astronomical unequally spaced time-series: contribution of time-frequency and time-scale analyses. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 6, (pp. 473–6). Hong Kong. ISSN 1520-6149.

Russell, Andrew I (2002). *Regular and Irregular Signal Resampling*. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.

Sachs, Joachim and Meyer, Michael (2001). Mobile Internet - performance issues beyond the radio interface. In *Aachen Symposium on Signal Theory (ASST)*. Aachen, Germany.

Sands, Kenneth E F; Appel, Marvin L; Lilly, Leonard S; Schoen, Frederick J; Mudge, Jr, Gilbert H; and Cohen, Richard J (1989). Power spectrum analysis of heart rate variability in human cardiac transplant recipients. In *Circulation*, vol. 70:pp. 76—82.

Scargle, Jeffrey D (1982). Studies in astronomical time series analysis II: Statistical aspects of spectral analysis of unevenly spaced data. In *The Astrophysical Journal*, vol. 263:pp. 835–853.

Shapiro, Harold S and Silverman, Richard A (1960). Alias-free sampling of random noise. In *Journal of the Society for Industrial and Applied Mathematics*, vol. 8(2):pp. 225–248.

Souders, T Michael; Flach, Donald R; Hagwood, Charles; and Yang, Grace L (1990). The effects of timing jitter in sampling systems. In *IEEE Transactions on Instrumentation and Measurement*, vol. 39(1):pp. 80–85. ISSN 0018-9456.

van Steenis, H. G. and Tulen, J. H. M. (1991). A comparison of two methods to calculate the heart rate spectrum based on non-equidistant sampling. In *International Conference of the IEEE Engineering in Medicine and Biology Society*. Philadelphia, Pennsylvania, USA.

Stevens, W Richard (1994). *TCP/IP illustrated vol 1: The protocols*. Addison Wesley.

Tarczynski, Andrzej and Allay, Najib (2004). Spectral analysis of randomly sampled signals: Suppression of aliasing and sampler jitter. In *IEEE Transactions on Signal Processing*, vol. 52(12):pp. 3324–3334. ISSN 1053-587X.

Tronarp, Otto (2003). *Quality of Service in Ad Hoc Networks by Priority Queuing*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3343.
    URL *http://www.ep.liu.se*

Vandewalle, Patrick; Sbaiz, Luciano; Vandewalle, Joos; and Vetterli, Martin (2004). How to take advantage of aliasing in bandlimited signals. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. Montreal, Canada.

Verbeyst, Frans; Rolain, Yves; Schoukens, Johan; and Pintelon, Rik (2006). System identification approach applied to jitter estimation. In *Instrumentation and Measurement Technology Conference*, (pp. 1752 – 1757). Sorrento, Italy. ISSN 1091-5281.

Wallin, Johan (2005). *Catheter-Compensated Blood Pressure Measurement*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3620.
URL *http://www.ep.liu.se*

Wikström, Martin (2005). *Compensating for Respiratory Artifacts in Blood Pressure Waveforms*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3654.
URL *http://www.ep.liu.se*

Wojtkiewicz, Andrzej and Tuszyński, Michal (1992). Application of the Dirichlet transform in analysis of nonuniformly sampled signals. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. San Fransisco, California, USA.

Yao, Kung and Thomas, John B (1967). On some stability and interpolatory properties of nonuniform sampling expansions. In *IEEE Transactions on Circuits and Systems [legacy, pre - 1988]*, vol. 14(4):pp. 404–408. ISSN 0098-4094.

Zakon, Robert Hobbes' (2003). Hobbes' Internet timeline v6.0.
URL *http://www.zakon.org/robert/internet/timeline/*

Zhao, Haichuan and Wu, Jian Qiu (2005). *Implementation and simulation of HSDPA functionality with ns-2*. Master's thesis, Department of Electrical Engineering, Linköpings universitet, Sweden. LiTH-ISY-EX-3647.
URL *http://www.ep.liu.se*

Zhu, Chengyu; Yang, Oliver W W; Aweya, Lames; Ouellette, Michel; and Montuno, Delfin Y (2002). A comparison of active queue management algorithms using the OPNET modeler. In *IEEE Communications Magazine*, vol. 40(6):pp. 158–167. ISSN 0163-6804.