

Enhanced Network-State Estimation using Change Detection

Erik Hartikainen^{1,2} and Svante Ekelin^{2,3}

¹ *Dept. of Science and Technology, Linköping University, Norrköping, Sweden*

² *Network Control Lab, Ericsson Research, Stockholm, Sweden*

³ *Dept. of Computer Science and Electronics, Mälardalen University, Västerås, Sweden*

Abstract

This paper presents the concept of change detection for filter-based network-state estimation. This could be useful in various contexts; two examples are network management and adaptive applications. In particular, it is shown that the performance of available-bandwidth estimation can be significantly enhanced by employing a change-detection technique in conjunction with a filter-based estimator. By using filter-based approaches, it is possible to track the state of communication systems, and to estimate network properties in real-time. A virtue of filter-based methods is the ability to enhance the estimation performance by combining them with change detection. This makes it feasible to overcome the trade-offs regarding speed of adaptation to changes versus stable estimation. We discuss filtering and change detection in general, and illustrate the power of this combination with the filter-based available-bandwidth estimator BART enhanced by the light-weight change-detection test CUSUM.

1. Introduction

1.1. Overview

In recent years, there has been a strong interest in the field of network management. Networks and their use, as well as their misuse, become increasingly more complex. There is a need for developing tools for providing network operators and applications with comprehensive information regarding fault diagnosis, performance tuning, network monitoring etc.

In order to achieve and maintain good network performance, it is necessary that decisions and actions are based upon accurate and reliable information, preferably obtained in real-time. However, the tools that are used for this purpose typically need to be configured to perform well under specific conditions, and consequently perform suboptimally when the

conditions are different.

One interesting area within network management is bandwidth measurements. In particular, there exist several tools for estimating the end-to-end available bandwidth along network paths, which can be useful when monitoring and adapting the network traffic flow to the unutilized capacity. However, due to the dynamic and time dependent characteristics of network traffic, it is indeed a challenge (not least if real-time estimation is required) to always deliver high quality estimates, as the network state can be subject to both fast and slow variations of different magnitudes.

As a possible way of accomplishing continuous available-bandwidth monitoring of a path in a packet-switched communication network, a filter-based approach BART (Bandwidth Available in Real-Time) was suggested and partly evaluated in [1]. BART uses active probing and Kalman filtering in order to maintain and update an estimate of the current available bandwidth. It may be configured for optimum performance with regard to the expected variability of the system state [2].

Using BART with a static configuration limits the range of system-state variability where estimation performance is good, just as for many other statistical estimation tools. The problem is, however, that the state variability is not always constant. Typically, it is constant and low for most of the time, but transiently and unpredictably the variability increases sharply when there is a real change of the system state, such as a sudden increase or decrease of the available bandwidth. The weakness with a static configuration often manifests as either fluctuating estimates when the real state is rather stable, or slow adaptation in case of abrupt changes in the system, depending on the choice of fixed estimator tuning.

In order to obtain filter-based tools featuring both fast adaptation to real changes and stable estimation in the normal case of slowly changing or stationary system state, change detection may be used. A change-detection technique has the ability of testing the

hypothesis that a change has occurred versus that a change has not occurred in the system state. Hence, it is appropriate to tune the filter for stability by default, and adjust for agile adaptation only when the change-detection test indicates a change.

Sudden link failures causing re-routing, rapidly changing quality of radio links in wireless communications, and various types of network attacks are examples of events that could abruptly influence the available-bandwidth situation, and probably also other network properties. As will be shown in this paper, the performance of BART can be significantly improved in such situations, when assisted by a change-detection technique.

Although we will exemplify our reasoning with available-bandwidth experiments, we believe that filter-based estimation and change detection could be useful for many other applications within computer networking, especially when reliable and accurate real-time estimation is required.

1.2. Related work

Besides BART [1], several other alternatives exist for measuring the available bandwidth in packet-switched networks, e.g. [3, 4, 5, 6, 7, 8, 9, 10]. These tools are not filter based, and the majority do not have the capability of providing real-time estimates.

Regarding change detection for filter-based tools, a number of techniques have been proposed throughout the years, e.g. [11, 12, 13, 14, 15]. A comparative study on change-detection techniques for automotive applications has been presented in [16].

In computer networking, change detection together with a filter-based estimator has been used in [17] for estimation of round-trip time for use in congestion control.

1.3. Paper organization

In the next section of this paper, we provide the reader with some background theory regarding filter-based estimation and the available-bandwidth estimation method BART.

In the third section, we define the problem and limitation of using statistical estimation methods in systems with highly irregular characteristics, and in particular, the trade-off between stable and agile estimation.

The fourth section introduces a possible solution to the problem, i.e. using change detection together with filter-based estimation. Also, we describe how a simple change-detection technique, the Cumulative Sum (CUSUM) test, can be integrated with BART.

The fifth section describes experiments performed with BART assisted by the CUSUM test. Measurements have been carried out in a controlled laboratory network as well as over the Internet.

Finally, the paper is summarized with a general discussion and conclusions drawn from the experiments.

2. Background

This section will provide the reader with background theory which hopefully will facilitate the understanding of the remaining part of the paper. First, the basic concept regarding filter-based estimation will be introduced, and thereafter, a short introduction to BART will be given.

2.1. Filter methods

In a filter-based approach, the state of a system is estimated from repeated measurements of some quantity dependent on the system state. This requires models of how the system state evolves from one measurement occasion to the next, and also how the measured quantity depends on the system state. The system equations can be expressed as¹

$$x_k = f(x_{k-1}) + w_{k-1} \quad (1)$$

$$z_k = h(x_k) + v_k \quad (2)$$

where x is the state of the system, z is the measured quantity, w is the process noise and v is the measurement noise. The functions f and h represent the system evolution model and the measurement model, respectively. The subscript refers to discrete time.

A filter is a procedure which takes a previous estimate \hat{x}_{k-1} and a new measurement z_k as input, and calculates a new estimate \hat{x}_k of the system state. A compelling property of filters is that they are capable of producing estimates in real-time, i.e. tracking the system state. For each new measurement, the previous estimate is updated.

If the functions f and h are linear, and if both the process noise and the measurement noise are Gaussian and uncorrelated, there is an optimal filter, namely the Kalman filter [18]. Experience has shown that Kalman filters often work very well, even when these conditions are not strictly met.

In this linear case the system equations (1) and (2) can be expressed using matrices:

$$x_k = Ax_{k-1} + w_{k-1} \quad (3)$$

¹ This can be formulated more generally, when the system is also influenced by a control input.

$$z_k = Hx_k + v_k \quad (4)$$

From the previous estimate and the new measurement, the Kalman filter equations allow estimation of the system state x and the error covariance matrix P as:

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (5)$$

$$P_k = (I - K_k H)P_k^- \quad (6)$$

where

$$\hat{x}_k^- = A\hat{x}_{k-1} \quad (7)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (8)$$

and

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1}. \quad (9)$$

The superscript minus sign indicates the a priori estimate at discrete time k . For further details, see [18].

Kalman filtering can be understood as a process where there are two phases of calculation in each iteration. First, there is a “prediction” phase, where the previous estimate evolves one discrete time step according to the system model (7). Then, there is a “correction” phase, where the new measurement is taken into account (5). One also computes the updated error covariance matrix P_k of the state estimate (6).

The difference $z_k - H\hat{x}_k^-$ in (5) is known as the residual (or innovation), which is essential in the “correction” phase. The residual reflects the deviation of the actual measurement from what is predicted according to the measurement model and the evolved state estimate.

The Kalman gain K_k , given by (9) and appearing in (5) and (6), can be interpreted as the relative weight given to the new measurement as opposed to the pure expected evolution of the previous estimate. As can be seen from (8) and (9), the Kalman gain increases as either Q increases or R decreases. These required inputs to the Kalman filter are the covariances of the process noise w and measurement noise v , respectively. An intuitive understanding of the importance of these quantities may be acquired by the following arguments:

- Large variations of the noise in the system model (high Q) imply that the prediction according to the system model is likely to be less accurate, and the new measurement should be weighted heavier.
- Large variations in the measurement noise (high R) imply that the new measurement is likely to be less accurate, and the prediction should be weighted heavier.

2.2. BART (Bandwidth Available in Real-Time)

The BART method for available-bandwidth estimation [1] makes use of active probing and Kalman filtering. Updated estimates are produced based on probe-packet measurements, which are performed at randomized rates in order to improve the statistical estimation properties. Each measurement consists of sending a sequence of pairs of packets, which are time-stamped on sending and on arrival. As the measured quantity z we use the average relative increase in packet-pair time separation, also referred to as the inter-packet separation strain ε . Furthermore, it is possible to compute the variance of ε , which can be used as an estimate of R . From a simple fluid-traffic, first-come-first-served network model [10], the expectation value of ε is zero when the probing rate u is less than the available bandwidth B , and grows in proportion to the overload when the probing rate is larger:

$$\varepsilon = \begin{cases} 0 & (u < B) \\ \alpha(u - B) = \alpha u + \beta & (u \geq B) \end{cases} \quad (10)$$

In order to use dimensionless quantities in the BART filtering, u is normalized with respect to the chosen maximum probe intensity u_{max} .

It should be noted that the Kalman-filter method is very “forgiving”, and good results are often produced even when the ideal conditions are slightly broken. So, even if a real system displays characteristics which deviate somewhat from the piecewise linear system curve (see (10) and Figure 1), the resulting available-bandwidth estimate is not automatically invalidated.

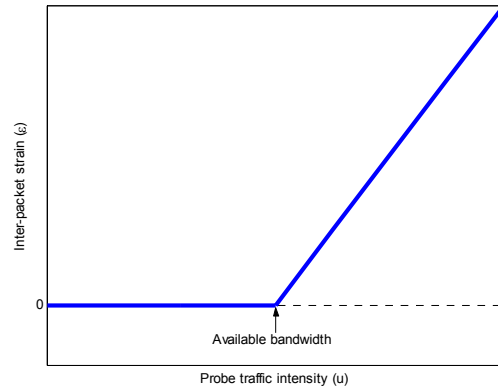


Figure 1. Based on measurements of the inter-packet separation strain and the intensity of the probe traffic, it is feasible to estimate the available bandwidth.

The model (10) allows for application of a Kalman filter in the overload region, when we represent the state of the system by a vector containing the two

parameters of the sloping straight line

$$x = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}. \quad (11)$$

We may then write, for the measurement z_k of the strain at discrete time k ,

$$z_k = Hx_k + v_k \quad (12)$$

where

$$H = [u \quad 1]. \quad (13)$$

Also, we write for the evolution of the system state

$$x_k = x_{k-1} + w_{k-1} \quad (14)$$

which means that we may apply the Kalman filter formalism with $A = I$ (the identity matrix).

When the filter estimates the system state variables α and β , we immediately obtain an estimate of the available bandwidth B , which corresponds to the breakpoint in the piecewise linear curve in Figure 1.

3. Problem

3.1. Stability versus agility

When dealing with dynamic systems, a static configuration of statistical measurement tools typically cannot optimally deal with both slowly changing conditions and rapid fluctuations. A tool configured for stability will suffer in case of sudden changes in the system state, due to slow adaptation. A tool with agile properties delivers fluctuating estimates in a steady-state situation, due to the measurement noise.

If there is no compelling need for always obtaining reliable estimates, it could be sufficient to use a tool configured for either stability or agility. As an example, for an alarm system, the main concern is most likely to give fast and clear indications when abnormalities occur in the system, whereas a precise measurement of the actual system state could be less important. In other applications, reliable steady-state estimates could be of great importance, while the time required for adaptation to sudden changes is less significant.

However, we believe that many applications would benefit from tools which have the ability to deliver reliable estimates under various conditions.

As an illustration, consider Figure 2, where the available-bandwidth measurement tool BART is estimating the unutilized capacity of a network path. The dotted curve corresponds to a high-agility configuration². This is accomplished by choosing Q

² $\lambda = 1$ in the configuration of Q , see (20) in Section 5.

large. It is obvious that BART easily follows the sudden changes in the available bandwidth, whereas the precision of the estimates leaves a great deal to be desired, due to amplification of the measurement noise. The dashed curve corresponds to an opposite configuration of high stability³. Here, we can see that BART is insensitive to the measurement noise and delivers stable and reliable estimates in the interval from 0 to 750 seconds. However, the configuration does not allow for fast adaptation, which is obvious when considering the estimates after 750 seconds, where BART is very slow in adapting to the new system state.

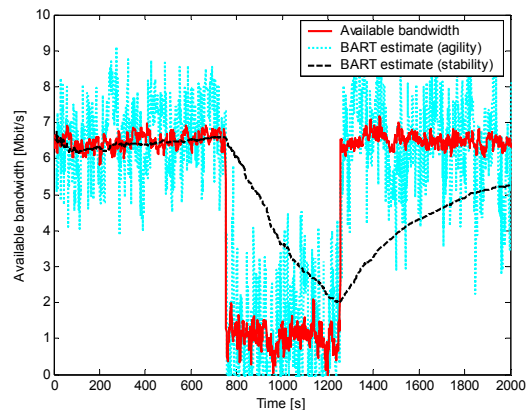


Figure 2. BART estimation of the available bandwidth using two configurations of Q , for different tracking ability.

Streaming of real-time data over a network path is an application which would probably benefit from a tool such as BART. Since congestion and packet loss are detrimental for real-time data, it would be beneficial if the sender could regulate the transmission rate with respect to the prevailing network conditions; for example by changing codec in case of audio or video streaming. A BART configuration corresponding to the dashed curve in Figure 2 would probably be suitable when estimating the available bandwidth over a network path, given slowly changing network characteristics. However, in case of abrupt changes, as in Figure 2, it is desirable that the estimates follow the variations such that a streaming application connected to BART has the possibility of adjusting the transmission intensity in case it is needed. An agile BART configuration is undesirable, since it constantly would trigger adjustment of the streaming codec due to exaggerated fluctuations in the available-bandwidth estimates. Nevertheless, it would be useful to transiently apply a configuration corresponding to the dotted curve in Figure 2, in order to rapidly follow abrupt changes in the network conditions.

³ $\lambda = 10^{-5}$ in the configuration of Q , see (20) in Section 5.

Although the BART configurations used in Figure 2 could be seen as two rather extreme choices (it is of course possible to choose Q such that the estimates become something in between the two curves in Figure 2), there is a clear trade-off between having stable but slowly adapting estimates and agile but noise-sensitive estimation properties.

4. Method

There are solutions to the problem described in Section 3. In the area of system identification using filter-based estimation methods, these solutions are often referred to as adaptive control and, more specifically, change detection algorithms (also known as fault detection).

In the following subsections, change detection will be introduced and exemplified by one particular algorithm; also, we will explain how this technique can be integrated with the previously described tool BART, for estimating available bandwidth.

4.1. Change detection

There are a number of existing filter-based methods available for detecting abrupt changes in dynamic systems, e.g., [11, 12, 13, 14, 15]. These methods can, in general, be divided into three classes: methods using one filter, where a whiteness test is applied to the filter residuals; methods using two filters, where a slow filter configured for stability runs in parallel with a fast filter configured for agility; methods using multiple filters, where each filter configuration is matched to a particular assumption regarding the abrupt change. For more information regarding these techniques and change detection in general, see [19].

There is a trade-off in complexity versus performance when comparing different change-detection techniques; in general, the more complex algorithm, the better result is obtained [16]. The computational complexity is often proportional to the number of filters required by the algorithm. A simple and intuitive technique is known as the Cumulative Sum (CUSUM) test [11, 19], which only uses information from one filter in order to make decisions regarding the hypotheses of change or no change in the system. This can be compared to, e.g., the full Generalized Likelihood Ratio (GLR) test [14, 19], where the number of required filters (and thus also the complexity) grows linearly in time.

When discussing change detection in general, it is not of great importance which particular filter to use. Instead, the majority of the available methods are only concerned about the residuals that are obtainable from

the used filter.

The characteristics of filter residuals can be examined for indications of the filter's state estimation being "off track". A filter residual is defined as the difference between the measured quantity at time k and the prediction of this using the a priori estimate of the system state at time k . In a Kalman filter, the residual is used in the "correction" phase; see the parenthesis in (5).

An important property of filter residuals is that, under certain model assumptions, they resemble white noise before a change occurs. This characteristic is often useful in change detection.

In an ideal system, the residuals would be zero before a change and non-zero after a change. However, due to measurement noise and process noise, which are unavoidable in statistical approaches, this is not the case in reality and the residuals cannot be predicted beforehand. Nevertheless, conclusions may be drawn from the statistical behavior of the residuals.

Assuming that there is no change in the system state, and given that the used model is correct, the residuals can be interpreted as independent stochastic variables with zero mean and a certain variance, which characterize white noise. After a change in the system state, the distribution of the residuals also changes. The challenge in statistical change detection is to design a hypothesis test, or "stopping rule", allowing to distinguish between random and systematic deviation of the residuals.

In the CUSUM test, which is a very simple change-detection algorithm, filter residuals are transformed into distance measures with different characteristics corresponding to desired design properties [19]. One distance measure (defined at time k), suitable for detecting changes in the mean of the filter residuals, is

$$s_k = \frac{z_k - H_k \hat{x}_k^-}{\sqrt{H_k P_k^- H_k^T + R_k}} \quad (15)$$

where the residual (i.e., the numerator) is normalized to unit variance in order to make the design somewhat more robust. This distance measure s_k is used to build up the test statistic

$$g_k = \max(g_{k-1} + s_k - v, 0) \quad (16)$$

which is used to decide whether the residuals are reflecting a positive trend; i.e., the predictions of the filter are systematically underestimating the measured quantity. The design parameter v is used to allow for a slowly varying system state; v is also referred to as the drift parameter.

Once the value of the test statistic g exceeds the

design threshold h , CUSUM will issue an alarm that a change has occurred in the system, and the filter procedure is recommended to act appropriately to handle the situation.

A suitable filter action when receiving a change-detection alarm is to momentarily increase the estimate of the process noise covariance matrix Q . By doing this, the filter will put more trust in the received measurement as compared to the previous estimated system state and, consequently, it will be quicker to follow variations in the system state. After adaptation, it is appropriate to return to a filter configuration characterized by stability, i.e., using a smaller Q .

4.2. BART using CUSUM

The available-bandwidth measurement tool BART has been implemented and evaluated with the change-detection technique CUSUM.

Since the residual of the BART Kalman filter can be both positive and negative, we use a two-sided test. Hence, at time k , the positive and negative test statistics are computed as

$$g_k^{pos} = \max(g_{k-1}^{pos} + s_k - v, 0) \quad (17)$$

and

$$g_k^{neg} = \min(g_{k-1}^{neg} + s_k + v, 0) \quad (18)$$

where $g_0 = 0$.

For each discrete time step, the test statistics are recalculated and the stopping rule is applied; the test statistics are compared to the chosen design threshold h :

$$\left. \begin{array}{l} \text{if } (g_k^{pos} > h) \text{ or } (g_k^{neg} < -h) \\ \text{then } \rightarrow g_k^{pos} = 0 \\ \quad \rightarrow g_k^{neg} = 0 \\ \quad \rightarrow \text{Alarm indication} \end{array} \right\} \quad (19)$$

The other design parameter, the drift v , can be seen as an indicator for what is expected to be regular random deviation between the measurements and the predictions. If the distance measure (15) is smaller than the drift parameter v , the observed residual will be interpreted as a normal statistical fluctuation and, consequently, not contribute to the CUSUM test statistics.

The choice of design parameters h and v affect the performance of the change detector, especially in terms of false alarm rate (FAR) and mean time to detection (MTD).

It should be noted that h and v do not necessarily have to be equal for the positive and the negative CUSUM test (i.e., we could have different h^{pos} , h^{neg} , v^{pos} , v^{neg}). However, for our purposes we have not had any compelling reason to introduce extra parameters by allowing them to differ.

When running BART (configured for stability) without change detection, the estimates in case of abrupt changes in the system can appear as illustrated by the dashed curve in Figure 2. It is obvious that BART is not responding quickly when rapid changes occur in the system, due to its configuration for stability.

When running BART with CUSUM for the scenario in Figure 2, and the design threshold $h = \infty$ (i.e., no alarm will ever occur), the positive and negative test statistics behave as in Figure 3, using different drift parameter v .

In Figure 3 it is clear that the positive test statistic (solid curves) starts to increase after 750 seconds, due to the sudden drop in the available bandwidth. The test statistic behaves differently with respect to the chosen drift parameter v ; larger v requires larger residuals in order to increase the test statistic. After 1250 seconds, the system returns to its previous state and, consequently, the positive test statistic decreases due to the lack of positive contributing distance measures. Instead, the filter residuals are facing a negative trend (since the BART estimate is below the true available bandwidth in Figure 2), which can be seen in Figure 3 by studying the negative test statistic (dashed curves).

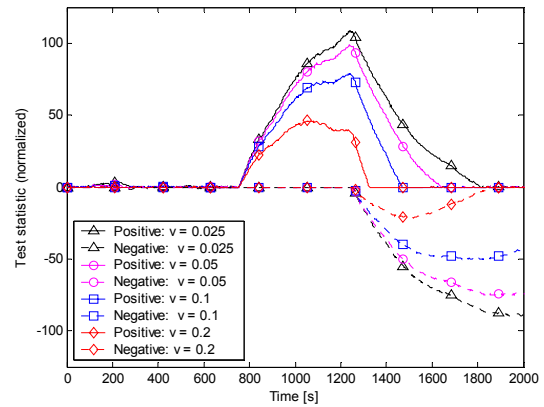


Figure 3. The CUSUM test statistics (both positive and negative) for different v when BART (configured for stability) is exposed to abrupt changes in the system, according to Figure 2.

From Figure 3, we can see that for the used drift v , a threshold value of e.g. $h = 50$ is probably an inconvenient choice, since the CUSUM test would not even alarm for the case when $v = 0.2$, and for the other cases, the MTD would be very large. By decreasing h , the performance is expected to improve, in terms of

shorter time to detection. However, going too low would cause false alarms; e.g. $h = 2$ would result in false alarm after approximately 200 seconds, when using $\nu = 0.025$.

5. Results

In order to evaluate the performance of BART when applying change detection, experiments have been performed in a controlled laboratory network and over an Internet path.

5.1. Experiment setup

In the laboratory network, a testbed consisting of two Extreme Summit routers has been used, see Figure 4. We have studied several traffic cases including different cross-traffic aggregation levels and statistical distributions of the inter-packet arrival times. However, for the scope of this paper and for clarity, we limit the illustrations of the results to different versions of one particular cross-traffic case. General conclusions drawn from this scenario are applicable to all other cases that have been under consideration.

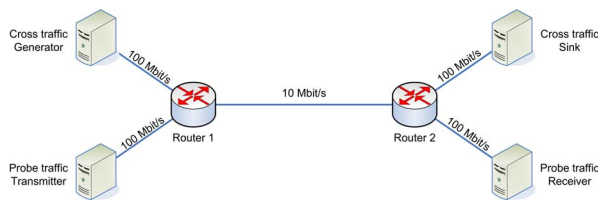


Figure 4. An illustration of the testbed setup that was used in the evaluation of BART with change detection.

The chosen testbed scenario for this paper emulates a cross-traffic case with approximately 100 simultaneously active traffic streams, which transmit UDP packets⁴ with inter-arrival times following a Pareto distribution (shape parameter = 1.9). New connections arrive according to a Poisson process and remain active according to a Pareto distribution (shape parameter = 1.5, mean = 1.0 second). The cross-traffic generator (which is the Ericsson proprietary tool IPTrafGen) is configured such that all active connections utilize (on average) an equal and predetermined amount of the bandwidth capacity.

In order to construct abrupt changes in the available bandwidth, a second instance of the same cross-traffic scenario suddenly affects the testbed for a time interval of 500 seconds. This additional traffic flow has statistical properties as described above, except that the

traffic intensity for each connection is somewhat higher for this latter case.

By abruptly adding 100 new traffic streams along the path, the available bandwidth will rapidly decrease. A similar effect could occur in a real network in case of, e.g., a denial-of-service attack or a nearby link failure (causing re-routing of the traffic).

For the testbed measurements, the true cross traffic was recorded using the standard tool `tcpdump`, while the probe-traffic receiver host recorded the series of strain measurements z_k . Based upon the recorded cross traffic, the true available bandwidth could be calculated by subtracting the cross-traffic intensity from the tight link capacity (which was known due to the setup). The cross-traffic intensity was obtained by averaging the content of the cross-traffic traces, using a sliding window with averaging time scale $\tau = 4$ seconds.

In the final experiment presented in this paper, BART with and without change detection estimates the available bandwidth over an Internet path in Sweden. The probe traffic was transmitted from a sender at Linköping University, which connects to the Swedish University Network (SUNET), to a probe receiver at Ericsson Research in Stockholm, connected to the Internet service provider Telia. From `traceroute`, we know that the path included 16 layer-3 hops.

In the Internet experiment, additional cross traffic was added along the path, in order to create sudden changes to the available bandwidth. This was accomplished by using `tcpreplay`⁵ together with a traffic trace obtainable from MAWI⁶. The time stamps of the trace file were re-scaled with different scaling factors, in order to get cross-traffic intensities of diverse magnitudes, which in turn caused steps of various sizes in the available bandwidth.

For all experiments in this paper, BART has been configured to transmit sequences of pairs of 1500-byte probe packets, organized as trains of 17 packets (16 pairs). The traffic intensity for each probe train was randomly chosen using a uniform distribution from 1 Mbit/s to 20 Mbit/s in the testbed measurements, and 10 Mbit/s to 100 Mbit/s in the Internet experiment. The inter-departure time between two consecutive probe trains was set to one second, which yields an average probe-traffic overhead of 0.2 Mbit/s, in both cases.

In the used implementation of BART, the refinement possibility of providing the Kalman filter with a precision measurement of R has not been used; instead, this filter input parameter was set to $R = 1$ for

⁴ The distribution of the synthetic cross traffic packet sizes roughly corresponds to observations from Sprint: <http://ipmon.sprintlabs.com/packstat/packetoverview.php> (June 2006)

⁵ <http://tcpreplay.synfin.net/> (June 2006)

⁶ <http://tracer.csl.sony.co.jp/mawi/samplepoint-B/2005/200504211400.html> (June 2006)

all measurements, i.e., from the filter's perspective all strain measurements were considered to be of equal quality and treated accordingly.

Regarding the process noise covariance parameter Q , which is crucial for the tracking ability of BART, the following simple form has been used:

$$Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} = \lambda \cdot I = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \quad (20)$$

It should be noted that Q , which is a symmetric matrix, has three degrees of freedom and Q_{11} and Q_{22} do not necessarily have to be equal. For a more detailed discussion regarding Q in BART, see [2].

In this paper, the used Q differs depending on desired characteristics of the BART estimation. In the remainder of this section, we will refer to the notation $Q_{\text{stability}}$ and Q_{alarm} when describing the steady-state estimation of BART and the characteristics in case of a change-detection alarm, respectively.

Regarding $Q_{\text{stability}}$, $\lambda = 10^{-5}$ has been used for all experiments in this paper; for Q_{alarm} , different values of λ were applied.

For all measurements where BART uses change detection, the two-sided CUSUM test has been applied with design threshold $h = 3$ and drift parameter $\nu = 0.05$, unless otherwise is specified.

5.2. Testbed measurements

In Figure 5, we illustrate the trade-off between FAR and MTD when BART is estimating available bandwidth supported by CUSUM. As mentioned previously, choosing a low threshold value h in the CUSUM stopping rule guarantees fast detection in case of a change in the system state, although one has to be aware of the greater risk of false alarms. If false alarms are unacceptable, it is recommended to choose a rather high value of h ; more test statistic is needed before exceeding the threshold, which also implies that one has to accept slower detection.

Here, two rather extreme threshold values h are used in order to illustrate the effects of different design choices.

The thick solid curve illustrates the scenario when a low threshold value is used for the CUSUM stopping rule, $h = 0.75$. During this measurement interval of 2000 seconds, a total of 50 alarms are issued from the CUSUM test, although only two are justified (i.e. corresponding to actual steps in the available bandwidth). Consequently, the configured stability property of BART is somewhat overridden, since CUSUM frequently triggers BART to transiently switch to a Q with agile properties ($\lambda = 1$). The

advantage of this threshold is the rapid response once the actual changes occur in the system; hence, the time to detection is very low.

The dashed curve depicts the opposite situation, when a rather large threshold value is used, $h = 50$. In this case, only two alarms are indicated during the 2000 second interval, and they are both due to the actual step changes taking place after 750 and 1250 seconds, respectively. The disadvantage of having a large threshold is clearly the time to detection, which in Figure 5 is approximately 150 seconds for both alarms.

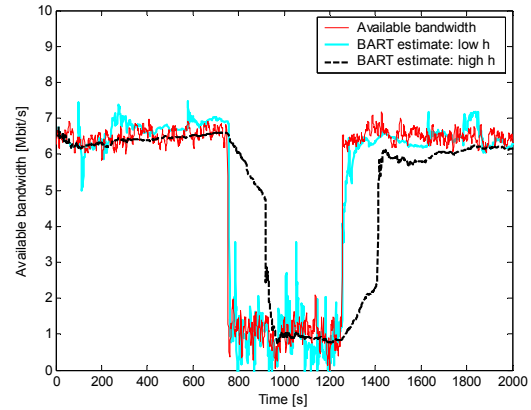


Figure 5. BART using CUSUM with two different threshold values h , in order to exemplify the trade-off between false alarms and time to detection ($\lambda = 1$ for one iteration in case of CUSUM alarm).

When CUSUM issues an alarm, it is interesting to investigate what Q matrix should be chosen in order to provide BART with enough agility such that the adaptation to the new state is successful. For the evaluation in Figure 6, the only parameter that differs between the four subfigures (a) – (d) is the momentarily used (one single iteration) process noise covariance matrix Q_{alarm} .

The diagonal elements of Q are related to the tracking ability of the corresponding state variables (in BART, Q_{11} affects the tracking of α and, likewise, Q_{22} influences β , see (11) and (20)). Subfigure (a) shows the performance of BART with CUSUM when Q_{alarm} is assigned the value of 10^{-3} on the diagonal, i.e., $\lambda = 10^{-3}$ in (20). Although the Kalman filter becomes more agile, it is obviously not enough to quickly track the new system state. By increasing Q_{alarm} (see (b) and (c), where $\lambda = 10^{-1}$ and $\lambda = 10^1$, respectively) the agility of the filter increases and the adaptation is much quicker. Increasing Q_{alarm} even more, see (d) where $\lambda = 10^3$, BART can be seen as overreacting in case of an alarm, however, the estimates are back on track after a short time.

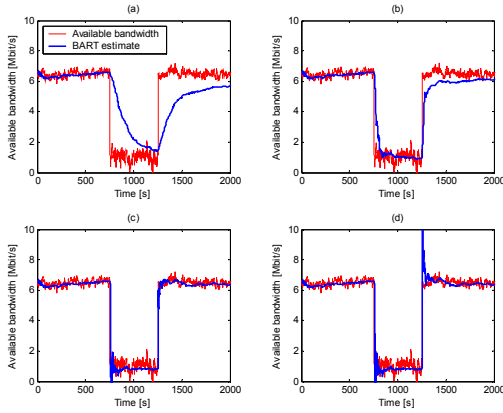


Figure 6. BART (configured for stability, $\lambda = 10^{-5}$) with CUSUM. In presence of change detection alarm, a different λ is momentarily used in (a) – (d). In case of alarm: (a) $\lambda = 10^{-3}$, (b) $\lambda = 10^{-1}$, (c) $\lambda = 10^1$, and (d) $\lambda = 10^3$.

In Figure 7, the estimates of the system state variables (α and β) of BART are shown for the four cases in Figure 6. The values of α and β are crucial in order to deliver reliable estimates of the available bandwidth.

In the testbed traffic scenario considered in this paper, the abrupt changes in the available bandwidth occur due to changes in the cross-traffic intensity, as approximately 100 additional traffic streams suddenly appear/disappear along the network path of interest. Since the bottleneck link capacity remains constant, the value of the state variable α is expected to remain constant throughout the measurements (α describes the slope in Figure 1, which in the network model is equal to the inverse of the bottleneck link capacity). The state variable β is, however, expected to change when the cross-traffic changes.

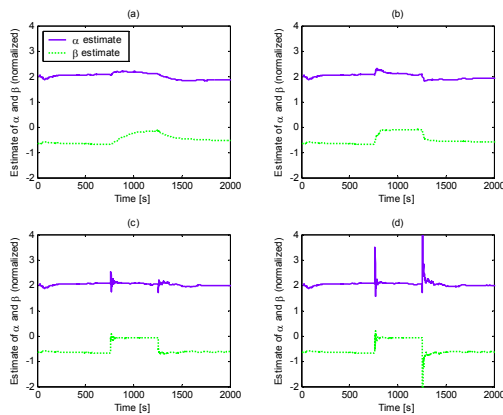


Figure 7. The estimate of state variables α and β when BART (configured for stability, $\lambda = 10^{-5}$) is used with CUSUM. In presence of change detection alarm, a different λ is momentarily used in (a) – (d). In case of alarm: (a) $\lambda = 10^{-3}$, (b) $\lambda = 10^{-1}$, (c) $\lambda = 10^1$, and (d) $\lambda = 10^3$.

In Figure 7 (a) – (d), we can study the estimation and the adaptation of the state variables α and β as the abrupt changes occur in the system state, according to Figure 6. In subfigure (a), BART was provided with a fairly weak agility injection in case of CUSUM alarm, $\lambda = 10^{-3}$ for one filter iteration. Although α and β are provided with equal opportunities for changing their values ($Q_{11} = Q_{22} = \lambda$ in (20)), α basically keeps its value, whereas β slowly adapts to the new circumstances. The reason for this slow adaptation is the low value of λ . When increasing λ , as in subfigures (b) – (d), we can observe a much more distinct and determined adaptation of the state variables, for case (d) also causing spurious transients in the adaptation.

Consequently, one way to achieve fast and successful adaptation in case of change-detection alarm is to select a suitable level, Q_{alarm} , to which momentarily increase Q . Another approach, which makes the choice of Q_{alarm} less sensitive, is to vary the duration for which the agile Q remains active in the Kalman filtering; it does not necessarily have to be for only one filter iteration.

Recall from Figure 6 (b) that $\lambda = 10^{-1}$ for one filter iteration appears to be too weak in order to at once accomplish a successful adaptation. This can be seen clearly at 1250 seconds. In Figure 8, the experiment from Figure 6 (b) is repeated, but now with Q_{alarm} active for 15 consecutive filter iterations (instead of only one) in case of a CUSUM alarm. As can be seen, repeated injections of an agile Q make it easier for the filter to adapt to the new system state. However, an issue when using this approach could be that applying Q_{alarm} for too long may cause unnecessary fluctuations of the estimates, although that potential problem is not apparent in this particular case.

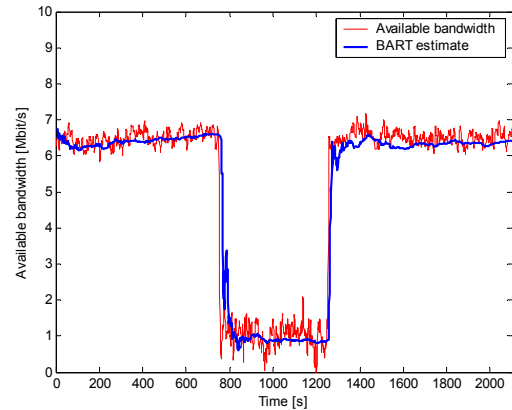


Figure 8. BART (configured for stability, $\lambda = 10^{-5}$) with CUSUM. In presence of change detection alarm, $\lambda = 10^{-1}$ for 15 consecutive filter iterations.

5.3. Internet measurements

Finally, we have performed measurements over an Internet path.

For simplicity, the CUSUM test for the Internet evaluation keeps the parameter choice from the testbed measurements, i.e., $h = 3$ and $\nu = 0.05$. In case of a CUSUM alarm, the process noise covariance matrix Q_{alarm} is used for only one iteration with $\lambda = 1$, which also turned out to be a fairly suitable choice in the testbed evaluation, as illustrated in Figure 6 (b) – (c).

The “true” available bandwidth shown in Figure 9 should rather be seen as an indication. As the measurement is performed over an Internet path, we do not have access to all intermediate routers between the probe sender and receiver. Hence, it is impossible to be completely sure about the actual available-bandwidth situation. Instead, the available bandwidth curve is calculated from a long-time average and the additional cross traffic that we injected along the network path (i.e., we have knowledge of the magnitude of the abrupt changes in the experiment, caused by cross traffic generated by us). During the time interval from 0 to 300 seconds, additional cross traffic of average intensity around 35 Mbit/s is injected, and in the interval from 450 to 1350 seconds, cross traffic corresponding to 15 Mbit/s is added along the network path. Otherwise, no additional cross traffic is used.

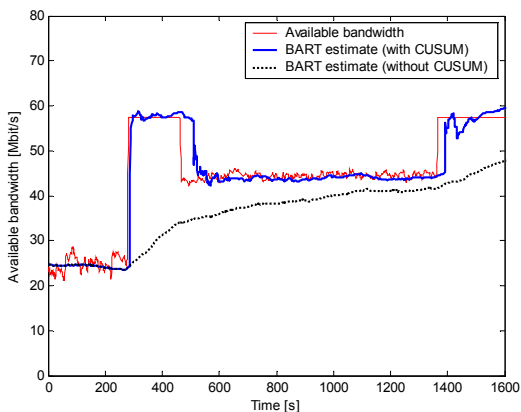


Figure 9. Internet experiment where BART (configured for stability, $\lambda = 10^{-5}$) is estimating the available bandwidth with and without CUSUM. Rapid changes occur in the system, due to manual activation and deactivation of additional cross traffic along the Internet path.

When no change detection is used, the response of BART (configured for stability) to the abrupt changes in the system is rather slow, see Figure 9. However, by including the CUSUM test, BART is provided with alarm indications and quick adaptation is possible.

The time to detect the abrupt change after 450 seconds is notable. This can be taken care of by, e.g.,

decreasing the design threshold h in the CUSUM stopping rule; however, this could also increase the rate of undesired false alarms. Another approach would simply be to configure BART such that sampling occurs more often. In Figure 9, BART is transmitting 17 probe packets each second, causing a probe-traffic overload of 0.2 Mbit/s over an Internet path with the minimum link capacity of 100 Mbit/s. By allowing more probe traffic in the network, BART would be able to probe more frequently, which also would result in faster detection in case of sudden changes.

An explanation for the slow detection around 450 seconds could be found by considering an aspect of the current BART implementation. Measurements with probe-traffic intensities below the current estimate of the available bandwidth are discarded, since these strain measurements most likely are to the left of the break point in the piecewise linear measurement model, illustrated in Figure 1 (the sloping straight line is the region of interest in BART). In this experiment, the probe-traffic intensities are randomly and uniformly chosen in the interval from 10 Mbit/s to 100 Mbit/s. Consequently, more than every other measurement is expected to be unutilized when the current estimate of the available bandwidth is around 60 Mbit/s, which is the case after approximately 450 seconds. The time to detection is shorter for the third change around 1350 seconds, and even shorter for the first abrupt change after 300 seconds. This seems to be reasonable, since a larger fraction of the measurements are taken into account as the BART estimate of the available bandwidth decreases.

If throwing away measurements can be seen as a weakness in the current implementation of BART, this could easily be improved by including feedback from the receiver to the sender. Thus, the BART receiver could constantly inform the sender about the current estimate of the available bandwidth, and the sender would be able to adapt the range of probing rates such that useful strain measurements are received. This would most likely improve the time to detection in Figure 9, without increasing the probability of false alarms.

6. Discussion and conclusions

In this paper, we have addressed the problem of accurately estimating communication network properties in systems with irregular characteristics. For statistical estimation methods, there is a clear trade-off between noise insensitivity and the ability of fast adaptation to sudden changes.

A major virtue of filter-based methods is the ability

to enhance performance by combining them with change detection. Hence, as a solution to the above problem, we have suggested using filter-based estimation tools assisted with a change-detection technique.

The simple change-detection technique CUSUM was implemented and tested together with the available-bandwidth estimation tool BART, in both a laboratory network and over an Internet path.

We have discussed the design properties of the CUSUM test, and how the performance is affected with respect to different choices of the used parameters.

Attention has been paid to possible and suitable actions when the used filter-based tool receives alarm indications from the applied change-detection technique. The focus has been on transiently altering the filter parameter Q when a change is considered to have occurred.

Overall, we believe that filter-based estimation with change detection is a promising approach for various applications in communication networks, especially when accurate real-time estimates of a system state are required.

By integrating CUSUM with BART, we have observed a significant enhancement of the estimation performance, when the network state is subject to abrupt changes.

A possible continuation of this work is to evaluate and compare several change-detection techniques in applications suitable for data communications. One should recall that the CUSUM test is a rather crude approach. Hence, it is likely that there is potential for further performance enhancement, by utilizing a more sophisticated change-detection technique.

References

- [1] S. Ekelin, M. Nilsson, E. Hartikainen, A. Johnsson, J.-E. Mångs, B. Melander, and M. Björkman, "Real-time measurement of end-to-end available bandwidth using Kalman filtering," in *Proc. 10th IEEE/IFIP Network Operations and Management Symposium*, 2006.
- [2] E. Hartikainen and S. Ekelin, "Tuning the temporal characteristics of a Kalman-filter method for end-to-end bandwidth estimation," in *Proc. 4th IEEE/IFIP Workshop on End-to-End Monitoring Techniques and Services*, 2006.
- [3] V. Ribeiro, R. Riedi, G. Baraniuk, J. Navratil, and L. Cottrell, "pathChirp: efficient available bandwidth estimation for network paths," in *Proc. Passive and Active Measurement Workshop*, 2003.
- [4] M. Jain and C. Dovrolis, "Pathload: a measurement tool for end-to-end available bandwidth," in *Proc. Passive and Active Measurement Workshop*, 2002.
- [5] J. Navratil and R.L. Cottrell, "Abwe: a practical approach to available bandwidth estimation," in *Proc. Passive and Active Measurement Workshop*, 2003.
- [6] R. Carter and M. Crovella, "Measuring bottleneck link speed in packet-switched networks," Technical Report 96-006, Boston University, 1996.
- [7] N. Hu and P. Steenkiste, "Evaluation and characterization of available bandwidth probing techniques," in *Proc. IEEE JSAC Internet and WWW Measurement, Mapping, and Modeling*, 2003.
- [8] G. Jin, G. Yang, B.R. Crowley, and D.A. Agarwal, "Network characterization service (NCS)," Lawrence Berkely National Lab Report 47892, 2001.
- [9] J. Strauss, D. Katabi, and F. Kaashoek, "A measurement study of available bandwidth estimation tools," in *Proc. ACM SIGCOMM Internet Measurement Conference*, 2003.
- [10] B. Melander, M. Björkman, and P. Gunningberg, "A new end-to-end probing and analysis method for estimating bandwidth bottlenecks," in *Proc. IEEE Globecom*, 2000.
- [11] E.S. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, 1954.
- [12] U. Appel and A.V. Brandt, "Adaptive sequential segmentation of piecewise stationary time series," *Information Sciences*, vol. 29, 1983.
- [13] M. Basseville and A. Benveniste, "Design and comparative study of some sequential jump detection algorithms for digital signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 31, 1983.
- [14] A.S. Willsky and H.L. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems," *IEEE Transactions on Automatic Control*, vol. 21, 1976.
- [15] F. Gustafsson, "The marginalized likelihood ratio test for detecting abrupt changes," *IEEE Transactions on Automatic Control*, vol. 1, 1996.
- [16] F. Gustafsson, "A comparative study on change detection for some automotive applications," in *Proc. European Control Conference*, 1997.
- [17] K. Jacobsson, N. Möller, K.-H. Johansson, and H. Hjalmarsson, "Some modeling and estimation issues in control of heterogeneous networks," in *Proc. 16th International Symposium on Mathematical Theory of Networks and Systems*, 2004.
- [18] G. Bishop and G. Welch, "An introduction to the Kalman filter," *SIGGRAPH*, Course 8, 2001.
- [19] F. Gustafsson, *Adaptive Filtering and Change Detection*. John Wiley & Sons, Ltd, 2000.