社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2003－DSM－31　(17)
2003／11／26

内容と数量に基づくパケット選択プロセッサの設計と実現

高木 淳史 †　加藤 和夫 †　片山 喜章 †　高橋直久 †

概要

ネットワークが高速になりトラヒック量が増大するにつれて，パケットキャプチャを用いたネットワークモニタが困難になってきている．ネットワークモニタシステムが分析，診断のために必要となるパケットのみを選択してキャプチャすれば，システム内部に流れるパケット流量を抑えることができ，高速ネットワークに対応できるようになる．本稿ではこのようなパケットキャプチャのためのパケット選択プロセッサを提案する．パケットヘッダの内容に基づくパケット選択（内容照合）とネットワーク上のパケット流量に基づくパケット選択（数量照合）を連携させることで，特定のフィルタ条件に適合するパケットが大量に現れた場合に選択したり，流量が多いパケットでも特定のフィルタ条件に適合する場合には選択しないことが可能になる．また，従来の数量照合における false negatives（パケット流量が多いにもかかわらず多いと判定できない問題）の発生を削減するための手法として，スライド式多重計数表を提案する．提案プロセッサにより，DoS 攻撃の一つである SYN Flood の検出が可能であることを示す.

Design and Implementation of Packet Selection Processor Based on Quantity and Contents

Atsushi Takagi†　Kazuo Kato†　Yoshiaki Katayama†　Naohisa Takahashi†

Abstract

It is difficult to capture all packets in high-speed network because of increasing packet. If packet capture system selects capturing packets, it can deal high-speed network. In this paper, we propose a packet selection processor which selects a packet required for a monitor. It combines filter based packet maching (based on contents of packet header) and threshold based packet maching (based on quantity of packets). Thus, it can select the capture-packets which matches specific filter conditions appears in large quantities, and select the non-capture-pakckets which large quantities packet but mathces specific filter condition. We also propose slide-multi-table, the way that reduce false negatives (packets quantity is large but system can't detect). We simulate our prosessor in software, we obtain our prosessor can detect SYN Flood, one of the DoS attack.

1　はじめに

ネットワークの安定運用のためには，悪意のあるホストからの攻撃や，ネットワーク機器の動作をモニタすることが不可欠である．モニタの方法として，ネットワークを流れる実運用トラヒック中のパケットをキャプチャし，その結果を解析して情報を得る方法 (Passive 計測法) が広く用いられている [1].

パケットのキャプチャはネットワークが高速になると困難となる．理由として，a) 記憶装置へのアクセス速度に対してネットワーク速度が高速であるため，パケット保存処理が間に合わなくなる，b) 通信の高速化によるトラヒックの増大により，保存用バッファが短時間であふれてしまう，などが上げられる．

高速ネットワークにおいて長時間のパケットキャプチャを可能にするためのアプローチとして，

(1) 高速な CPU と大容量の記憶装置を持つ専用ハードウェアを使用する
(2) トラヒックの一部のパケットを選択してキャプチャ，解析することで元のネットワークの状態を推定する

という二種類の方法がある．

(1) のアプローチでは，どれだけ大量の記憶装置を搭載したとしても定常的な監視をするには限界がある．また，多地点で観測する場合には大規模なキャプチャシステムでは不向きである．一方，(2) のアプローチでは，パケット選択のロジックを 1 チップ化しキャプチャシステムを小規模化することができれば，定常的，多地点の監視ができると期待できる．

本研究グループでは，高速ネットワークのモニタを可能にするためのツールとして，スケーラブルパケットキャプチャシステム SPACE を開発中である [2]. SPACE ではパケットヘッダの切り出し，パケットの選択，タイムスタンプ付与などの機能を通信機能の一部として 1 チップに納めて実装することで，汎用 PC の NIC（Network Interface Card) 上にパケットキャプチャの基本機能を実現することを目指している．

SPACE ではネットワークモニタに必要なパケットをキャプチャし，それ以外は出来るだけ廃棄する．これにより，PC への取り込みのボトルネックを解消し，ソフトウェアによる処理の負荷を軽減させる．SPACE では次のような条件を満たすパケットについて，キャプチャするか廃棄するかを選択できるシステムを目指している．

(1) パケットヘッダなどの内容が特定の条件を満たすパケット　ex.) ルータ設定異常，特定サービスの通信
(2) 一定時間当たりのパケット流量が異常なほど多いパケット　ex.)IP フラッディング攻撃

†名古屋工業大学大学院 工学研究科
Graduate school of Engineering, Nagoya Institute of Technology

(3) 上記二つの条件をあわせ持つパケット
ex.) 特定サービスのアクセス集中，DoS 攻撃
このようなパケットをキャプチャあるいは廃棄できるようにするには，パケットヘッダの内容とパケットの流量をキャプチャの条件として指定できる必要がある．

本稿では，SPACE においてキャプチャすべきパケットを選択するためのプロセッサを提案する．提案プロセッサの特徴は，(1) 内容と数量に基づくパケット選択ができる，(2) ポジティブな近似を行う，(3)1 チップ程度の規模で実現，の 3 点である．

(1) 内容と数量に基づくパケット選択　パケットヘッダの値をそれぞれ満たすべき条件（フィルタ）と照合する機能，および一定時間当たりの特定パケットの到着数を計数する機能を連携させる．具体的には，照合に成功したフィルタの一定時間当たりの成功回数を計数することで実現する．

また，内容の照合に成功した場合にとるべき動作として 4 種類のアクションを定義する．このアクションにより，特定の内容の条件を満たすパケットが大量に現れた場合に選択したり，流量が多いパケットでも特定の内容条件に適合する場合には選択しないことが可能になる．

(2) ポジティブな近似　提案プロセッサでは，選択処理の高速化および少リソース化のためにユーザが指定したキャプチャ条件に対しポジティブな近似をする．ポジティブな近似とは，キャプチャ条件を近似する際に，パケットを廃棄する条件が大きくならないように近似することで，キャプチャすべきパケットは全てキャプチャすることを保証する近似である．ポジティブな近似により，キャプチャすべきパケットを SPACE が廃棄してしまい，ネットワークモニタで攻撃が検知できなくなるというような事態を回避する．

提案プロセッサでは，パケット数を保持するためにスライド式多重計数表という手法を提案し，false negatives（本来は流量が多いと判定されなければならないパケットが流量が少ないと判定されてしまうこと）の発生を軽減させ，ポジティブな近似を可能にする．

(3)1 チップ程度の規模で実現　必要なリソースを削減するため，内容に基づくパケット選択に選択近似機能を有する空間分割型パケット分類法 [3] を用いる．この手法は，パケット分類を作表計算（事前に実行可能な計算）と索表計算（分類の実行）の二つの部分計算に分割することで，パケット分類を高速に行う．また，空間分割型パケット分類法に近似の機能を加えることで，制限されたリソースのもとで照合を可能とする．

また，パケットの数量に基づくパケット選択ではパケット数の計数にハッシュテーブルを用いて必要なリソースを削減する．また，ハッシュ値の衝突回避の処理を行わないことで，ノイズを含むが高速な照合が可能となる．

本稿の構成は以下のとおり．まず，2 章で関連研究について述べ，3 章，4 章で提案するパケット選択プロセッサの詳細について述べる．また，5 章でスライド式多重計数表の理論的評価および選択プロセッサのソフトウェアシミュレーションによる評価を行い，最後に 6 章で結論および今後の課題について述べる．

2　関連研究

選択的パケットキャプチャの手法として，Cisco 社の NetFlow[4] がある．NetFlow では，受信した全パケットのうち一部のパケットをランダムにキャプチャし (サンプリング)，元のトラヒックの状態を推定している．

しかし，ランダムサンプリングではモニタの対象ではないパケットまでキャプチャしてしまうため，モニタの対象となるパケットがあらかじめ分かっている場合は非効率であり，推定も不正確になってしまう．提案プロセッサは，あらかじめモニタの対象となるパケットを指定して，モニタに必要なパケットを選択的にキャプチャできるので，効率的で正確な推定が可能になると期待できる．

Estan ら [5] は，パケット流量が多いフローを識別する手法を提案している．フローとは，同じ送信元ホスト／宛て先ホスト間のパケットを同じクラスとするもので，具体的には 5 つのフィールドの組（送信元 IP アドレス，送信元ポート番号，宛て先 IP アドレス，宛て先ポート番号，プロトコル）などを用いている．この方式では，QoS 評価などでフローごとの特性を詳細に解析する場合に適しているが，DDoS 攻撃のような多数のホストから一つの攻撃対象ホストへ大量のトラフィックを送信する攻撃を検出する場合は，複数のフローを合わせて分析する必要がある．

それに対し，本稿ではパケットを以下のようにクラス分けを行う．

- 同じ送信元ホストのパケットは同じクラス
- 同じ宛て先ホストのパケットは同じクラス
- 同じ選択条件に適合するパケットは同じクラス

提案プロセッサではこれらのクラスごとにパケット数を計数する．これにより，先ほどの DDoS 攻撃の例では，同じ宛て先に大量のパケットが現れるので，提案プロセッサでは攻撃を検知可能である．また，比較的簡単な機構で実現可能である．

文献 [5] では，並列ハッシュテーブルにパケット流量を保存することで，パケット流量が多いフローを識別する．この手法は，時間をある一定時間で量子化し，量子化された一つの時間区間内に受信したパケットのサイズからそのパケットが属するフローのパケット流量を判定している．

この Estan らの手法は false negatives は存在しないとしているが，これはフローが十分長く続くことを仮定しているためであり，フローが十分長く続かない場合には false negatives が発生する可能性がある．本稿で提案するスライド式多重計数表方式は，この false negatives を削減する効果がある．

桐村ら [6] は，ネットワークモニタを FPGA 上で実装することで，ハードウェアであるが検出項目の再構成が可能なモニタシステムを提案している．検出したい項目ごとにモジュールをあらかじめ生成し，FPGA 回路にそのモジュールを搭載することで検出項目の追加，変更を行っている．しかし，FPGA 回路を変更するには専用の装置が必要であり，容易に変更を行うことはできない．

それに対し本手法では，提案プロセッサに与える探索表を変更することにより，パケット選択条件を追加，変更できる．これは回路自体を変更するものではないため，デバイスドライバ経由で指定することができ，専用の装置を必要とせず容易に追加，変更を行うことができる．

3　パケット選択条件の指定法

SPACE ではパケットを受信した際に，与えられた選択条件に基づいてそのパケットを選択するか決定する．本章では，パケット選択条件の記述法について定義する．選択条件は数量条件 (Threshold Condition : 以下 Th.c) と内容条件（以下：フィルタ）の二つに分けることができる．

3.1　数量に基づく選択条件

数量条件 Th.c は，数量 τ と時間 Δ をからなる．Δ 時間当たりの同じクラスに属するパケット数が τ 個以上であった場合，そのパケットは集中していると定義

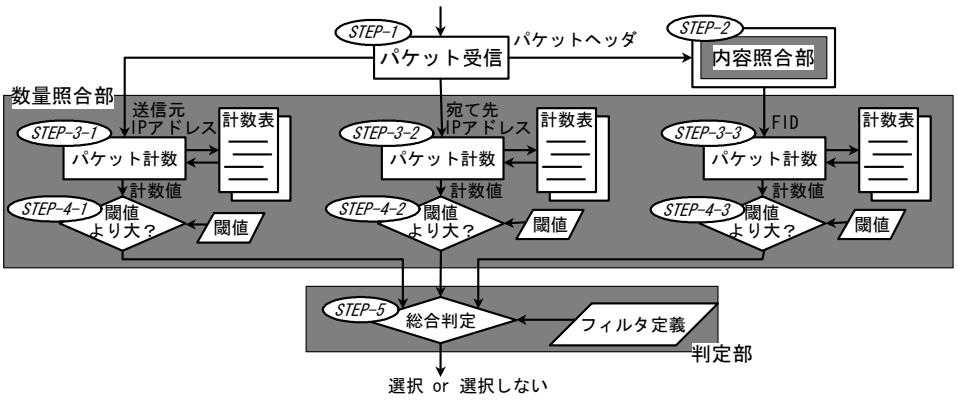


図 1: パケット選択プロセッサの構成

表 1: 内容と数量に基づく 4 つのアクション

アクション	フィルタに適合した場合の動作
ACCEPT	選択する
WEAKLY-ACCEPT	集中していれば選択
WEAKLY-REJECT	集中していれば廃棄
REJECT	選択しない

する．今後，τ のことを閾値，Δ のことをモニタ単位時間という．

フラッディング攻撃を行っているホストや帯域を占有しているホストのパケットだけを選択したい場合では，あらかじめ選択条件としてそれらのホストを指定することはできない．そこで，そのような集中的なパケットを送信している特定のホストを指定するために，”ある値”を意味する「ANY」という値を定義する．例えばポート番号 80 番のパケットを集中して送信しているホストが居た場合にそのパケットを選択したいときは，「Src IP Address ANY，TCP Port 80」と指定する．

閾値 τ とモニタ単位時間 Δ を決定する方式として，本稿では事前のネットワーク観測からの学習値から決定する方式を提案する．具体的には，対象とするネットワークで短時間パケットキャプチャをし，そのときの 1 秒当たりの平均パケット数 $\overline{N}$ をあらかじめ求めておき，τ を $\overline{N}$ の一定割合の値として定める．たとえば，$\overline{N}$ = 1000[個/秒] のとき，その 10%を占めるパケットが現れた場合に短時間に集中したと判断し，τ/Δ = 100 と決定する．

3.2 内容に基づく選択条件

フィルタは，条件，識別子 (FilterID : 以下 FID)，アクションの 3 つからなる．条件にはパケットヘッダの各フィールドが満たすべき値の条件を記述する．あるパケットのヘッダがこの満たすべき条件を全て満たした場合，そのフィルタに適合する，あるいはそのフィルタは照合に成功すると言う．アクションは，パケットがフィルタに適合した際にとるべき動作を指定する．

既存の多くのフィルタリングでは，パケットのキャプチャ／廃棄のアクションとして ACCEPT（そのパケットを選択）または REJECT（そのパケットを廃棄）の二種類を提供している．提案プロセッサではこれに加え，WEAKLY-ACCEPT（集中していれば選択：以下 W-A）と WEAKLY-REJECT（集中していれば廃棄：以下 W-R）という二つの指定を与える（表 1）．

WEAKLY-ACCEPT により，DoS 攻撃のようにパケットが集中したときにそのパケットをキャプチャすることが可能になる．また WEAKLY-REJECT により，アクセスが集中にした場合にモニタシステムにいたずらに負荷を掛けることを回避することが可能となる．

4 提案プロセッサの構成と動作

提案するパケット選択プロセッサの構成を図 1 に示す．提案プロセッサは以下の 3 つの機能からなる．

- **内容照合部:** 内容に基づく照合を行う
- **数量照合部:** 数量に基づく照合を行う
- **判定部:** 上記二つの照合結果から，最終的にそのパケットを選択するかどうかを決定する

照合プロセッサの動作は以下の通りである．

STEP-1 パケットを受信した際に，そのパケットの送信元／宛て先 IP アドレスをパケット計数ブロックに，パケットヘッダを内容照合部にそれぞれ渡す
STEP-2 内容の照合を行い，その結果として適合したフィルタの FID をパケット計数ブロックに渡す
STEP-3 IP アドレスと FID に対してそれぞれパケット数を計数する
STEP-4 計数値と閾値を比較判定する
STEP-5 3 つの判定結果とフィルタ定義から，受信したパケットを選択するか決定する

以下に，それぞれの詳細を述べる．

4.1 内容照合部

内容照合部では，選択近似機能を有する空間分割型パケット分類法 [3] に基づき，フィルタ照合を行う．空間分割型パケット分類法は高速な照合処理が可能だが大量のメモリが必要になるという欠点がある．文献 [3] に示したように，空間分割に近似を適用することで，必要メモリ量を 1 チップに収まる程度に削減する．

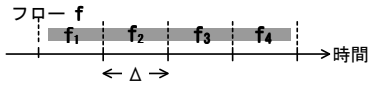
内容照合部は入力として受信したパケットのヘッダを受け取り，ユーザから指定されたフィルタのなかで，ANY 以外の条件について，受信したパケットに適合するものがあるか照合する．照合の結果，受信したパケットに適合するフィルタがあれば，そのフィルタの識別子 FID を数量照合部へ転送する．

4.2 数量照合部

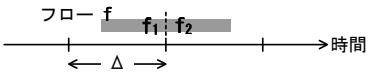
ここで，送信元 IP アドレスが同じパケットは同一クラスに属するとみなす．同様に，宛て先 IP アドレスが同じパケット，あるいは同じフィルタに適合するパケットはそれぞれ同じクラスに属するとみなす．

数量照合部では，上記クラスごとに到着パケットを計数し，結果を計数表に格納する．IP アドレス別のパケット数を計数により，パケットが集中している IP アドレスの判定（ANY の判定）を行い，また FID を計数により，そのフィルタに適合するパケットが集中して現れているかを検出（内容と数量の判定）する．

計数表の実現にはハッシュテーブルを用る．ここで，ハードウェアの簡単化と高速な照合のために，ハッシュ値の衝突時に回避処理は行わない．これにより，実際のパケット数は閾値未満であるが，閾値よりも大きい



(a) 継続時間：長



(b) 継続時間：短

図 2: 継続時間と一つの区間内で数えることができるパケット

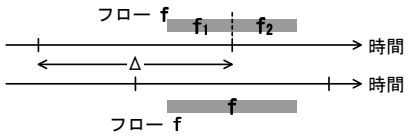


図 3: スライド式多重計数表を使用した場合

と判定する誤差が発生する可能性がある．しかしこれはポジティブな近似であるため，提案プロセッサではこれを許容する．

数量照合部はパケット計数ブロック（図 1，STEP-3）と計数値判定ブロック（同 STEP-4）のからなる．パケット計数ブロックは送信元／宛て先 IP アドレスおよび FID のいずれかを入力として受け取り，その値をキーとしてハッシュ値を求め，ハッシュ値から計数表のエントリを求める．計数表から現在のパケット数を得たのち計数値を 1 増加し，計数値判定ブロックに渡す．

計数値判定ブロックは，計数値とあらかじめ指定された閾値 τ とを比較判定し，計数値が τ よりも多い場合に T(true) を返す．

計数表の実現方式を次節で述べる

4.2.1 スライド式多重計数表

文献 [5] では，時間 Δ 当たりのフロー（文献 [5] では前述の 5 つのフィールドの組み，送信元／宛て先ホストのペア，送信元／宛て先 AS のペアの 3 種類をフローとして定義している）のパケット数が閾値を超えているか判定するため，時間を Δ 間隔毎に分割し，その間隔中に受信したパケット数を閾値と比較している．

この方法では，フローの継続時間が Δ よりも十分長い場合に都合が良い．継続時間が十分長ければ，一部の（あるいはほとんどの）分割された区間で，そのフローの Δ 時間当たりのパケット数を得られるためである（図 2(a)）．

逆に言えば，フローの継続時間が Δ より十分長くないと Δ 時間当たりのフローのパケット数が正確に計数できない（図 2(b)）．このようなフローは，本来なら閾値を超えているため選択されるべきパケットであっても，閾値を超えていないと判定し選択できないという問題，つまり false negatives が起こり得る．

本稿ではこの問題に対し，計測時間をずらした複数の計数表を用いるスライド式多重計数表方式を提案する．この方式では複数の計数表で同時にパケット数を計数し，いずれかの計数値が τ より大きければ集中していると判定する．こうすることで，ある計数表でで false negatives が起こり得る場合に別の計数表でカバーするので (図 3)，false negatives の発生を抑えることができる．

ただし，スライド式多重計数表では複数の計数表を使用するので，通常の計数時と比べてメモリ使用量が多くなる．

4.3 判定部

総合判定ブロック (STEP-5) は STEP-4 で出力された各条件（送信元 IP アドレス／宛て先 IP アドレス／FID) の閾値判定結果を受け取る．受信したパケットが適合したフィルタのアクションをフィルタ定義から参照し，各閾値判定結果とアクションからパケットを選択するか決定する．

5 評価

5.1 スライド式多重計数表

提案したスライド式多重計数表について，計数表が二重の場合について定性的な評価をする．Δ に対してあるクラスのパケット数が集中している時間が十分長くない場合に，スライド式多重計数表を使用する場合と使用しない場合とで，どのような継続時間，個数のパケットが来たときにどれだけの確率でパケットが選択されるかを評価する．

まず，以下の値を定義する．

- Δ :モニタ単位時間
- τ :閾値
- T :同一クラスのパケット数が集中している時間（以下 継続時間）
- n :同一クラスのパケット数
- $N = \begin{cases} n & [T < \Delta] \\ \frac{\Delta}{T} n & [T \geq \Delta] \end{cases}$:Δ 時間当たりのパケット数

ここで，パケットの到着間隔は常に一定と仮定する．また，N を場合分けしているのは，$T < \Delta$ のときに $N = (\Delta/T)n$ とすると，実際に受信したパケット量よりも N が大きい値になってしまうためである．

さらに，これらの値から，パケット数，継続時間を表す係数として以下を定義する．

- $k_\Delta = T/\Delta$:Δ に対する継続時間の割合
- $k_\tau = N/\tau$:τ に対するパケット数の割合

5.1.1 理論値の計算

パケット数を閾値以上と判定する確率 p_c の理論値を求める．まず，同一クラスのパケットがどのタイミングで来たらどれだけの数のパケットを一つの区間の中で数えられるかを考える．最初のパケットの到着時刻を x，一つの区間内で数えることができるパケット数を y として，横軸 x，縦軸 y としてグラフ化したものが図 4 である．一つの区間内で数えることができるパケット数 y がシステムの閾値 τ 以上だった場合に，そのパケットは閾値以上と判定する．このとき，x 全体（0 ~ Δ）に対する $y \geq \tau$ となる x の割合がパケット数が閾値以上と計数できる確率，つまりパケットを選択する確率となる．

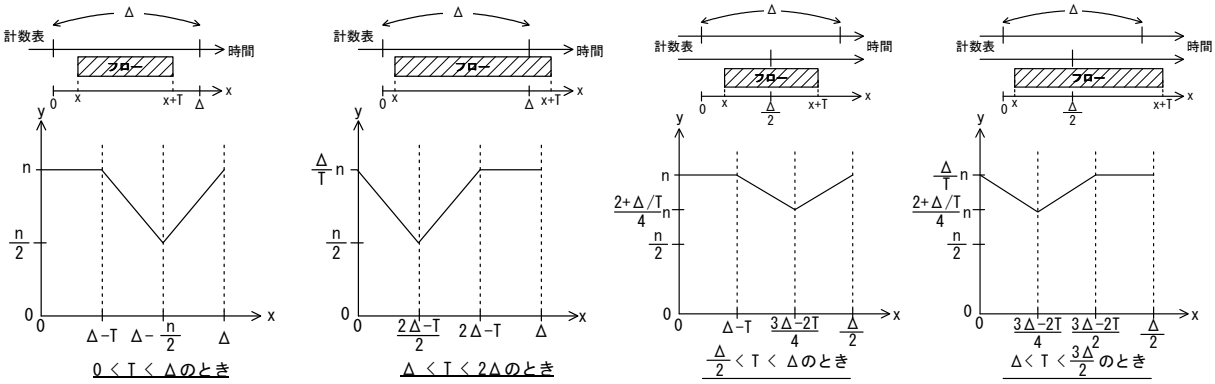
5.1.2 考察

まず，次のような理想的な選択プロセッサを使用した場合について考えてみる．すなわち，モニタ単位時間時間当たりのパケット数 N が閾値 τ 以上の場合は必ずそのパケットを選択し，τ 未満の場合は必ず選択をしないという選択プロセッサを考える．このプロセッサを用いたときの k_τ と p_c の関係を図示すると図 5(a) となる．この図より，$k_\tau < 1$ のときに $p_c > 0$ であれば false positives（パケット流量が少ないにもかかわらず閾値以上と判定されること），$k_\tau \geq 1$ のときに $p_c < 1$ であれば false negatives であることが分かる．

つぎに，各種 k_Δ の値に対して 5.1.1 節の理論の元にパケットを選択する確率を計算すると図 5(b)〜5(f) を得る．これらの図では，横軸を k_τ，縦軸を選択確率 p_c としてスライド式多重計数表を使用した場合と使用しない場合を合わせてグラフにしたものを示す．これらの図より，スライド式多重計数表を使用したほうが，どの k_Δ の値に対しても常に false negatives が少なく，理想値に近い特性を得られることが分かる．

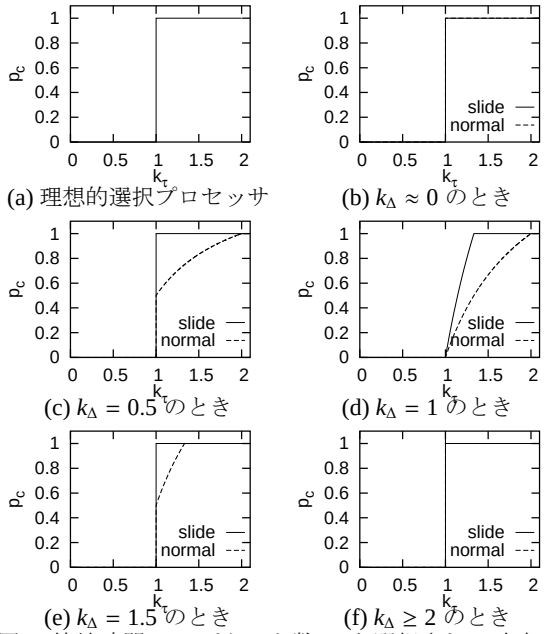
5.1.3 false negatives を無くす閾値の推定

前節よりスライド式計数表を使用すれば，false negatives の数が減少することが分かった．しかし，false



(a) スライド式多重計数表を用いない場合　　(b) スライド式多重計数表を用いた場合

図 4: 集中するパケットの開始時刻 x と 1 区間中で計数可能パケット数 y



(a) 理想的選択プロセッサ　(b) $k_\Delta \approx 0$ のとき

(c) $k_\Delta = 0.5$ のとき　(d) $k_\Delta = 1$ のとき

(e) $k_\Delta = 1.5$ のとき　(f) $k_\Delta \geq 2$ のとき

図 5: 持続時間 k_Δ，パケット数 k_τ と選択される確率 p_c の関係

negatives は無くなったわけではない．ポジティブな近似では false negatives を無くさなければならない．

そこで，false negatives を無くすため，ユーザから指定された閾値に対し，プロセッサで使用する閾値を低い値に設定するという手法を用いる．この手法により，ユーザが指定した閾値に満たない流量のパケットを選択できる．この手法では，ユーザから指定された閾値に対する false positives を増加させてしまうが，その代わりに false negatives を無くすことができるようになる．以下，ユーザから指定された閾値をユーザ閾値 τ_u，プロセッサで使用する閾値をプロセッサ閾値 τ_p とする．

プロセッサ閾値 τ_p をユーザ閾値 τ_u に対してどの程度低くすればいいかは，false negatives が発生する（つまり $p_c < 1$ となる）k_τ の最大値に関係する．

具体的には，それぞれ $k_\tau = 2$[スライド未使用時]，$k_\tau = 1.33$[スライド使用] である（図 5(d)）．よって，それぞれ $\tau_p = (1/2)\tau_u$[スライド未使用時] と $\tau_p = (3/4)\tau_u$[スライド使用時] にすると $p_c < 1$ となる k_τ の最大値を $k_\tau = 1$ にでき，このとき false negatives が発生しない（図 6）．この場合にも，図より，スライド式計数表を使用した場合のほうが false positives が少なく，理想値（図 5(a)）により近いといえる．

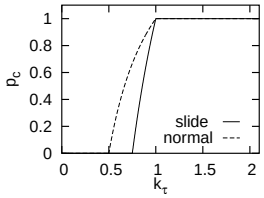


図 6: false negatives を無くすようにプロセッサの閾値を変更した場合のパケット選択確率 [$k_\Delta = 1$ のとき]

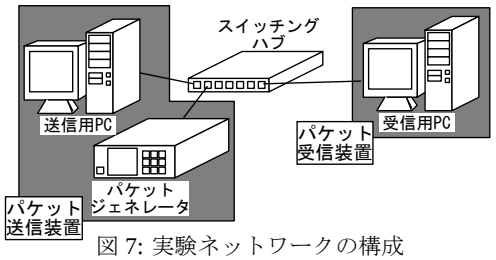


図 7: 実験ネットワークの構成

5.2　パケットの選択

提案するプロセッサをソフトウェアで実装し，以下の 2 点について評価を行う．

- 3.1 節で示した学習値による閾値の決定法の妥当性
- モニタ単位時間 Δ の選択結果に対する感度

感度とは，Δ の値の設定がパケット選択結果にどれだけ影響するかを示す値であり，以下のように定義する．

$$\text{感度} = \left| \frac{\text{パケット選択数の変化量}}{\text{モニタ単位時間}\Delta\text{の変化量}} \right| \qquad (1)$$

5.2.1　実験環境

実験ネットワークの構成は図 7 の通り．パケットはトラヒックジェネレータ (Anritsu MD1230A[7]) とパケット送信用 PC を用いて生成し，それを提案プロセッサが動作する受信用 PC で受信する．送信用，受信用 PC の環境は表 2 に示す．また，ネットワークは 1000Base-T Ethernet を使用した．

本実験では，通常のトラヒックに DoS 攻撃のパケットを混入させた場合に，DoS 攻撃のパケットがどれだけ選択できたかを求めた．通常のトラヒックとして，MAWI[8] で公開されているトラヒックデータを NetPoke [9] により実験用 PC から送信した．また DoS 攻撃として，トラヒックジェネレータから SYN Flood を模擬したパケットを送信した．

実験のステップは以下の通りである．

STEP-1 実験 PC 上のプロセッサに選択条件を与える
STEP-2 パケットジェネレータと送信用 PC でパケットを送信し，受信用 PC で受信する
STEP-3 提案プロセッサがパケットを選択する

表 2: 実験用 PC 環境

OS	RedHat Linux 7.3
MotherBoad	Supermicro X5DPE-G2
CPU	Intel Xeon 2.4MHz x 2
Memory	3 GBytes
Hard Disk	Maxtor DiamondMax16 80GB ATA/133 x 2

表 3: 実験時に送信したパケット

送信パケット内容	送信数 [個]	送信時間 [s]	送信数 [個/s]
SYN Flood	424,677	900	471.9
通常トラヒック	4,246,775	900	4,718.6
合計	4,671,452	—	5,190.5

STEP-4 どのパケットをどれだけ選択したかを評価する

5.2.2 実験用パケット

実験時に送信したパケット数を表 3 に示す．通常トラヒックとして，MAWI の samplepoint-B における 8 月 25 日（14:00〜14:15 の 15 分間）のデータを使用した．また疑似 SYN Flood として，TCP SYN フラグビットを 1 としたパケットを特定の宛て先に対し通常トラフィックの 10%の数だけ送信した．

5.2.3 実験で用いた選択条件

本実験で使用したパケット選択条件を表 4 に示す．感度の評価のため，モニタ単位時間は 0.1〜50[s] の間の 6 つの値を使用した．ユーザ指定閾値は，学習値による決定法に基づき，通常トラフィックの最初 3 分間における 1 秒あたりのパケット数の平均値である，471.2[個/s] から決定した．

提案システムでポジティブな近似を行うため，スライド式計数表を使用し，ユーザ指定閾値 τ_u の 3/4 をプロセッサの閾値 τ_p として使用した．

5.2.4 選択結果

実験によって得られた各パケットの選択数を表 5 に示す．表 5 より，全ての場合において選択したパケットの 99%以上は SYN Flood のパケットという結果となり，SYN Flood の選択に成功しているといえる．これより，学習値による閾値 τ の決定は効果があることが示された．通常トラヒックを若干数選択しているが，選択したパケットを検査した結果，これはハッシュ値の衝突による誤差であることが分かった．

表 6 にモニタ単位時間 Δ の選択パケット数に対する感度の結果を示す．Δ が 5[s] 以上の場合には，選択数（約 200,000 個）に対して感度が小さい，すなわち Δ の値を多少変化させても選択パケット数はほとんど変化しないことが分かった．

表 4: 実験に使用した選択条件

(a) Th.c の定義

モニタ単位時間 Δ[s]	ユーザ指定閾値 τ_u[個]	プロセッサ設定閾値 τ_p[個]
0.1	47	35
0.5	236	177
1.0	471	353
5.0	2,356	1,767
10.0	4,712	3,534
50.0	23,559	17,669

(b) フィルタの定義

FID	条件	アクション
1	SrcIP ANY & SYN flag	W-A

表 5: モニタ単位時間 Δ の設定値と選択したパケット数

Δ[s]	総数 [個]	通常 [個]	SYN[個]	SYN 割合 [%]
0.1	338,678	510	338,678	99.85
0.5	240,186	94	240,092	99.99
1.0	228,164	44	228,120	99.98
5.0	215,911	38	215,882	99.99
10.0	213,333	32	213,333	99.99
50.0	202,691	28	202,663	99.99

通常：通常トラフィック　SYN：SYN Flood

表 6: モニタ単位時間 Δ の感度

Δ[s]	感度 [個/s]
0.1 〜 0.5	246,465.00
0.5 〜 1.0	23,944.00
1.0 〜 5.0	3,059.50
5.0 〜 10.0	509.80
10.0 〜 50.0	266.75

6 おわりに

内容と数量に基づくパケットの選択を実現するためのパケット選択プロセッサを提案した．既存の数量に基づく照合で問題となる false negatives を削減するスライド式多重計数表を提案した．評価として，まずスライド式多重計数表の二重の場合における定性的な評価を行い，スライド式多重計数表は false positives の削減に効果を発揮することを示した．また，提案プロセッサをソフトウェアで作成し，評価実験により SYN Flood が選択できることを示し，この実験において次の 2 点を示すことにより，提案プロセッサの 2 つのパラメータ τ，Δ を比較的容易に設定できることを示した．

- 監視対象ネットワークのトラヒックに基づいて設定した τ/Δ の値が有効である
- Δ の値を多少変化させても選択パケット数はほとんど変化しないような Δ の値域がある

これらの結果は特定のトラヒックに対する評価実験から得られたものであるので，他の各種トラヒックについての実験が必要である．

現在，提案プロセッサを搭載した GbE 対応のネットワークインターフェイスボードの開発をすすめている．今後はプロセッサのハードウェア化と，高速ネットワークでの選択動作の評価を進める予定である．また，今回提案したスライド式多重計数表における定量的な評価と，テーブルを三重以上にした場合の評価を行う予定である．

参考文献

[1] 鶴正人, 尾家祐二. インターネットの特性計測技術とその研究開発動向. 情報処理学会 学会誌, Vol. 42, No. 02, pp. 192–197, Feb 2001.

[2] 加藤和夫, 大須賀怜, 高木淳史, 片山喜章, 高橋直久. スケーラブルパケットキャプチャシステムの実現法. 第 5 回インターネットテクノロジーワークショップ, 2003.

[3] 大須賀怜, 加藤和夫, 片山喜章, 高橋直久. 高速パケットキャプチャのための選択近似機能を有する空間分割型パケット分類器の実現と評価. 第 31 回分散システム／インターネット運用技術研究会, 2003.

[4] Cisco NetFlow. http://www.cisco.com/warp/public/732/Tech/netflow.

[5] Cristian Estan and George Varghese. New directions in traffic measurement and accounting. *In Proceedings of ACM SIGCOMM'02*, pp. 323–336, August 2002.

[6] 桐村昌行, 高本佳史, 森亮憲, 安本慶一, 中田明夫, 東野輝夫. 高速ネットワーク向けネットワークモニタ回路の設計と実装. 情報処理学会論文誌, pp. 1593–1603, June 2003.

[7] アンリツ株式会社 MD1230A. http://www.anritsu.co.jp/.

[8] MAWI Working Group Traffic Archive. http://tracer.csl.sony.co.jp/ mawi/.

[9] NetPoke – Tcpdump File Replay Utility. http://www.ll.mit.edu/IST/ideval/tools/tools_index.html.