SAROS:A Self-Adaptive Routing Oblivious Sampling Method for Network-wide Heavy Hitter Detection

Enhan Li

2024.8

北京航空航天大學
杭州创新研究院

中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY ,CHINESE ACADEMY OF SCIENCES

Background Heavy Hitter Detection

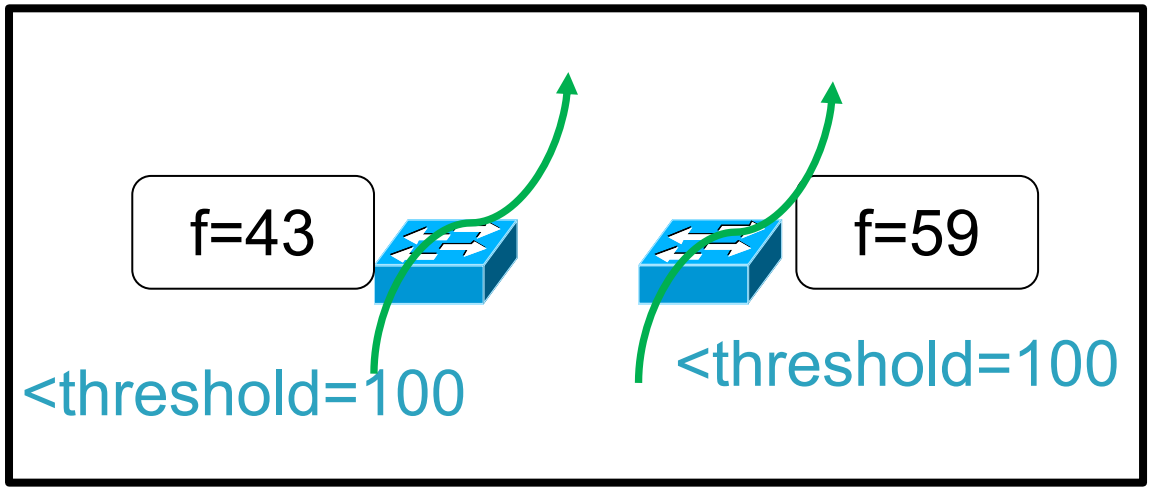
- **Heavy Hitter (HH)：** a flow with the same identifier has a proportion which is greater than a certain threshold to network volume

$$\frac{f}{V} \geq \theta \quad \text{(f: size of flow, V:network volume, } \theta\text{:threshold, } 0<\theta<1\text{)}$$

- **Significance:**
  - Large network flows carry a large part of the total traffic flow
  - Increase the completion time of small flows that are sensitive to delays
  - The main source of congestion

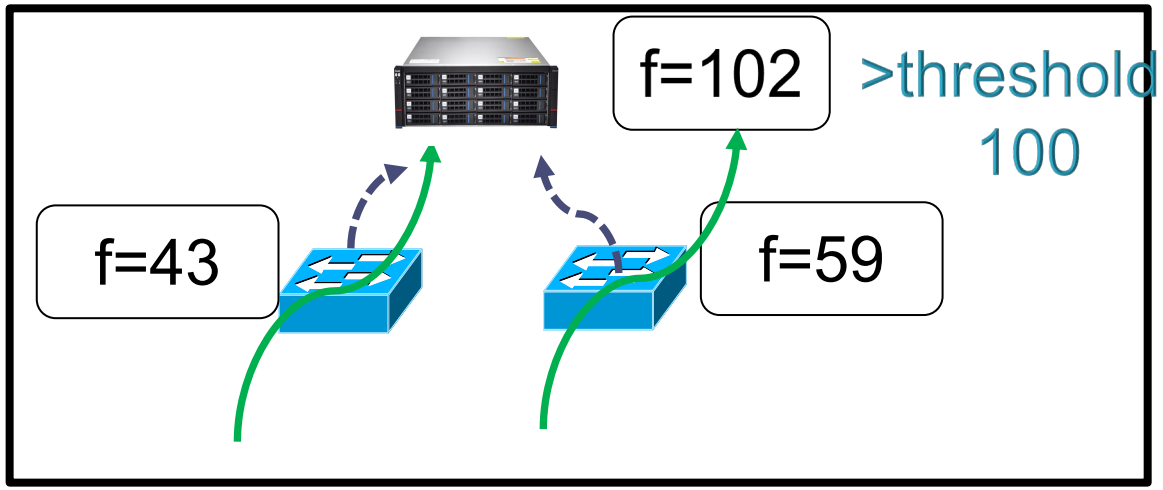
Background Network-Wide Heavy Hitter Detection

- **Heavy Hitter(HH):** a flow which proportion to network volume is greater than a certain threshold $\frac{f}{V} \geq \theta$（0<θ<1）



Detect on Single Network Measurement Point

miss HH f



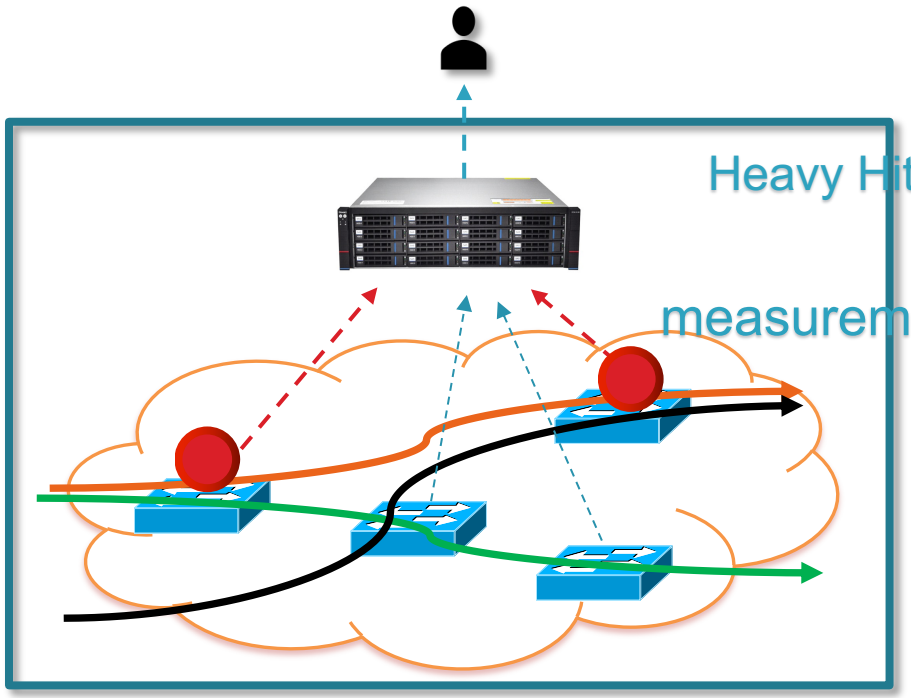
Network-wide Heavy Hitter Detection for more Accuracy

get HH f

Background - SDN & Programmable Data Plane

SDN → Open controll plane programmability

Programmable Data Plane → Open data plane programmability



Central Controller

Merge Network-Wide Information

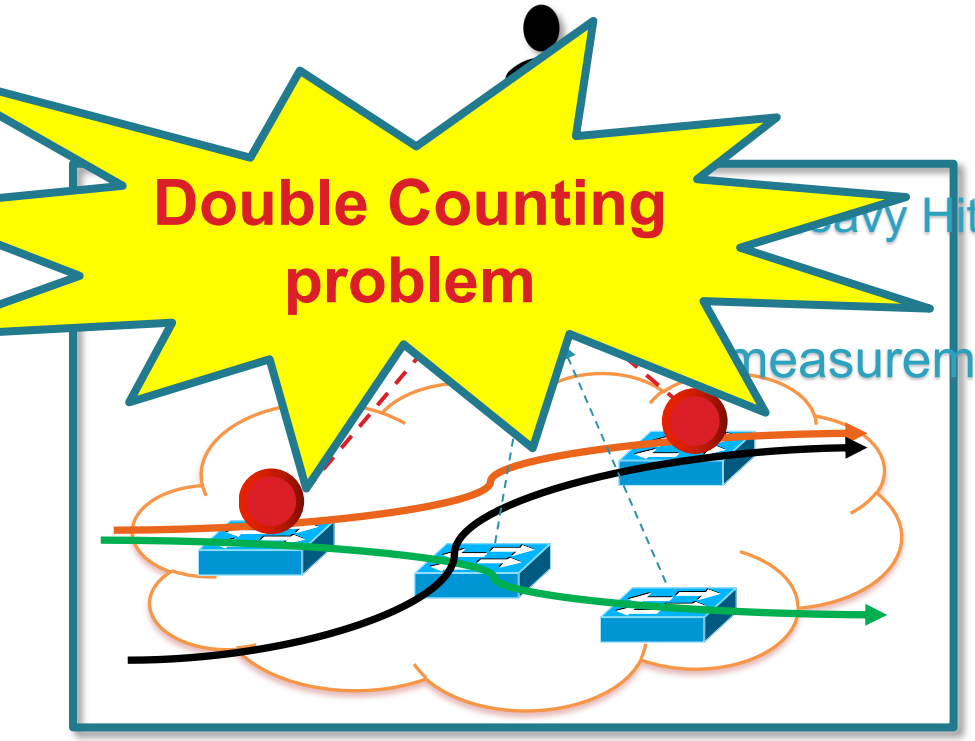
Programmable network Measurement Points（NMPs）

Deploy special data structures /algorithms for sampling

Background - SDN & Programmable Data Plane

SDN → Open controll plane programmability

Programmable Data Plane → Open data plane programmability



Heavy Hitter

measurement information

Central Controller

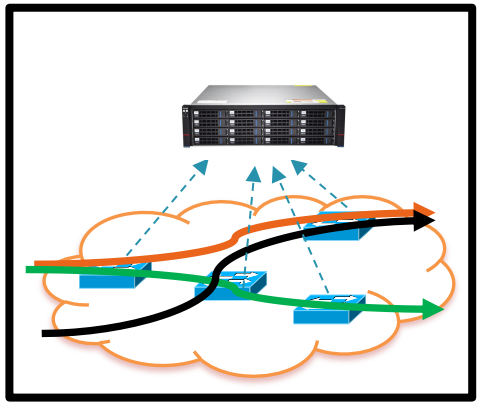
Merge Network-Wide Information

Programmable network Measurement Points（NMPs）

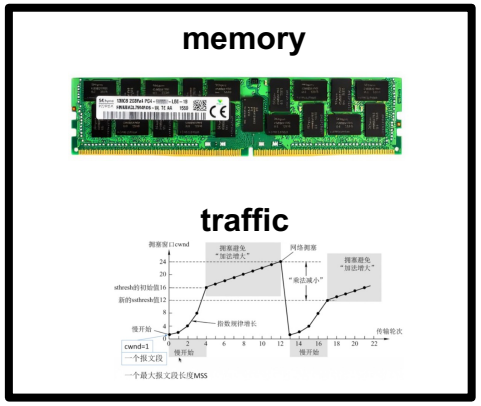
Deploy special data structures /algorithms for sampling

Challenges

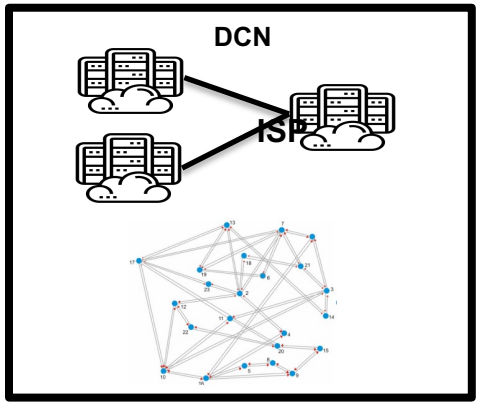
- **Chanllenges:**
  - Packets double counting problem in routing oblivious situation
  - Balance detection accuracy and data plane storage use at the same time
- **Goals:**



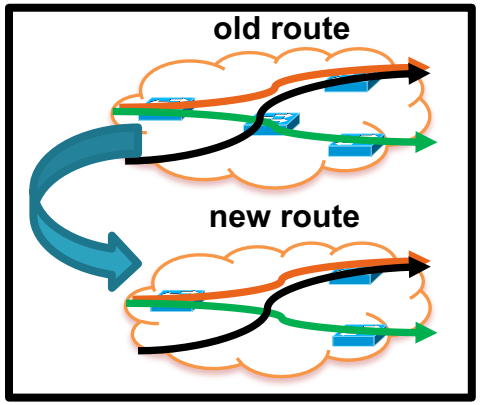
Solve packet double counting



Keep low memory & low traffic



Can be deployed in real network

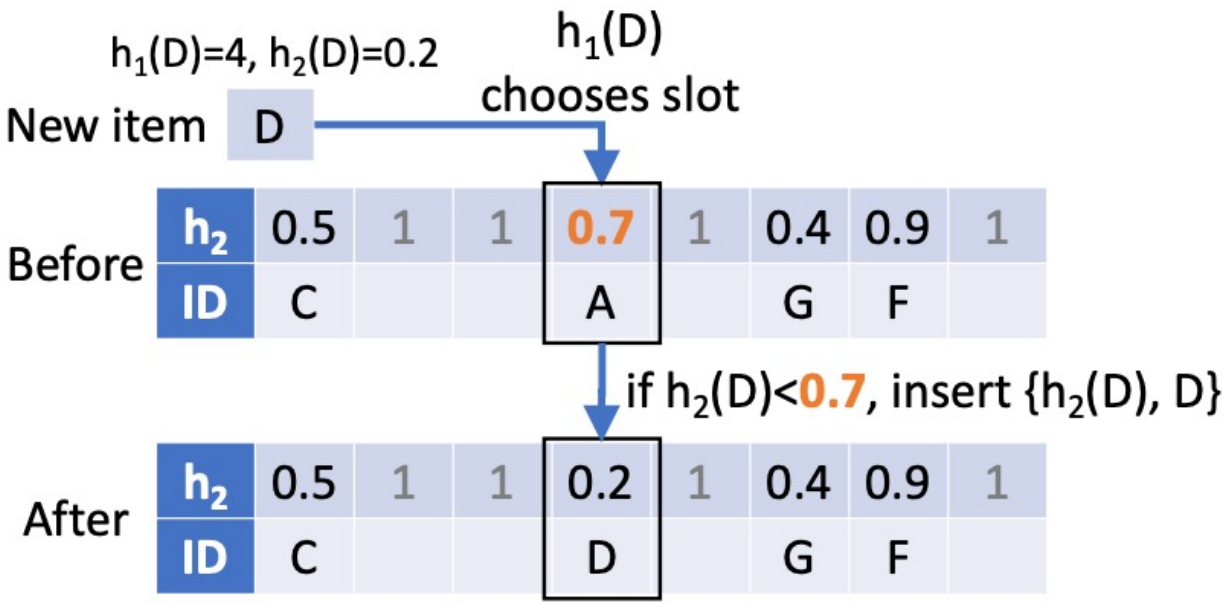


Response to routing updates quickly

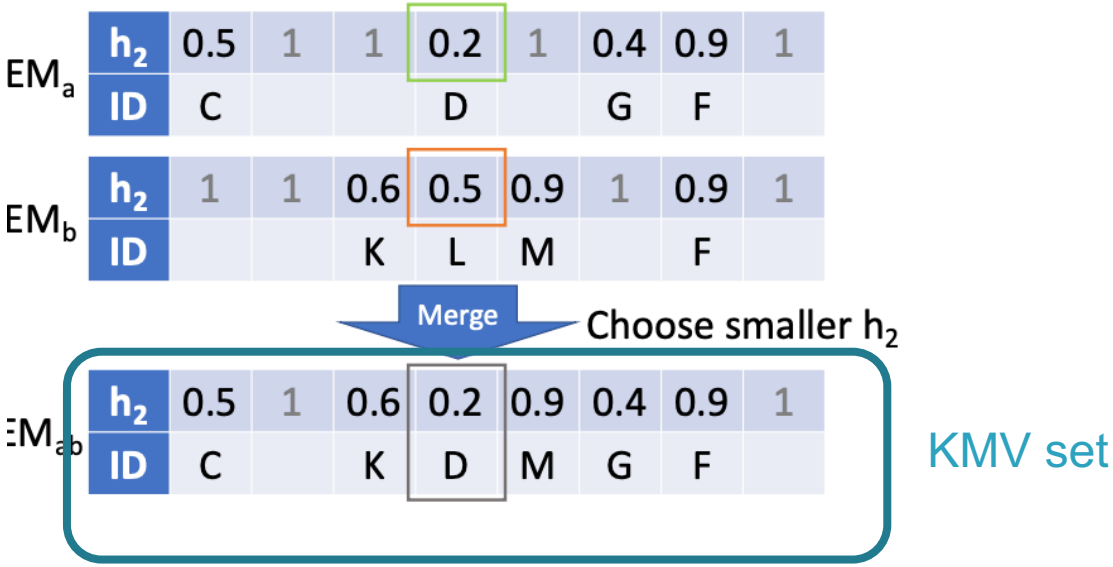
Existing Method

AROMA [IFIP 2020]

- Key Point：Use K Minimum Value(KMV) algorithm to solve packet double counting problem
- Disadvantages: Needs to mantain flow key information in dataplane (asociated with k) while P4 data plane has limited memory)



Update on Data Plane



Merge on Control Plane

Review K Minimum Value(KMV) Algorithm(1/2)

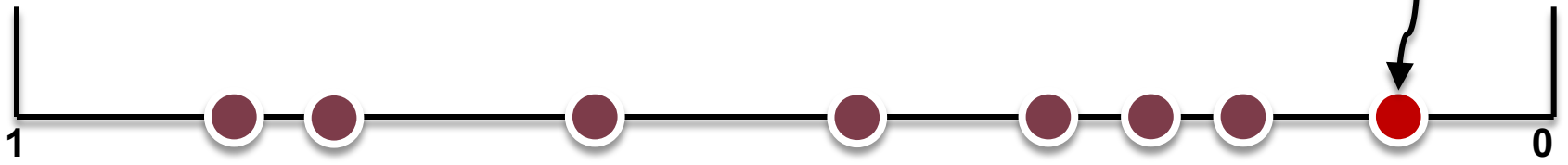
k=1

actural different packets num = 8

estimate different packets num = $\frac{1}{0.1}$ = 10

average space = 0.1

minimum hash value =0.1



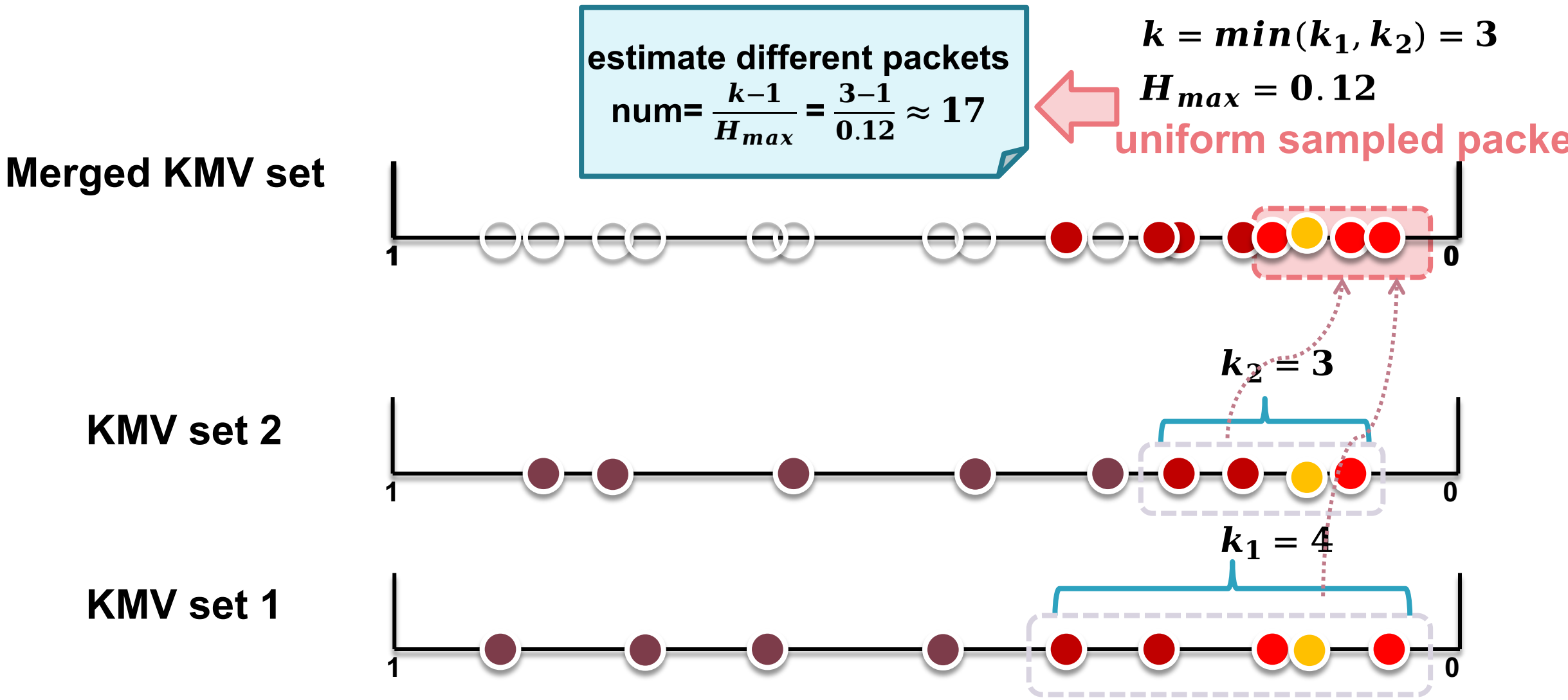
k=3

actural num of different packets = 8

estimate different packets num= $\frac{k-1}{H_{max}}$ = $\frac{3-1}{0.3}$ = 6.7

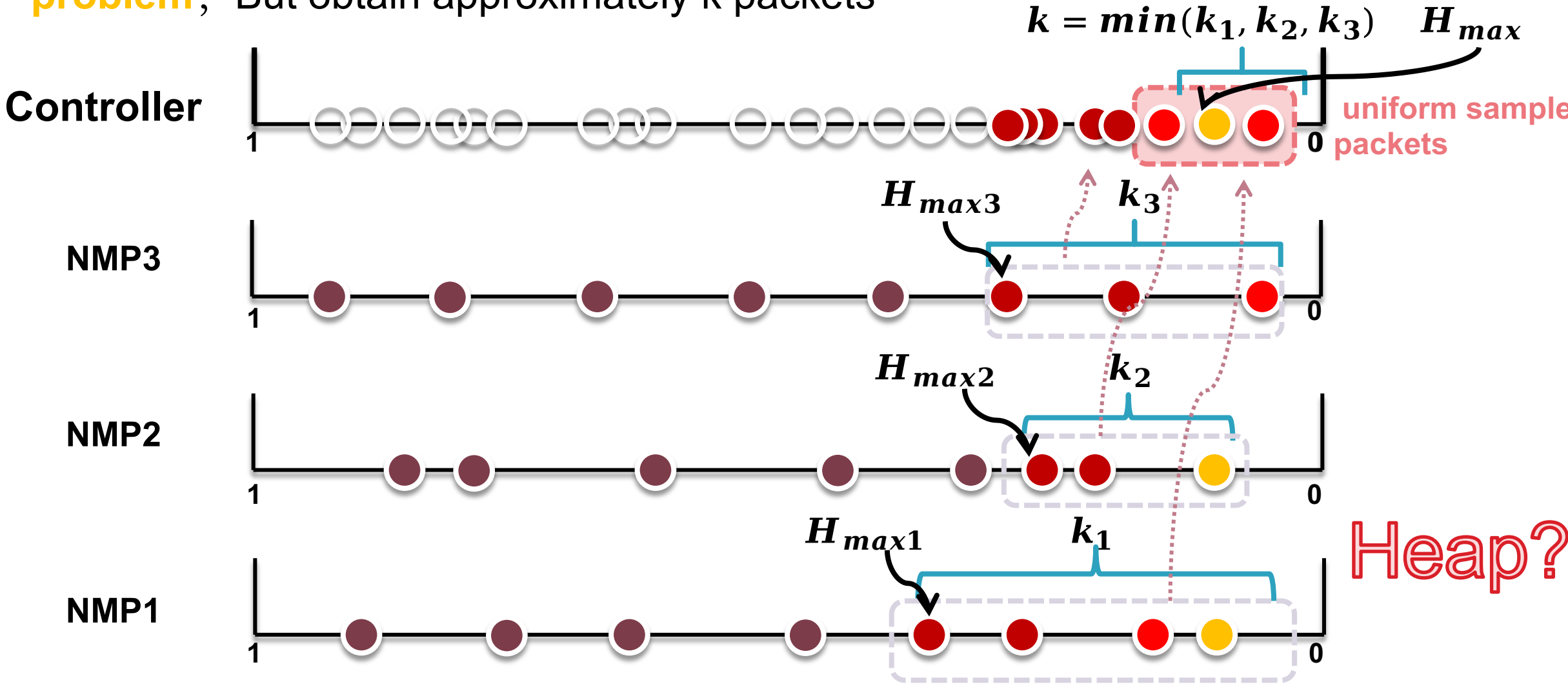
kth minimum hash value=0.3

Review K Minimum Value Algorithm(1/2)



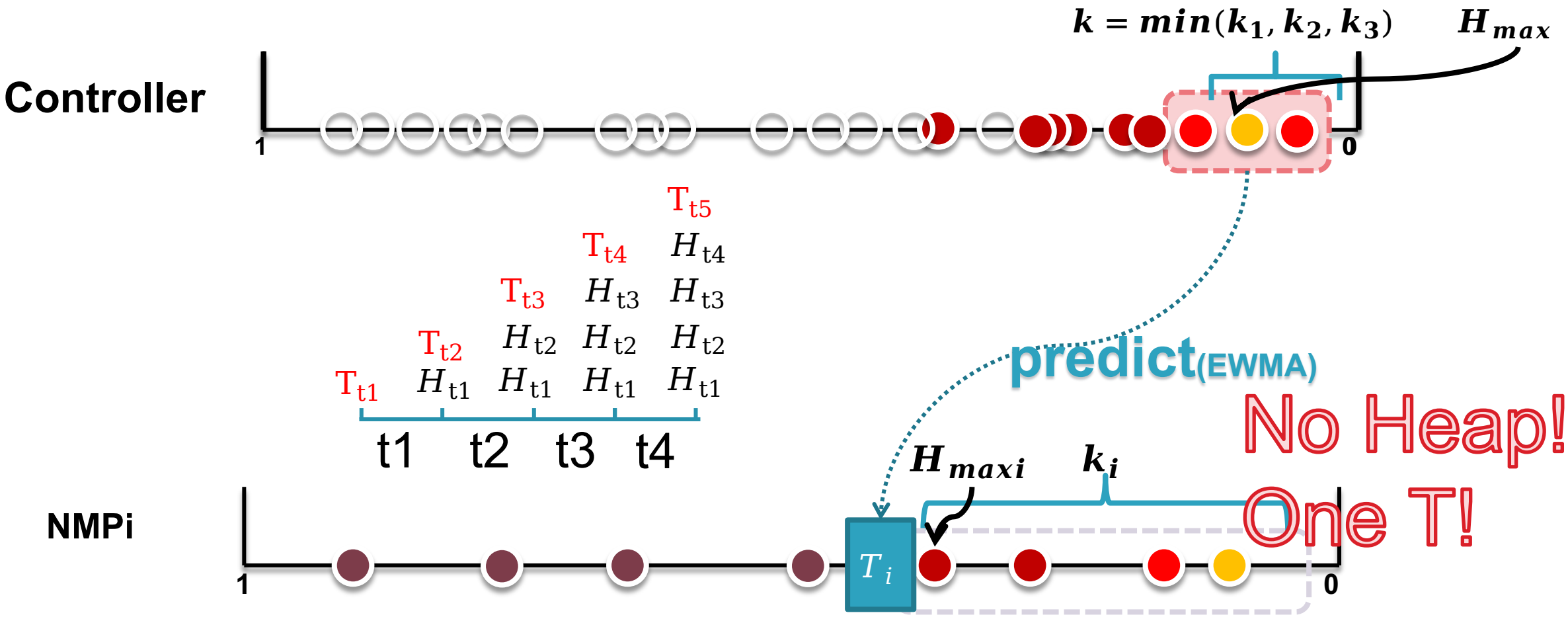
Insight1

- KMV algorithm for uniform sampled packet solves packets double counting problem; But obtain approximately k packets



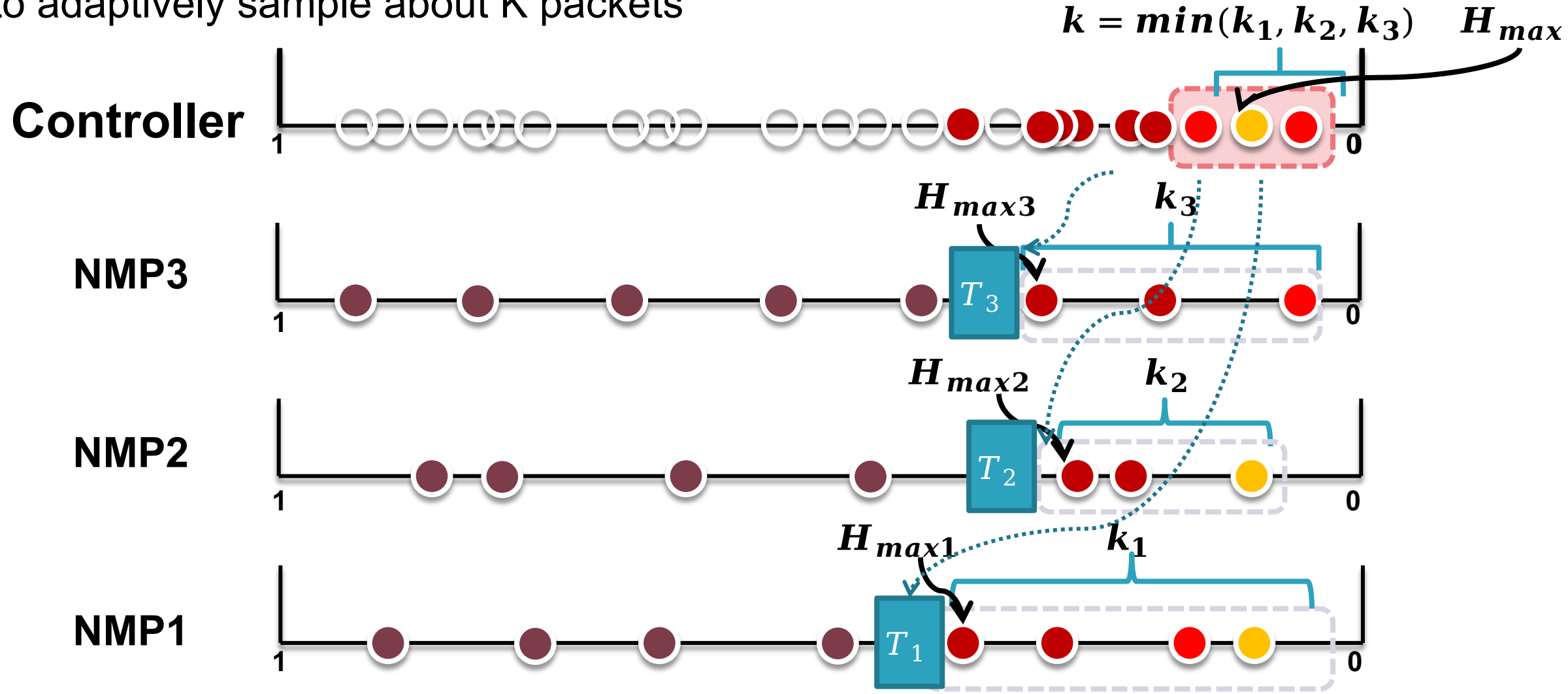
Insight2

- Control plane predict kth smallest hash value T(32bit) for each NMP to adaptively sample about K packets in next time interval.

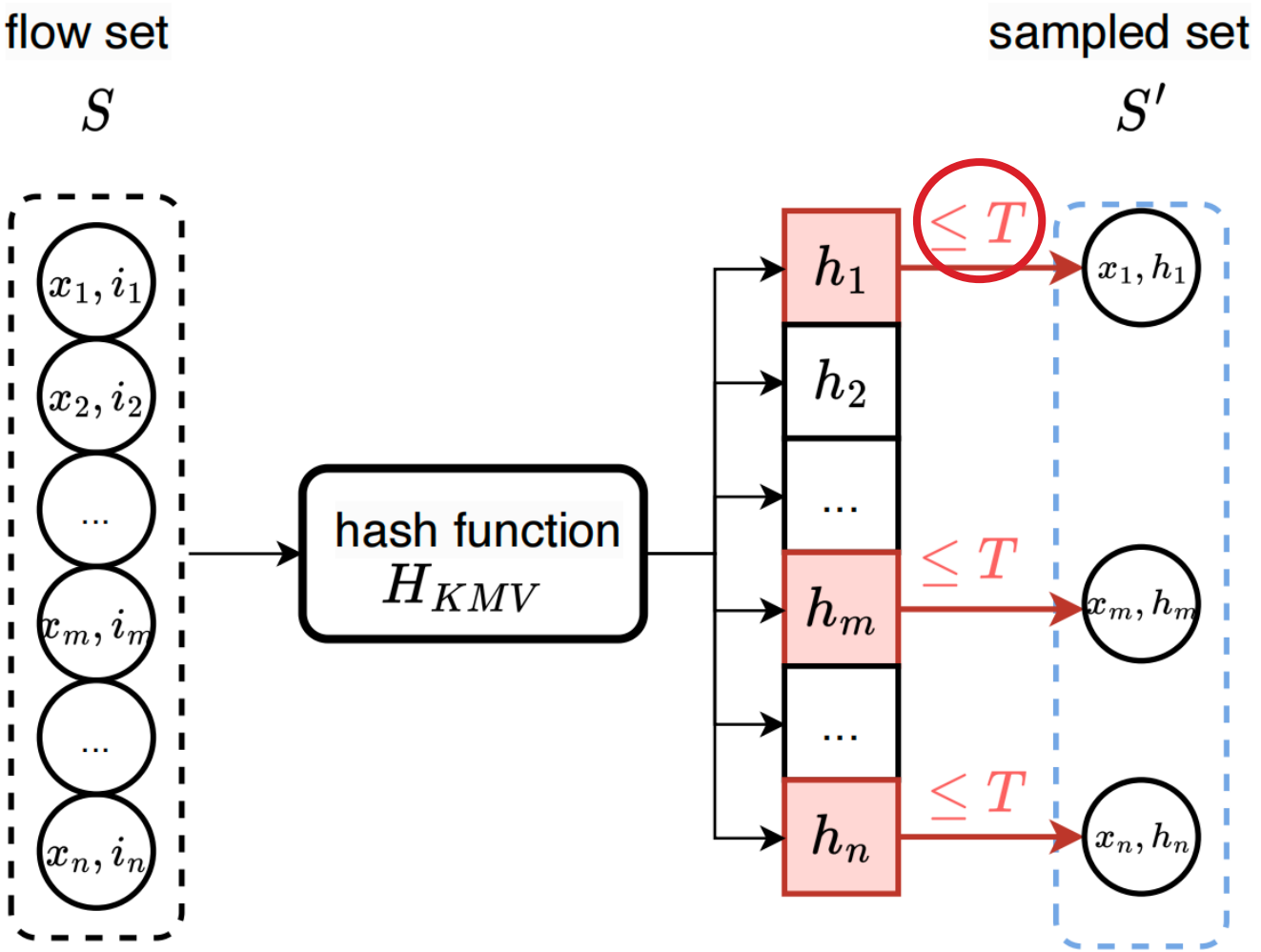


Insight2

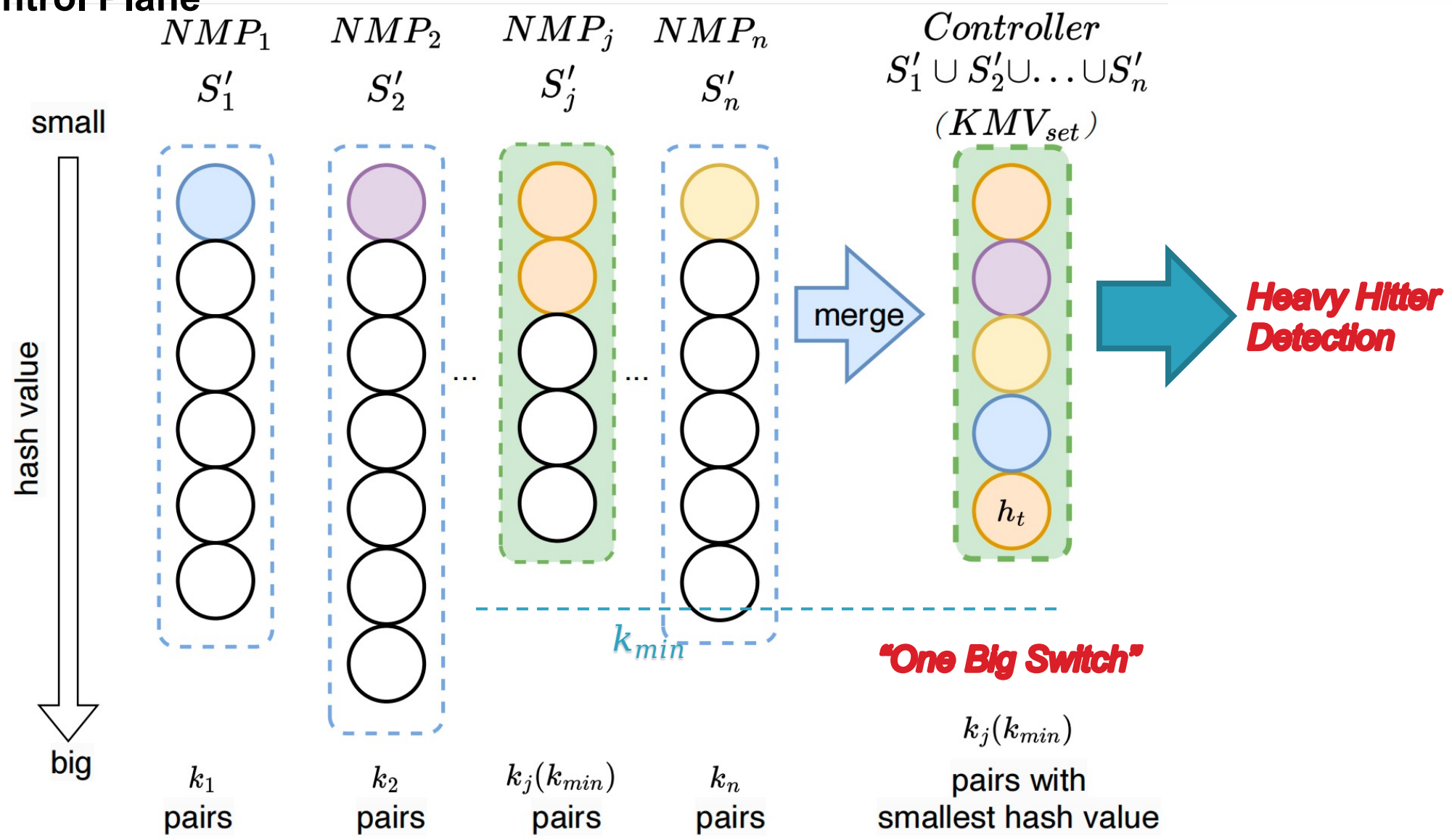
- Key Points2： Control plane predict kth smallest hash value (T) for each NMP to adaptively sample about K packets



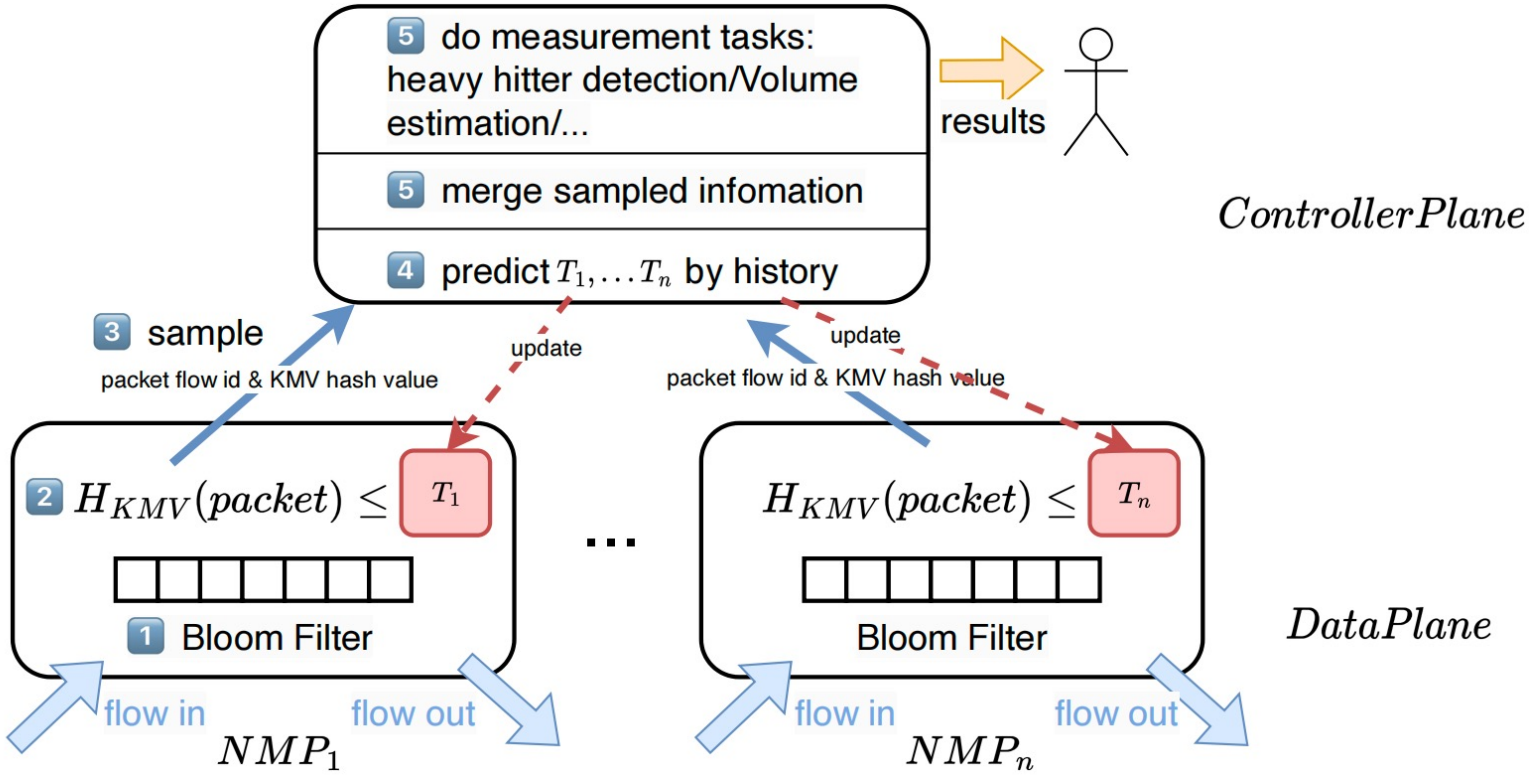
SAROS Dataplane



SAROS Control Plane



SAROS system design



KeyPoint1:KMV

① bloom filter： Partial de-duplication

② KMV： Global de-duplication

④ Predict new threshold

⑤ Merges sampled info

accuracy

KeyPoint2: Self-adaptive sampling

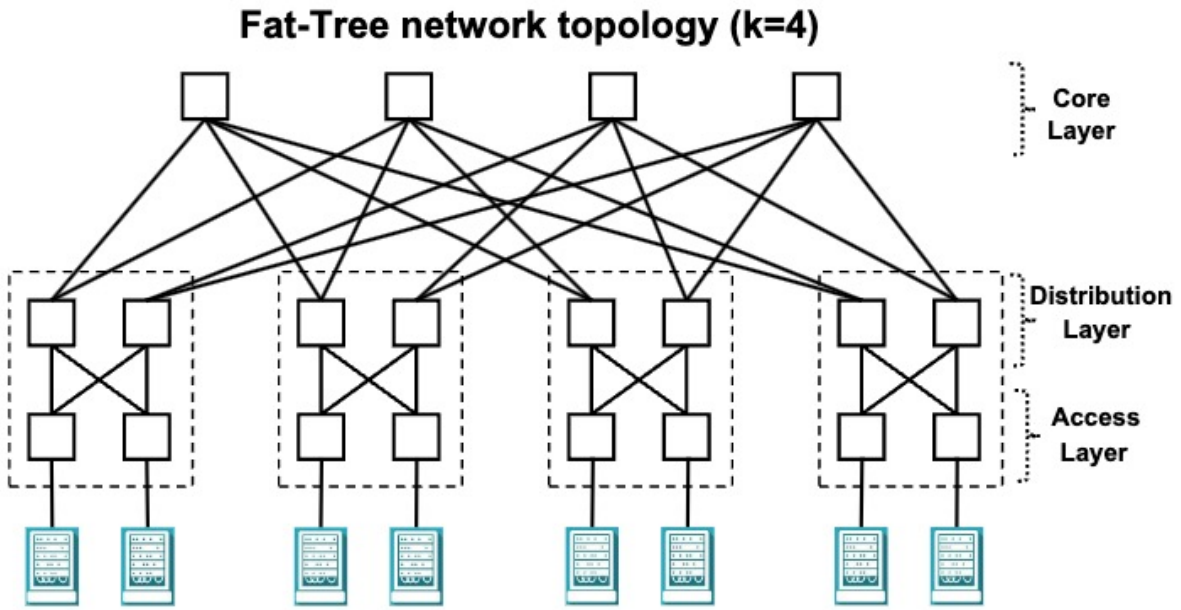
③ sampling

⑥ HH detection

memory use

Evaluation Setup

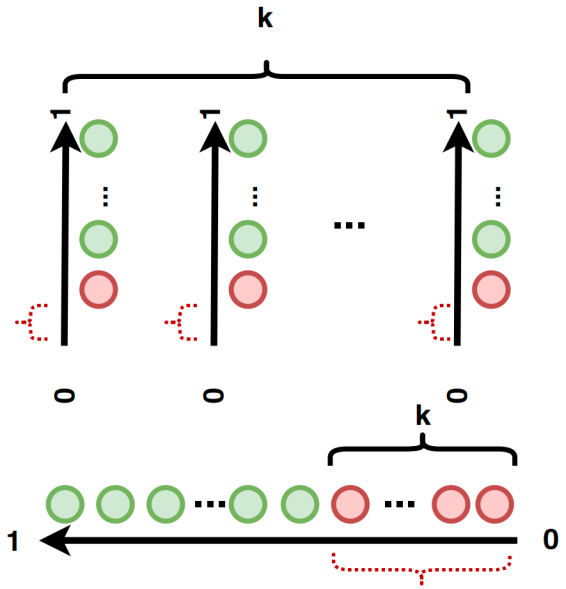
- **Dataset**:
  - mawi1，mawi2，univ1
  - time interval：500s，0.02v
- **Topology**:
  - Fattree with 4 pods
  - **bmv2+mininet**
- **Compared method**:
  - AROMA (existing routing oblivious method)

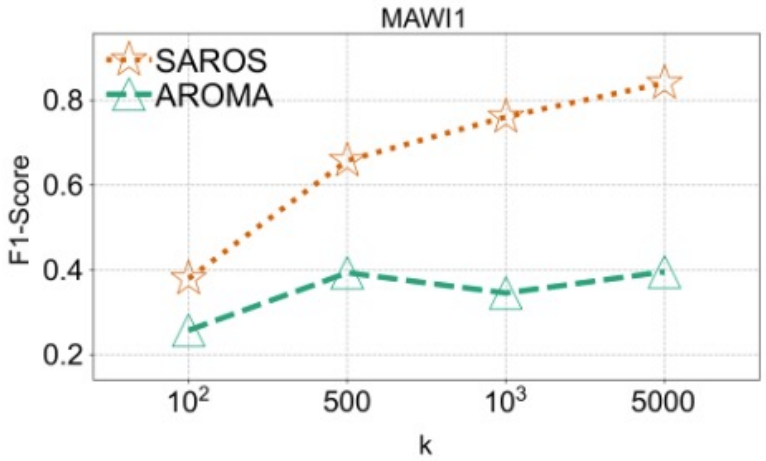


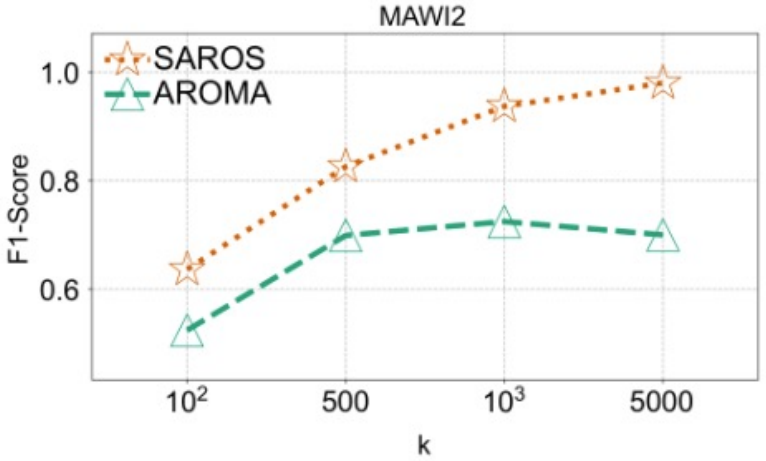
- **Evaluation Metrics**:
  - Network Wide Heavy Hitter Detection：F1-Score (the larger the better)
  - Memory Use: memory in data plane (the smaller the better)
  - Message Number: the message bettewn controll plane and data plane(the smaller the better)

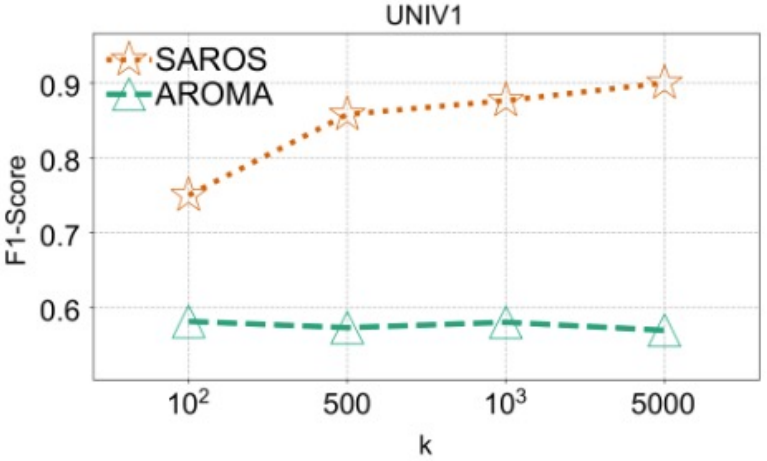
HH-detection Accuracy

- Network-Wide Heavy Hitter detection（F1-Score）
  - SAROS has improved the heavy hitter detection accuracy by 10~40%
  - Hash conflicts make the set of packets in the bucket less similar to the distribution of the set of packets passing through that NMP, resulting in inaccurate AROMA detection





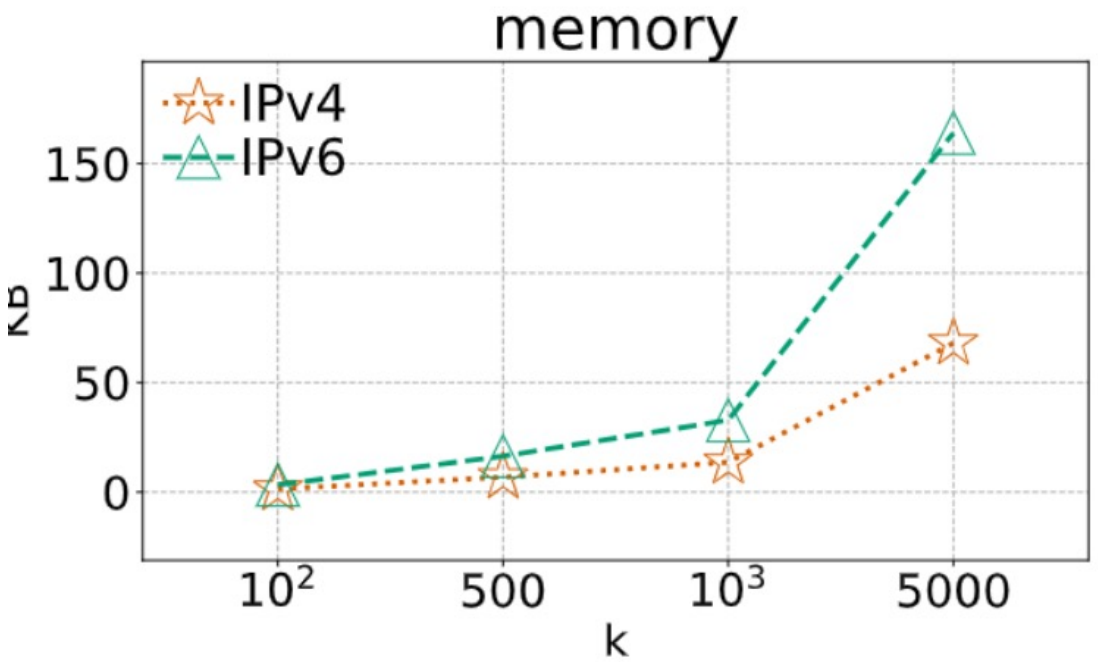




Memory Efficiency

- SAROS has reduced memory consumption by 99%
- AROMA stores k groups of <five-tuple, hash> on the data plane
- SAROS stores 32bit hash values on the data plane

IPv4:
when k=1200000
SAROS: 32bit
AROMA: 20MB

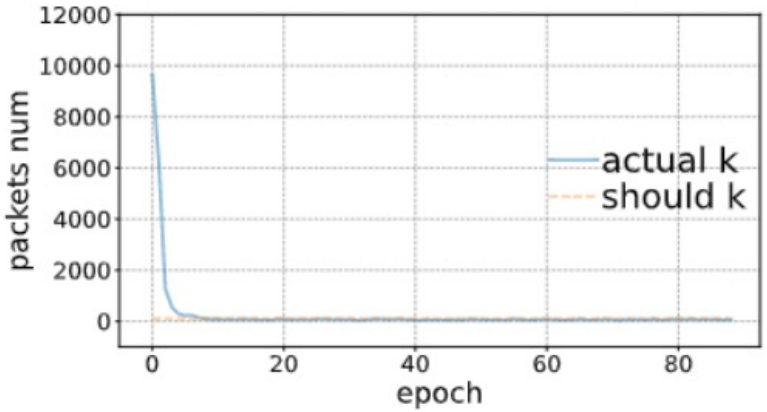


IPv6:
when k=400000
SAROS: 32bit
AROMA: 20MB

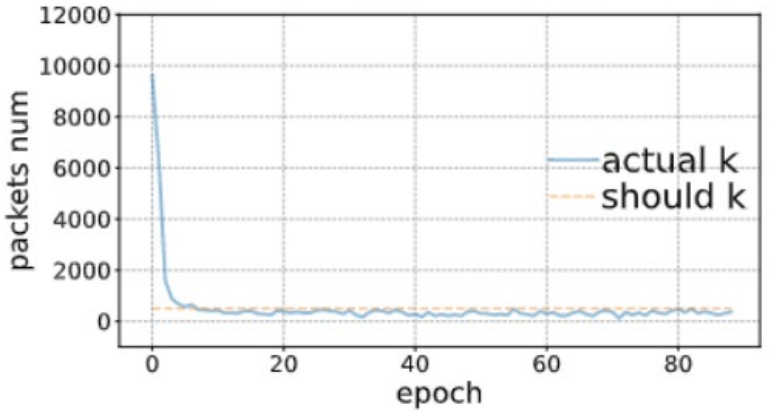
Status of convergence

- about 5 time intervals

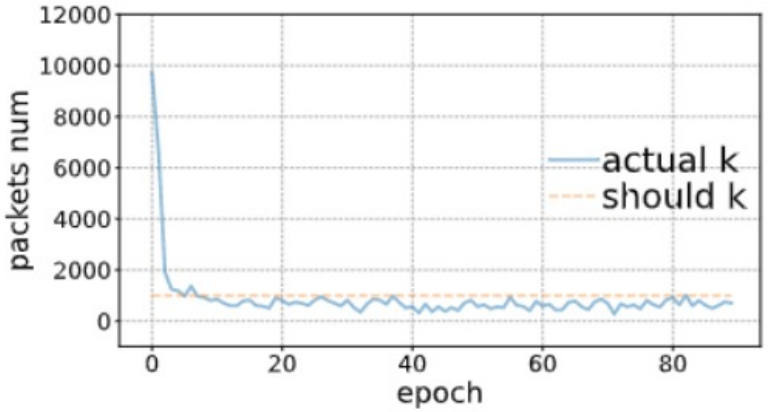
k=100



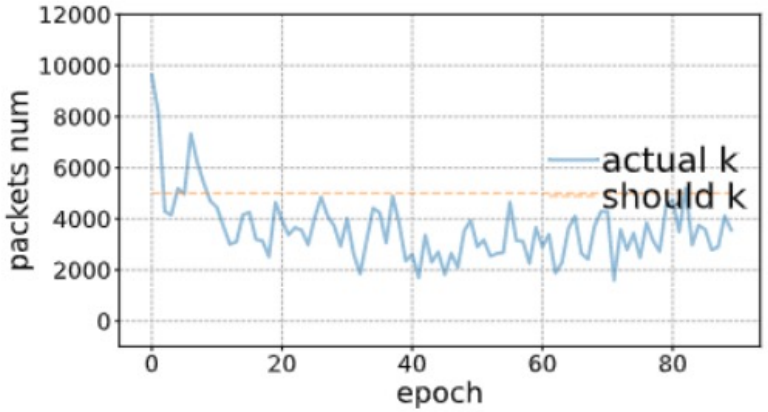
k=500



k=1000

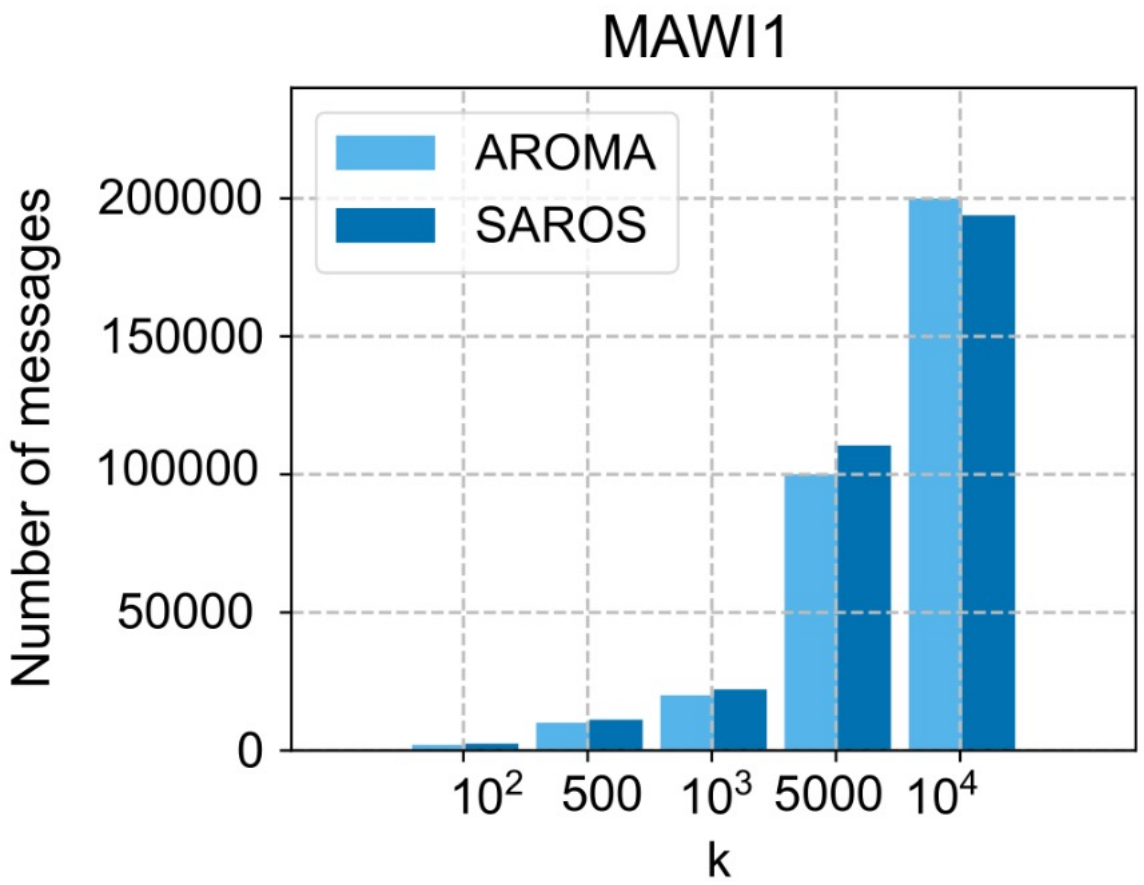


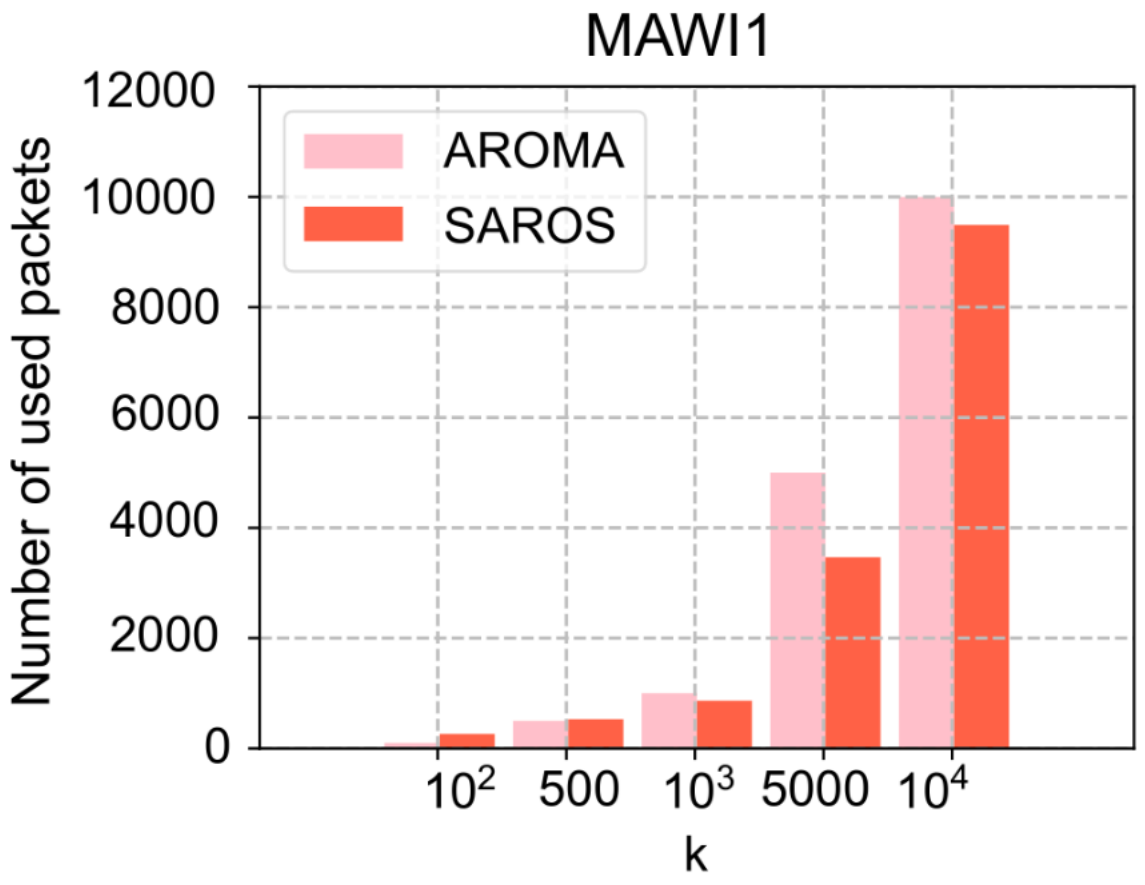
k=5000



Message Num

- SAROS has lower message number than AROMA





Contributions & Future Work

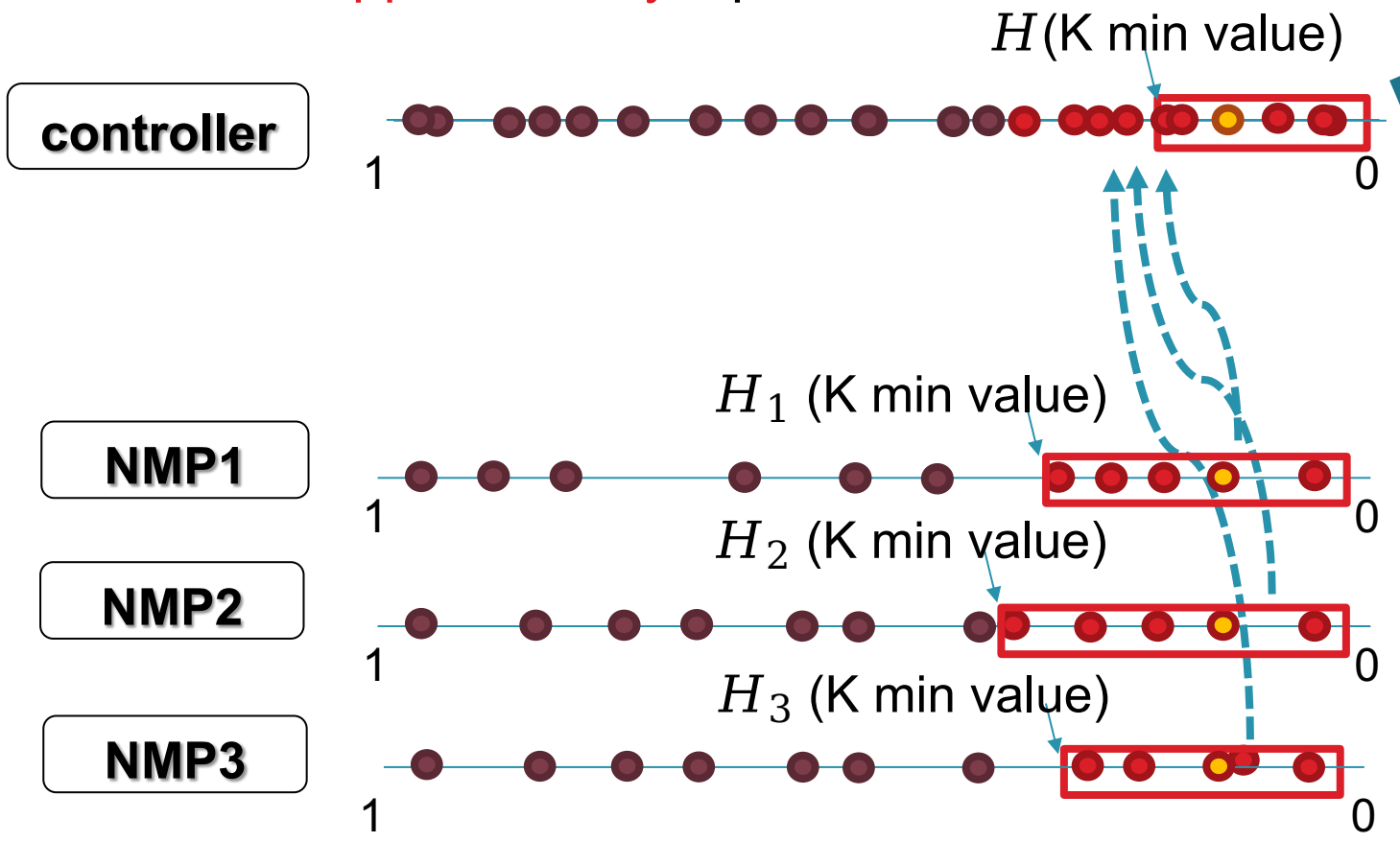
- **Contribution**
  - We propose a double-counting-free sampling mechanism for network-wide heavy-hitter detection, which ensures detection accuracy in routing oblivious cases.
  - We propose a dynamic sampling threshold prediction method in the SAROS control plane, which calculates appropriate sampling thresholds for each NMP according to network traffic distribution.
  - Compared with existing solutions, SAROS has improved the heavy hitter detection accuracy by 10~40%, and reduced memory consumption by 99%.
- **Future Work**
  - SAROS now uses EWMA to simply predict sampling threshold and we will try to explore more prediction methods in the future, such as transformers.
  - In the future, we will expand SAROS to more measurement tasks like superspreader, DDos .....

Thanks

Q&A

SAROS

- Key Points1:
  - KMV algorithm for uniformly sampling solves packets double counting problem
  - But obtain approximately k packets



Volume Estimation:

$$\widehat{V} = (K-1)/H$$

Heavy Hitter Detection:

Global Threshold:

$$T_{global} = \widehat{\theta V}$$

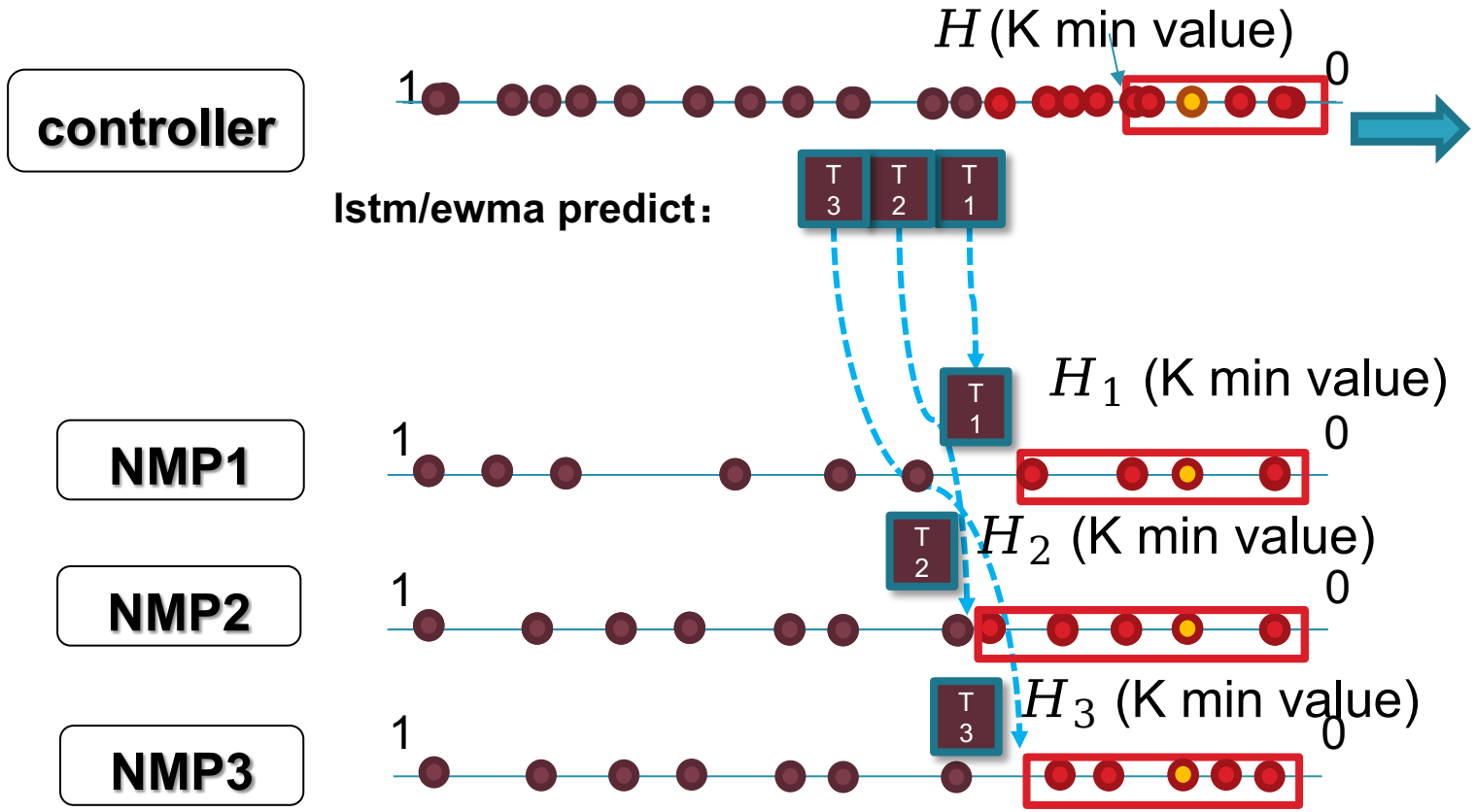
Heavy Hitter:

Flow frequency larger than global threshold 👉Heavy Hitter

how to sample kth smallest hash in programmable switch?

SAROS

- Key Points 2:
  - Control plane predict kth smallest hash value (T) for each NMP to adaptively sample about K packets
  - Guaranteed small dataplane memory use



sample K_1 packets
max hash=H_1

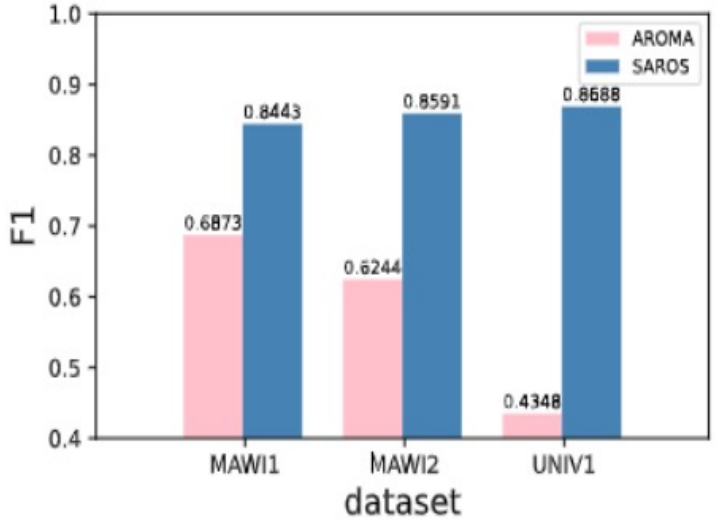
should sample *K* packets
should max hash is $H_1^{'} = \frac{KH_1}{K_1}$

use $H_1^{'}$ sequence to predict T_1 for sampling kth smallest hash

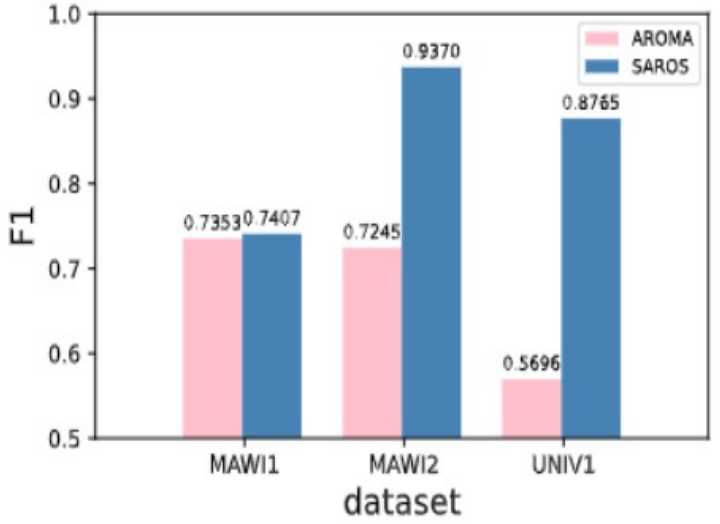
Sensitivity Analysis

- **Sensitivity Analysis(Dataset)**
  - AROMA performs the best in MAWI1, followed by MAWI2, and performs the worst in UNIV1 dataset.
  - SAROS exhibits stronger stability in different

- **Sensitivity Analysis(θ)**
  - SAROS has higher F1 scores than AROMA
  - The Error% values of SAROS and AROMA are similar.



θ=0.05



θ=0.01