トラヒックモニタリング高速化技術の検討

Traffic Monitoring Method for High Speed Network

正村　雄介[†a)]　　渡辺　義則[†b)]　　池田　尚哉[†c)]

Yusuke SHOMURA[†a)], Yoshinori WATANABE[†b)], and Naoya IKEDA[†c)]

概要　通信事業者等の広域ネットワークでは、P2P による帯域の占有や DDoS・Worm 等異常フローの発生が問題となり、ネットワークに流れるトラヒック解析技術への要求が高まっている。トラヒックを詳細に解析するには多数の統計情報が必要となるが、ネットワーク回線の高速化が進むと、統計情報を更新する際のメモリアクセスがボトルネックとなり解析に必要な統計情報の収集が困難となる。本研究では、トラヒックの統計情報更新に必要となるメモリアクセス回数を削減する二段階集約方式に加え、メモリアクセスを高速化する複数データ一括更新方式を提案する。

キーワード　DDoS，Worm，P2P，フロー統計，異なり数

1. まえがき

近年、ADSL(Asymmetric Digital Subscriber Line) や FTTH(Fiber To The Home) 技術によるアクセス回線のブロードバンド化に伴い、広帯域な常時接続環境が一般利用者に急速に普及し、ネットワークの利用形態が多様化している。従来の Best Effort 型データ通信だけでなく、音声・動画や基幹業務のトランザクションデータなど、通信の品質保証が必要なデータも通信され始めている。一方、不正なパケットを大量に送信し、サーバやルータをサービス停止状態とする DoS(Denial of Service) 攻撃、DDoS(Distributed Denial of Service) 攻撃の発生、ウィルスや Worm による異常トラヒックの発生、及び、P2P ファイル交換による帯域占有が問題となっている。特に通信事業者の広域ネットワークでは、網管理者が、これらのトラヒックを判別し、QoS 処理やフィルタリング処理を実施し、ネットワークを管理する必要がある。

この様な網管理者の業務を支援するため、DDoS、Worm、P2P 過大トラヒック等の通信帯域を圧迫するフローの検出技術が求められている。そこで我々はデータマイニング技術を用いたトラヒックモニタリング技術の開発を進めてきた [1]。本技術は異常フローの解析に有効な統計情報をレイヤ４までのヘッダ情報から収集する技術であり、統計情報収集に必要な計算処理は非常に少ない。そのため、高速・大容量のネットワークを監視するのに適した技術である。しかし、現在のネットワーク回線はメモリアクセス速度の向上を上回る速さで高速化しているため、メモリアクセス速度がネックとなり解析に必要な統計情報の収集も困難となってきている。

本稿では、この課題に対応するため、ボトルネックとなるメモリアクセス速度を改善する二段階集約方式と一括更新方式を提案し、10Gbps 回線へ適用可能であるかを評価した。

2. 従来技術と課題

トラヒック解析の手法として、専有帯域の大きい特徴的なトラヒックを抽出する方法や、通信相手数、使用ポート数に着目し種類を判別する方法がある。本章では、それら従来技術を説明し、本研究で解決しようとする課題について述べる。

専有帯域の大きい特徴的なフローの抽出には、データマイニングで用いられるバスケット解析技術が適用できる [2], [3]。バスケット解析は頻繁に出現する項目の組み合わせを抽出する手法であり、バスケット解析で言うところの 5tuple 「送信元 IP アドレス (SIP)、宛先 IP アドレス (DIP)、プロトコル番号 (PRT)、送

† アラクサラネットワークス株式会社　〒 212-0058 神奈川県川崎市幸区鹿島田 890 新川崎三井ビル
ALAXALA Networks Corporation　890 Kashimada, Saiwai, Kawasaki, Kanagawa, 212-0058 Japan
a) E-mail: yusuke.shomura@alaxala.com
b) E-mail: yoshinori.watanabe@alaxala.com
c) E-mail: naoya.ikeda@alaxala.com

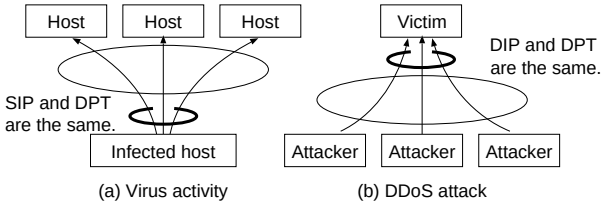

図 1　異常フローの振る舞い

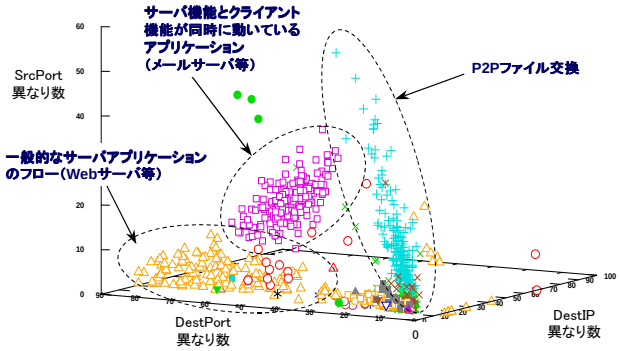

図 2　ホストの振る舞い分析

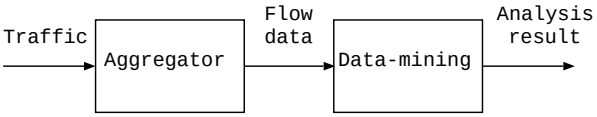

図 3　二段階集約方式

信元ポート番号 (SPT)、宛先ポート番号 (DPT)」を解析すれば流量の少ない DDoS, Worm 等のフローも抽出することが可能となる。例えば、Worm の拡散は、感染ホストが複数のホストに特定のポート番号でアクセスする特徴がある。従って、SIP と DPT が一致するパケットが多数存在するため、感染ホストのアドレスと攻撃ポート番号の組み合わせが頻繁に出現するものとして Worm が抽出可能である (図 1(a))。DDoS 攻撃も特定の DIP と DTP の組み合わせが頻繁に出現するものとして抽出可能である (図 1(b))。

また、上記手法を小メモリで高速に行う Stream Mining 技術も研究されており [2], [4], [5]、リアルタイムの特徴トラヒック抽出を実現している。

一方、統計情報を用いたトラヒックの種類判別において、異なり数を計測することの重要性が着目されている [1], [6]〜[8]。ホスト毎に通信相手やポート番号の異なり数を計測することで、ホストの振舞いを分類し、トラヒックの種別を判別できる。また、異なり数が非常に大きく現れる DDoS、Worm、スキャン等の判別も可能である [9]。

我々が開発してきたモニタリング技術 [1] は上記 2 つの方法を組み合わせトラヒック解析を行う。例えば、送信帯域の大きい送信元に関し、異なり数情報を用いホストの振る舞いを分析すると、一般的なサーバ、サーバ機能とクライアント機能が同時に動くホスト、P2P が動作するホスト等が容易に分類できる (図 2,A· 1,A· 2)。また、頻出する SIP と DPT の組み合わせについて、DIP の異なり数をみることで、Worm の判定も即座に可能となる。

このように、バスケット解析技術と異なり数の計測技術の組み合わせは高い解析性能を持つ。中でも [1] は、限られたメモリ量で動作し、演算処理が少ない。一方、パケット受信時に更新する統計情報は 15 種類前後と多く、メモリアクセスは多い。従って、適用する回線速度が高速になるにつれ、メモリアクセスのボトルネックが顕在化してくる。

上記の考察から、本稿ではメモリアクセス回数の削減及びメモリアクセス速度の高速化について検討した。

3. アルゴリズム

本章では、前節で示したトラヒック解析を高速回線で実施するための高速化技術を提案する。本提案は、二段階集約方式と複数データ一括更新方式の二方式からなる。二段階集約方式は、トラヒック情報をフロー毎の統計情報に集約するフロー集約部 (aggregator) と解析処理を行うデータマイニング部に分離し処理する (図 3)。また、複数データ一括更新方式は、データマイニング部を高速化する。本提案は、フロー集約部をハードウェアで実装し、データマイニング部を PC 上に実装することを想定した方式である。

3.1 フロー集約部

P2P 等の帯域を占有するアプリケーションによるフローのパケット数は非常に多い。フロー集約部でこのようなトラヒックの情報を 1 つのフロー情報へ集約し後段のデータマイニング部で一括処理することで、データマイニング部のメモリアクセス回数を削減する。

フロー集約部の集約アルゴリズムを図 4 に示す。フロー集約部では、固定サイズのキャッシュを用意し、サンプルパケット到着時にフロー (5tuple) 単位の統計情報を更新する。統計情報を格納するエントリは、ハッシュ法を拡張した Hash2 [2] を用い決定する。フロー

```
Algorithm Aggregator
Variable
    Cache[Cache_size]: fixed-size table
begin
    Create empty cache;
    j = 0;
    while (input Transaction)
        increment packet_cnt by 1;
        i = index of items in cache; (calculated by hash2)
        if (cache_flowID[i] != flow ID in Transaction)
            if (cache_cnt[i] > 0)
                report statistics in Cache[i];
            make new entry on Cache[i];
        increment cache_cnt[i] by 1;
        if (cache_cnt[i] ≧ thresh_hold)
            report statistics in Cache[i];
            cache_cnt[i] = 0;
        if (packet_cnt % round_robin == 0)
            if (j > Cache_size);
                j = 0;
            report statistics in Cache[j];
            cache_cnt[j] = 0;
            increment j by 1;
end
```

図 4　フロー集約部のアルゴリズム

集約部で収集したフロー単位の統計情報は、以下３つの条件によりデータマイニング部へフローデータとして出力する。

- 統計情報のパケット数が閾値に到達した場合、その統計情報を出力
- 新規フロー情報格納のためエントリを上書きする場合、既に格納されていた統計情報を出力
- 一定パケット受信ごとに、ラウンドロビンで統計情報を出力

上書きされるエントリを出力することで、フロー集約部によるトラヒック情報の欠落を防ぐ。また、キャッシュに含まれている統計情報をラウンドロビンで出力することで、特定フローの情報がフロー集約部に長期間滞留することを防ぐ。

3.2　データマイニング部

データマイニング部は、複数データ一括更新方式を用いた実装である。複数データ一括更新方式は、更新するフローデータを一定時間蓄積し一つの統計項目に対し蓄積したフローデータ全てを連続的に更新する方法である。複数のフローデータを一括して更新することで、CPU 内のキャッシュを一つの統計項目に関する情報処理で占有できる。従って、キャッシュヒット率が向上し、メモリアクセス速度の向上を図れる。

データマイニング部は、フロー集約部から受信したフローデータを２種類のバッファに蓄積する。バッファはフロー集約部からの受信用バッファと、統計情報の更新用バッファであり、両者を切り替えながら利用する。すなわち一括処理する数まで受信用バッファにフローデータを蓄積した後、受信用バッファを統計情報の更新用バッファとして用いてマイニング処理に用いる。その間、前の処理タイミングで更新用バッファとして用いたバッファを受信用バッファに用いる。

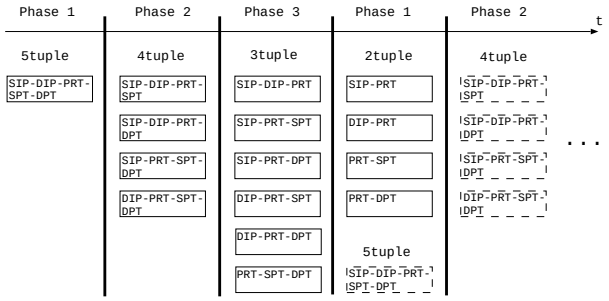

図 5　データマイニング部の統計更新処理

データマイニング部は、以下３つの統計更新フェーズからなり、各フェーズを順番に繰り返す (図 5)。ここで、x tuple とは、5tuple に含まれる 5 項目から、x 個の項目を抽出したものであり、抽出した項目の組合せ全てが一致するトラヒック毎に統計情報を収集する。また、プロトコル単位で解析するため、5tuple のうち PRT は必ず抽出するものとする。

Phase1:
　更新用バッファ１のフローデータに関し 5tuple を更新し、更新用バッファ２のフローデータに関し 2tuple を更新する。統計情報の更新処理が全て完了すると Phase2 へと移行する。

Phase2:
　更新用バッファ１のフローデータに関し 4tuple を更新する。統計情報の更新処理が全て完了すると Phase3 へと移行する。

Phase3:
　更新用バッファ１のフローデータに関し 3tuple を更新する。統計情報の更新処理が全て完了すると Phase1 へと戻り、バッファを切り替える。

各フェーズの処理は処理対象とする tuple 毎にスレッド化されて複数ある CPU に割り付ける。以上により複数 CPU による並列処理と、一括処理によるキャッシュアクセスの局所化が行われ、高速化に寄与する。tuple 毎に一括処理する CPU を割り当てる事でメモ

表 1　トラヒック情報

Trans-Pacific line 2008/03/18	21:00 - 21:15	21:15 - 21:30	21:30 - 21:45	21:45 - 22:00
Number of packets	14.93M	14.03M	15.13M	12,93M
Number of flows	789,676	804,483	924,709	718,532
Avg. packet length	594.5	567.2	595.6	570.5
Avg. packets per flow	18.91	17.44	16.36	17.99
Pct. of SYN packets	4.06%	5.62%	5.52%	5.01%

リアクセスの局所化を行い、メモリアクセスを高速化する事が特徴である。

4. 評 価 実 験

提案方式を適用したプロトタイプに MAWI [10] 公開のトラヒック (表 1) を入力とし、評価実験を行った。評価項目は、処理速度と計測精度の 2 点である。なお、評価は CPU:Intel Core2 Duo 3.0GHz(cache 6144K) × 2、メインメモリ:8G bytes の PC を用い評価した。

4.1 処理速度の評価

二段階集約方式と複数データ一括更新方式の有効性についてそれぞれ評価した。まず、二段階集約方式を評価するため、フロー集約部での情報圧縮率を測定した (図 6,7)。ここで、情報圧縮率を (出力フローデータ数) ／（入力パケット数）とする。

図 6 は、キャッシュ量と閾値を評価した図である。閾値の評価のためラウンドロビンでの出力は OFF とした。フロー集約部に 64k のキャッシュを用意すれば情報圧縮率、すなわちデータマイニング部での更新回数を 1/5 以下に減らせることが分かる。ここで、64k のキャッシュ量は MAWI のトラヒックで約 30 秒間、10G の回線であれば 6~7 ミリ秒間で流れるフロー数に相当する。また、閾値は 40 より大きく設定しても情報圧縮率の向上は薄いことが分かる。これは、フローの平均パケット数が 20 弱であり、閾値に到達するフロー数が少ないためである。

図 7 は、ラウンドロビンによる出力を評価した図である。横軸のラウンドロビン出力頻度は、x パケット受信毎に 1 エントリ出力することを表す。10 パケット受信毎に 1 エントリの出力と頻繁にしても、集約率にほとんど変化はみられない。これは、ラウンドロビンで出力するとメモリ内に空きエントリが増え、結果として無駄な上書きの頻度が減るためである。

次に、複数データ一括更新方式を評価するため、全パケットの処理に要した処理時間を測定した（図 8)。

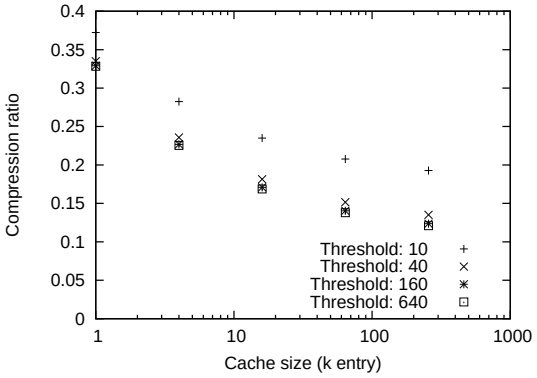

図 6　情報圧縮率の評価 (キャッシュ量)

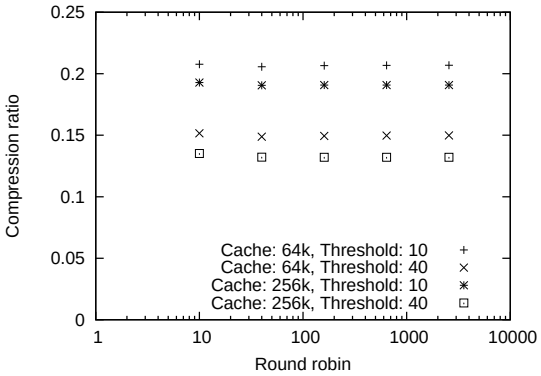

図 7　情報圧縮率の評価 (ラウンドロビン)

図中 B は何パケットを一括処理するかを示している。キャッシュ量を 2^{21} エントリにした時、512 パケットを一括処理する場合では 70 秒以上かかるものが、32768 パケットを一括処理する場合は 40 秒以下になっており、一括処理するフローデータ数を多くするほど、キャッシュヒット率が向上し高速化していることが分かる。

最後に、提案システムの処理性能を図 9 に示す。本実験では、フロー集約部からの出力を HDD に蓄積し、HDD から読込み処理した時の性能を示す。適用前の処理性能は 386kpps であり、1024k キャッシュのフロー集約部で約 3.8Mpps と 10 倍の処理速度向上となる。トラヒックの平均パケット長を用い計算すると、18Gbps の通信帯域 (bps) までフルパケットでモニタリングできることが分かった。以上の結果から 10Gbps 回線に充分適用可能であると考える。

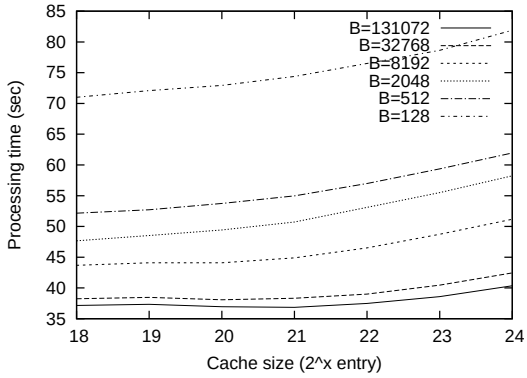

図 8　一括処理数と処理時間

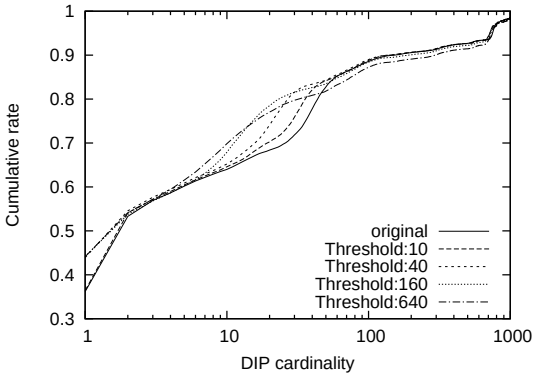

図 10　異なり数の分布 (閾値)

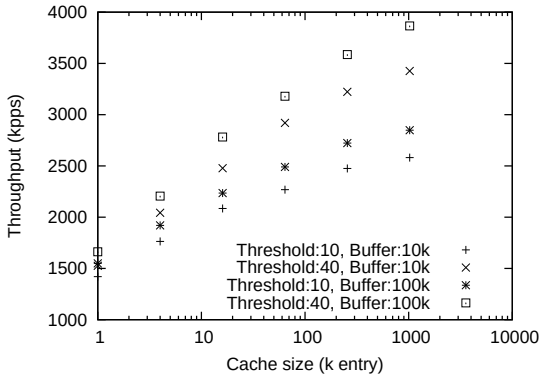

図 9　処 理 性 能

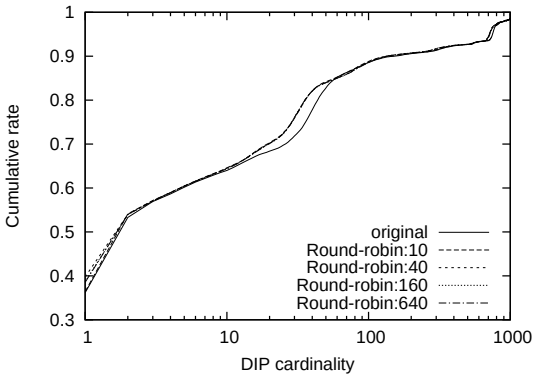

図 11　異なり数の分布 (ラウンドロビン)

4.2　計測精度の評価

フロー集約部でトラヒック情報を集約すると、データマイニング部で統計情報を更新する回数、時刻、順序が変化する。更新回数と順序の変化により、計測期間中に表れた種類数である異なり数情報に影響を及ぼす。そこで、提案方式適用前の異なり数分布と比較することで、提案方式が異なり数情報に与える影響を評価した (図 10,11,12)。異なり数の分布を示すため、縦軸はデータマイニング部から出力された統計情報の積算比率とした。また、データマイニング部ではパケット数が 1000 以上となる統計情報を出力した。

図 10,11 より、閾値は異なり数の分布に最も影響を与える要素であることが分かる。異なり数情報からホストの挙動を詳細にモニタリングする場合、閾値はフローの平均パケット数より小さい値への設定が必要となる。また、処理速度優先で閾値 40 とした場合でも、粗い解析をするには充分の精度と考えられる。

図 11 は、ラウンドロビン出力頻度の異なり数への影響を評価したものであり、キャッシュ量 64k, 閾値 10 の結果である。ラウンドロビン出力頻度は、異なり数の分布にはほとんど影響しないことが分かる。ラウンドロビン出力が減ると上書きによる出力が増加するため、フロー集約部に滞在する時間は変わらず、異なり数への影響も少ない。

図 12 は、閾値 10、ラウンドロビン 10 とし、キャッシュ量の影響を評価した図である。キャッシュ量が増えるにつれ、異なり数はやや小さくカウントされる傾向がある。これは、キャッシュ量が増えると、上書きによる出力とラウンドロビンによる出力が減り、滞在時間が延びるためである。

以上の結果から、閾値を小さく設定すると計測精度にはほとんど影響しないことが分かった。また、図 9,

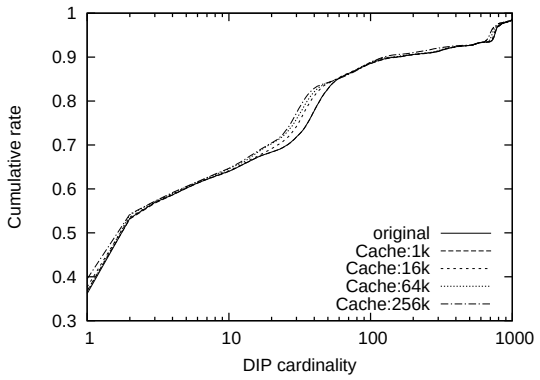

図 12　異なり数の分布 (キャッシュ量)

図 10 から速度と計測精度はトレードオフの関係にあり、重要となるパラメータは閾値である。閾値の適切な値は、流れるトラヒックの種類によって変化する。モニタリングする地点で最適な閾値の設定方法は今後の課題である。

5. まとめと今後の課題

本稿では、トラヒックモニタリングを高速回線へ適用するため、ボトルネックとなるメモリアクセスを高速化する二段階集約方式と複数データ一括更新方式を提案した。試作評価した結果 10Gbps 回線のモニタリングを実現する目処を得た。

本稿の内容は、フロー集約部とデータマイニング部をソフトウェアで実装し、それぞれを評価したものである。フロー集約部をハードウェアで実装し、システム全体としての性能を実トラヒックで評価することが今後の課題である。

謝辞　本稿は，独立行政法人新エネルギー・産業技術総合開発機構の委託事業「次世代高効率ネットワークデバイス技術の開発事業」の研究成果を含む。

文　献

[1] Y. Shomura, Y. Watanabe, and K. Yoshida, “Analyzing the number of variety in frequent found flows,” Transaction of the Institute of Electronics, Information and Communication Engineers, vol.E91-B, no.6, pp.1896–1905, 2008.

[2] K. Yoshida, S. Katsuno, S. Ano, K. Yamazaki, and M. Tsuru, “Stream mining for network management,” Transaction of the Institute of Electronics, Information and Communication Engineers, vol.E89-B, no.6, pp.1774–1780, 2006.

[3] E.D. Demaine, A. Lopez-Ortiz, and J.I. Munro, “Frequency estimation of internet packet streams with limited space.,” Proc. of the 10th Annual European Symposium on Algorithms, 2002.

[4] A. Metwally, D. Agrawal, and A.E. Abbadi, “Efficient computation of frequent and top-k elements in data streams.,” ICDT, pp.398–412, 2005.

[5] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules,” Proc. 20th Int. Conf. Very Large Data Bases, VLDB, ed. J.B. Bocca, M. Jarke, and C. Zaniolo, pp.487–499, Morgan Kaufmann, 12–15 1994.

[6] T. Mori, R. Kawahara, N. Kamiyama, K. Ishibashi, and T. Abe, “Detection of worm-infected hosts by communication pattern analysis,” Technical Report of the Institute of Electronics, Information and Communication Engineers, pp.1–6, 2005 (in Japanese).

[7] K. Keys and C. Estan, “A robust system for accurate real-time summaries of internet traffic,” SIGMETRICS Perform. Eval. Rev, pp.85–96, ACM, 2005.

[8] A. Lakhina, M. Crovella, and C. Diot, “Mining anomalies using traffic feature distributions,” SIGCOMM Comput. Commun. Rev., vol.35, no.4, pp.217–228, 2005.

[9] Q. Zhao, A. Kumar, and J. Xu, “Joint data streaming and sampling techniques for detection of super sources and destinations,” Proc. ACM SIGCOMM Internet Measurement Conference, pp.77–90, 2005.

[10] Mawi WG, “http://www.wide.ad.jp/wg/mawi/,” 2007.

付　録

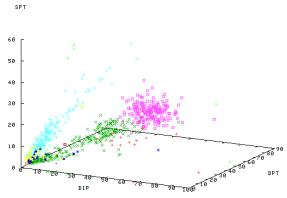

図 A·1　ホストの振る舞い分析 2

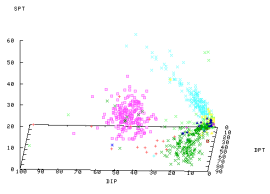

図 A·2　ホストの振る舞い分析 3