IIJ Technical WEEK 2012

フロー集約によるトラフィック計測

長　健二朗
IIJイノベーションインスティテュート　技術研究所

加藤碧、本多倫夫、徳田英幸（慶応義塾大学）

2012-11-15

IIJ
Internet Initiative Japan

コンパクトなトラフィック情報

- トラフィックの監視の必要性
  - 利用状況の把握
  - 異常検出 （トラフィック集中、攻撃、設定ミスなど）
- 簡潔なサマリ情報の必要性
  - 問題発見に十分で、かつ、見落としが起きない情報量

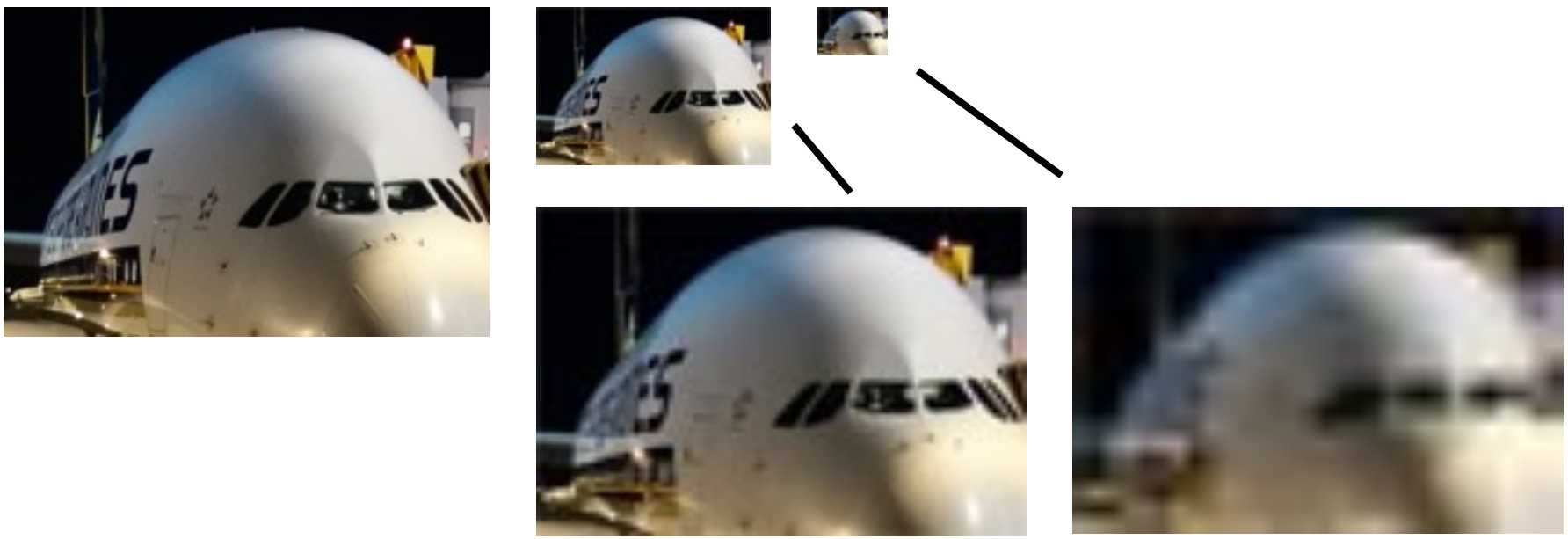
フロー集約

- フロー：5-tuple (src_IP, dst_IP, src_port, dst_port, protocol)
  - 単一のTCPセッションなど
- フローを集約することで、重要なパターンを抽出
- 集約フロー：共通属性で集約されたフロー
  - 例：TCPセッション from 10.0.0.0/29:80 to 10.1.0.0/24:*
  - 例：ICMPトラフィック全体

Flow	10.0.0.7:80 - 10.1.0.2:3003 TCP
Aggregated Flow (src/dst address)	10.0.0.0/29:80 – 10.1.0.0/24:3003 TCP
Aggregated Flow (+ dst port)	10.0.0.0/29:80 – 10.1.0.0/24:* TCP

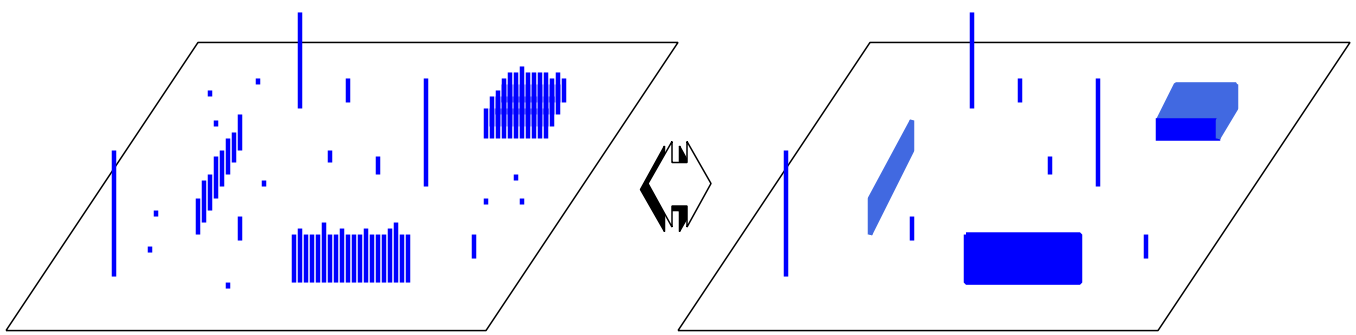
情報理論から見たフロー集約

- データ圧縮：画像圧縮との類似点
  - 高解像度 (情報量大) ↔ 低解像度 (情報量小)
  - 情報量（エントロピー）の符号化



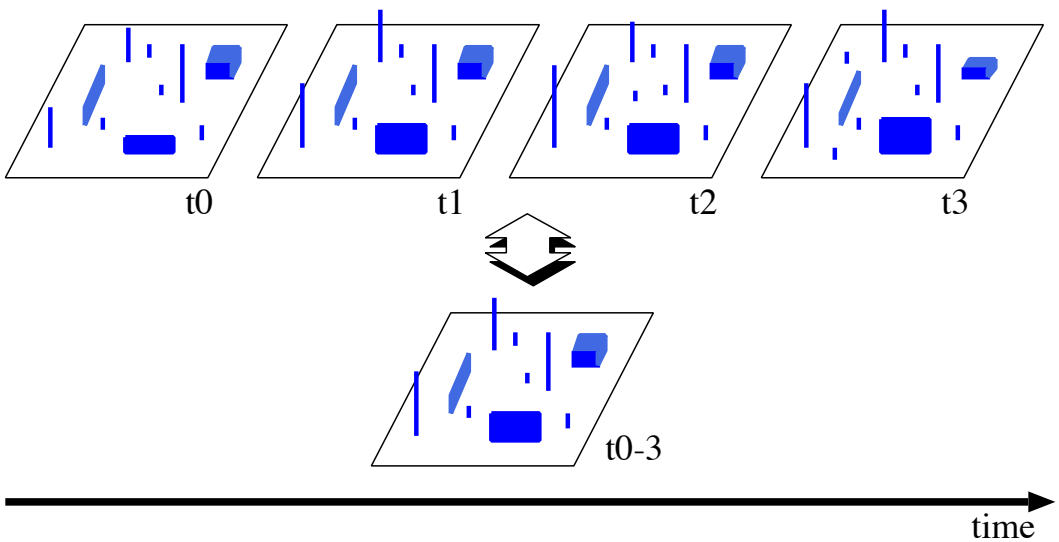
空間的フロー集約

- ２次元の例 (src_IP, dst_IP) （実際には５次元）
  - 値：トラフィック量（パケット量）
- 画像圧縮との違い
  - 空間に対して疎ら（空間的可視化は難しい）
  - 点の集合：限られた集約通信パターン
    - １対多：縦線分、多対１：横線分、多対多：矩形

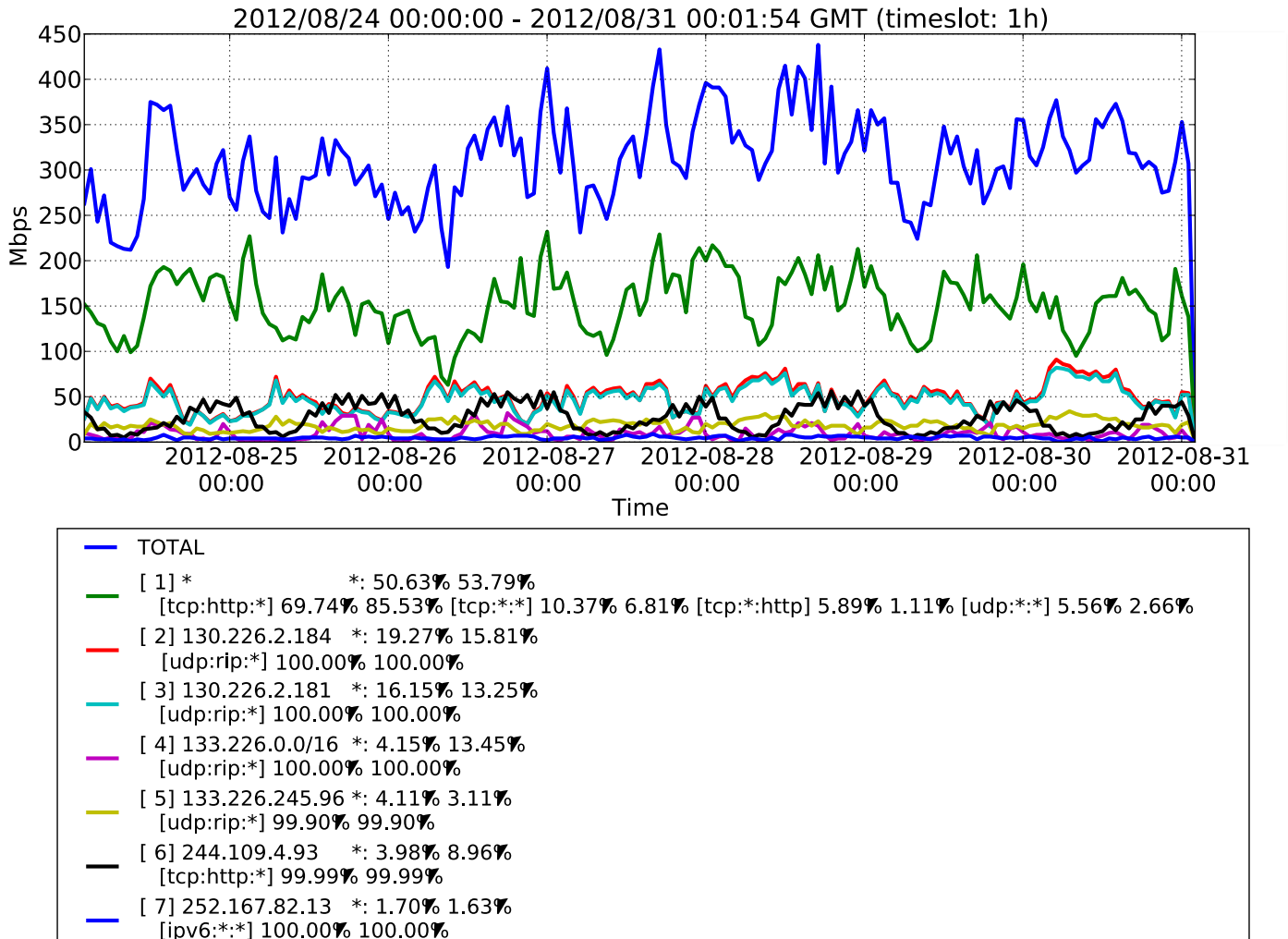


時間方向への集約

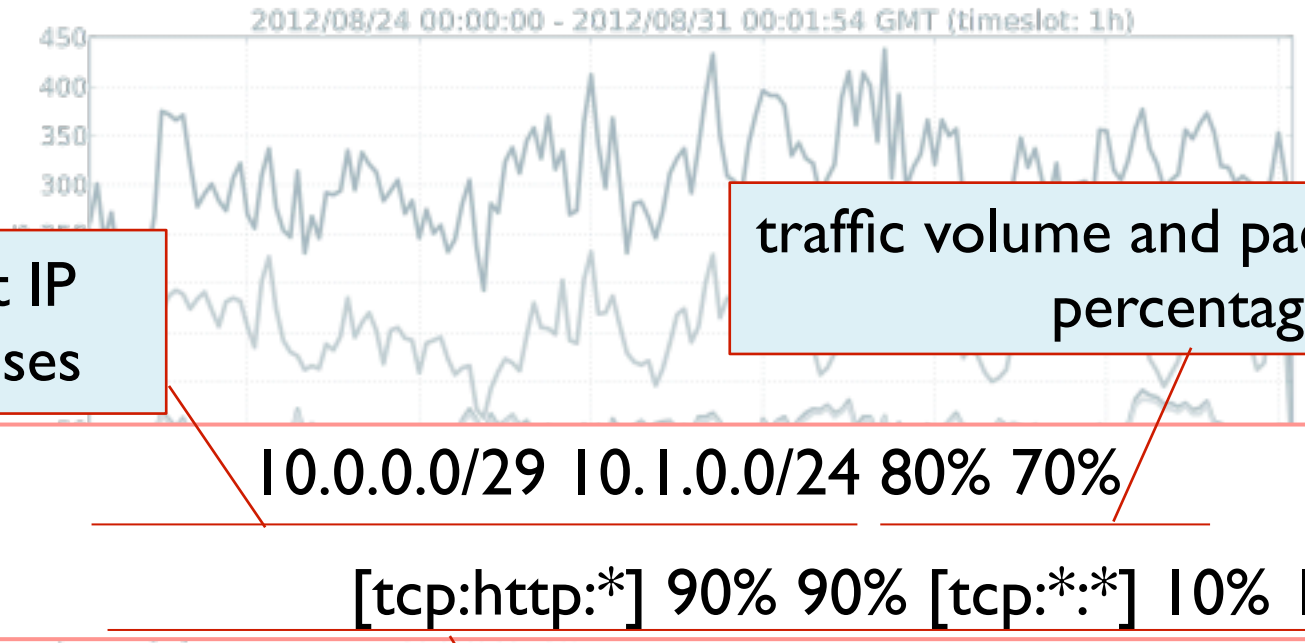
- 同じ考え方で時間方向の粒度を変えることが可能



集約フローの可視化のイメージ



集約フローの表現



src/dst IP addresses

traffic volume and packet counter percentage

10.0.0.0/29 10.1.0.0/24 80% 70%

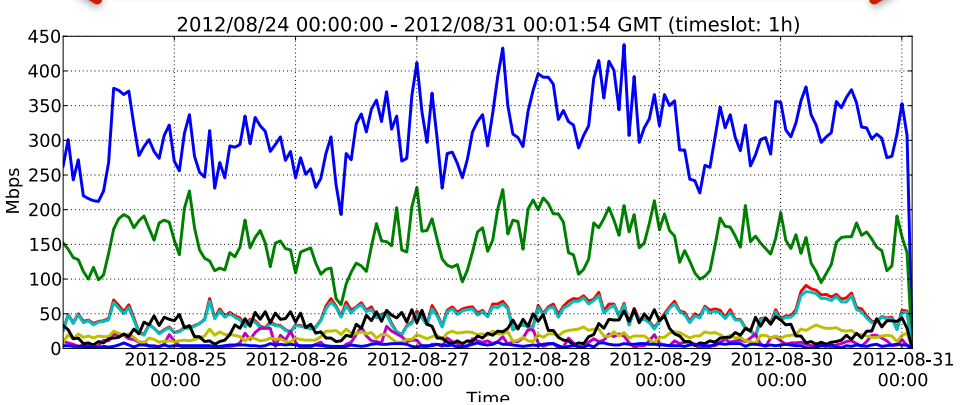
[tcp:http:*] 90% 90% [tcp:*:*] 10% 10%

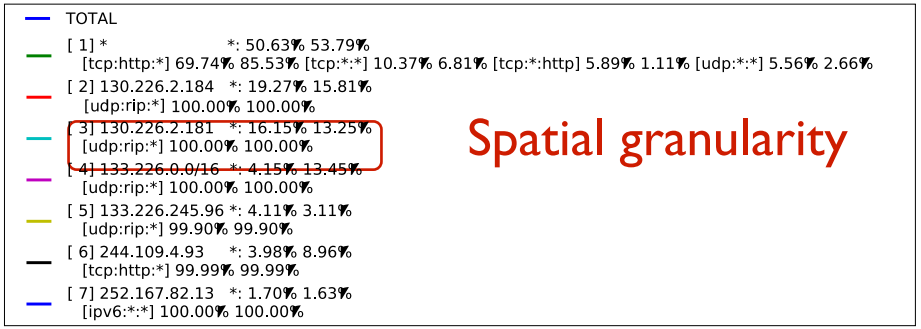
decomposition of protocol and src/dst port numbers within IP address pair

集約粒度変更

- 既存の集約フローの可視化ツールでは粒度変更は面倒
  - 処理速度と柔軟性のトレードオフ
- 自由に粒度変更可能なツールが欲しい
  - 時間方向：表示期間と時間粒度
  - 空間方向：フロー集約の粒度

Temporal granularity

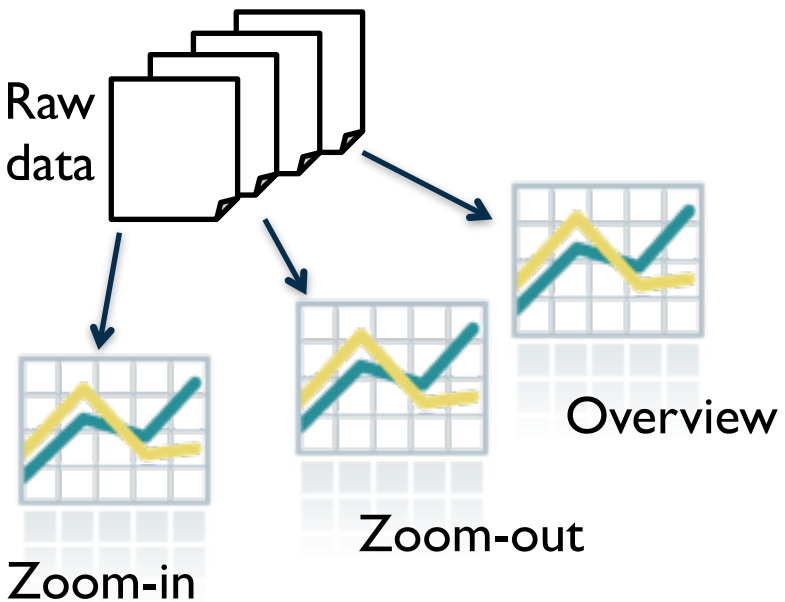




Spatial granularity

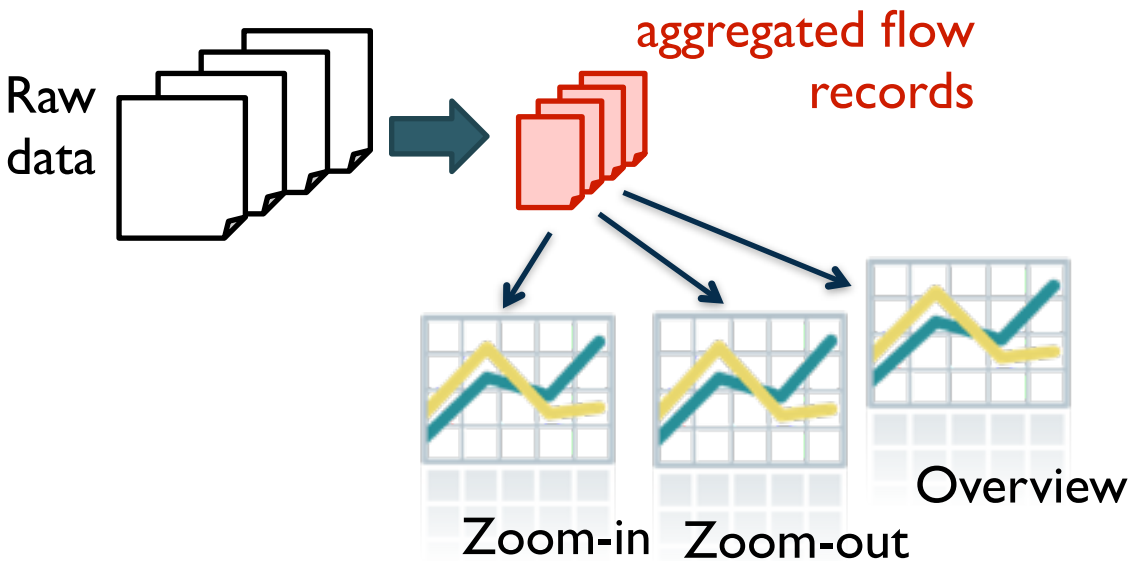
技術的チャレンジ

- 広大な多次元空間でフローを対話的にクラスタリング
  - 数百万フローから10個程度の集約フローを作成
- ビュー（粒度）を変更する際のオーバヘッドの削減



Agurim

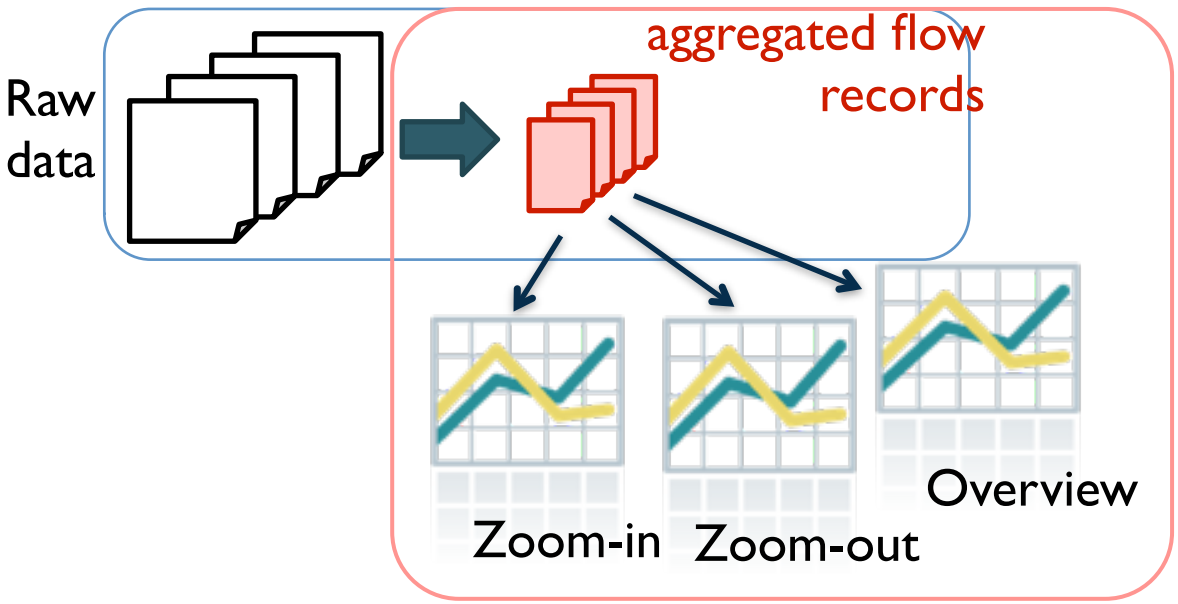
- 柔軟で効率良い多次元フロー集約近似アルゴリズムの提案
  - 対話型の視覚化の実現
  - 効率：生データから再利用可能な細粒度集約フロー生成
  - 柔軟性：再粒度集約フローを再集約して、ビューに必要な集約フローを生成



Agurimの概要：２段階フロー集約

一次集約：効率重視のフロー集約

再利用可能な細粒度集約フロー情報の生成



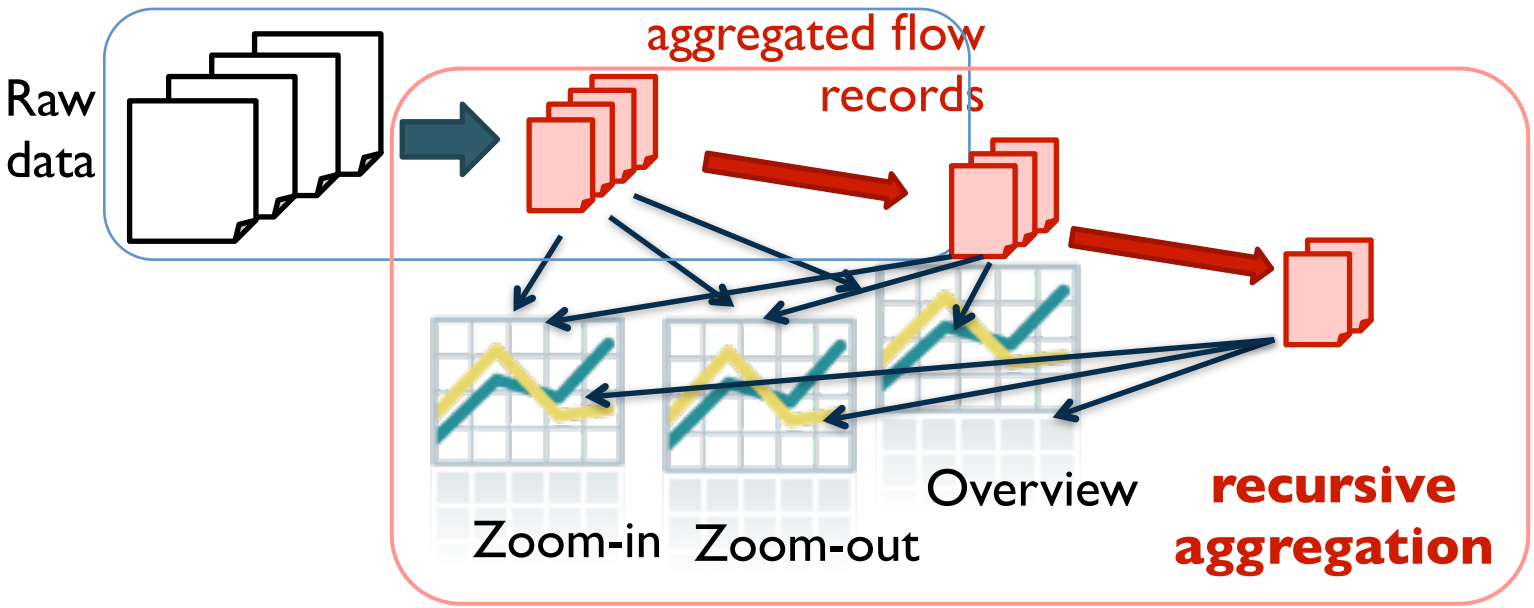
二次集約：表示用の柔軟性重視のフロー集約

プロットに必要な時間空間粒度で集約フロー情報を再集約

Agurimの概要：多段階フロー集約

一次集約：効率重視のフロー集約

再利用可能な細粒度集約フロー情報の生成



二次集約：表示用の柔軟性重視のフロー集約

プロットに必要な時間空間粒度で集約フロー情報を再集約

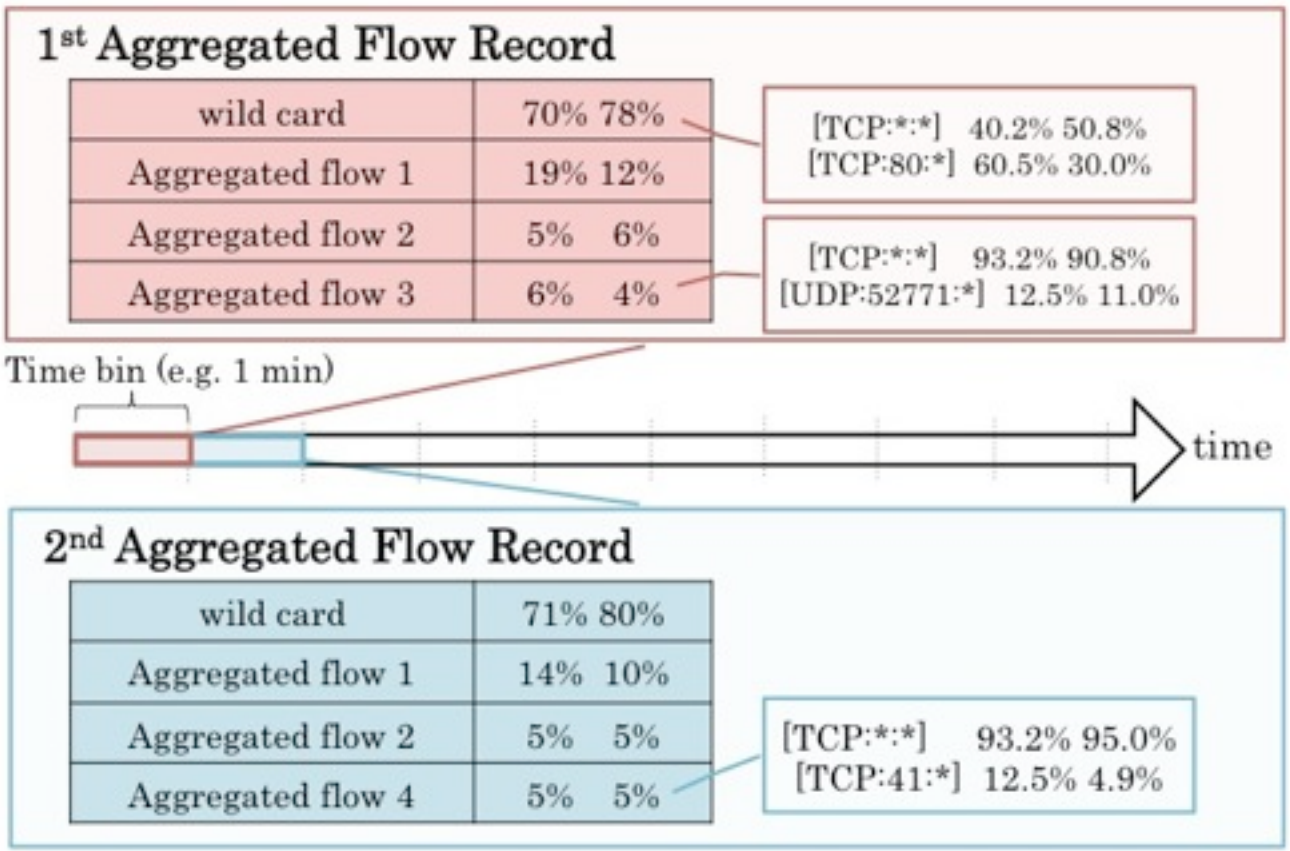
再帰的集約：再集約結果をさらに集約

関連研究

	Monitoring tool	Multi-dimensional Flow Aggregation?	Recursive Aggregation?
1	Aguri	no	yes
2	AutoFocus	yes	no
3	ProgME	yes	no
4	HHH	yes	no
5	Multi-dimensional HHH	yes	no
	Agurim	**yes**	**yes**

[1] K. Cho, R. Kaizaki, and A. Kato. "Aguri: An aggregation-based traffic profiler" In Quality of Future Internet Services, 2001.
[2] C. Estan, S. Savage, and G. Varghese. "Automatically inferring patterns of resource consumption in network traffic" In ACM SIGCOMM 2003.
[3] L. Yuan, C.-N. Chuah, and P. Mohapatra. "Progme: towards programmable network measurement" In ACM SIGCOMM,2007
[4] Y. Zhang, S. Singh, S. Sen, N. Duffield, and C. Lund. "Online identification of hierarchical heavy hitters: algorithms, evaluation, and applications" In ACM IMC 2004
[5] G. Cormode, F. Korn, S. Muthukrishnan, and D. Srivastava. "Diamond in the rough: finding hierarchical heavy hitters in multidimensional data". In ACM SIGMOD, 2004

一次集約の概要



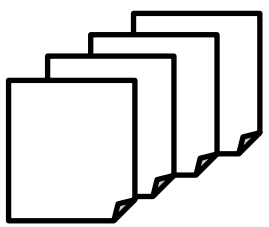
- フロー属性ごとにツリー構造で集約
- 時間スロットごとに集約フローを生成

一次集約の概略：2パス アルゴリズム

- 5個の属性空間を独立に集約
- 集約された属性を使ってパケットを集約フローに分類
- 1パス実装：直前のタイムスロットの集約属性の使用

[1st-pass]
Aggregate each attribute separately

[2nd-pass]
Match each packet against aggregated attributes



Raw data

Aggregated attributes for src address

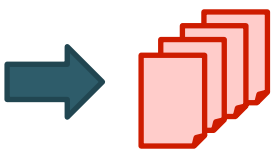
$S_1, S_2, \ldots, S_m$

Aggregated attributes for dst address

$D_1, D_2, \ldots, D_n$

dst\src	S_0	S_1	S_2		S_m
D_0	300	12	77	⋯	⋯
D_1	28	0	14		
D_2	59	0	0		
⋮				⋱	
D_n					⋱

(unit: KB)

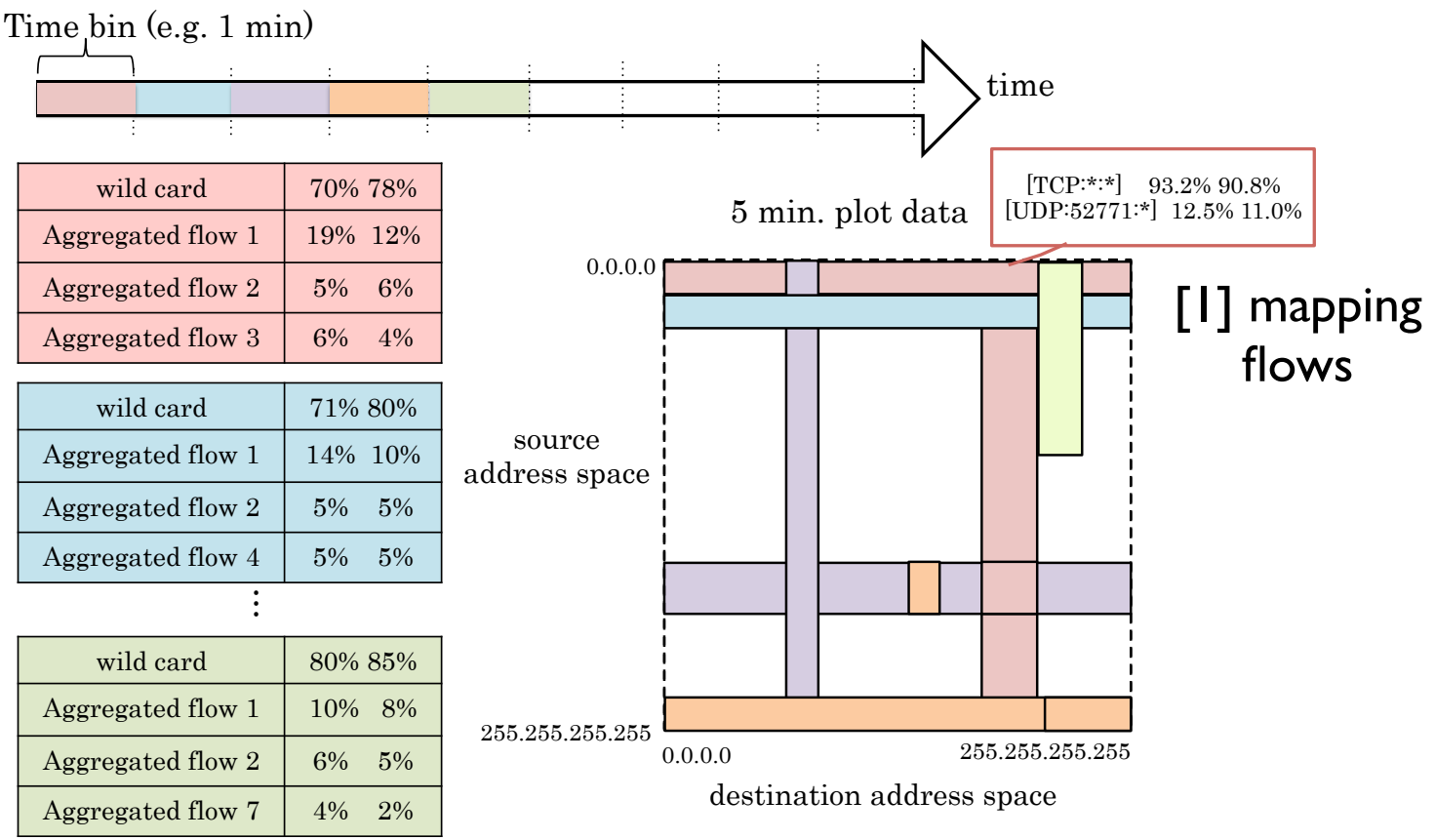


aggregated flow records

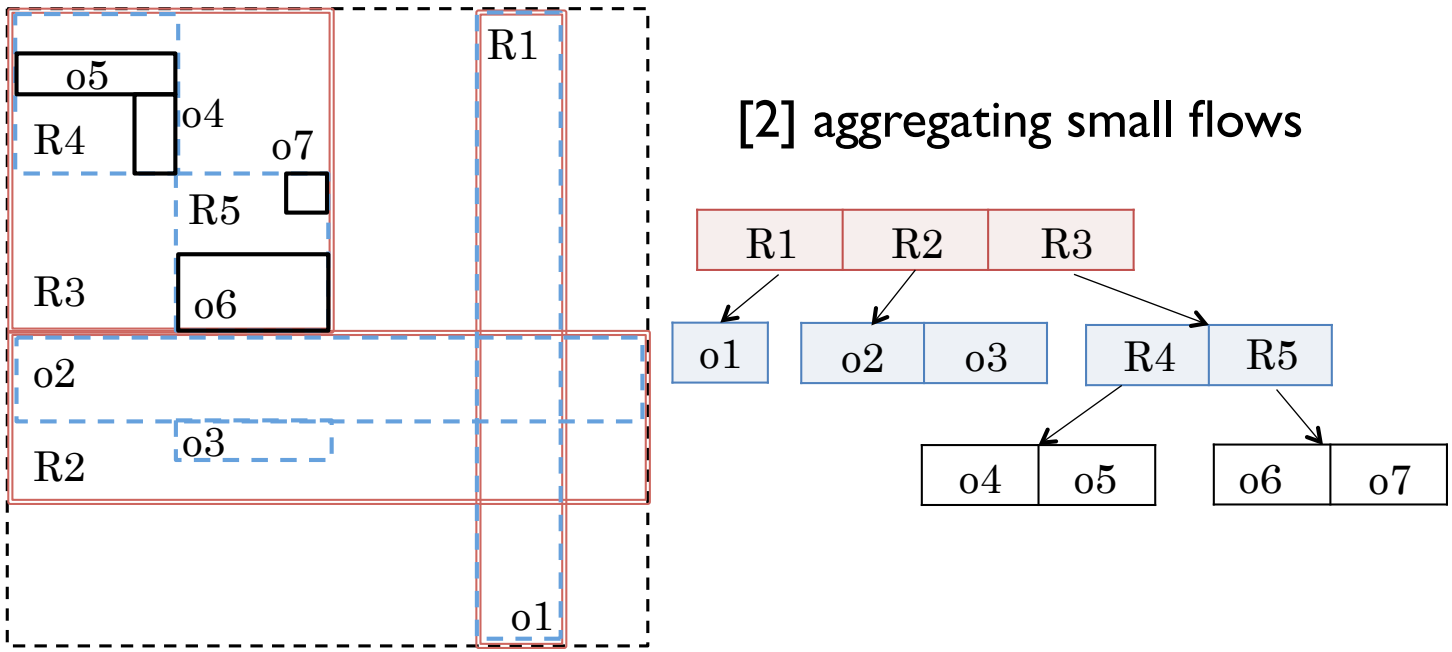
フロー空間を1/10000程度に圧縮

二次集約の概要

- 指定された期間の集約フローレコードを抽出
- 量的に小さい集約フローをスーパーセットとなるフローに集約
- 多次元データ集約にR-tree構造を利用

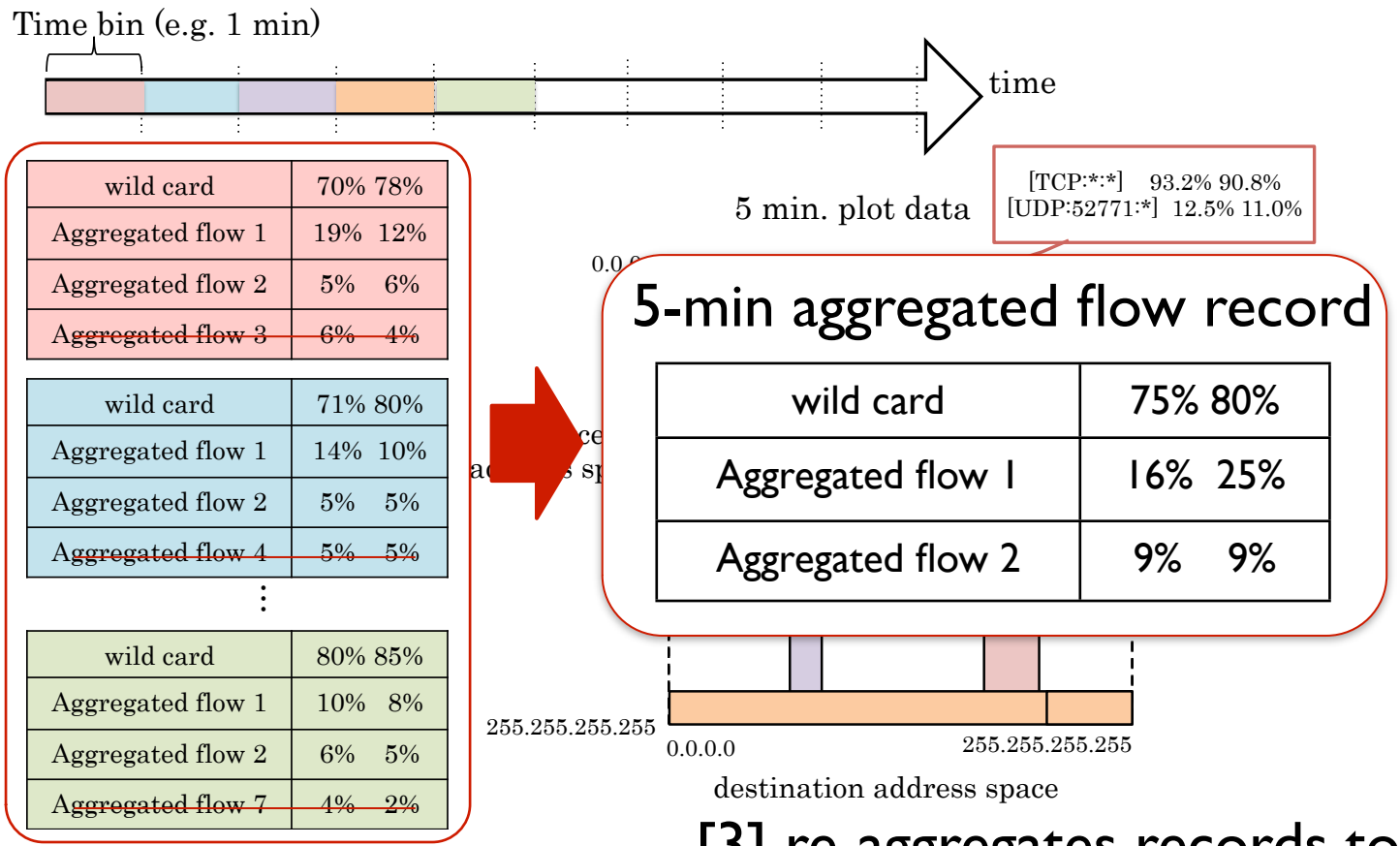


二次集約の概要：R-tree



- R-tree：多次元空間の領域表現のデータ構造
  - 親ノードの領域はすべての子ノード領域を包含
  - 最小包囲矩形（MBR）を見つけ、そこに集約
- 領域クエリのためのデータ構造を、包含関係の管理に応用

二次集約の概要：再帰的集約

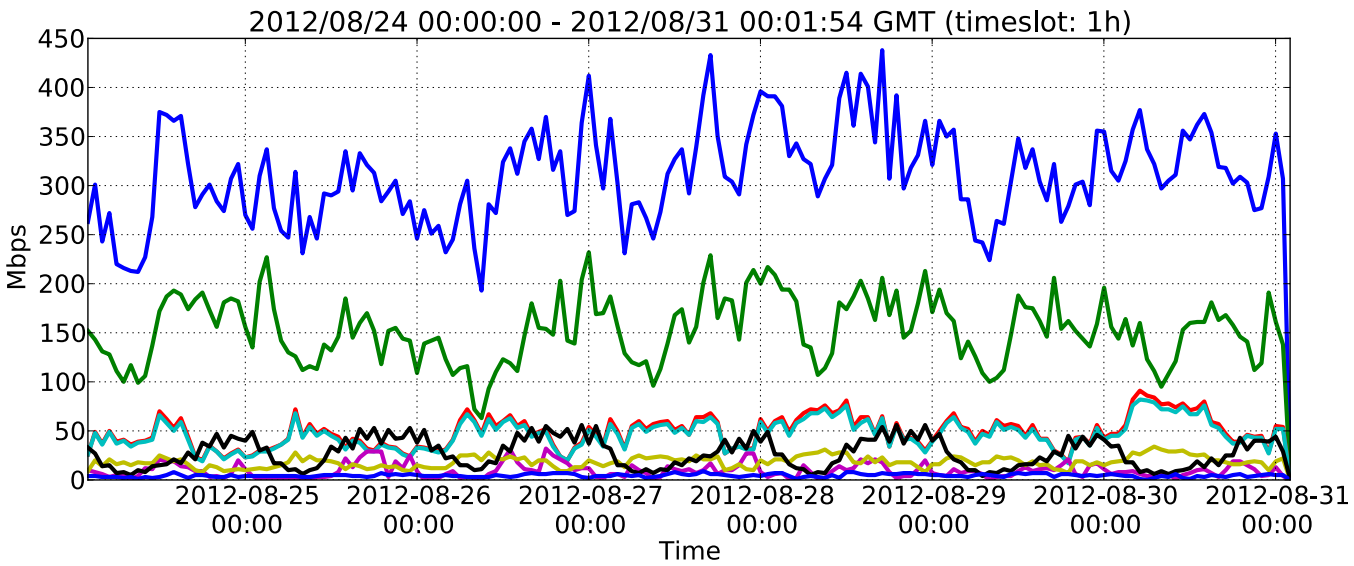


5 x 1min aggregated flow records

[3] re-aggregates records to reduce computation overhead

- Agurim aggregates short and significant flows by flow re-aggregation

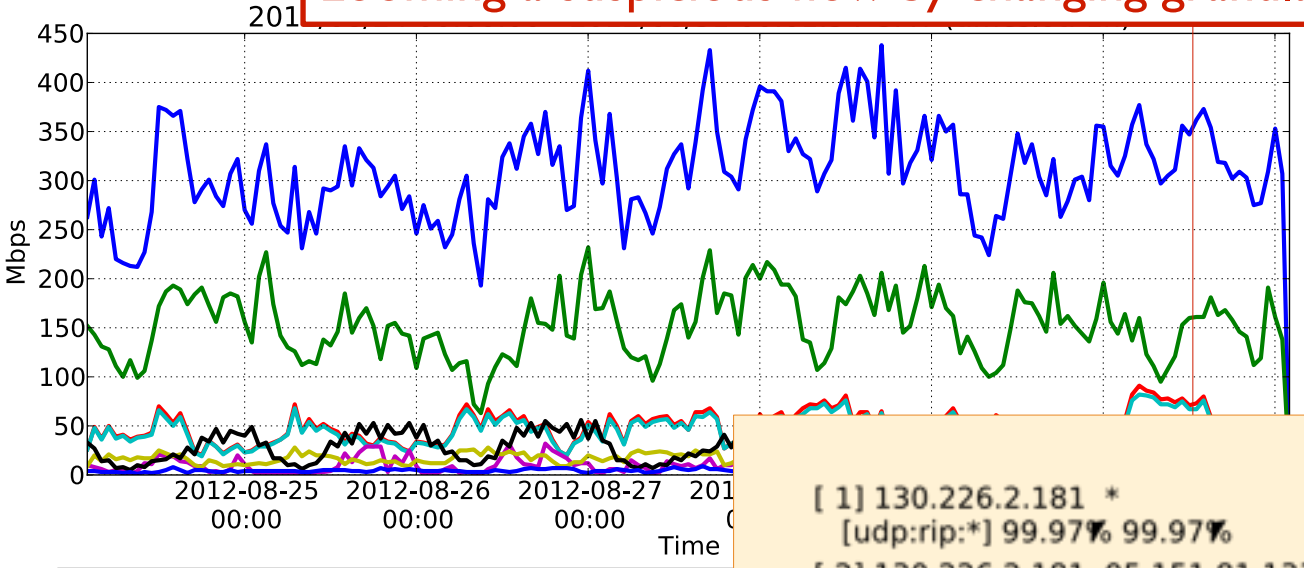
Agurim対話型グラフ



- Significant UDP traffic using RIP port
- From several srcs to many dsts

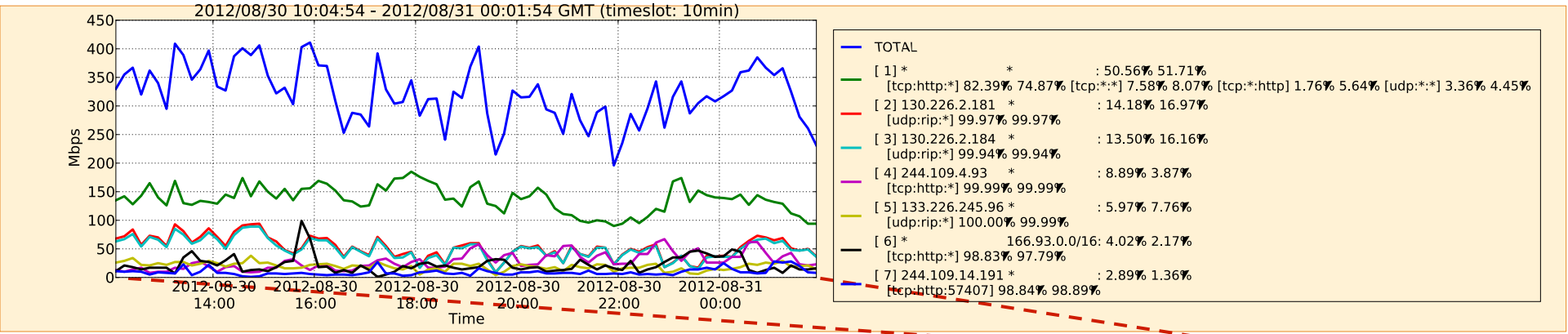
Agurim対話型グラフ

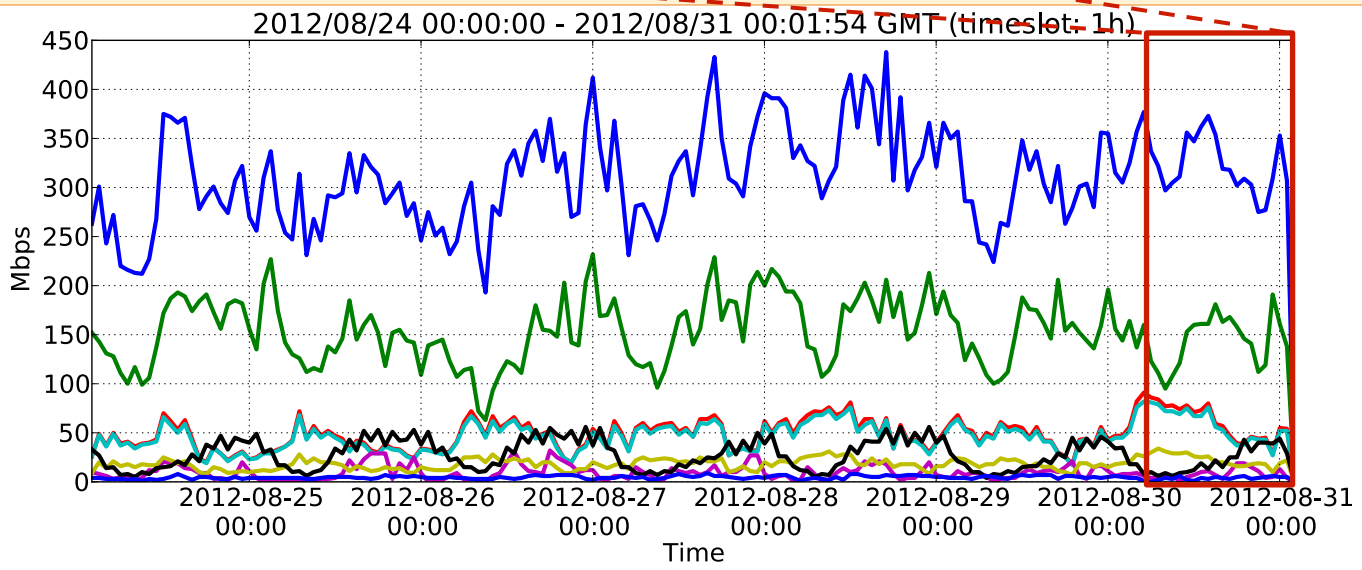
zooming a suspicious flow by changing granularity

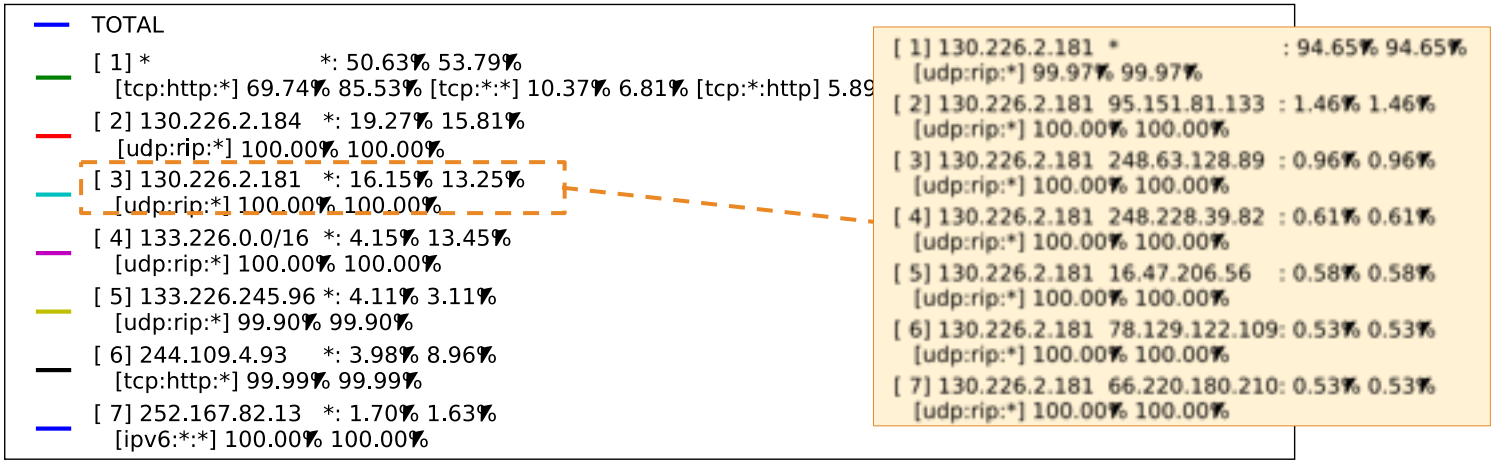


TOTAL
[1] * *: 50.63% 53.79%
[tcp:http:*] 69.74% 85.53% [tcp:*:*] 10.37% 6.81%
[2] 130.226.2.184 *: 19.27% 15.81%
[udp:rip:*] 100.00% 100.00%
[3] 130.226.2.181 *: 16.15% 13.25%
[udp:rip:*] 100.00% 100.00%
[4] 133.226.0.0/16 *: 4.15% 13.45%
[udp:rip:*] 100.00% 100.00%
[5] 133.226.245.96 *: 4.11% 3.11%
[udp:rip:*] 99.90% 99.90%
[6] 244.109.4.93 *: 3.98% 8.96%
[tcp:http:*] 99.99% 99.99%
[7] 252.167.82.13 *: 1.70% 1.63%
[ipv6:*:*] 100.00% 100.00%

[1] 130.226.2.181 * : 94.65% 94.65%
[udp:rip:*] 99.97% 99.97%
[2] 130.226.2.181 95.151.81.133 : 1.46% 1.46%
[udp:rip:*] 100.00% 100.00%
[3] 130.226.2.181 248.63.128.89 : 0.96% 0.96%
[udp:rip:*] 100.00% 100.00%
[4] 130.226.2.181 248.228.39.82 : 0.61% 0.61%
[udp:rip:*] 100.00% 100.00%
[5] 130.226.2.181 16.47.206.56 : 0.58% 0.58%
[udp:rip:*] 100.00% 100.00%
[6] 130.226.2.181 78.129.122.109: 0.53% 0.53%
[udp:rip:*] 100.00% 100.00%
[7] 130.226.2.181 66.220.180.210: 0.53% 0.53%
[udp:rip:*] 100.00% 100.00%


2012/08/30 10:04:54 - 2012/08/31 00:01:54 GMT (timeslot: 10min)
Mbps
Time
TOTAL
[1] * * : 50.56% 51.71%
[tcp:http:*] 82.39% 74.87% [tcp:*:*] 7.58% 8.07% [tcp:*:http] 1.76% 5.64% [udp:*:*] 3.36% 4.45%
[2] 130.226.2.181 * : 14.18% 16.97%
[udp:rip:*] 99.97% 99.97%
[3] 130.226.2.184 * : 13.50% 16.16%
[udp:rip:*] 99.94% 99.94%
[4] 244.109.4.93 * : 8.89% 3.87%
[tcp:http:*] 99.99% 99.99%
[5] 133.226.245.96 * : 5.97% 7.76%
[udp:rip:*] 100.00% 99.99%
[6] * 166.93.0.0/16: 4.02% 2.17%
[tcp:http:*] 98.83% 97.79%
[7] 244.109.14.191 * : 2.89% 1.36%
[tcp:http:57407] 98.84% 98.89%

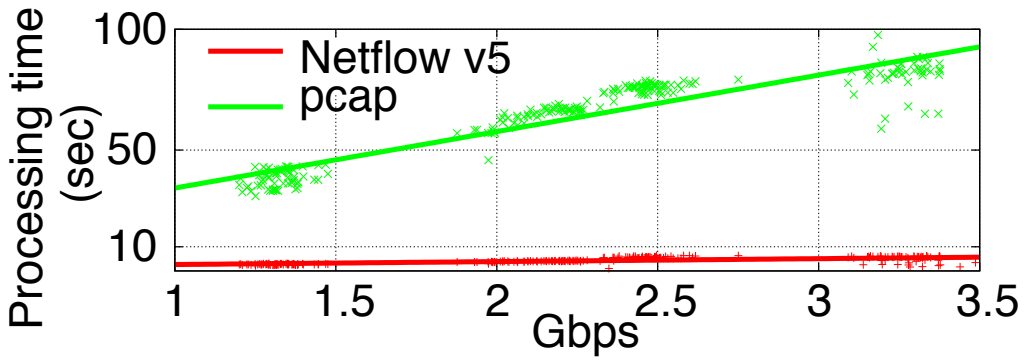

2012/08/24 00:00:00 - 2012/08/31 00:01:54 GMT (timeslot: 1h)
Mbps
Time


TOTAL
[1] * *: 50.63% 53.79%
[tcp:http:*] 69.74% 85.53% [tcp:*:*] 10.37% 6.81% [tcp:*:http] 5.89
[2] 130.226.2.184 *: 19.27% 15.81%
[udp:rip:*] 100.00% 100.00%
[3] 130.226.2.181 *: 16.15% 13.25%
[udp:rip:*] 100.00% 100.00%
[4] 133.226.0.0/16 *: 4.15% 13.45%
[udp:rip:*] 100.00% 100.00%
[5] 133.226.245.96 *: 4.11% 3.11%
[udp:rip:*] 99.90% 99.90%
[6] 244.109.4.93 *: 3.98% 8.96%
[tcp:http:*] 99.99% 99.99%
[7] 252.167.82.13 *: 1.70% 1.63%
[ipv6:*:*] 100.00% 100.00%
[1] 130.226.2.181 * : 94.65% 94.65%
[udp:rip:*] 99.97% 99.97%
[2] 130.226.2.181 95.151.81.133 : 1.46% 1.46%
[udp:rip:*] 100.00% 100.00%
[3] 130.226.2.181 248.63.128.89 : 0.96% 0.96%
[udp:rip:*] 100.00% 100.00%
[4] 130.226.2.181 248.228.39.82 : 0.61% 0.61%
[udp:rip:*] 100.00% 100.00%
[5] 130.226.2.181 16.47.206.56 : 0.58% 0.58%
[udp:rip:*] 100.00% 100.00%
[6] 130.226.2.181 78.129.122.109: 0.53% 0.53%
[udp:rip:*] 100.00% 100.00%
[7] 130.226.2.181 66.220.180.210: 0.53% 0.53%
[udp:rip:*] 100.00% 100.00%

一次集約の性能評価

- pcap(tcpdump) と Netflowデータによる処理時間

Dataset: 4-hour trace data collected on 10Gbps link from Tier 1 ISP to CAIDA
timebin: 1 min.
CPU: Intel Core i5 @2.5GHz



- Netflow: 3Gbps 60秒間のデータ処理は約10秒
  - すでにある程度フロー集約がされているため高速
- Pcap: 1Gbps 60秒間のデータ処理は約40秒
  - パケットごとに属性ツリーを検索する処理のオーバヘッドが大

二次集約の性能評価

- プロットデータを生成するのにかかる時間を計測

Dataset: 7-day-long aggregated flow records collected on 150Mbps transit link of WIDE backbone
Time bin of records: 1 min
Time resolution of plot: 1 hour

time period in the entire view	observed unique aggr. flows	processing time
12-hour	2,178	0.44 sec
1-day	3,796	1.35 sec
3-day	9,858	13.46 sec
1-week	23,065	75.77 sec

- 処理時間は集約フロー数に対して指数関数的に増加
  - 問題は、指定フロー数に到達するまで、最小フローを見つけては集約するループ制御
  - 再集約処理の最適化を検討中
    - 小さいフローは１パスでまとめて集約
    - １時間、１日という粗粒度のデータをあらかじめ用意して利用

まとめ

- フロー集約：フローデータの圧縮技術という側面
- Agurim：フロー集約の現実的な近似アルゴリズム
  - 多次元フロー集約により柔軟な集約粒度変更を実現
  - プロトタイプの性能：1日分のプロットは1.5秒で描画
- 今後のとりくみ
  - 実装のオープンソース化
  - さらなる性能改善
  - 集約精度の評価

Midori Kato, Kenjiro Cho, Michio Honda, Hideyuki Yokuda. “Monitoring the Dynamics of Network Traffic by Recursive Multi-dimensional Aggregation”. OSDI2012 MAD Workshop. Hollywood, CA. October 2012.