

Adaptive CUSUM for Anomaly Detection and Its Application to Detect Shared Congestion

Sheng-Ya Lin, Jyn-Charn Liu, Wei Zhao

Department of Computer Science

Texas A&M University

Technical Report: TAMU-CS-TR-2007-1-2

January 22, 2007

Email: {shengya, jcliu, w-zhao}@tamu.edu

Abstract—It is a major challenge for a detection algorithm to maintain high detection probability and low false alarms simultaneously. In this paper, we propose an adaptive CUSUM algorithm (ACS) to robustly detect an *anomaly*, which is defined as system behavior that deviates from its expected values. By embedding a sliding model control (SMC) controller into a CUSUM detector, ACS effectively prevents unlimited build-up of the *accumulator*, i.e., the y_n variable in CUSUM, during the anomaly period. This way, ACS effectively detects *change points* at on-set and termination of an anomaly period, while satisfying the requirements of detection and false alarm time.

For general performance evaluation, we found that ACS is more stable than the Neyman-Pearson detector, and also overcomes the tardy detection of anomaly termination of the CUSUM algorithm. We further reduce the problem of shared congestion to that of anomaly detection, where shared congestion is regarded as anomaly and independent congestion as normality, using the delay correlation of two probing streams as metric. Existing solutions employed accumulated samples to detect shared congestion. We proposed a sliding window scheme to capture the change of congestion states. The simulation results showed that window scheme with ACS can detect the switch of the shared vs. independent congestion for each detection instance.

Index Terms—change point, CUSUM, sliding mode control, adaptive CUSUM, shared congestion.

I. INTRODUCTION

Anomaly is a deviation of a meteorological quantity value from the expected (mean) value of a system under observation. Timely, robust detection of anomalous changes in network traffic is important for enforcement of bandwidth allocations, detection of denial of service (DoS) traffic, and numerous traffic management functions. CUSUM [1] is a broadly adopted algorithm for detection of abrupt traffic flow change [2]–[5], especially for DoS packet flooding. CUSUM is good at detecting the abrupt change of the mean of an observed sequence. By adjusting parameters of CUSUM, one can make tradeoff between the detection sensitivity and the degree of noise-resilience in an anomaly detection system.

Several different techniques such as pattern matching and data mining have been developed for Internet traffic anomaly detection. However, these approaches need to use a large amount of samples to train the detector and are sensitive to training noise. Different from above techniques, we proposed an ACS algorithm, a CUSUM-based technique, to detect shared congestion where packet loss or time delay can be used as

a feedback signal. Techniques based on time delay [6]–[8] usually have more robust detection outcomes than those using packet loss [6], [9], which are limited to drop-tail queues and lossy links. To detect switching between two congestion states either from shared to independent or vice versa, we propose a sliding window scheme that captures the change of congestion states based on the cross-correlation of one-way delays. Simulation results shows that sliding window scheme based on ACS has precise, responsive detection of anomalies for both TCP and UDP traffic flows.

II. CUMULATIVE SUM (CUSUM) ALGORITHM

Being a nonparametric methodology, a CUSUM detector detects *change points*, i.e., the rising and falling edges of the *anomalous period*, which denotes the duration that the anomaly lasts, without prior knowledge on the distribution of the traffic flow. When the mean of a subset of samples becomes higher than a threshold, a change point is registered, and the level of change is cumulated into an *accumulator*. The accumulator begins to discharge when the other change point is detected at end of the anomaly period.

Let the observed traffic flow be denoted by a random process $\{X_n\}$ such that

$$X_n = \alpha + \xi_n I(n < m \vee k < n) + (h + \zeta_n) I(m \leq n \leq k), \quad (1)$$

where m is the *rising* change point (RCP) and k is the *falling* change point (FCP), $\xi = \{\xi_n\}_{n=1}^{\infty}$, $\zeta = \{\zeta_n\}_{n=1}^{\infty}$ are random sequences such that $E(\xi_n) = E(\zeta_n) \equiv 0$, $h > 0$. When the anomaly does not exist,

$$E(X_n) = \alpha, \quad (2)$$

In practice, x_n is usually transformed to \tilde{X}_n , i.e., (3), such that \tilde{X}_n has a negative mean value, but it will become positive when the rising change point is detected.

$$\tilde{X}_n = X_n - \beta, \quad (3)$$

where β is the upper bound of α . The *accumulator* y_n is defined as

$$\begin{aligned} y_n &= (y_{n-1} + \tilde{X}_n)^+, \\ y_0 &\equiv 0, \end{aligned} \quad (4)$$

where $f(\cdot)^+$ denotes $\max(0, f(\cdot))$. An anomaly is detected at time slot n when y_n becomes greater than an alarm threshold Γ , and $d_\Gamma(\cdot) = 1$:

$$d_\Gamma(y_n) = I(Y_n > \Gamma), \quad (5)$$

where $I(\cdot)$ denotes the indicator function.

There exist two types of errors in detection. (i) Type I error (false alarm): a normal example is diagnosed as abnormal, and (ii) Type II error (miss): an anomaly occurs but not be detected. Usually, the period of Type I error is called *false alarm time*. *Detection time* is a sensitivity metric indicating the interval between occurrence and detection of an anomaly. Multiple false detections of RCP's and FCP's may happen, and the ratio of the time that an anomaly is correctly identified is called *probability of detection* P_D . On the contrary, the ratio of the period that the anomaly is not perceived is called the *probability of miss* P_M . For the asymptotical optimality of CUSUM in one of its worst scenarios, such as Gaussian random process, [2], [3], [5] suggested to set

$$\beta = h/2 + \alpha, \quad (6)$$

where h is the lower bound of the mean increase after an anomaly occurs. Then alarm threshold Γ can be obtained as:

$$\begin{aligned} \tau_\Gamma &= \inf\{n : d_\Gamma(y_n) = 1\}, \\ \Gamma &= (h - |\alpha - \beta|) \cdot (\tau_\Gamma - m)^+, \end{aligned} \quad (7)$$

where $\tau_\Gamma - m$ is the detection time. (7) also shows the relationship between Γ , h and the detection time. Increasing Γ reduces false alarms but will lead to longer detection time. Lowering β reduces detection time but increases false alarm time, because it speeds up increase of y_n , but slows down the decrease of y_n . Intuitively, CUSUM works like an integrator such that y_n is proportional to the length of the anomaly period. One cannot simultaneously minimize the detection time and false alarm time of the CUSUM, based on its energy preservation principle.

III. ADAPTIVE CUSUM

In this section, we propose an Adaptive CUSUM (ACS) architecture for responsive detection of anomaly, both at onset and termination of an anomaly period. Our basic technique is to make variable β *buoyant*, meaning that the value of β is adjusted based on the detector output, to prevent the CUSUM accumulator y_n from oscillating. Selection of the control paradigm is critically dependent on the behaviors of the target system. Given the bursty nature of traffic flows, it is highly unlikely to find tractable and precise mathematical models to characterize the system dynamics to be integrated into most control theories, except for the sliding model control (SMC) [10]–[13]. SMC is known for its tolerance to inaccuracy of behavioral models, and its only major design requirement is the *relative degree* of the system. The relative degree of a system is r if its input variable is obtained after its objective function is differentiated r times [12], [13]. A key feature of our scheme is that it allows one to use the SMC controller output (detector outcomes) to adjust its ACS input (and that of the SMC controller,) so that the CUSUM accumulator in

ACS will become independent of the duration of the anomaly, while maintaining its responsiveness to both RCP and FCP.

A. Sliding Mode Control

A SMC controller consists of two major control rules - linear and switching control laws. The linear control law is derived from the equivalent control, which controls the input after it is filtered by a low pass filter. The goal of equivalent control is to keep the motion trajectory on the *sliding manifold* $s(t) = 0$, where $s(t)$ is termed *switching function*, and eventually retain $s(t) = \dot{s}(t)$ once sliding manifold is reached at time t under ideal sliding motion. SMC primarily consists of two motion phases; one is *reaching phase* during which trajectories are driven to the sliding manifold, and the other is *sliding phase* during which the motion is designed to move toward the equilibrium point over sliding manifold, as shown in Fig. 1. Due to the time delay in the physical

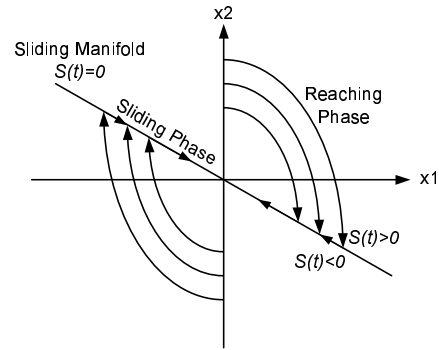


Fig. 1. Phase portrait under sliding mode control.

system, the sliding motion is usually a chattered trajectory. The switching control is used for driving the trajectory back to the sliding manifold if the system dynamics satisfies the reachability condition $s(t)\dot{s}(t) < 0$, which can be proved using the Lyapunov function candidate $V(s) = s^2(t)/2$. A system is asymptotically stable if its Lyapunov function satisfies (8).

$$\begin{aligned} V(0) &= 0, \\ V(s) &> 0 \text{ in } D - \{0\}, \\ \dot{V}(s) &< 0 \text{ in } D - \{0\}, \end{aligned} \quad (8)$$

where $D \subset R^n$ is a domain containing the equilibrium point $s = 0$ and $V : D \rightarrow R$ is a continuously differential function. By taking the first derivative of $V(s) = s^2(t)/2$ and letting it satisfy (8), we get $s(t)\dot{s}(t) < 0$, the reachability condition. Though assuring the trajectory of reaching the sliding manifold eventually, reachability condition does not imply the time needed to reach the line. To guarantee a finite reaching time, η -reachability [14], [15] in (9) need to be satisfied:

$$s(t)\dot{s}(t) < -\eta|s|, \quad (9)$$

where η is a positive number

Reachability is closely related to switching function $s(t)$. The construction of switching function depends on the relative degree of the system dynamics. Thereby, given a control

system of relative degree r , a switching function is constructed as

$$s(t) = \sum_{i=0}^{r-2} k_i \theta^i + \theta^{r-1}, \quad (10)$$

where θ is the objective function and k_i 's are selected such that $s(t)$ is Hurwitz, i.e. the roots of the characteristic function are all on the left half plane for stabilizing the linear switching function $s(t)$.

B. ACS Detector

We introduce a *buoyant variable* $\tilde{\beta}_n$, which floats above its lower bound $\underline{\beta} = \beta$ in (3) and then replace β with $\tilde{\beta}_n$ to secure

$$\begin{aligned} \tilde{X}_n &= X_n - \tilde{\beta}_n, \\ \underline{\beta} &= \min_{n \in \mathbb{Z}^+} (\tilde{\beta}_n). \end{aligned} \quad (11)$$

To stabilize the value of y_n in the anomaly period, an objective value $(1 + \varepsilon)\Gamma$, where ε parameterizes the duration of false alarm, is used for y_n to converge by adjusting the buoyant variable $\tilde{\beta}_n$, which changes in response to detector states. Convergence behaviors of y_n can be characterized by its responsiveness at RCP and FCP. Fig. 2 illustrates the architecture of ACS. Given measurements $\vartheta_1(n)$ and $\vartheta_2(n)$ in time slot n , DR is the difference ratio defined as $\Delta_n/g_n(\vartheta_1, \vartheta_2)$, where $g_n(\vartheta_1, \vartheta_2)$ is the function of $\vartheta_1(n)$ and $\vartheta_2(n)$ and Δ_n is the difference of measurements, which may be $\vartheta_1(n) - \vartheta_2(n)$, $\vartheta_1(n) - \vartheta_1(n-1)$ or $\vartheta_2(n) - \vartheta_2(n-1)$, etc. In each time slot, y_n is updated by the accumulation of DR . The equilibrium point of y_n lies at $(1 + \varepsilon)\Gamma$ for an expected false alarm time. Thereby the objective function θ becomes

$$\theta = y_n - (1 + \varepsilon)\Gamma. \quad (12)$$

Let τ_f denote the first time that detector perceives anomaly termination, we have

$$\tau_f = \inf\{n : d_\Gamma(y_n) = 0, d_\Gamma(y_p) = 1, p \leq n\}. \quad (13)$$

An ideal false alarm time $(\tau_f - k)^+$ can be obtained by substituting $\varepsilon\Gamma = (h - |\alpha - \beta|)(\tau_f - k)^+$ into (7). The design parameter ε becomes

$$\varepsilon = \frac{(\tau_f - k)^+}{(\tau_\Gamma - m)^+}. \quad (14)$$

Since the relative degree of CUSUM is one, switching function $s(t)$ can be constructed using (10) such that

$$s(t) = \theta. \quad (15)$$

The sliding manifold is used as a track to guide state variables toward the equilibrium point $y = (1 + \varepsilon)\Gamma$ if Hurwitz condition is satisfied. Condition (9) must hold to guarantee that the trajectory of y_n can reach the manifold, and this leads to the condition $(y_n - (1 + \sigma)\Gamma)(\tilde{\beta} - DR) > \eta|s|$. As a result,

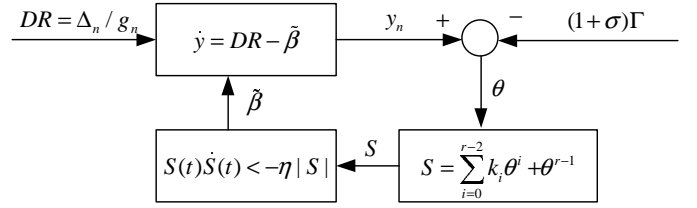


Fig. 2. Control model of adaptive CUSUM.

a control law using $\tilde{\beta}_n$ as a control variable to stabilize y_n at RCP is

$$\begin{aligned} \tilde{\beta}_n &= \check{\beta}_n + \hat{\beta}_n, \\ \check{\beta}_n &= \frac{-\Delta_n}{g_n(\vartheta_1, \vartheta_2)}, \\ \hat{\beta}_n &= \eta \times \text{sgn}(y_n - (1 + \varepsilon)\Gamma), \end{aligned} \quad (16)$$

where $\check{\beta}_n$ is the linear control law, $\hat{\beta}_n$ the switching control law, $\text{sgn}(\cdot)$ the signum function, $s \times \text{sgn}(s) = |s|$, and $\eta > 0$. The magnitude of η is set to $\varepsilon\Gamma$ to ensure that y_n is bounded stable in the range of $[\Gamma, (1 + 2\varepsilon)\Gamma]$ during the anomaly period. Recall that $\underline{\beta}$ is the lower bound of $\tilde{\beta}_n$, we have $\tilde{\beta}_n = \max(\underline{\beta}, \check{\beta}_n)$ and the control law (16) of ACS is refined as

$$\begin{aligned} \tilde{\beta}_n &= \begin{cases} \check{\beta}_n + |\hat{\beta}_n| & \text{if } \text{sgn}(s) > 0 \\ \check{\beta}_n - |\hat{\beta}_n| & \text{if } \text{sgn}(s) < 0 \end{cases}, \\ \tilde{\beta}_n &= \max(\underline{\beta}, \check{\beta}_n), \\ \hat{\beta}_n &= \varepsilon\Gamma. \end{aligned} \quad (17)$$

Lowering the false alarm rate is expected to increase the missing rate. For robust detection of the subtle signal change at presence of noise, η is also made a dynamic variable, which is adjusted by *direction change* of the recent and current switching function $s(n)$. That is, if $s(n-1) \times s(n) > 0$, it means that $s(n)$ and $s(n-1)$ are in the same direction, otherwise they are different. η is dynamically adjusted by the rule of additive increase multiplicative decrease (AIMD) for reducing the interference of high variant noises:

$$\eta(n) = \begin{cases} \eta(n-1) + \kappa & \text{if } s(n) \times s(n-1) > 0 \\ \eta(n-1) \times \psi & \text{if } s(n) \times s(n-1) \leq 0 \end{cases} \quad (18)$$

where κ and η are two constant variables. Rapid decrease of η when the direction of $s(n)$ changes reduces the oscillation of the sliding trajectory. Gradual increase of η avoids the excessive control leading to false alarms. By integrating the sliding mode control with the adaptive law of η , the pseudo code of ACS is depicted in Algorithm 1.

IV. IMPACT OF METRICS AND DETECTORS

When the ratio of two inputs variables is used for anomaly detection, extreme large/small values may cause numerical problems for certain ratio metrics. In this section, different metrics and detectors are compared for their false alarm time, indication stability and robustness to parameters.

Algorithm 1 Adaptive CUSUM algorithm

Input: X_n
Output: $d_\Gamma(\cdot)$

- 1: **loop**
- 2: $s(n) \leftarrow y_n - \Gamma(1 + \varepsilon)$ {switching function}
- 3: **if** $y_n \geq \Gamma$ **then** {entry of the anomaly period}
- 4: {adaptive law of η }
- 5: **if** $s(n) \times s(n-1) > 0$ **then**
- 6: $\eta(n) \leftarrow \eta(n-1) + \kappa$
- 7: **else**
- 8: $\eta(n) \leftarrow \eta(n-1) \times \psi$
- 9: **end if**
- 10: **else** {departure of the anomaly period}
- 11: $\eta(n) = \varepsilon\Gamma$
- 12: **end if**
- 13: {sliding mode control}
- 14: **if** $s(n) > 0$ **then**
- 15: $\tilde{\beta}_n \leftarrow \tilde{\beta}_n + |\hat{\beta}_n|$
- 16: **else**
- 17: $\tilde{\beta}_n \leftarrow \tilde{\beta}_n - |\hat{\beta}_n|$
- 18: **end if**
- 19: $\tilde{\beta}_n \leftarrow \max(\beta, \tilde{\beta}_n)$
- 20: {CUSUM computing}
- 21: $\tilde{X}_n \leftarrow X_n - \tilde{\beta}_n$
- 22: $y_n \leftarrow (y_{n-1} + \tilde{X}_n)^+$
- 23: $d_\Gamma(y_n) \leftarrow I(Y_n \geq \Gamma)$
- 24: **end loop**

A. Effect of Ratio Range on CUSUM

A number of metrics have already been used for CUSUM to detect anomaly in specific domains. [3] uses the difference ratio $DR_1 = \Delta_n/\vartheta_1(n)$ as the measure metric and [2], [5] use $DR_2 = \Delta_n/\vartheta_2(n)$, where $\Delta_n = \vartheta_1(n) - \vartheta_2(n)$, $\vartheta_1(n)$ and $\vartheta_2(n)$ are the packet number of type p_1 and p_2 at time slot n respectively. The ranges of DR_1 and DR_2 are unbounded and asymmetric. i.e. $DR_1 \in (-\infty, 1]$ and $DR_2 \in [-1, \infty)$. DR_1 tends to decrease y_n and thus may lead to the longer detection time. On the contrary, DR_2 has longer false alarm time due to slow decrease of y_n . Therefore we consider the third metric DR_3 which is equally bounded and denoted as $\Delta_n/(\vartheta_1(n) + \vartheta_2(n)) \in [-1, 1]$. The three ratio metrics are summarized as follows:

$$\begin{aligned}
 DR_1 &= (\vartheta_1(n) - \vartheta_2(n))/\vartheta_1(n), \\
 DR_2 &= (\vartheta_1(n) - \vartheta_2(n))/\vartheta_2(n), \\
 DR_3 &= (\vartheta_1(n) - \vartheta_2(n))/(\vartheta_1(n) + \vartheta_2(n)),
 \end{aligned} \tag{19}$$

where $DR_i = 0$ if $(\vartheta_1(n), \vartheta_2(n)) = (0, 0)$. CUSUM using DR_3 as a metric behaves like a Canberra distance¹, which is widely employed in multivariate analysis, except for its directionality and a shifted amount of β . To better understand the effect of ratio metrics upon CUSUM, three anomaly processes are generated numerically according to (1) and their parameters are depicted in Table I, where $X_n = DR_2$, ξ_n and ζ_n are time series of Gaussian noise. Proc² has a shorter

anomaly duration, lower signal-to-noise ratio (SNR) contrasted with Proc¹. Proc³ is the worst case which consists of the worse parts of Proc¹ and Proc².

TABLE I
BENCHMARK PROCESSES FOR EVALUATION OF RATIO METRICS.

Process	α	h	RCP	FCP	ξ_n	ζ_n
Proc ¹	0	6	8	30	N(0, 0.10)	N(0, 0.10)
Proc ²	0	1	8	20	N(0, 0.25)	N(0, 0.25)
Proc ³	0	1	8	30	N(0, 0.25)	N(0, 0.25)

For zero-mean additive noise, the SNR is defined as

$$\begin{aligned}
 \text{SNR(dB)} &= 10 \log_{10}(E/\sigma^2), \\
 &= 20 \log_{10}(h/\sigma),
 \end{aligned} \tag{20}$$

where E is the signal energy, h is the mean of the change and σ the standard deviation of the noise. Statistically, the detecting accuracy is proportional to SNR since the increase of E/σ^2 also increases the distance between the mean of the distribution of the noise and that of the tainted signal. In Table I, SNRs of Proc¹ and Proc² are 27.99 and 6.02 dB respectively. To compare the difference between DR_1 , DR_2 and DR_3 , we apply an equalization transformation to (19), so that

$$\begin{aligned}
 DR_1 &= X_n/(1 + X_n), \\
 DR_2 &= X_n, \\
 DR_3 &= X_n/(2 + X_n).
 \end{aligned} \tag{21}$$

In CUSUM, detection time, false alarm time, and threshold Γ , are correlated. Threshold Γ can be obtained according to (7) for a specified detection time. If the detection time of CUSUM using DR_1 in Proc¹ is set to 3, then Γ_1 is obtained by $\Gamma_1 = \frac{3 \times h_1}{2} = \frac{1.5 \times h}{1+h} \approx 1.286$. The equivalent thresholds of DR_2 and DR_3 can be derived in a similar way. Table II lists CUSUM parameters of DR_i for the identical detection time $\tau_\Gamma - m = 3$.

TABLE II
EQUIVALENT PARAMETERS FOR IDENTICAL DETECTION TIME.

Process	Metric	α	h	β	Γ
Proc ¹	DR_1	0	0.857	0.429	1.286
	DR_2	0	6.000	3.000	9.000
	DR_3	0	0.750	0.375	1.125
Proc ² , Proc ³	DR_1	0	0.500	0.250	0.750
	DR_2	0	1.000	0.500	1.500
	DR_3	0	0.333	0.167	0.501

Let $\zeta(x) = \ddot{f}(x)$ denote the curvature of a function $f(x)$, it is easy to see that DR_i have different curvatures. The larger curvature a DR_i has, the more unbalance between the decreasing and increasing slope, thus influencing its corresponding false alarm time. If $X_n < \sqrt[3]{4} + \sqrt[3]{2} = 2.487$, then $|\zeta(\frac{X_n}{2+X_n})| > |\zeta(\frac{X_n}{1+X_n})| > \zeta(X_n)$ i.e. $|DR_1 - \delta| - |DR_1 + \delta| > |DR_3 - \delta| - |DR_3 + \delta| > |DR_2 - \delta| - |DR_2 + \delta|$ where $\delta \approx 0^+$. If $X_n > 6$, the curvatures of equivalent DR_i are almost identical. For $DR_1 \in (-\infty, 1]$, the difference between

¹Canberra distance $\text{Canb}(X, Y) = \sum_{k=1}^n \frac{|X_k - Y_k|}{X_k + Y_k}$

infimum and supremum is large, so that it is relatively easier for y_n to decrease in comparison with DR_2 and DR_3 . In Fig. 3, CUSUM use distinct metrics for detecting change points of Proc¹. Fig. 3(a) shows that the rate of the decrease to the increase for y_n are $DR_1 > DR_3 > DR_2$. In Fig. 3(b), given a detection time=3, all metrics detect RCP at time slot $n = 11$, and false alarm time for DR_1 , DR_2 and DR_3 are 18, 21 and 20.

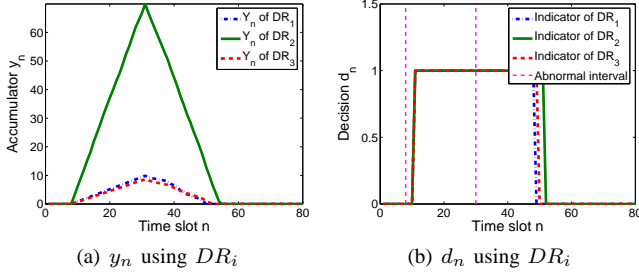


Fig. 3. CUSUM detection in Proc¹.

Fig. 4(a) shows that the histogram of y_n for the case that the noise is added to the signal input. Fig. 4(b) indicates the anomaly period of CUSUM using DR_1 , DR_2 and DR_3 are $[10, 25]$, $[10, 34]$ and $[10, 30]$. Because its low-pass filter like behavior, the change point can be reliably detected in the presence of noise. However the standard CUSUM still suffers from the build-up effects of y_n , i.e. the false alarm time depends on the period of anomaly. Comparing Fig. 3(b) with Fig. 4(b), it shows that CUSUM using DR_1 detects FCP in Proc² at time slot $n = 25$, and its false alarm time $\Delta n = 25 - 10 = 15$ is shorter than $\Delta n = 48 - 30 = 18$, the false alarm time using DR_1 in Proc¹. When based on other metrics, the false alarm time of CUSUM in Proc² is also shorter than that in Proc¹.

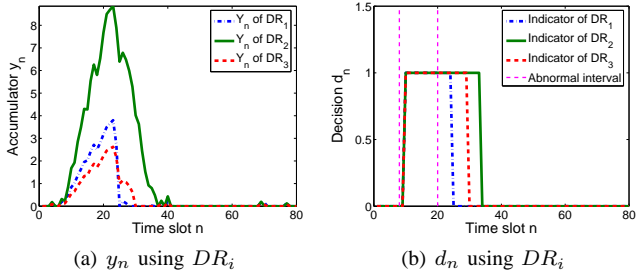


Fig. 4. CUSUM detection in Proc².

B. Comparison of Detectors

In this section, we compare the general performance of ACS with that of a well known statistical detector, the Neyman-Pearson (NP) detector, under the worst workload of the three cases, Proc³. Using the time series of (1), the problem of NP detection can be formulated as the hypothesis test as (22).

$$\begin{aligned} H_0 &: X_n = \xi_n, \\ H_1 &: X_n = h + \zeta_n, \end{aligned} \quad (22)$$

where h is the signal of DC level of amplitude 1; ξ_n and ζ_n are zero-mean white Gaussian noise with variance $\sigma^2 = 0.25$

for Proc³. H_0 is the noise-only (normal) hypothesis, and H_1 is the signal-present (anomaly) hypothesis. The likelihood ratio test (LRT) is used to test H_0 and H_1 based on Theorem 4.1.

Theorem 4.1 (Neyman-Pearson [16]): The highest probability of detection P_D for H_1 , given probability of false alarm $P_{FA} = \omega$, occurs when the likelihood ratio $L(x)$ is such that

$$L(x) = \frac{p(x; H_1)}{p(x; H_0)} > \gamma, \quad (23)$$

where the threshold γ is obtained by

$$P_{FA} = \int_{\{x: L(x) > \gamma\}} p(x; H_0) dx = \omega.$$

A NP detector decides a sample x abnormal if $L(x) > \gamma$, and normal otherwise. Given $P_{FA} = \omega$, NP detector has the maximal probability of detection P_D , or the minimal probability of miss P_M . For comparison between CUSUM and ACS, NP criterion is modified using Theorem 4.2 to minimize P_{FA} , given $P_M = \hat{\omega}$. In the case of Proc³, requiring the detection time $\tau_T - m = 3$ in the anomaly period $k - m = 22$ for CUSUM can be translated as setting $P_M = \frac{(\tau_T - m)}{(k - m)} = 0.136$ for the NP detector.

Theorem 4.2: To minimize probability of false alarm P_{FA} for a given probability of miss $P_M = \hat{\omega}$, the detector decide H_1 if

$$L(x) = \frac{p(x; H_1)}{p(x; H_0)} > \hat{\gamma},$$

where the threshold $\hat{\gamma}$ is derived by

$$P_M = \int_{\{x: L(x) < \hat{\gamma}\}} p(x; H_1) dx = \hat{\omega},$$

Proof: See Appendix A ■

Based on (22), NP detector decides H_1 if $X_n > 0.451$ and decides H_0 if $X_n < 0.451$ by substituting $h = 1$, $\sigma = 0.5$ and $\hat{\omega} = 0.136$ into Theorem 4.2. See details in Appendix B.

Given a specific value on the probability of miss, the objective of detectors is to minimize the probability of false alarm. Fig. 5(a) depicts the random process of Proc³ of which SNR is 6.02 dB. Given $P_M = 0.136$, the P_{FA} of NP detector is 0.172. The detection outcome of NP detector is shown in Fig. 5(b) which oscillates significantly.

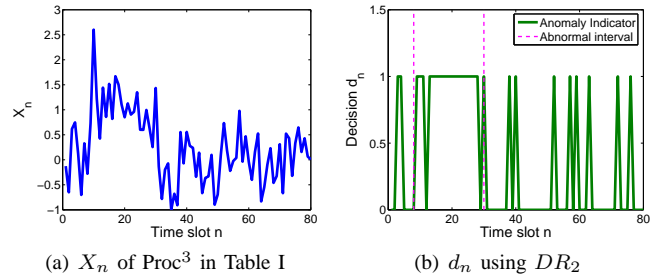


Fig. 5. NP detection in Proc³.

Fig. 6(a) shows the y_n of CUSUM accumulated in the anomaly period, i.e. $n \in [8, 30]$. Compared with NP detector, the outcome of CUSUM is highly stable. In Fig. 6(b), the detection time of CUSUM is 3 (time slots), which is equivalent to $P_M = 0.136$. The false alarm time is 17, i.e. $P_{FA} = 0.293$, which is worse than that of NP detector.

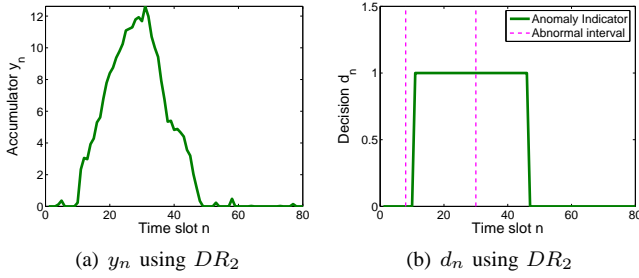


Fig. 6. CUSUM detection in Proc³.

Despite its resilience to noise, a main drawback of CUSUM is that y_n continues to build up during the anomaly period. Using the switching function $s(t)$, ACS stabilizes y_n and also avoids the disturbance of noise. Fig. 7(a) shows that y_n of ACS clipped to an upper bound in the anomaly period. See Fig. 6(a). ACS intelligently adjusts its control magnitude according to the direction change of the switching function to smooth the curve of y_n . Combining aforementioned factors together, ACS can minimize the missing rate with a stable false alarm time. Parameters of ACS used for Fig. 7 are $\Gamma = 1.5$ in (7),

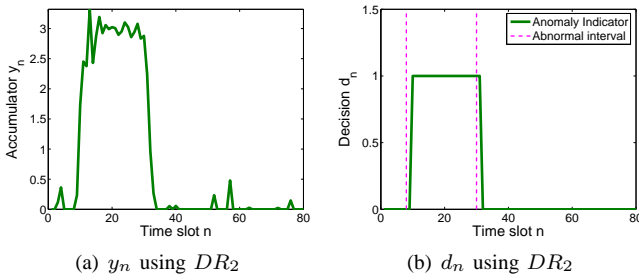


Fig. 7. ACS detection in Proc³.

$\varepsilon = 1$ in (14), $\psi = 0.5$ and $\kappa = \frac{\varepsilon \times \Gamma}{16} = 0.094$ in (18). In Fig. 7(b), P_{FA} of ACS is 0.051 and P_M is 0.136. Therefore P_{FA} of ACS is lower than that of NP detector, of which P_{FA} has been proved statistically optimal. From Fig. 5(b) and Fig. 7(b), it shows that on-off frequencies of ACS are also fewer than those of NP. Furthermore ACS greatly reduces the false alarm time compared with CUSUM.

C. Robustness

The dynamics of y_n dominates the detection results of CUSUM and ACS. It is of interest to see if the metric transformation attains approximated curves of y_n . The domain of the transformation function should be in $(-\infty, \infty)$ for including distinct ranges of ratio metrics. For further minimizing the effect of outliers biasing the detection, the range of transformed metrics should be bounded so that hyperbolic tangent $\tanh(\cdot)$ of which the range is in $(-1, 1)$ and the domain is in $(-\infty, \infty)$ is used for transformation.

Since the first derivative of $\tanh(x)$ approaches 0 for $x \geq 4$, DR_i is scaled for distinguishing the normal and abnormal processes. For optimal detection performance, $\tanh(\beta')/\tanh(h') = \beta/h$ in (6) needs to be satisfied. Under the assumption of $\alpha \approx 0$, $\tanh(\beta')/\tanh(h') = 0.519 \approx 0.5$

for a given $h' = 0.4$. The larger h' is, the higher P_M becomes. Thus \overline{DR}_i , the normalization of DR_i becomes

$$\overline{DR}_i = \tanh(\varrho \times DR_i), \quad (24)$$

where ϱ is a scale factor such that $\varrho \times h = 0.4$. Parameters of the hyperbolic metric used for CUSUM and ACS are depicted in (25).

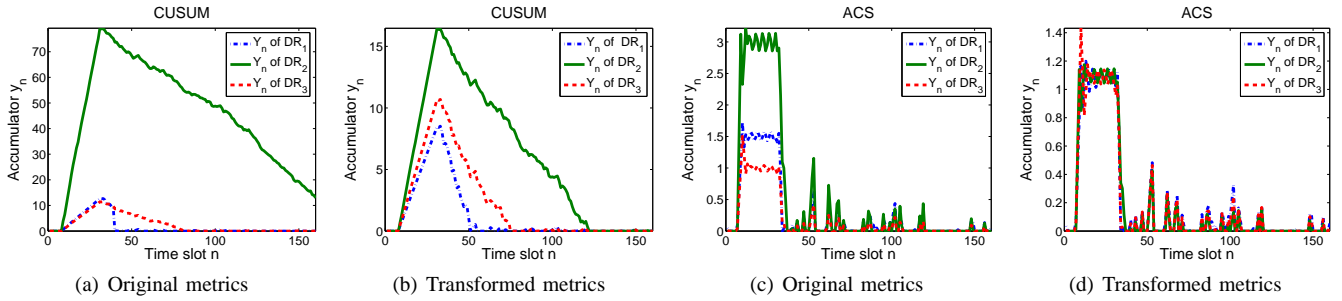
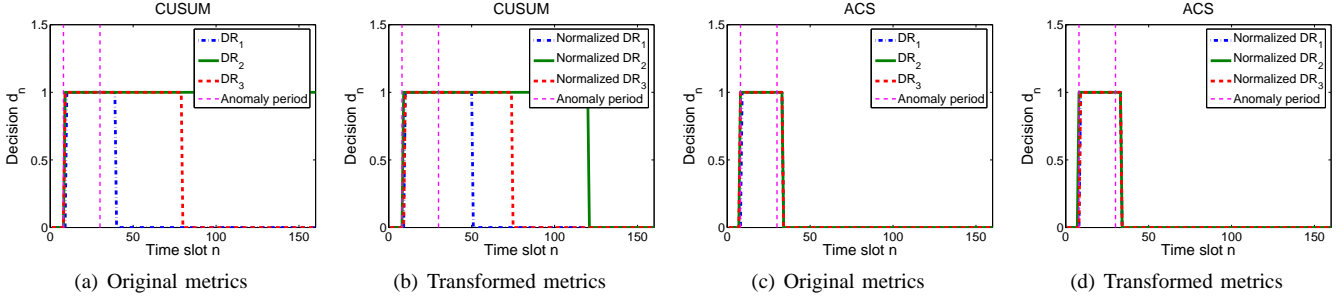
$$\begin{aligned} \overline{h} &= \tanh(\varrho \times h), \\ \overline{\beta} &= \tanh(\varrho \times \beta), \\ \overline{\Gamma} &= (\overline{h} - \overline{\beta}) \times (\tau_{\Gamma} - m). \end{aligned} \quad (25)$$

Both of original and transformed metrics are used in CUSUM and ACS to detect the anomaly in Proc³. However, parameter h is not a tight bound such that the effective change is 4 times of h . Fig. 8(a)-8(d) shows that y_n of normalized metrics are closer to one another than those of non-normalized ones regardless of CUSUM or ACS used. Comparing Fig.6(b) with Fig.9(a), the false alarm time using DR_2 as the metric is increased from 17 to 156 time slots so that CUSUM is sensitive to the parameter setting. In addition, transformation is incapable of improving the performance of all metrics in CUSUM, e.g. the detection of FCP using normalized DR_1 is more sluggish than that using the original one as shown in Fig. 9(a) and Fig. 9(b). On the contrary, ACS perceives the FCP earlier than CUSUM for DR_i , where $i = 1, 2, 3$, regardless of the metric transformation. In contrast, Fig. 9(c) and Fig. 9(d) show that ACS is robust to the parameter uncertainty and nearly independent of the metric used. Detection outcomes of CUSUM and ACS are summarized in Table III. In summary, metric transformation is incapable of improving the performance of all metrics in CUSUM such that ACS is more responsive than CUSUM even after metric transformation.

V. SHARED CONGESTION

In this section, we demonstrate how to apply ACS for detecting shared congestion. Two flows have shared congestion if they traverse the same *point of congestion* (PoC) [6]. If there exists no intersection between PoC's of the two flows, then their congestion are *independent*. Let the normal situation denote that two flows have independent congestion, then our goal is to detect the occurrence of shared congestion, which is referred to as a deviation from the normal condition.

In general, detecting shared congestion needs to select the proper measure signal and metric, and set the decision threshold of the metric. Two widely used measure signals are packet loss and time delay. Signals are often obtained by using a pair of active probing streams. A number of approaches were proposed for detecting share congestion. MP (Markovian probing) [6], decides the shared congestion by comparing the cross-correlation with auto-correlation of packet loss or one-way delay. BP (Bayesian Probing) [9] is a conditional probing technique different from MP for diagnosing the shared loss of packets. DCW (Delay correlation wavelet-denoising) [8], employs the wavelet technique to denoise time-delay sequences for increasing the correlation accuracy using fewer samples.

Fig. 8. y_n of CUSUM and ACS.Fig. 9. d_n of CUSUM and ACS.TABLE III
DETECTION OUTCOME OF CUSUM AND ACS.

Detection Algorithm	CUSUM						ACS					
	original			Transformed			original			Transformed		
	DR_1	DR_2	DR_3	DR_1	DR_2	DR_3	DR_1	DR_2	DR_3	DR_1	DR_2	DR_3
Detection Time of RCP	10	9	9	10	9	10	9	8	8	9	8	9
Detection Time of FCP	40	186	80	51	121	75	34	34	34	34	34	34

A. Optimum Sample Size

Detection of shared congestion is based on analysis of the correlation between two probing flows. The cross-correlation coefficient γ of signals is defined as

$$\gamma = \frac{\sum_{k=0}^{n-1} (X_k - \bar{X})(Y_k - \bar{Y})}{\sqrt{\sum_{k=0}^{n-1} (X_k - \bar{X})^2 \cdot \sum_{k=0}^{n-1} (Y_k - \bar{Y})^2}}, \quad (26)$$

where $\{X_k\}$ is a random sequence and \bar{X} denotes the mean of $\{X_k\}$. For reducing the costs of computing and communicating, the sample size should be small and still able to precisely express γ , which can be regarded as the ratio within the range $[-1, 1]$, such that $\gamma \approx 0$ if congestion are independent. The magnitude of γ indicates the intensity of share congestion. [6] used 3000 packets for computing γ (25 Hz in 120 sec). [8] reduced the sample size to 1000 (10 Hz in 100 sec). The large sample size helps express the cross-correlation precisely in the static condition, whereas it could smear the change points in the correlation analysis and also increase the computing cost. On the contrary, inadequate sampling is easily influenced by noise and lead to unstable detection outcomes.

Dell et al. [17] proposed a method to decide the optimum sample size as follows. Given a γ from n observations, if γ is not distributed normally, it can be converted to the normal approximation with standard deviation $1/\sqrt{n-3}$ using the

Fisher z transformation [18]:

$$z = \frac{1}{2} \ln \frac{1+r}{1-r}. \quad (27)$$

As a result, the number of samples required to distinguish γ from another specific correlation coefficient γ_0 [17] is given by

$$n = 3 + \frac{4C}{[\ln(\frac{1+r}{1-r} \times \frac{1-r_0}{1+r_0})]^2}, \quad (28)$$

where C is a function of P_{FA} and P_D and their corresponding values are listed in Table IV.

TABLE IV
VALUE OF PARAMETER C.

C		$P_D = 1 - P_M$	
		0.8	0.9
P_{FA}	0.05	7.85	10.51
	0.01	11.68	14.88

In (28), $\gamma_0 = 0$ represents the case of independent congestion and γ is the threshold indicating an occurrence of shared congestion. The magnitude of a cross-correlation interpreted by Cohen [19] is that 0.5 is large, 0.3 is moderate, and 0.1 is small. [8] shows that the detection outcome is insensitive to the cross-correlation threshold γ_{th} if $\gamma_{th} \in [0.3, 0.6]$ and adopts $\gamma_{th} = 0.512$ as a result of minimizing the detection error

rate of two distinct normal distributions using Bayesian theory. Thereby we reasonably choose $\beta=0.5$ in ACS and applied it to subsequent experiments, unless stated otherwise. If $P_{FA}=0.05$ and $P_D=0.9$ are assigned, $C=10.51$ can be looked up from Table IV. Subsequently, sample size $n=38$ is obtained by substituting C and γ_{th} into (28), i.e. $n=38$ is the optimum size to discriminate $\gamma_{th} = 0.5$ from $\gamma_0 = 0$.

B. Probing Mechanism

Active probing [8] employs a pair of synchronized probing streams $\{p_i\}$ and $\{q_i\}$ for collecting samples, where $i = 0, 1, 2, \dots$ and p_i denotes the i^{th} packet in stream p . When receiving a probe packet from the source, the destination records its one-way delay and replies an echo packet to the source for calculating the correlation coefficient γ . Since sample packets may be corrupted during the delivery, two approaches [20], [21] are utilized for addressing missing samples without drastically increasing computing complexity as follows. (i) Mean substitution: Any missing values is substituted with the mean of the non-missing values. Linear interpolation can be regarded as a variant of mean substitution, where the mean is calculated over adjacent samples. (ii) Pairwise deletion: Each γ is computed using cases with complete data for the pair of variable, i.e. a two-dimensional sample $s_i=(p_i, q_i)$ is valid only if neither p_i nor q_i is lost. Claim 5.1 provides a way to decide the number of probing packet pairs.

Claim 5.1: Assume that the loss probability of $\{p_i\}$ and that of $\{q_i\}$ are independent. Given a network of which maximum loss probability (MSP) is r_u , then the actually received sample size n_r from n_p probing packet pairs is no less than

$$n_r = (1 - 2r_u + r_u^2) \times n_p.$$

Proof: Denote \bar{p}_j the loss of packet p_j , then MSP's of (\bar{p}_j, q_j) and (p_j, \bar{q}_j) are r_u , and MSP of (\bar{p}_j, \bar{q}_j) is r_u^2 . Accordingly, the MSP of invalid samples is $2r_u - r_u^2$. Since the size of probing packet pairs is n_p , the actually received number of (p_j, q_j) is at least $(1 - 2r_u + r_u^2) \times n_p$. ■

Typically, the packet loss in Asia, Europe and North America averages between 4 and 15% daily [22]. Assuming the maximum loss rate is 15%, then $n_p \approx 54$ using $n_r=38$ and $r_u=0.15$ in Claim 5.1.

One-shot probing can detect the current congestion state randomly, whereas a continuous probing is necessary to detect change points of congestion states. For capturing the congestion dynamic, a windowed mechanism is proposed in Fig. 10 wherein n_o is the optimal sample size, r_s the sampling rate (Hz), t_s the largest RTT deviation (second) between p_i and q_i . The window size grows from 0 to n_o during the warm-

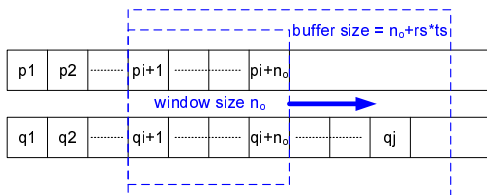


Fig. 10. Sample-gathering sliding window.

up period and remains fixed thereafter. The window slides only when receiving a valid sample $s_i = (p_i, q_i)$ and then samples within the window are used for calculating the γ of one-way delay. If either p_i or q_i becomes corrupted, s_i is discarded based on pairwise deletion. The buffer size for each of stream packets equals to $n_o + r_s \times t_s$ to accommodate the RTT offset between two probing streams. In following simulations, $n_o=38$, $r_s=20$ as well as $t_s=1$ are applied.

C. Simulation

A number of techniques such as MP, BP and DCW have been proposed for detection of shared congestion [6], [8], [9]. A performance comparison for them was detailed in [8]. In this work, we show that ACS can detect not only shared congestion, but also switching between shared and independent congestion in a responsive manner. Without considering the synchronization offset, a common source topology in Fig. 11 similar to those in [6], [8] is used to evaluate CUSUM, ACS, and DCW when the congestion conditions change dynamically.

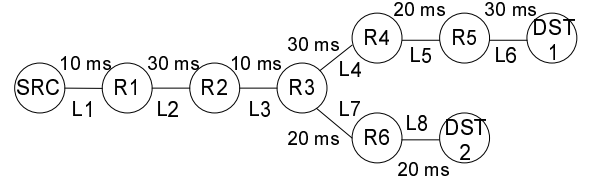


Fig. 11. Fork topology used in simulation.

In Fig. 11, each link has a bandwidth 1.5 Mb/s and a queue limit 50 packets. Two 20 Hz probing streams are transmitted from the source SRC to destinations DST₁ and DST₂, and each probing packet is echoed with its one-way delay back to SRC. Background traffic on each link are selected based on two scenarios.

TCP scenario: Let P_s denote the set of links L_1 to L_3 , and P_i the set of remaining links. In [0-35] sec, the network is simulated to have only independent congestion such that P_s is idle and the number of FTP flows on P_i are chosen uniformly between 15 and 25. At 35 sec, an RCP is injected so that shared congestion occurs to R₁, where 20 FTP flows traverse P_s and other links become idle. At 65 sec, FCP is injected so that only independent congestion is in the net, where flows on P_s cease and other links have flows between 15 and 25. The simulation period lasts for 100 seconds.

UDP scenario: Pareto on-off traffic, of which shape parameter is 1.2, are used in the background. Their burst and idle time are set uniformly between 0.2 and 3 seconds, and rates are between 20 and 40 kb/s. Congested links are uniformly assigned 71-100 UDP flows and non-congested ones are assigned 41-70 UDP flows. Similar to the TCP scenario, initially P_i is congested, but P_s is not. At the time epoch 35 sec, P_s becomes congested, while P_i does not. Eventually the congestion state returns to the initial situation at 65 sec.

The two scenarios are simulated using ns2 [23] for droptail queue and RED queue respectively. Parameters employed in distinct detectors are as follows. (i) CUSUM: $\beta = 0.5$ and $\gamma = 5$. (ii) ACS: $\epsilon = 1$, $\eta = 5$, $\psi = 0.5$, $\kappa = 0.3125$,

$\beta = 0.5$ and $\gamma = 5$. (iii) DCW: Similar to [8], Daubechies wavelet 6 basis in Matlab [24] is used in simulation to calculate wavelet coefficients of a sequence of data $\{X_k\}$ produced from ns2 traces. Given the sample size N and wavelet coefficients, the one-way delay noise is filtered by the soft threshold $T = \sigma\sqrt{2\log_e N}$, where $\sigma = \text{MAD}/0.6745$ and $\text{MAD} = \text{median}(|X - \text{median}(X)|)$. While the exact level of wavelet decomposition was not reported in [8], we found the detection performance was not significantly affected by this parameter. We chose the decomposition level 2 for its slightly better outcome, and the decision threshold is set to 0.512.

The simulation results are plotted in Fig. 12 and Fig. 13. Comparing Fig. 12(a) with Fig. 12(e), it shows that the network using the RED queue has less one-way delay than using the droptail queue. Fig. 12(c) and 12(d) plot the trajectories of accumulator y_n for CUSUM and ACS when a shared congestion occurs between 35 and 65 sec. Fig. 12(g) and 12(h) show that both of CUSUM and ACS can detect the RCP of shared congestion timely, whereas ACS is more responsive to the FCP of shared congestion than CUSUM. The time gap between the FCP detection time and the change of congestion condition (from shared to independent) at time epoch 65 is mainly caused by two factors. The first is the accumulator energy in detector, and the second is the time required to finish transmitting outstanding packets of the terminated flows, and to allow new flows to load up their transmission paths.

The correlation γ using wavelet-denoising delays is illustrated in Fig. 12(b) and 13(b) where its value is near 1 during the period of shared congestion, but very spiky for the independent congestion periods. One possible explanation of the spiky outputs is the relatively small window size being used in the simulation. But if the window size is increased, then detection performance of RCP and FCP is compromised. We further note that while mean correlation values aggregated from spiky values of experiment runs are usually much smoother, they represent the average detection behavior. On the contrary, both Fig. 12(b) and Fig. 13(b) represent the outcome from one experimental run. It gives a better understanding of the expected detector behaviors for each detection instance. The detection outcome of DCW in TCP and USP traffic are shown in Fig. 12(f) and Fig. 13(f) respectively.

In the background of UDP traffic, the one-way delay is not drastically affected by RED or droptail queue. See Fig. 13(a) and Fig. 13(e). Since no slow start for Pareto on-off flows, independent congestion can be precisely generated in simulation such that y_n of CUSUM and ACS in Fig. 13(c) and Fig. 13(d) respond the change point on time. Hence the false alarm time of CUSUM and ACS in Fig. 13(g) and Fig. 13(h) are less than that of the TCP case. Similar to results in the TCP scenario, ACS consistently outperforms CUSUM and DCW.

VI. CONCLUSION

In this paper, we propose a robust detector ACS, which reduces false alarm rate P_{FM} without lowering the detection probability P_D . We also show that the detection performance of CUSUM is not greatly improved by metric transformation.

ACS retains a good detection performance regardless of the metric type or transformation used. Furthermore, a sliding window with the optimal size can capture the dynamics of signals including its change points. Our simulation shows that ACS associated the small window size performs well in detecting the switch of congestion states.

VII. ACKNOWLEDGEMENTS

This work is supported in part by NSF CNS-5030210 and DUE-0516825.

REFERENCES

- [1] B. Brodsky and B. Darkhovsky, *Nonparametric statistical Diagnosis: Problems and Methods*. Netherlands: Kluwer Academic Publishers, 2000.
- [2] H. Wang, D. Zhang, and K. G. Shin, "Detecting SYN flooding attacks," in *Proceedings of IEEE Infocomm*, 2002.
- [3] T. Peng, C. Leckie, and K. Ramamohanarao, "Proactively detecting DDoS attack using source IP address monitoring," in *Proceedings of IFIP-TC6 conference on Networking*, Athens, Greece, May 2004.
- [4] T. Peng, C. Leckie, and R. Kotagiri, "Prevention from distributed denial of service using history-based ip filtering," in *Proceedings of IEEE International Conference on Communications*, Anchorage, Alaska, USA, August 2003.
- [5] H. Wang, D. Zhang, and K. G. Shin, "Change-point monitoring for the detection of DoS attack," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 4, pp. 193–208, 2004.
- [6] D. Rubenstein, J. F. Kurose, and D. F. Towsley, "Detecting shared congestion of flows via end-to-end measurement," in *Proceedings of ACM SIGMETRICS*, June 2000.
- [7] D. Katabi, I. Bazzu, and X. Yang, "A passive approach for detecting shared bottlenecks," in *Proceedings of the 10th IEEE International Conference on Computer Communications and Networks*, Oct 2001.
- [8] M. S. Kim, T. Kim, Y. Shin, S. S. Lam, and E. J. Powers, "A wavelet-based approach to detect shared congestion," in *Proceedings of ACM Sigcomm*, Portland, August 2004.
- [9] K. Harfoush, A. Bestavros, and J. Byers, "Robust identification of shared losses using end-to-end unicast probes," in *Proceedings of IEEE International Conference on Network Protocols*, November 2000.
- [10] J.-J. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, N.J: Prentice Hall, 1991.
- [11] C. Edwards and S. K. Spurgeon, *Sliding Mode Control - Theory and Applications*. Bristol, PA: Taylor and Francis Inc, 1998.
- [12] V. I. Utkin, *Sliding Mode in Control Optimization*. New York: Springer Verlag, 1992.
- [13] A. Isodori, *Nonlinear Control Systems: An Introduction*. New York: Springer Verlag, 1989.
- [14] J. Filipiak, *Modeling and Control of Dynamic Flows in Communication Networks*. New York: Springer-Verlag, 1988.
- [15] A. Isodori, *Lecture Notes on Nonlinear Control (Notes for a Course at the Carl Cranz Gesellschaft)*, Aug 1987.
- [16] S. M. KAY, *Fundamentals of Statistical Signal Processing*. New Jersey: prentice-Hall, Inc., 1998.
- [17] R. B. Dell, S. Holleran, and R. Ramakrishnan, "Sample size determination," *ILAR*, vol. 43, no. 4, pp. 207–213, 2002.
- [18] G. W. Snedecor and W. G. Cochran, *Statistical Methods*. Ames, Iowa: Iowa State University Press, 1989.
- [19] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. Mahwah, NJ: Lawrence Erlbaum Associates, 1988.
- [20] P. D. Allison, *Missing Data*. Thousand Oaks, CA: Sage Publications, 2002.
- [21] D. B. R. Roderick J. A. Little, *Statistical Analysis with Missing Data, Second Edition*. New York: Wiley-Interscience, 1990.
- [22] "Packet loss." [Online]. Available: <http://www.internettrafficreport.com>
- [23] "Ns2." [Online]. Available: <http://www.isi.edu/nsnam/ns/>
- [24] "Matlab." [Online]. Available: <http://www.mathworks.com/>

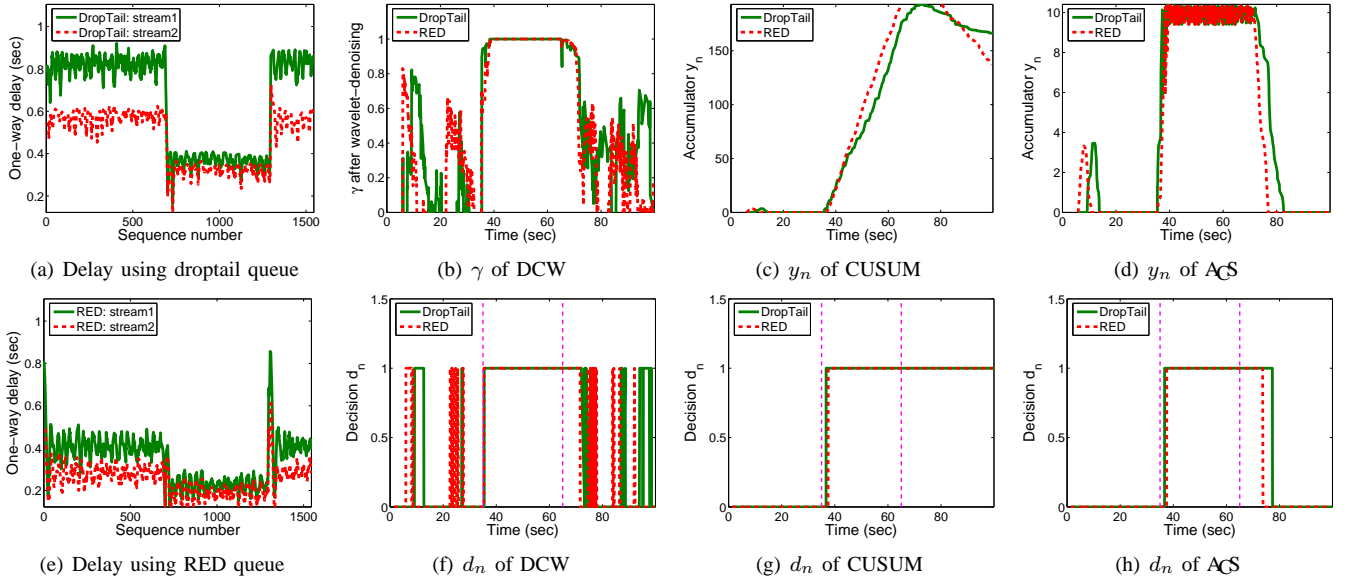


Fig. 12. Detection of congestion states in the background of TCP traffic .

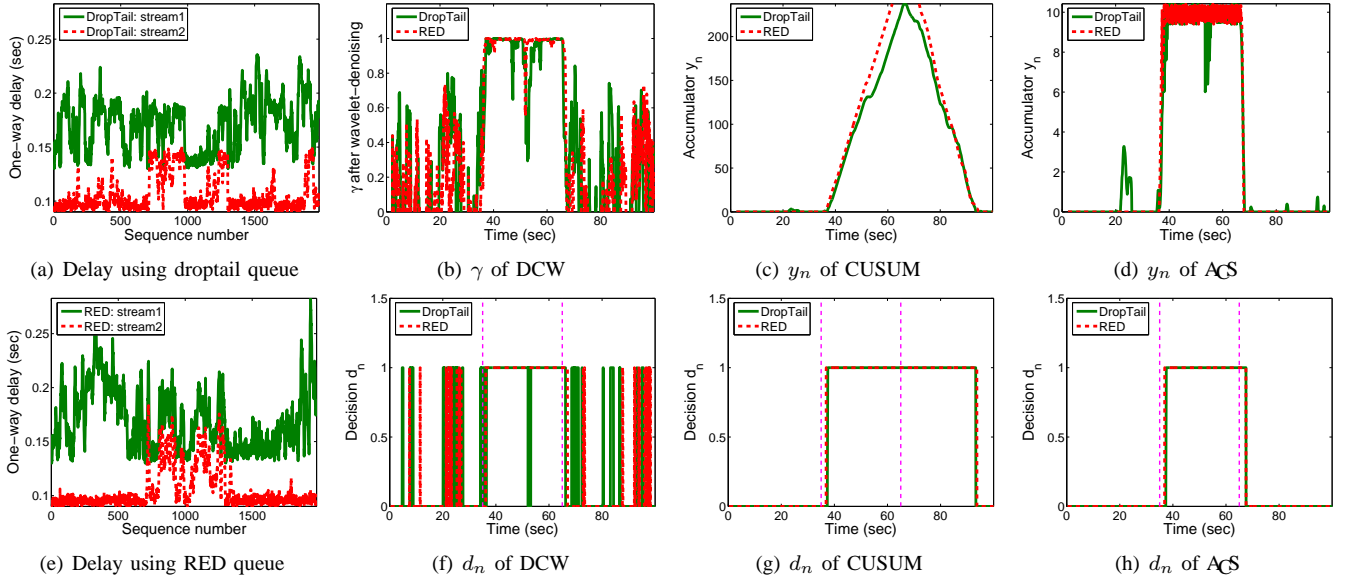


Fig. 13. Detection of congestion states in the background of UDP traffic .

APPENDIX A PROOF OF THEOREM 4.2

We use Lagrangian multiplier $\lambda > 0$ to minimize P_{FA} for a given P_M . Forming the Lagrangian

$$\begin{aligned}
 F &= P_{FA} + \lambda(P_M - \hat{\omega}) \\
 &= \int_{R_1} p(x; H_0) dx + \lambda \left(\int_{R_0} p(x; H_1) dx - \hat{\omega} \right) \\
 &= \int_{R_1} (p(x; H_0) - \lambda p(x; H_1)) dx + \lambda(1 - \hat{\omega}),
 \end{aligned}$$

where $R_0 = \{x : \text{decide } H_0\}$ and $R_1 = \{x : \text{decide } H_1\}$. To minimize F we should include x in R_1 if the integrand is negative for the value of x , or

$$p(x; H_0) - \lambda p(x; H_1) < 0.$$

When $p(x; H_0) - \lambda p(x; H_1) = 0$, x may be included in either R_0 or R_1 . However, the probability of this case approaches 0 for continuous PDFs. Thereby we let $\hat{\gamma} = 1/\lambda$, and decide H_1 if

$$\frac{p(x; H_1)}{p(x; H_0)} > \hat{\gamma},$$

where the threshold $\hat{\gamma}$ is obtained by $P_M = \hat{\omega}$.

APPENDIX B DERIVATION OF NP TEST

By the hypothesis test (22), the NP test can be found as follows. Given $P_M = \hat{\omega}$, H_0 can be decided using (23) if

$$\frac{p(x; H_1)}{p(x; H_0)} = \frac{\frac{1}{\sqrt{2\pi}} e^{-\frac{(x-h)^2}{2\sigma^2}}}{\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}} < \hat{\gamma},$$

or equivalently

$$e^{\frac{2xh-h^2}{2\sigma^2}} < \hat{\gamma}.$$

Taking natural logarithm on both sides, it is followed by

$$x < \frac{2\sigma^2 \ln \hat{\gamma} + h^2}{2h}.$$

Let $\check{\gamma} = \frac{2\sigma^2 \ln \hat{\gamma} + h^2}{2h}$, we can determine $\check{\gamma}$ by the P_M constraint as follows:

$$P_M = Pr\{x < \check{\gamma}; H_1\} = \hat{\omega}$$

such that

$$\int_{-\infty}^{\check{\gamma}} \frac{1}{\sqrt{2\pi}} e^{-\frac{(\tau-h)^2}{2\sigma^2}} d\tau = \hat{\omega}. \quad (29)$$

Given h , σ and $\hat{\omega}$, $\check{\gamma}$ can be obtained using (29). Eventually, the NP detector decides H_0 if $x < \check{\gamma}$ and H_1 if $x > \check{\gamma}$.