Performance Prediction Methods for Address Lookup Algorithms of IP Routers

Ryo Kawabe
Graduate School of Engineering Science
Osaka University
E-mail : r-kawabe@ics.es.osaka-u.ac.jp

Contents

Background

- An explosive growth in demand for high-speed packet transmission technologies
  - Link capacity
  - Packet forwarding capability of IP routers
- Address lookup
  - Requires complicated processing

⇒ To improve the router performance, it is necessary to make address lookup more efficient

Many address lookup algorithms have been proposed

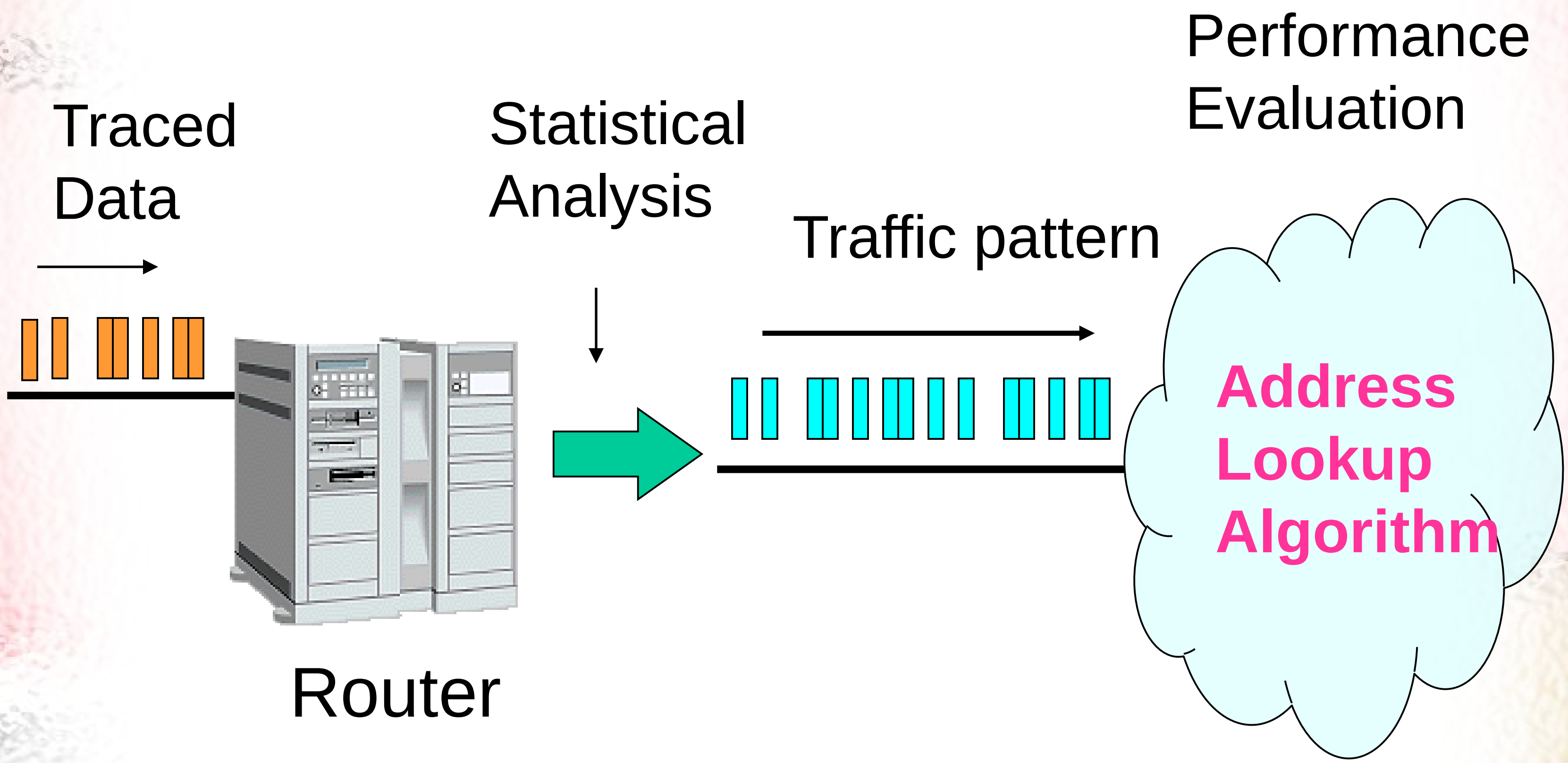
Performance Studies on Address Lookup Algorithms

- Worst-case performance
  - Easy to derive from the complexities
  - Important index to show its basic capability
  - Not always the best metric for the customers
- Average-case performance
  - Sometimes useful
  - Simulation using random address
    - Large difference between its result and the actual performance
  - Simulation using traced-data
    - Lack of generality because the amount is limited

Objective

- More realistic prediction of the average-case (actual) performance based on the statistical traffic analysis
  - models the actual traffic
  - generates traffic pattern
  - predicts the actual performance of address lookup algorithms

Overview of Our Proposed Method



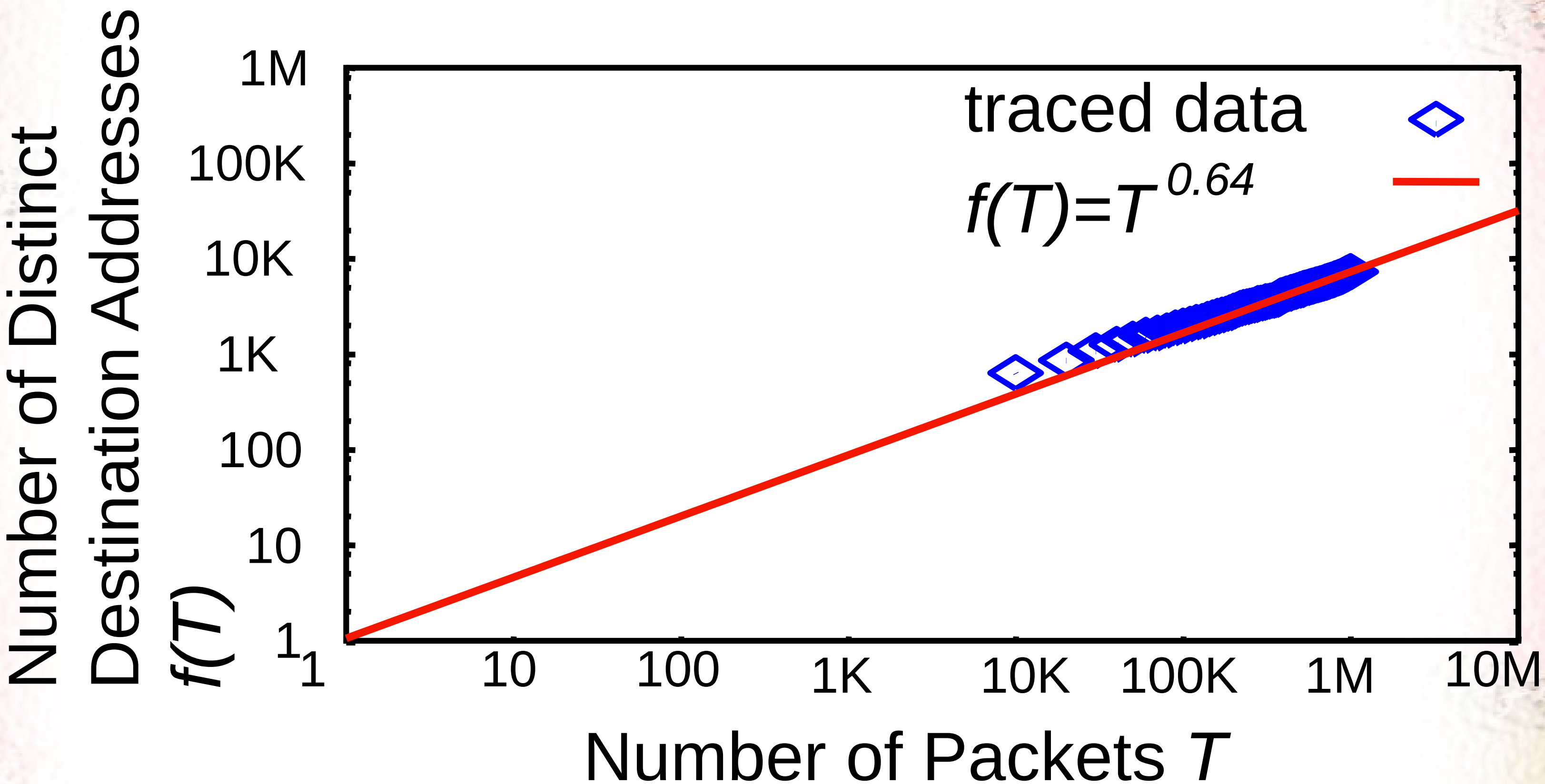
Generation of Traffic Pattern

Pseudo-address generation algorithm

- Pseudo-address generation algorithm [5]
  - Inverse Stack Growth Function (ISGF)
    - means the expected number of distinct destination addresses dependent on number of packets
  - LRU Stack Model
    - generates address sequences of packets, if ISGF is given

[5] M. Aida and T. Abe, “Pseudo-address generation algorithm of packet destinations for Internet Performance Simulation,” in Proceedings of IEEE INFOCOM 2001, pp.1425-1433, April 2001

ISGF (The Gateway on Osaka Univ.)



$$f(T) \cong T^{\alpha} \quad (T >> 1)$$

LRU stack model

LRU stack

A(1)
A(2)
:
A(i)
:
A(n)
:

The probability that next packet is A(i)

a_1 *(e.g. =50%)*

a_2

:

a_i

:

a_n

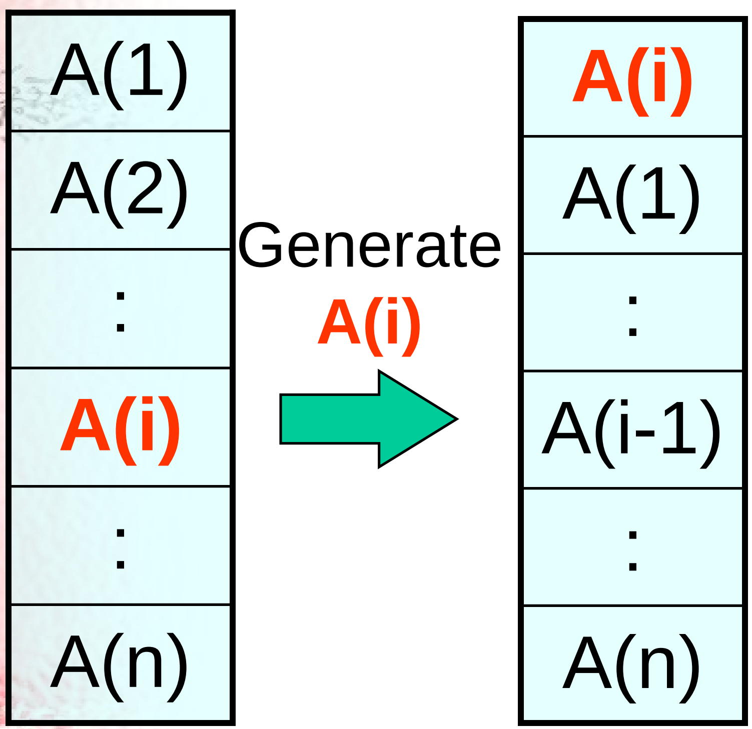
:

- i-th most recently arrived address A(i) is stored at i-th element of the LRU stack
- The probability a_i that next arriving address is A(i) in the LRU stack is determined by its position and ISGF.

$$a_i = \{((i-1)^{1/a}+1)^a - (i-1)\} - \{(i^{1/a}+1)^a - i\}$$

Address Generation per Packet (AGP)

LRU stack



1. Store distinct destination addresses of traced-data in the LRU stack
2. Generate the address according to a_i
3. Assign the address as the destination address of packets
4. Move the element to the top and return step 2.

$$a_i = \{((i-1)^{1/a}+1)^a - (i-1)\} - \{((i^{1/a}+1)^a - i\}$$

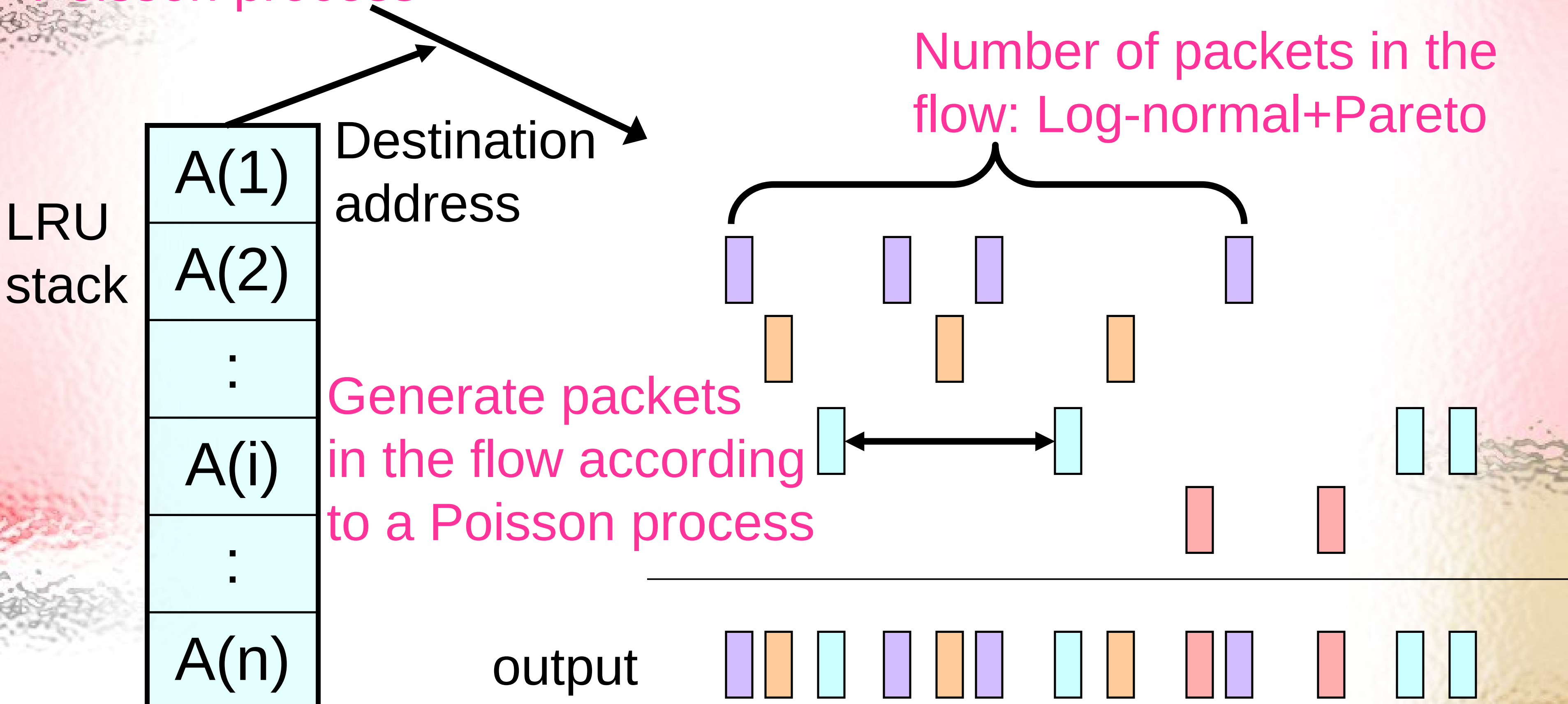
Our Approach

- AGP does not consider the flow characteristics (e.x. flow duration) of actual traffic

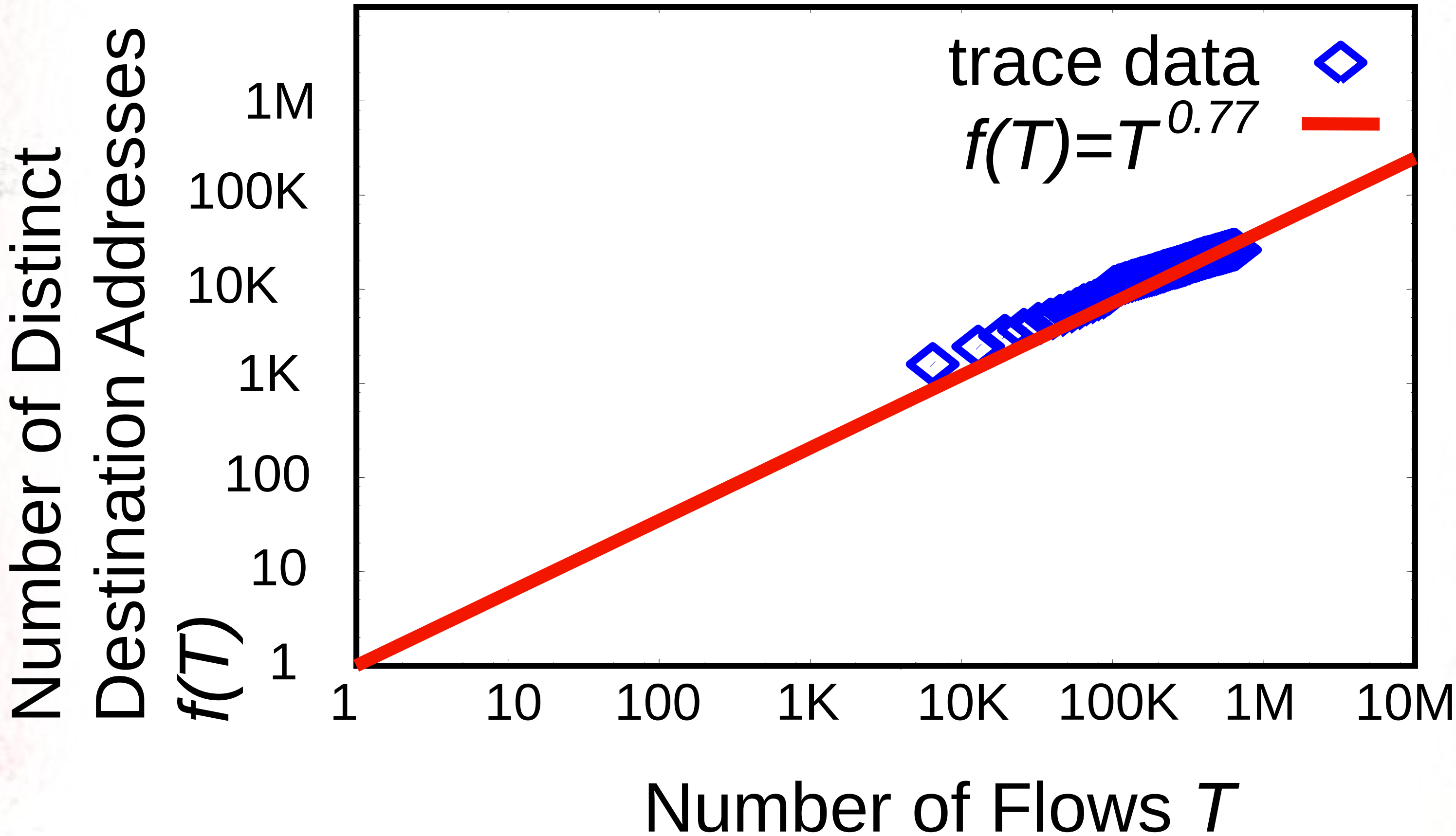
↓

- We propose a new method using flow-based address generation (called AGF)

Address Generation per Flow (AGF)



ISGF on Flows (The gateway on Osaka Univ.)



Performance evaluation through trace-driven simulation

Example Applications of Our Proposed Methods

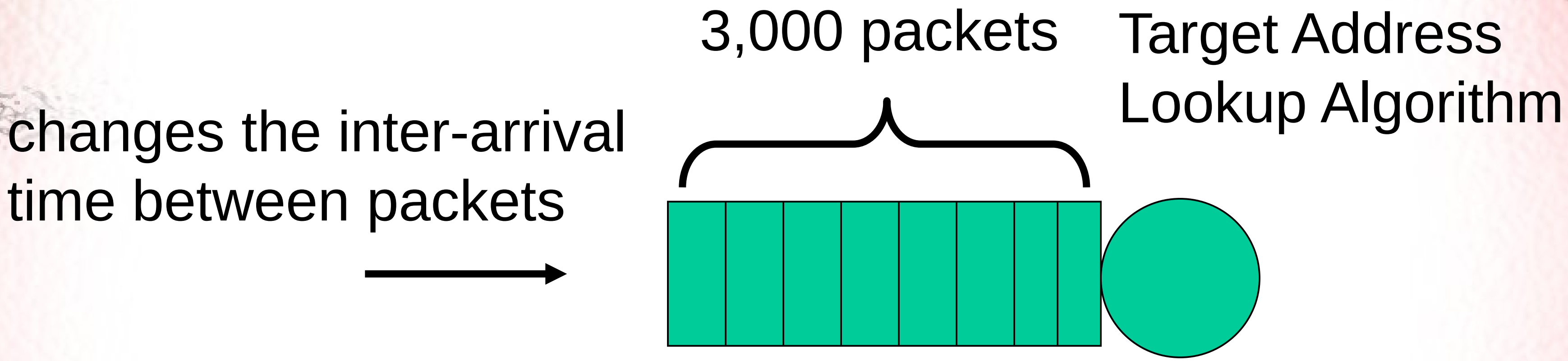
- Patricia Tree Search
  - Extended binary tree
- Pointer Cache Method [3]
  - CAM-based algorithm combining CAM and the Patricia Tree Search

[3] M. Uga, K. Shiomoto, "A fast and compact longest prefix look-up method using pointer cache for very long network address," in *Proceedings of IEEE ICCCN '99*, pp. 1240-1247, Apr. 1999

Address Generation Methods Evaluated in Simulation

- Actual traffic
  - A raw sequence of 10,000,000 packets from traced data
- Random traffic
  - 10,000,000 addresses randomly chosen 32-bit values
- Trace-Random traffic
  - 10,000,000 addresses randomly picked up the destination addresses of 1,000,000 packets from the traced data
- AGP and AGF
  - 10,000,000 packets generated based on 1,000,000 packets from the traced data

Performance Metrics



- Maximum Throughput
  - It is defined as the reciprocal of the minimum average of packet inter-arrival time if no packet is lost during the simulation
- Time-dependent input queue length
  - The number of packets queued in the buffer

Simulation Results

Maximum Throughput (Patricia Tree Search)

mpps: million packets per second

	Maximum Throughput	Error Rate
Actual	4.63 mpps	--
Random	8.33 mpps	79.9 %
Trace-Random	4.67 mpps	0.86 %
AGP	4.63 mpps	0.00 %
AGF	4.52 mpps	2.38 %

⇨ Trace-Random, AGP and AGF provide good estimation with low errors compared with Random

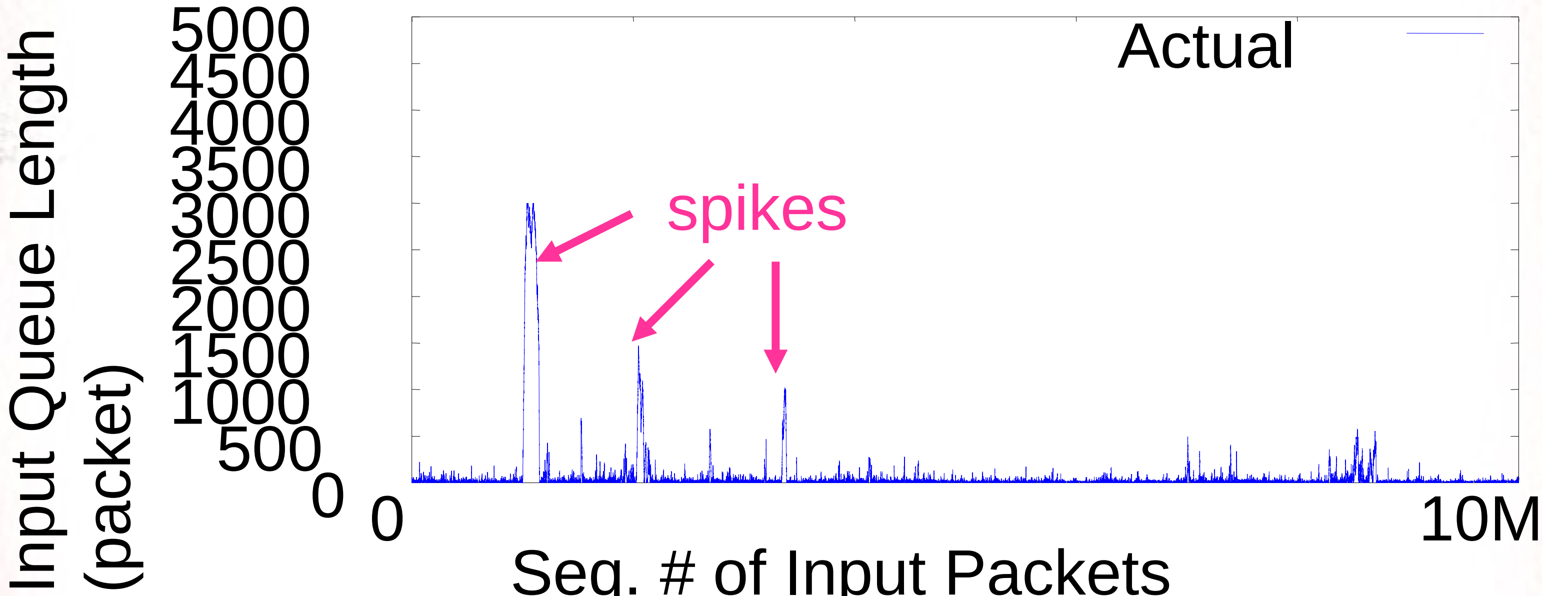
Maximum Throughput (Pointer Cache Method)

mpps: million packet per second

	Maximum Throughput	Error Rate
Actual	35.2 mpps	--
Random	58.8 mpps	67.0 %
Trace-Random	36.5 mpps	3.69 %
AGP	35.0 mpps	0.57 %
AGF	35.0 mpps	0.57 %

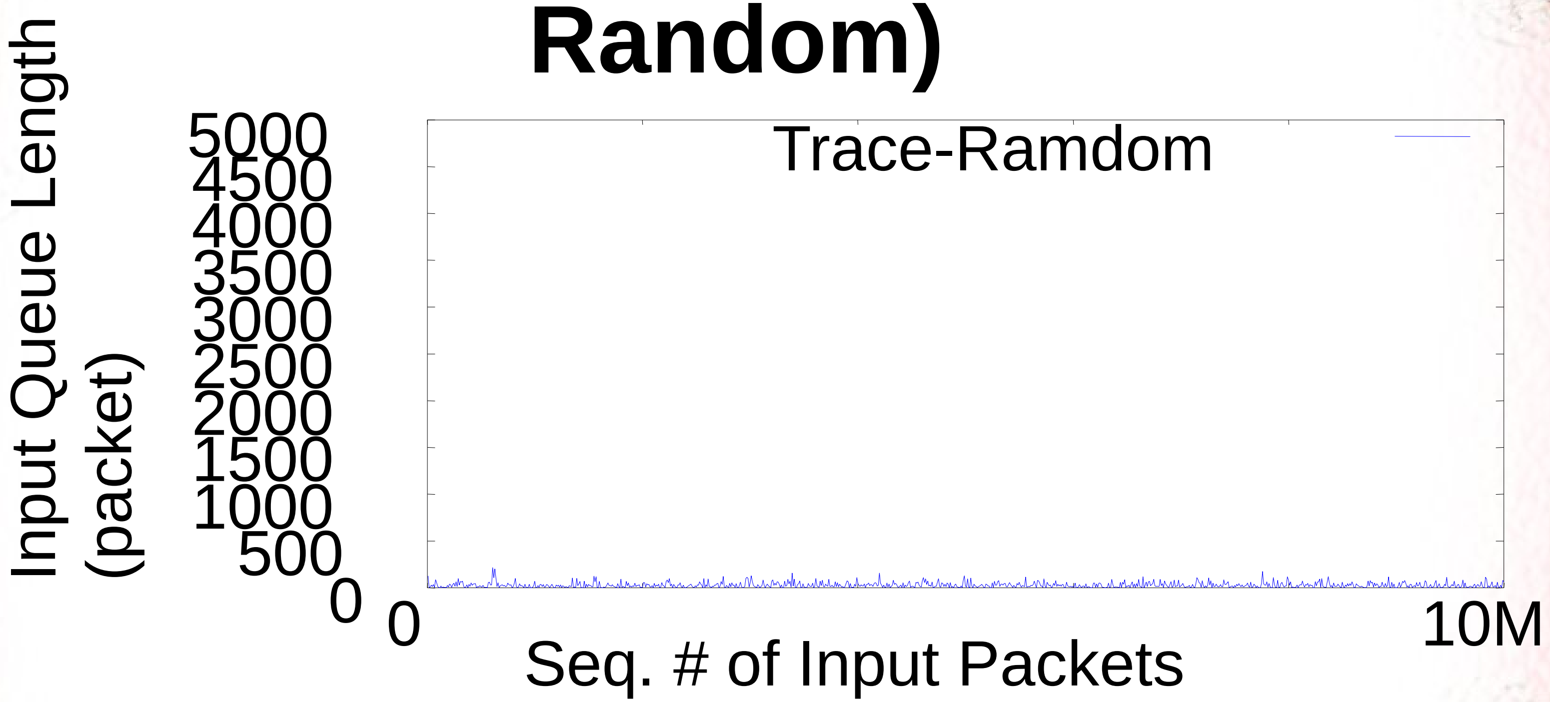
Our proposed methods can provide good estimation regardless of target address lookup algorithms

Transition of Input Queue Length (Pointer Cache Method: Actual)



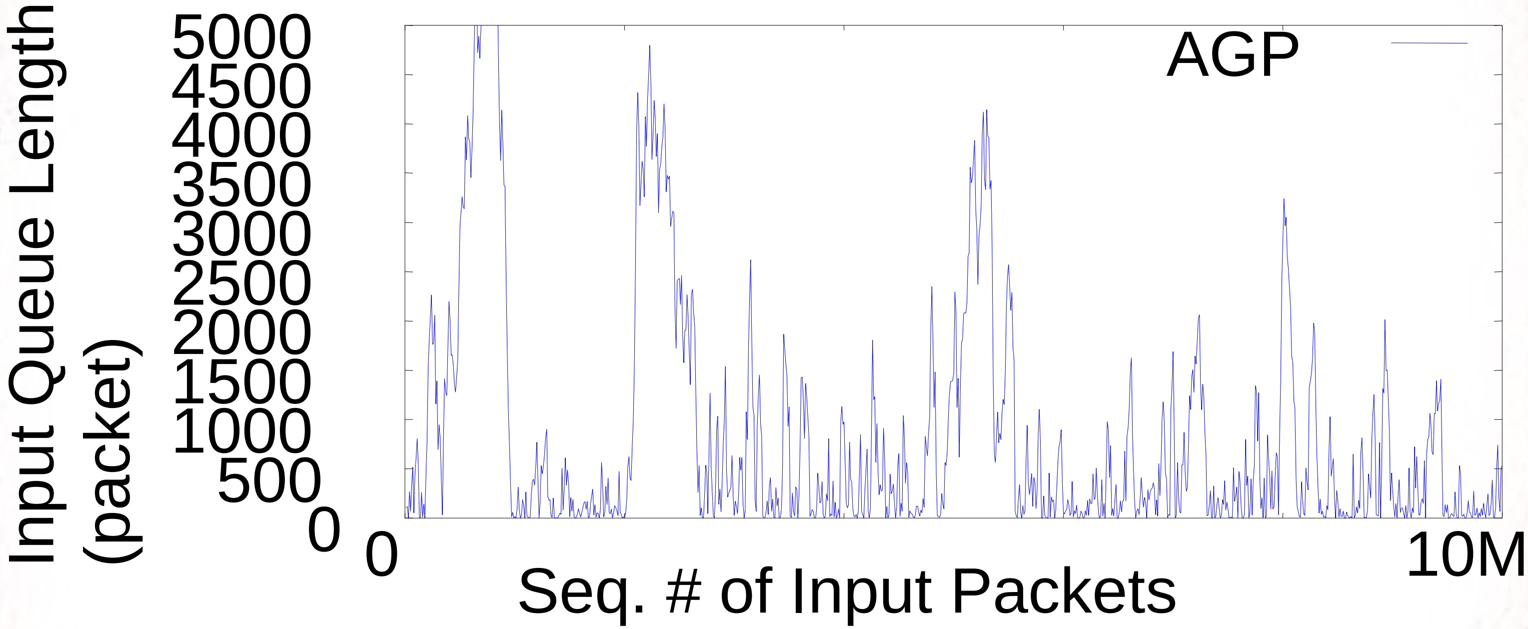
Its fluctuation is low (below 50 packets),
but, sometimes increase significantly (spike)

Transition of Input Queue Length (Pointer Cache Method: Trace-Random)



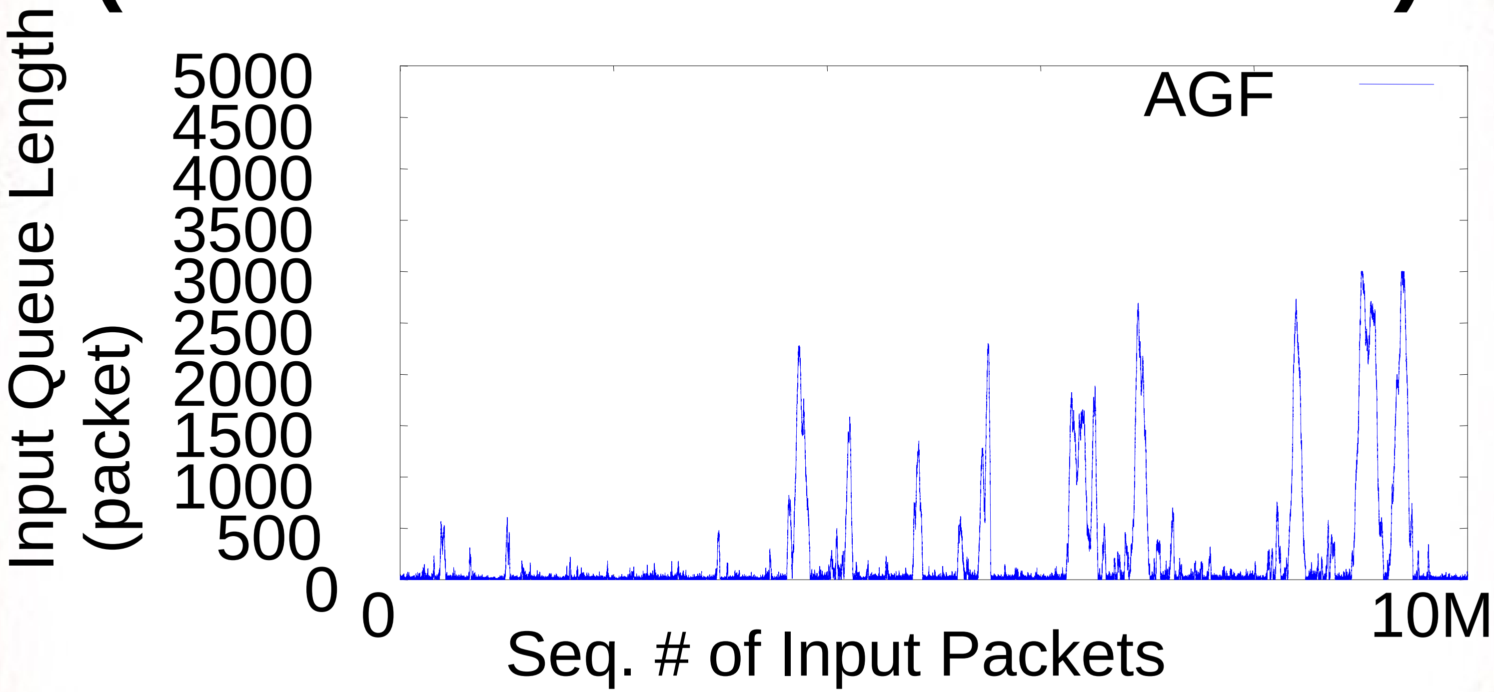
Any spikes does not appear

Transition of Input Queue Length (Pointer Cache Method: AGP)



Queue length always fluctuates
Any spikes cannot be observed

Transition of Input Queue Length (Pointer Cache Method: AGF)



More similar behavior to that of Actual case than those of Trace-Random and AGP

AGF has more spikes than Actual case

Conclusion and Future Topics

- Conclusion
  - Random traffic overestimates the performance
  - Trace-Random, AGP and AGF can provide good estimation of the Maximum throughput
  - In AGF, the time-dependent behavior of the input queue length is similar to Actual case
- Future Topics
  - Develop the analytic approach for quick prediction
  - Validate the applicability of AGF to the packet classification algorithms