



A Study on High-Speed and Scalable Packet Scheduling Algorithm for Achieving Fair Service

宮原研究室
博士前期課程 2年
牧 一之進



発表内容

- 研究の背景
- 研究の目的
- 高速かつスケラブルなパケット処理方式
(Hierarchically Aggregated Fair Queuing)
の提案
- シミュレーションによる性能評価
- まとめと今後の課題
- 今後の研究の進め方



研究の背景

- Best effort 環境においては、特定のユーザのみが帯域を占有してしまう可能性がある
- TCPフロー制御では、異なる環境（RTT、回線速度、TCPバージョン）では公平に動作しない、UDPとの共存が難しい

公平なインターネットの構築

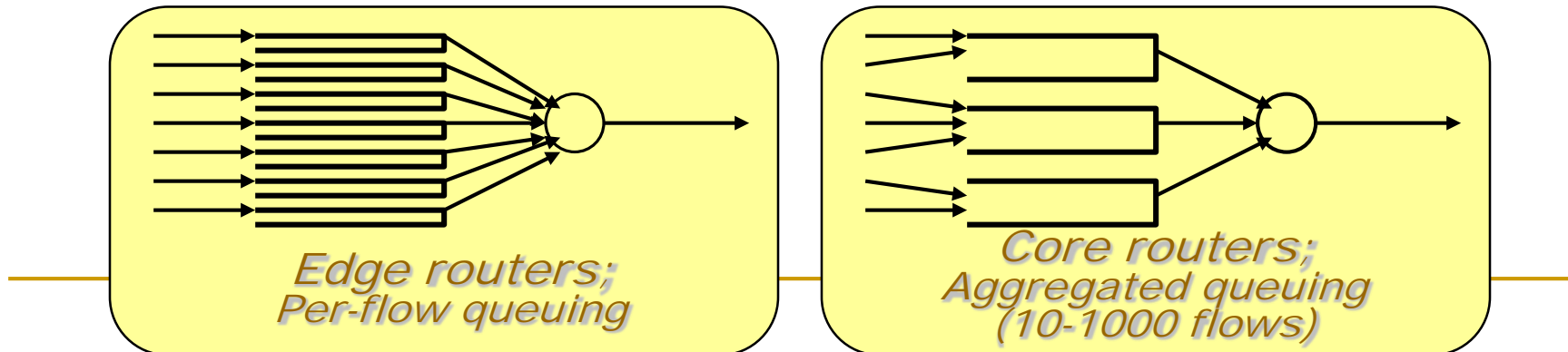
各ユーザに公平なサービスを提供

従来の研究

- フローごとの制御を行わない方式 (RED, SRED)
 - 実装は容易であるが、完全には公平なサービスを実現できない
- フローごとの制御を行う方式 (DRR, FRED)
 - 高い公平性を実現できるが、高速なバックボーンルータへの適用が困難である

エッジルータからコアルータまでスケラブルに
実装可能なパケットスケジューリング方式が必要

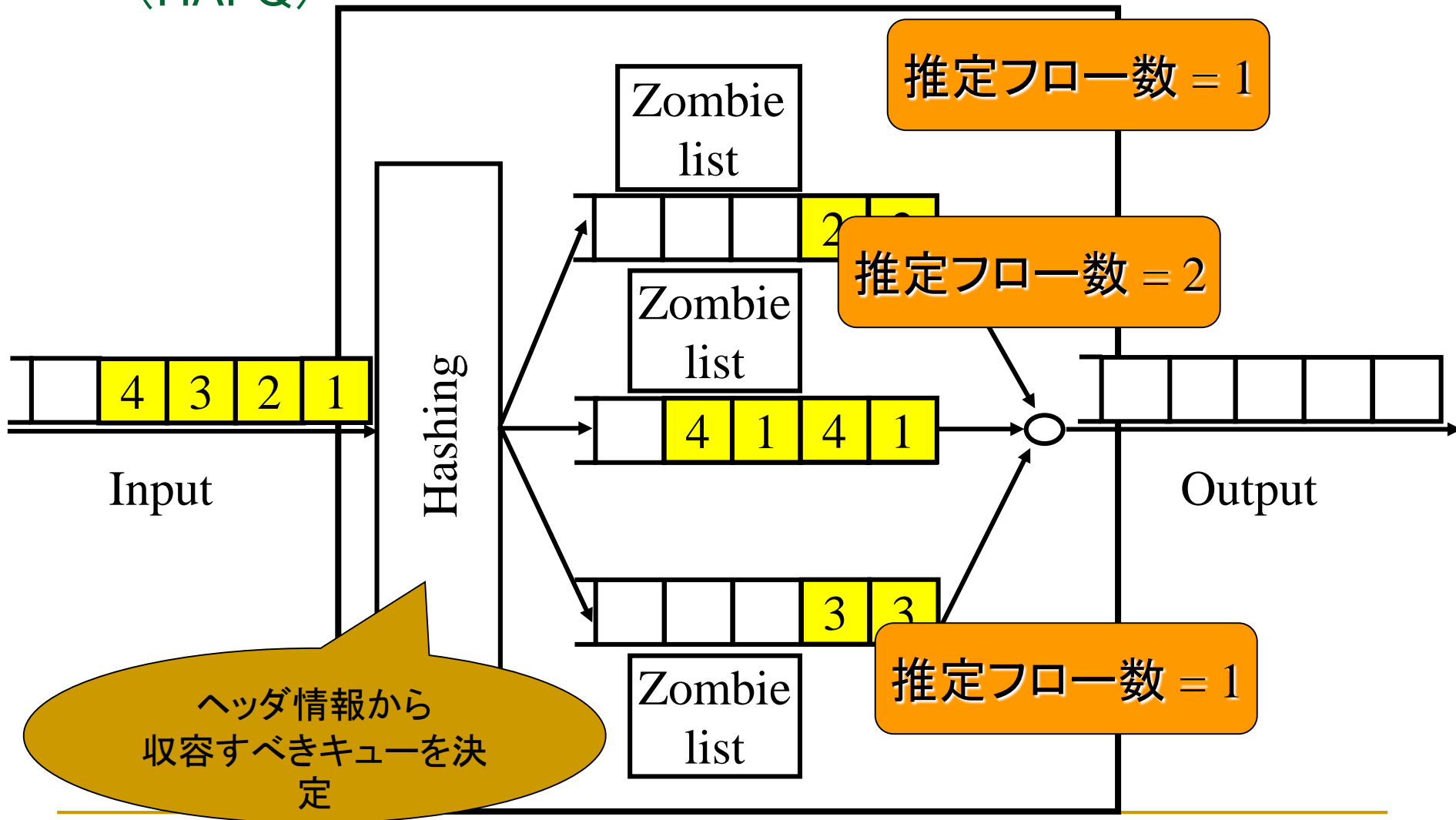
提案方式





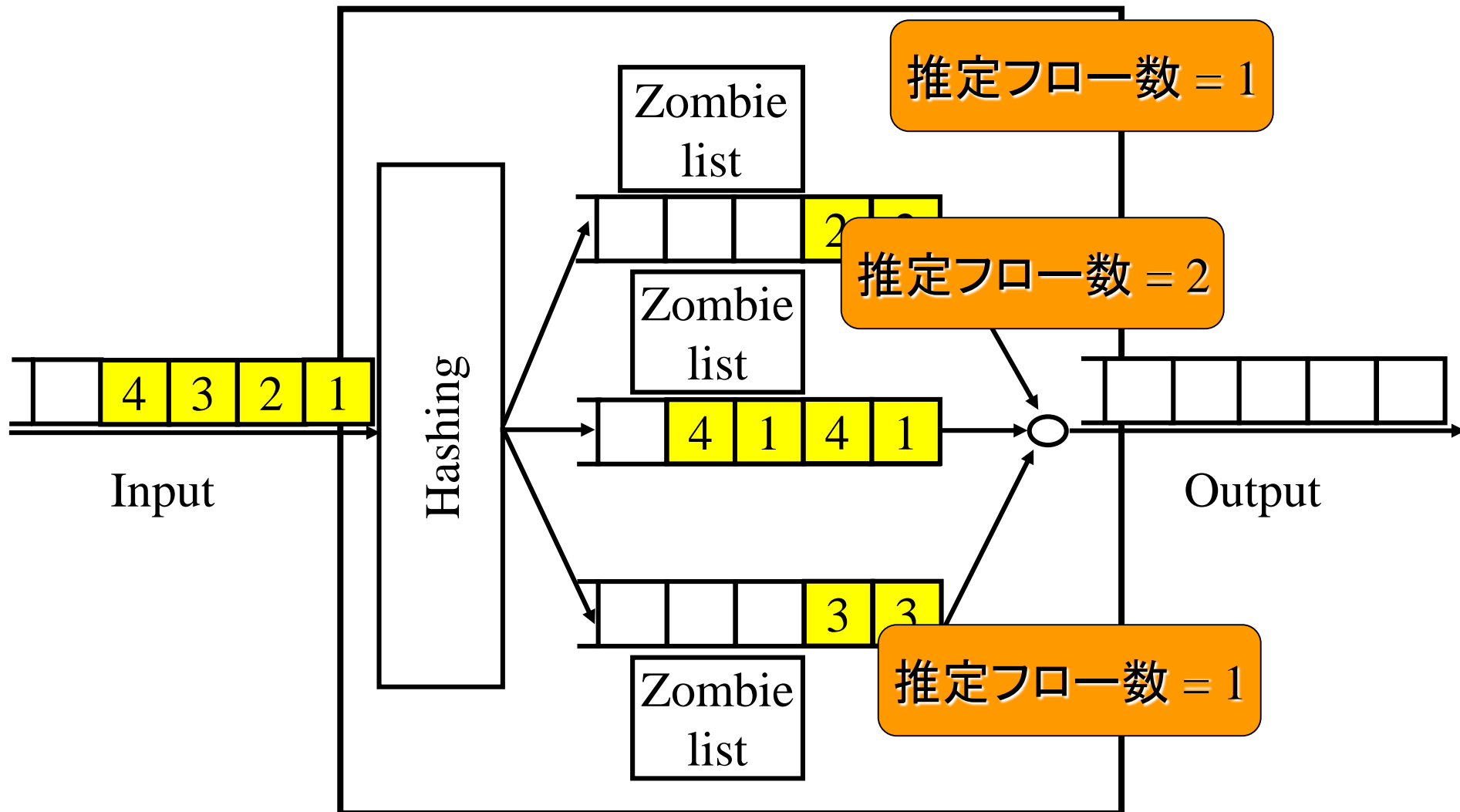
提案方式の概要（入力動作）

(HAFQ)



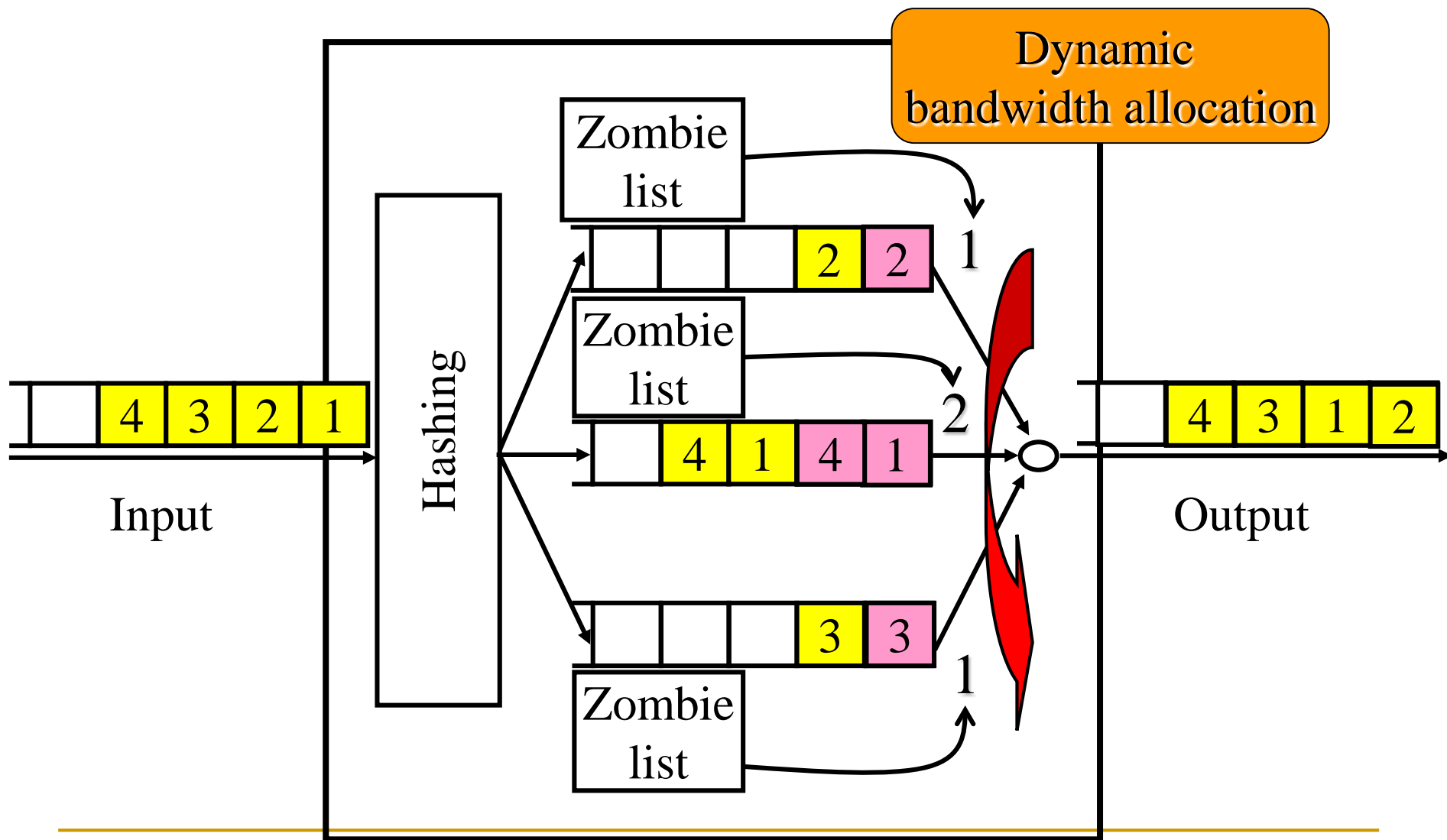


提案方式の概要（出力動作）





提案方式の概要（出力動作）

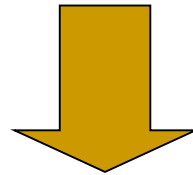




ゾンビリストとは？

- {flow Key, counter}を1つの組とする
- 過去に到着したフローに関する履歴を保持する（ただし、すべてのフローに関する情報は必要ない）

flow Key	counter
1	3
2	7
5	2
7	4

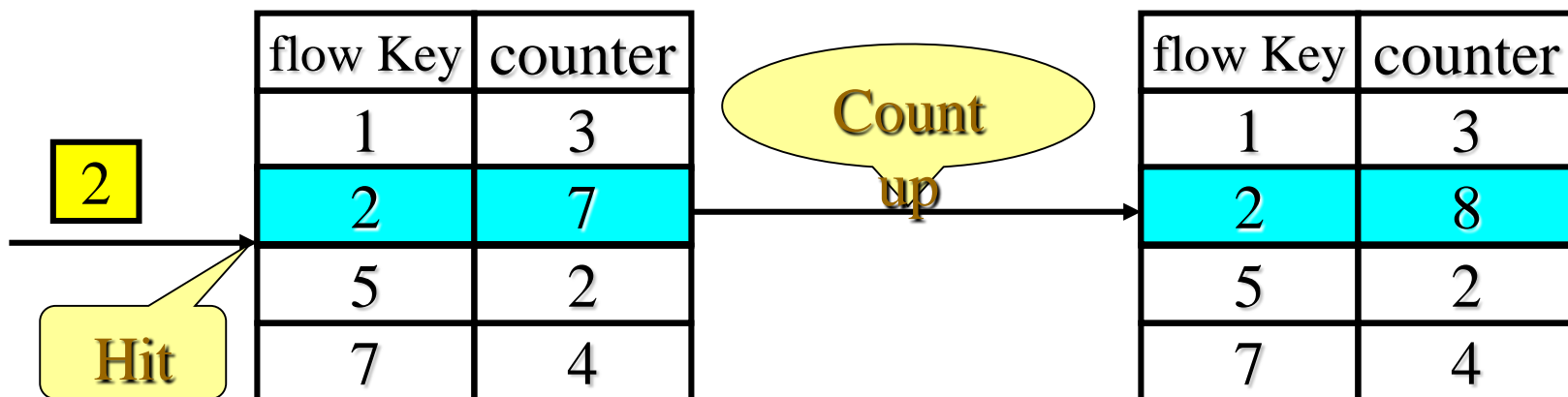


- ✓ そのキュー内に収容されたフロー数を推定できる
- ✓ 同一キュー内で、より多くの帯域を使用しているフローを発見できる



ゾンビリストの動作 (リスト内にKeyがあるとき)

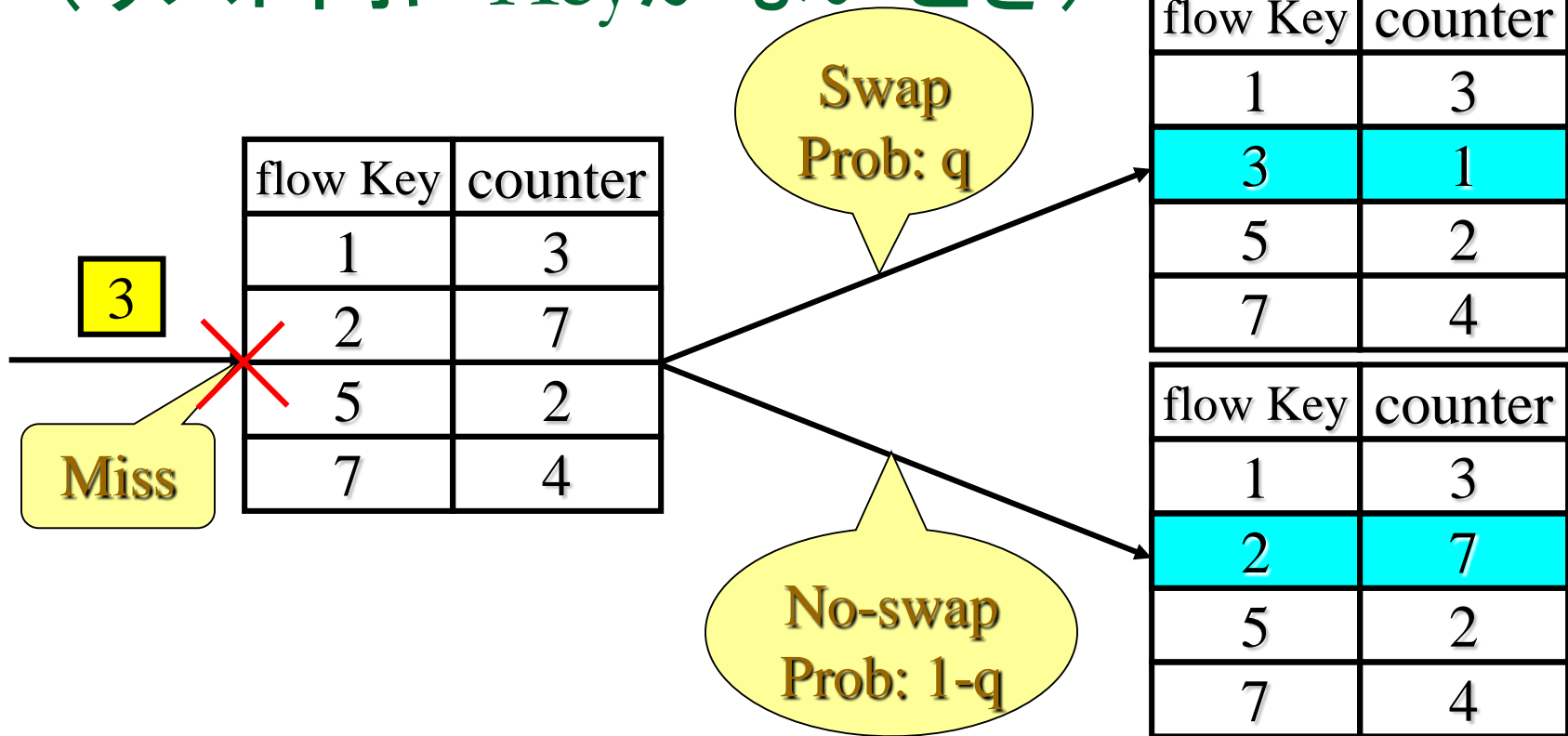
- パケットがルータに到着すると、ゾンビリストをすべて検索する



ゾンビリスト内に到着してきたフローのKeyがあれば、そのゾンビのカウンタ値を1増やす。



ゾンビリストの動作 (リスト内にKeyがないとき)



ゾンビリスト内に到着してきたフローのKeyがなければ、 q の確率で置き換えてカウンタ値を1に初期化する。
 $1-q$ の確率で何もしない。



フロー数推定方式の概要

$$\text{全フローの平均到着率} = \frac{\text{全フローの到着率の合計}}{\text{フロー数}}$$



$$\text{フロー数} = \frac{\text{全フローの到着率の合計}}{\text{全フローの平均到着率}}$$

※各フローの到着率に偏りがある場合も成立



- フロー数推定アルゴリズム
 - Step1 : 到着パケットが属するフローの到着率を計算
 - Step2 : 全フローの到着率の平均を計算
 - Step3 : 推定フロー数を計算



フロー数推定方式

■Step1: 到着パケットが属するフローの到着率を計算

■ゾンビリストのカウンタ最大値より到着率を統計的に推定

■カウンタ最大値はフローの到着率に比例

■ゾンビがSwapされたときのみ計算

■推定到着率 = (1-ヒット率)置き換え確率 / ゾンビ数(カウンタ最大値-1)

■Step2: 全フローの到着率の平均を計算

■推定到着率の移動時間平均を計算

■平均到着率 = a 平均到着率 + (1- a)計算した推定到着率

■レートの高いフローは高い割合で平均到着率に組み入れられる

■ a にバイアス $\Rightarrow a = \beta \times (\text{カウンタ最大値} / \text{推定レート})$

■Step3: 推定フロー数を計算

■推定フロー数 = $1 / \text{平均到着率}$



同一キュー内での優先廃棄

- 各キュー間の公平性は保たれる
 - 各フローのレートにはばらつきがあるため、同一キュー内の公平性は保たれない
- あるエントリのカウンタ値が大きければ大きいほど他のフローよりも多くの帯域を使用している可能性が高い

カウンタ値が大きいフローの packets を優先的に廃棄

flow Key	counter
1	2
2	10
5	2
7	3

ある期間内に
ルータに到着した

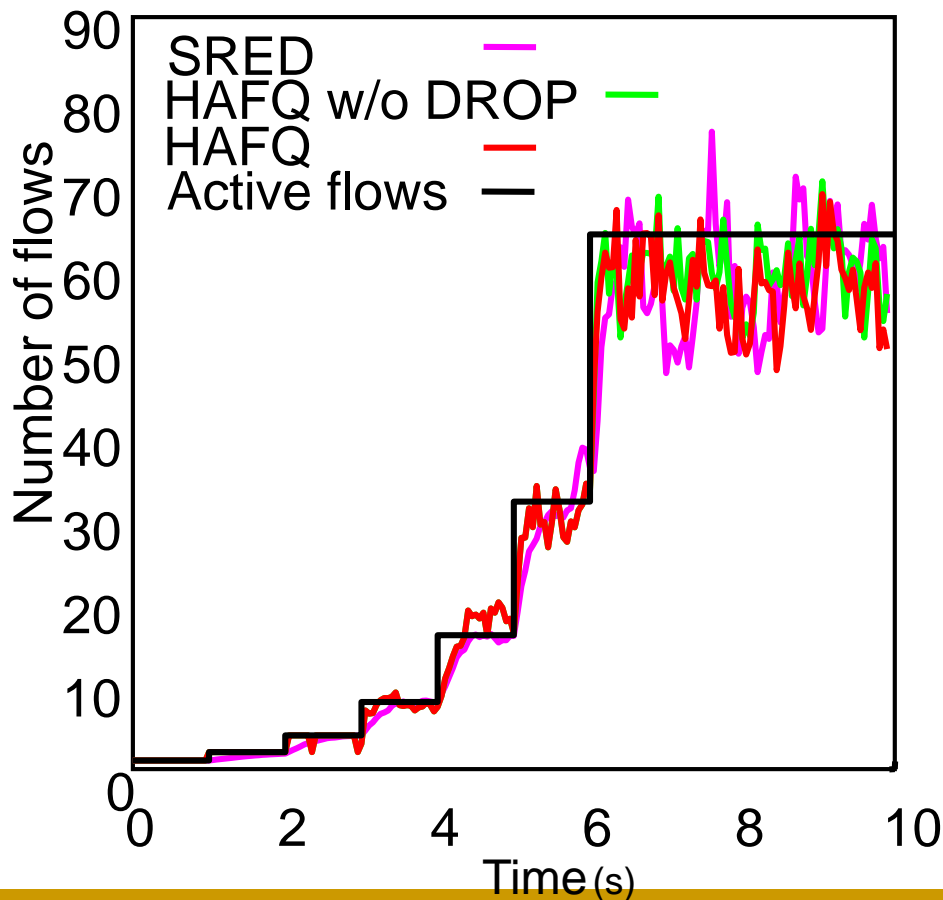
パケット数

平成13年度 修士論文発表会

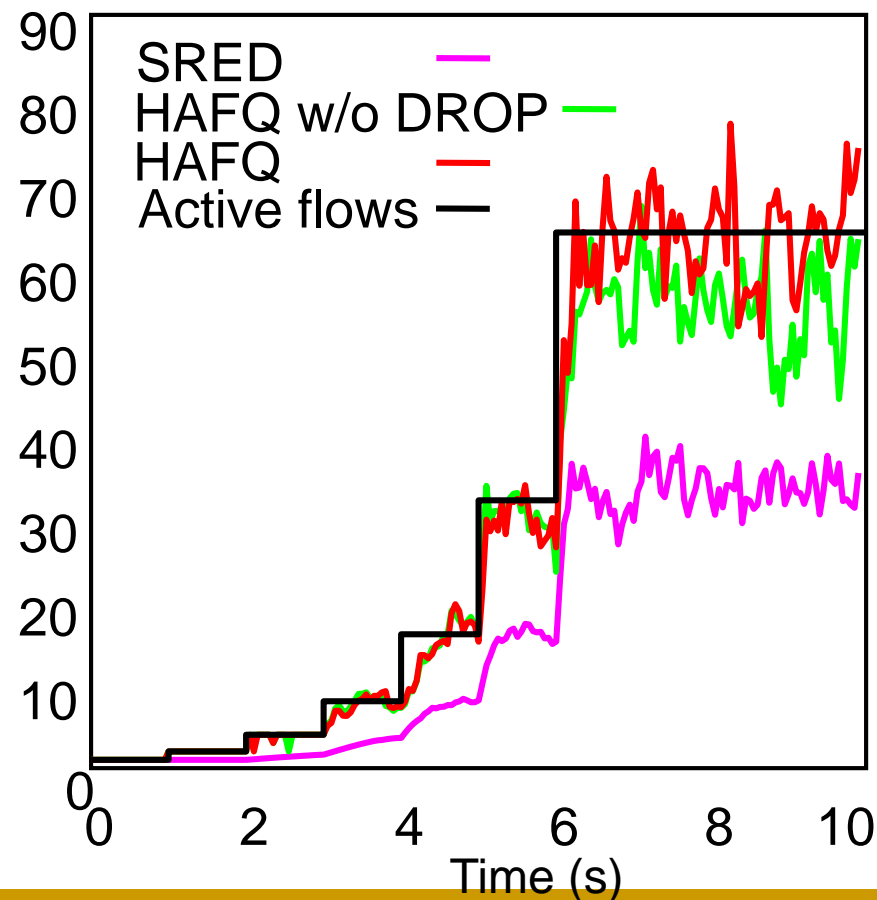


推定フロー数の評価

- 1秒毎にフロー数を2倍にしていく
TCPのみ64本



- TCP : 32本、UDP (3.2Mbps) : 32本



従来手法に比べてHAFQは推定フロー数が実際の数に近い

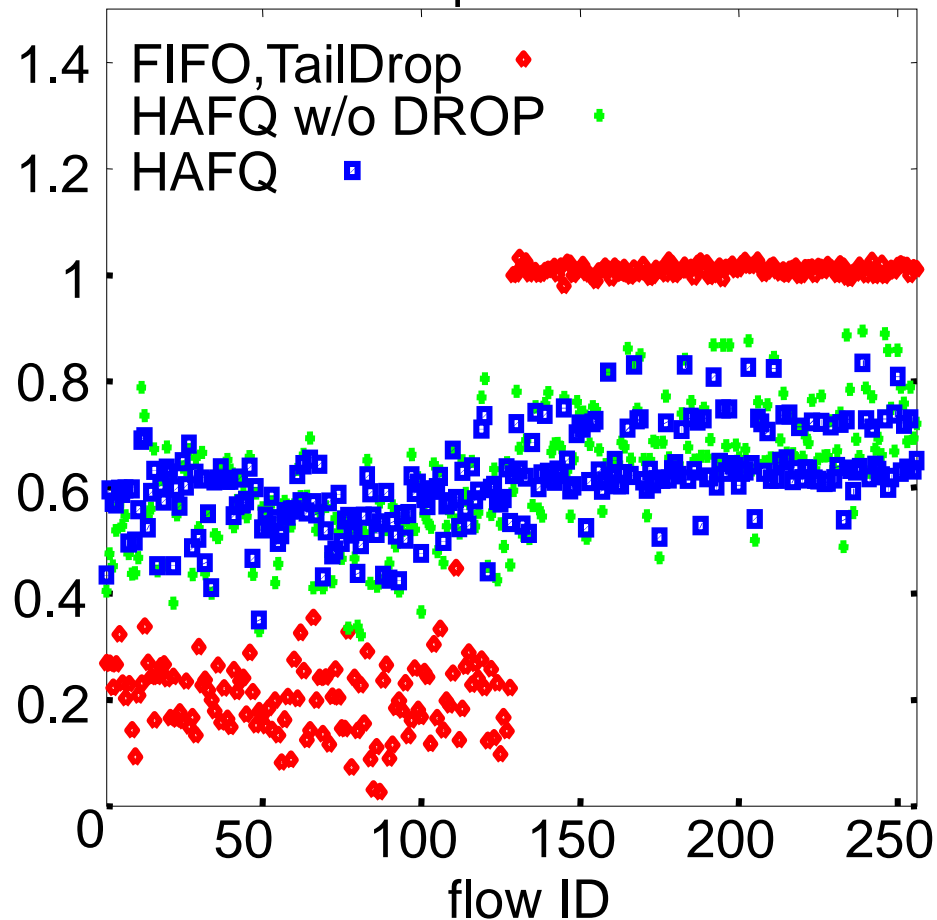
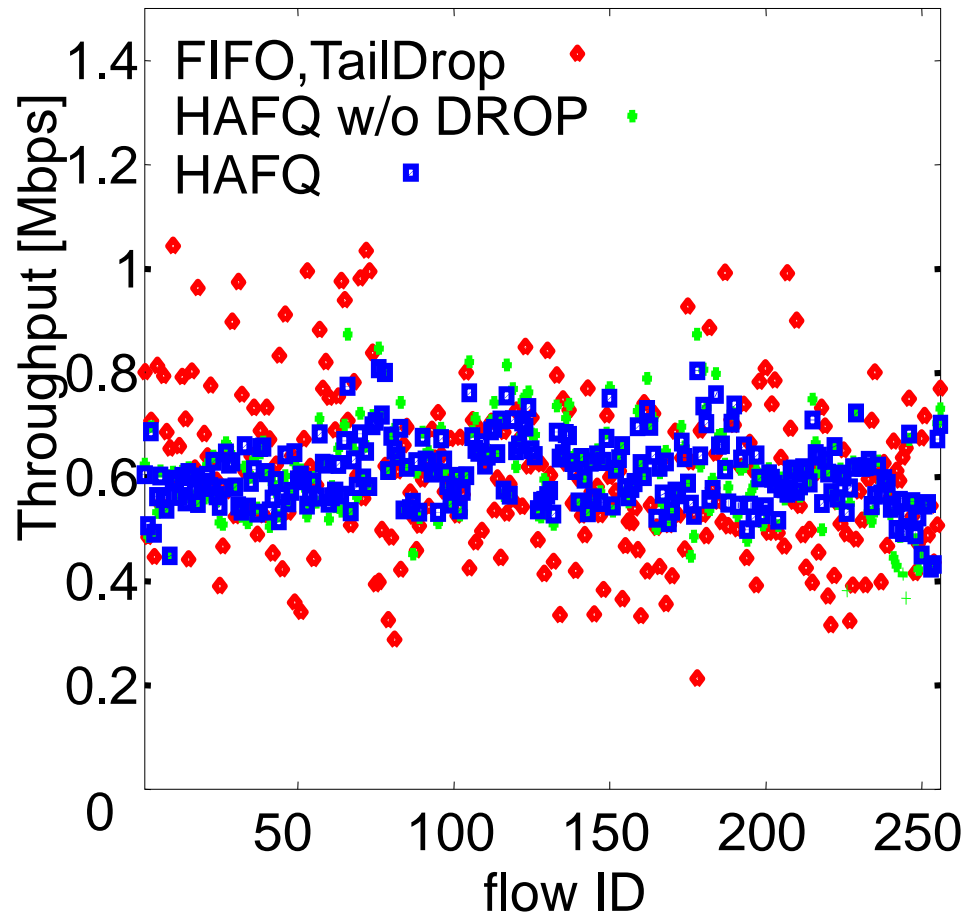


各フローのスループットの評価

●各フローは同時に送信を開始する
TCPのみ 256本

TCP : 128本

UDP (1.2Mbps) : 128本



レートの高いフローが存在する場合でも、HAFQでは高い公平性を実現



高速ルータの現状

- バックボーンルータにおいては、高速な処理速度が必要である。
 - 例：40 Gbps のルータの処理能力
10 ns / packet (40 byte / packet)
SRAM (off-chip) へのメモリアクセス
100 ns / メモリアクセス
- メモリアクセスがボトルネック
オンチップメモリの利用、メモリアクセスの削減
- スケジューリング計算、その他の制御も高速に処理する必要がある



実装評価の必要性

シミュレーションや解析による定性的な
評価では不十分

ハードウェアへの実装による定量的な評価が必要

ネットワークプロセッサ

- インテル社の IXP1200 を用いて評価
- ルータでの処理をプログラマブルに実現可能
- 実環境に近い状態で実装評価できる



まとめと今後の課題

■ まとめ

- エッジルータやコアルータの能力に応じてスケーラブルに実装可能なパケットスケジューリング方式の提案
- フローごとの優れた公平性を実現

■ 今後の課題

- 既存のルータと共存できるのか？
- IXP1200 評価ボードを用いた実装実験



今後の研究の進め方

■ 研究の目的

- ルータにパケットスケジューリング機構を持たせ、ネットワーク側での制御による公平性の向上、帯域占有の防止を実現する

■ 研究のゴール

- ルータでのパケットスケジューリング方式の提案と有効性の評価および実装
- 10 Gbps を超えるような超高速ルータへ適用できるルータの開発

パケットスケジューリング方式の例



■ エッジルータ

- TCP特性を考慮し、トランスポートレベルでの獲得帯域の公平化
- アクセス回線帯域の考慮
- TCPでないフローに対する強制的な制御
- コアルータでの制御との協調

■ コアルータ

- ステートレス制御