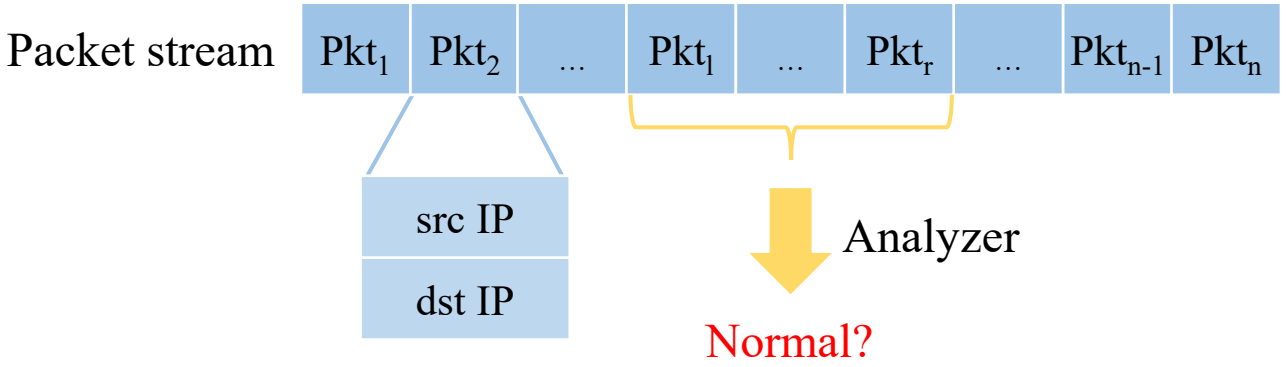
Sans: Streaming Anonymized Network Sensing

Ketai Zhao, *Yuhang Zhou*, Hongxu Pan, Zhibin Wang, Sheng Zhong, Chen Tian

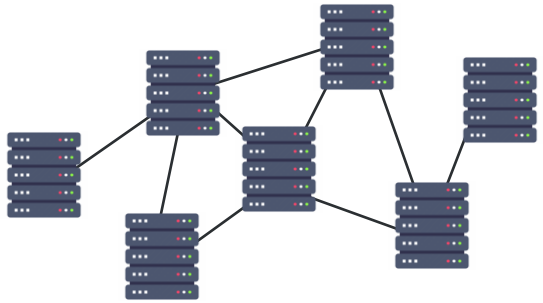
State Key Laboratory for Novel Software Technology, Nanjing University

Background

Large-scale network sensing



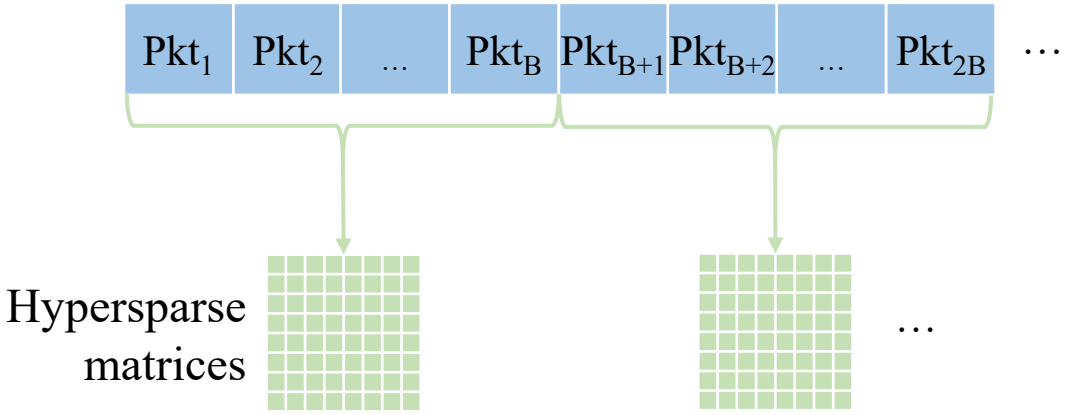
Social Recommendation Systems



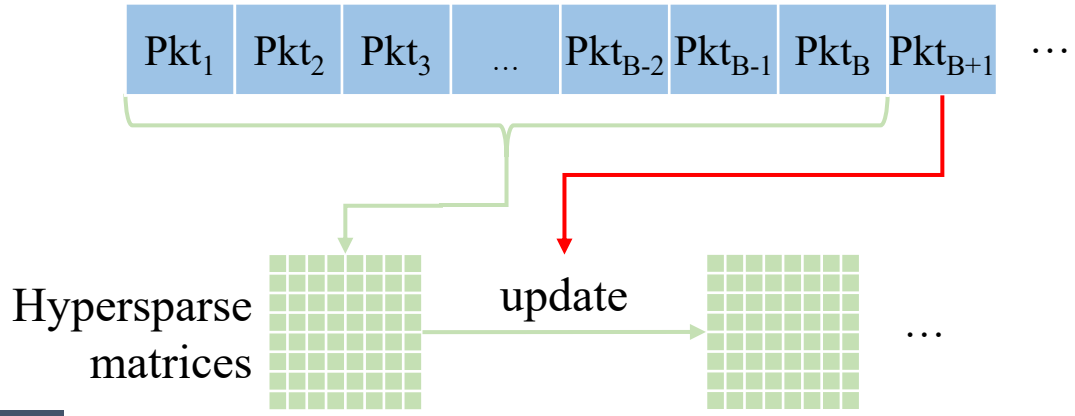
Network Traffic Analysis

Motivation and Challenge

Blocking:



Streaming:



# of updates ≈ # of packets
Existing data structures (CSR) is not efficient to perform such updates:

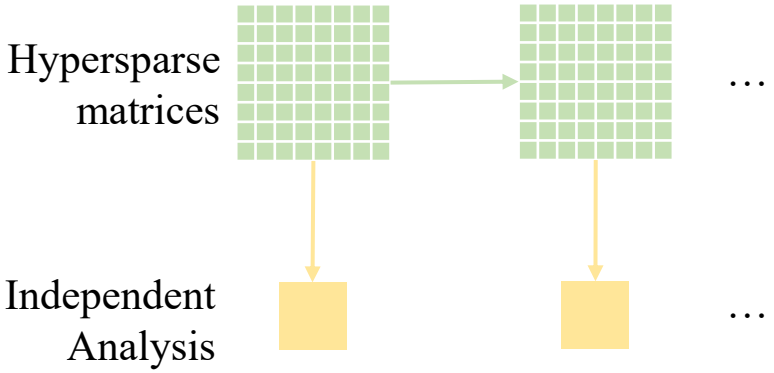
	IP 0	IP 1	IP 2	IP 3	…
IP 0		1		2→1	
IP 1		0→1			
IP 2			1		
IP 3	3		1		
…					

values	1	2	1	3	1
column indices	1	3	2	0	2
row offsets	0	2	3		

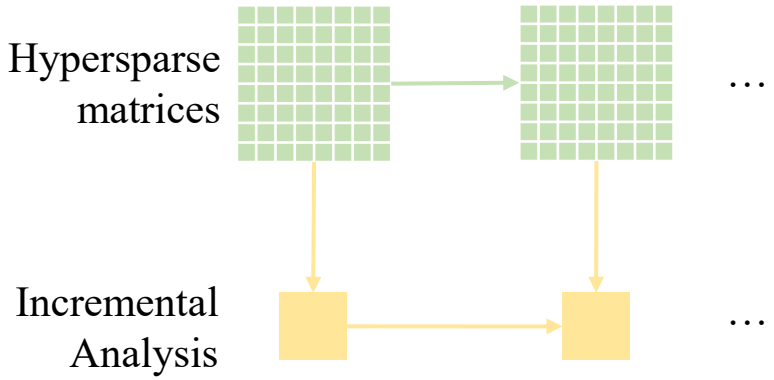
Dynamic, efficient, and compressed data structure

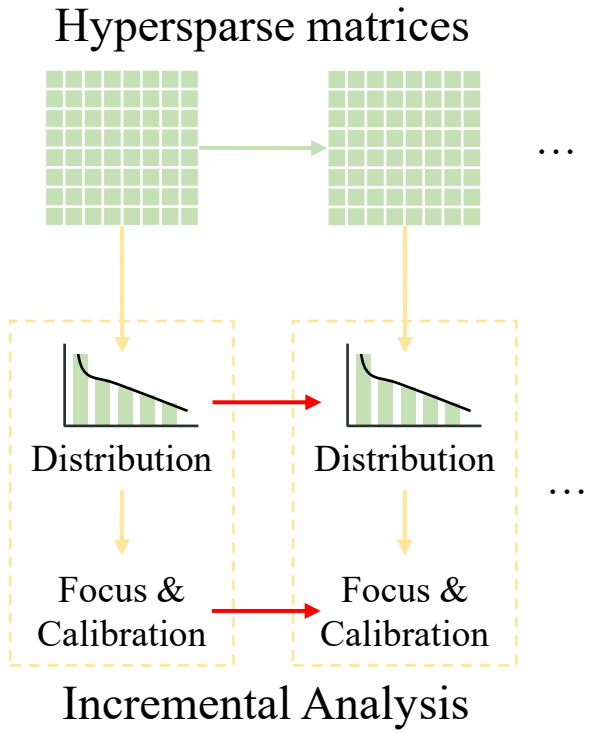
Motivation and Challenge

Independent:



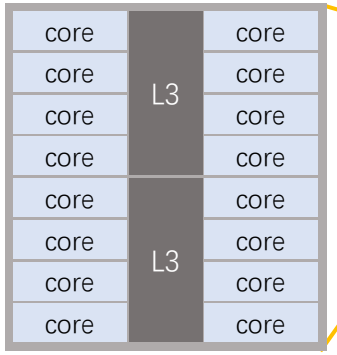
Incremental:



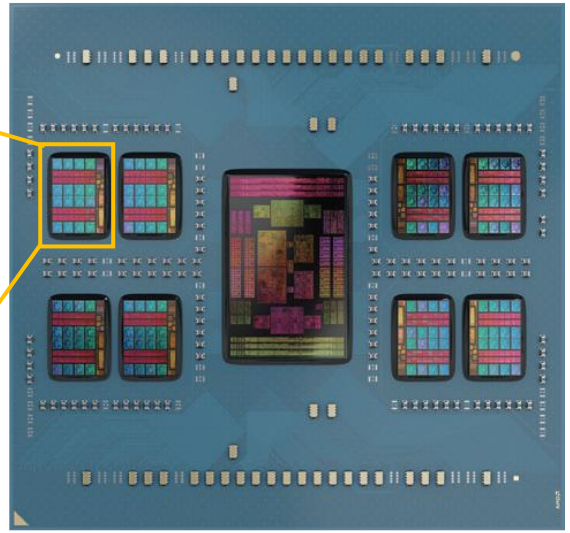


Incremental analysis algorithm

Motivation and Challenge



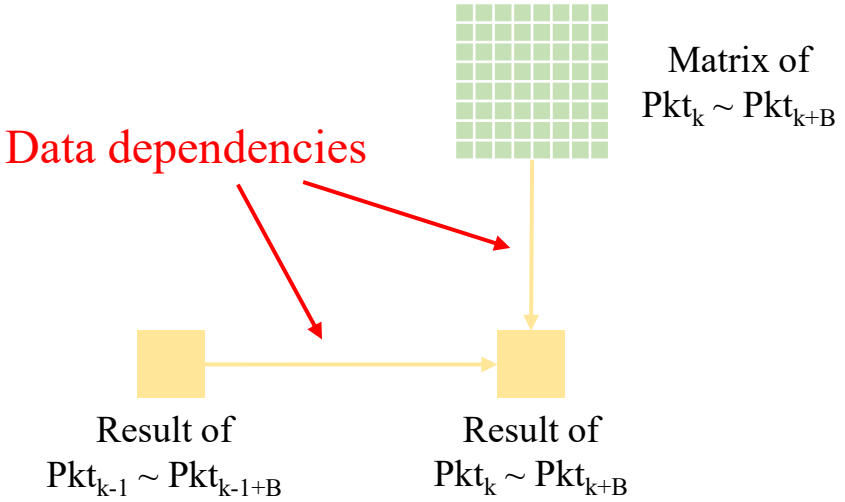
16 cores per die



8 dies

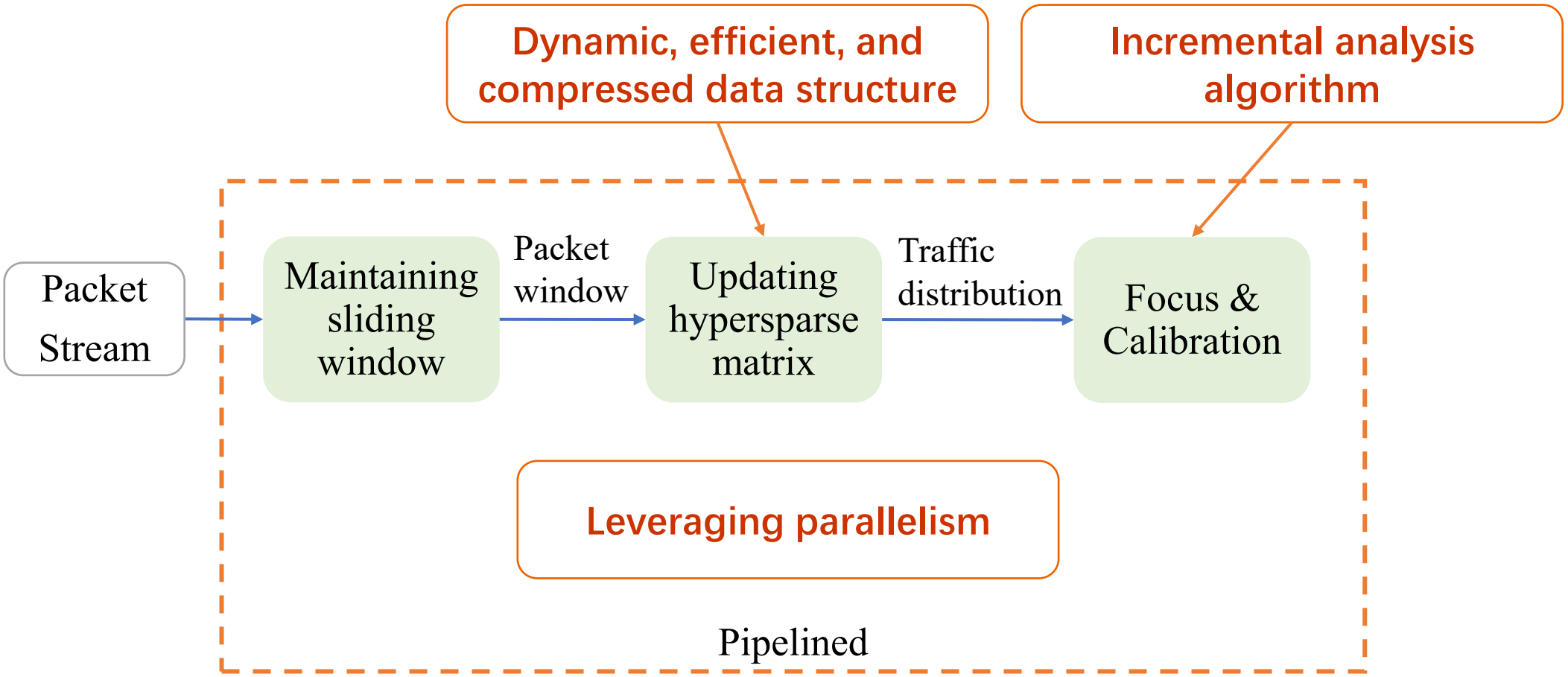
(AMD EPYC 9004 series processors)

Modern CPUs have more than 100 cores

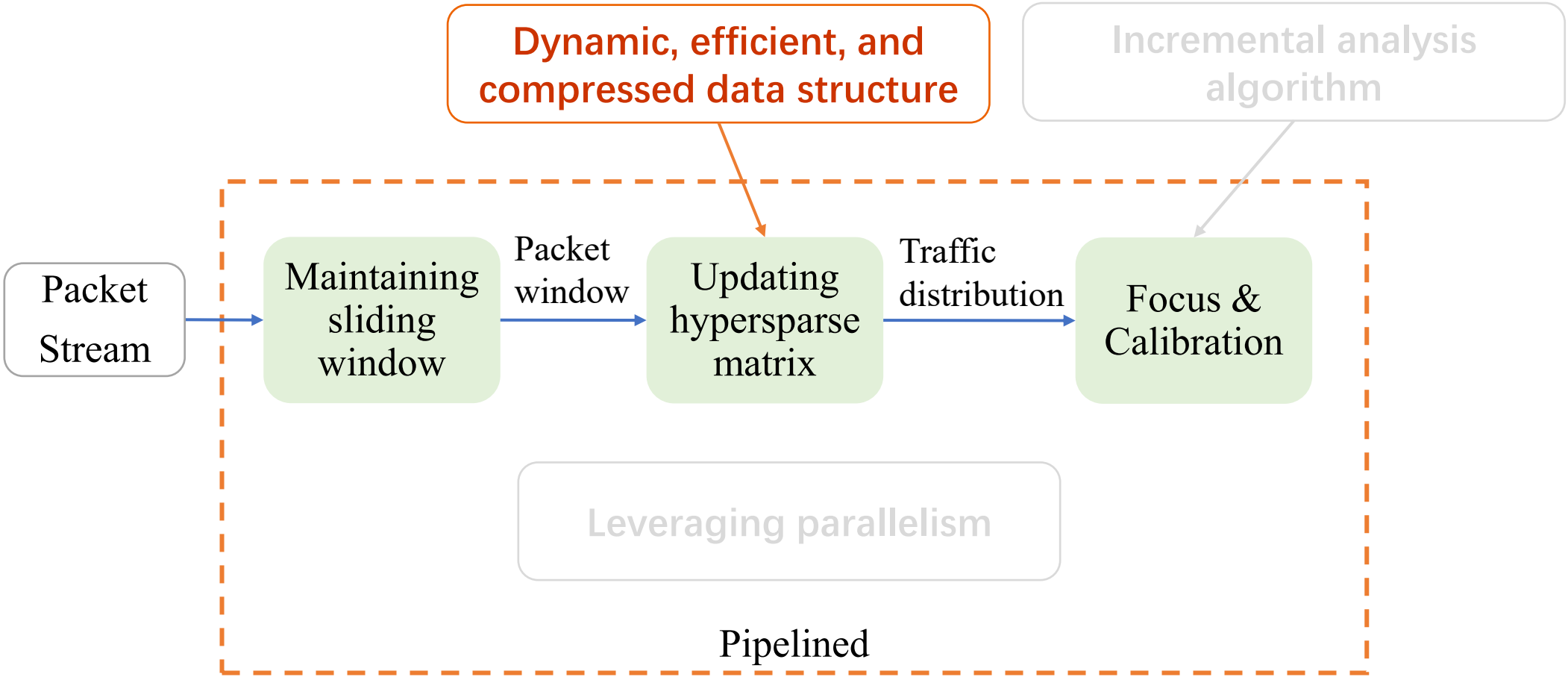


Leveraging parallelism

Sans: Streaming Anonymized Network Sensing



Sans: Streaming Anonymized Network Sensing



Data structure

(IP0, IP1)	(IP3, IP1)	(IP3, IP2)	…	(IPx, IPy)	…
1	3	4		2	

(a) Edge list

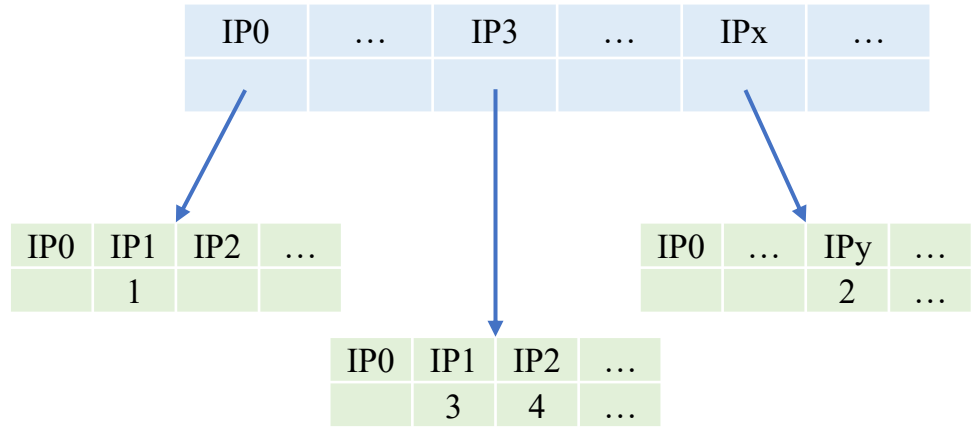
(IP0, IP1)	…	(IP3, IP1)	(IP3, IP2)	…	(IPx, IPy)	…
1		3	4		2	

(b) HashTable for edge

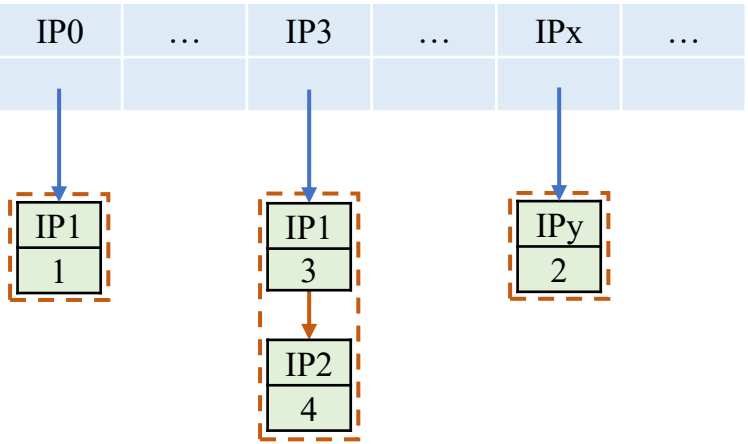
Values	1	3	4	…	2	…
Column Indices	1	1	2	…	y	…
Row offset	0	…	1	…	z	…

(z = # of values in row 0 ~ x-1)

(c) CSR



(d) HashTable of HashTable



(e) HashTable of list

Data structure

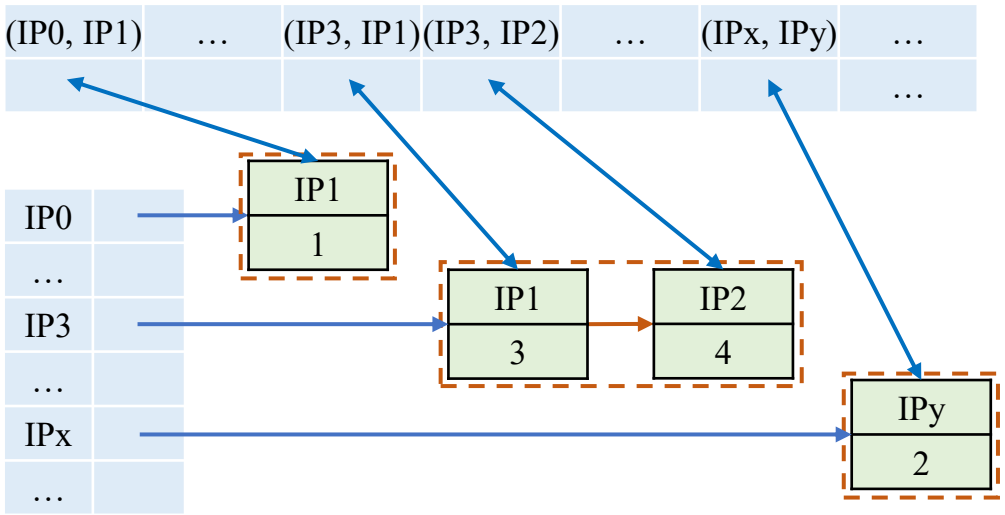
Category	Structure	Type	Add	Remove	Memory
Direct	Edge list	Static	O(|E|)	O(|E|)	Bounded
	HT for edge	Dynamic	O(1)	O(1)	Bounded
Hierarchical	CSR	Static	O(|E|)	O(|E|)	Bounded
	HT of HT	Dynamic	O(1)	O(1)	Unbounded
	HT of list	Dynamic	O(d(v))	O(d(v))	Bounded

Not support neighbor access

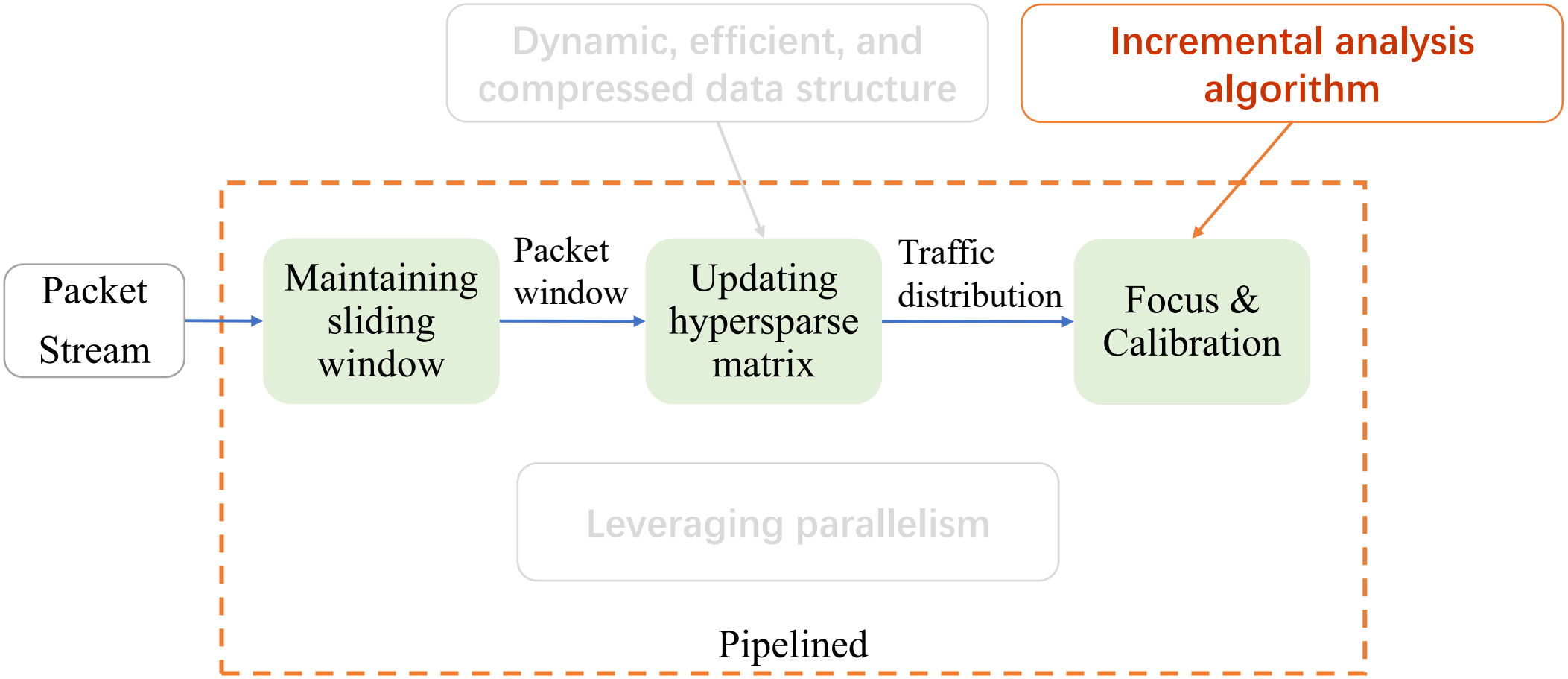
Support neighbor access

Combining HT for edge and HT of list:

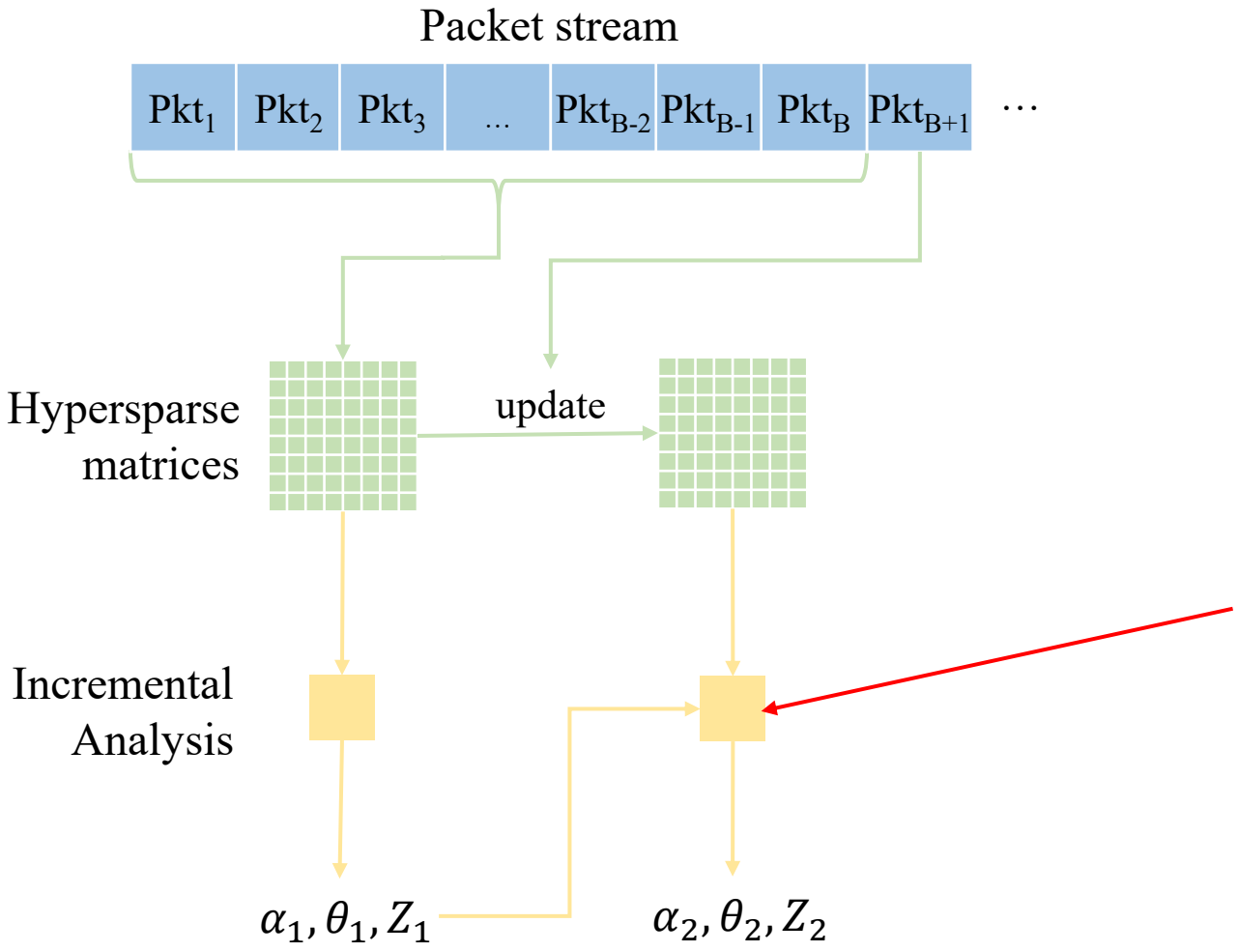
Add/Remove: O(1)
Support Accessing Neighbors



Sans: Streaming Anonymized Network Sensing



Incremental analysis algorithm



Zipf-Mandelbrot distribution:

$$P(d) = \frac{1}{Z}\left(\frac{1}{d+\theta}\right)^{\alpha}$$

Minimize:

$$MSE = \frac{1}{m}\sum_{d=1}^{m}(P(d) - y_d)^2$$

Gradient descent:

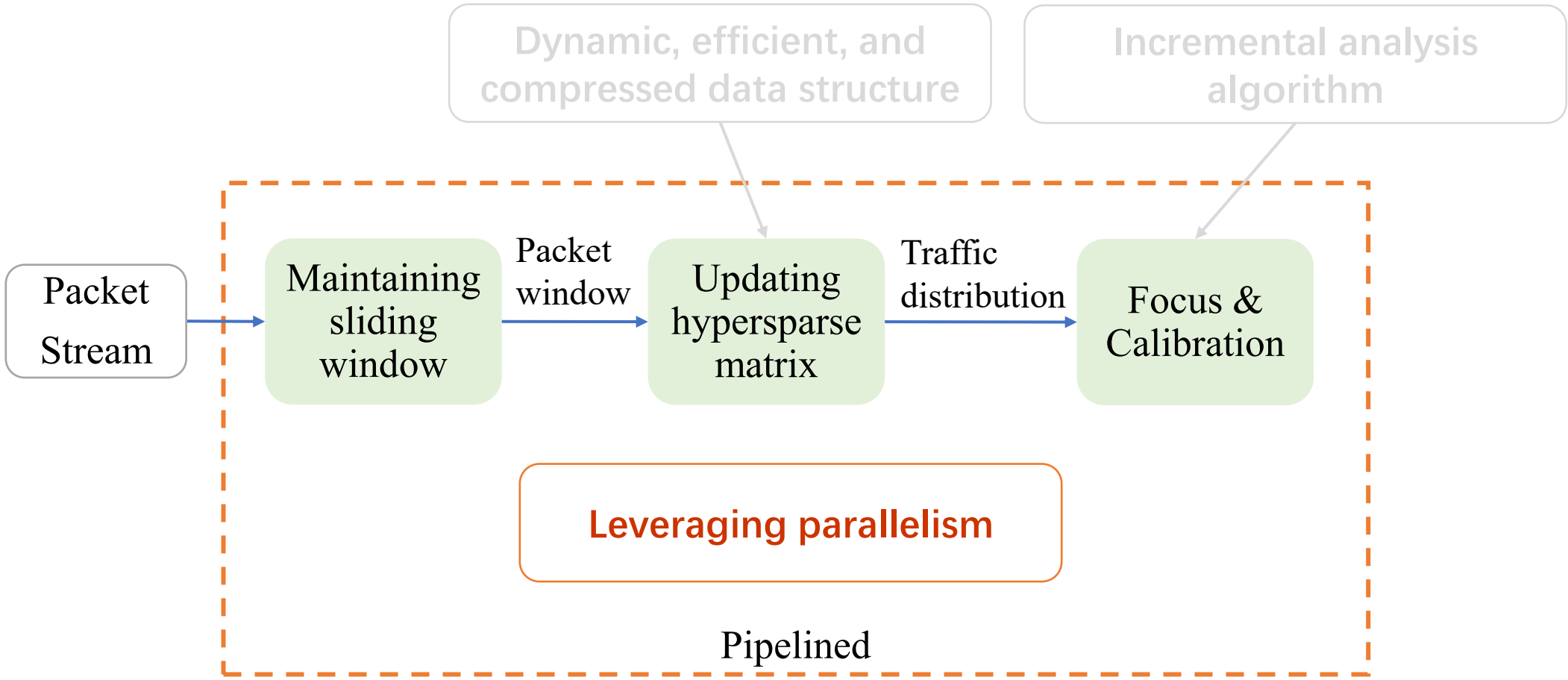
$$\alpha \leftarrow \alpha - \gamma \frac{\partial}{\partial \alpha} P(d)$$

$$\theta \leftarrow \theta - \gamma \frac{\partial}{\partial \theta} P(d)$$

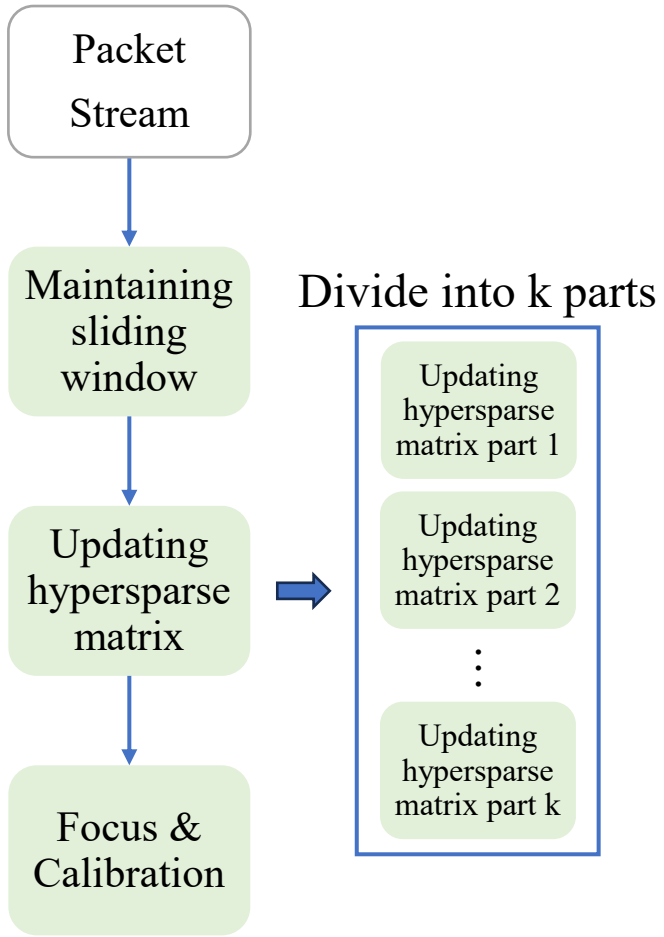
$$Z \leftarrow Z - \gamma \frac{\partial}{\partial Z} P(d)$$

(γ is a preset parameter)

Sans: Streaming Anonymized Network Sensing



Leveraging Parallelism



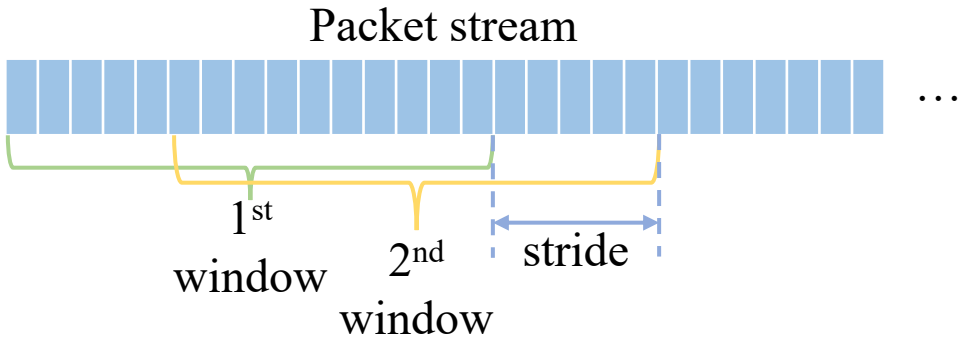
Use multithreading and pipeline to accelerate computing:

Thread 1	Maintaining sliding window	Maintaining sliding window	Maintaining sliding window	…	
Thread 2		Updating hypersparse matrix part 1	Updating hypersparse matrix part 1	Updating hypersparse matrix part 1	…
Thread 3		Updating hypersparse matrix part 2	Updating hypersparse matrix part 2	Updating hypersparse matrix part 2	…
⋮		⋮	⋮	⋮	
Thread k+1		Updating hypersparse matrix part k	Updating hypersparse matrix part k	Updating hypersparse matrix part k	…
Thread k+2			Focus & Calibration	Focus & Calibration	Focus & Calibration …

Experimental Evaluation

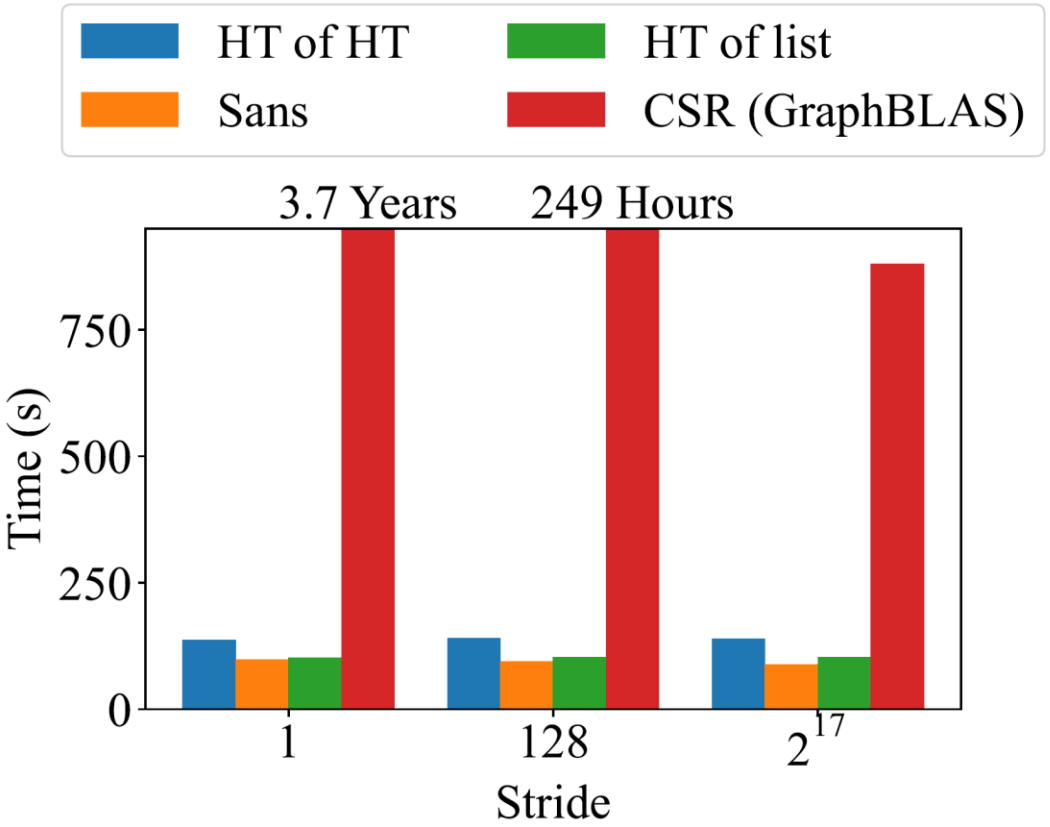
- **Hardware**
  - Intel Xeon Gold 6330 CPU (56 cores, 2.0GHz)
  - 512GB memory
- **Comparator**
  - CSR: GraphBLAS
- **Dataset**
  - Provided by GraphChallenge
  - Consists 2^{30} synthetic packets with random data

Experiment 1: Sans V.S. SOTA

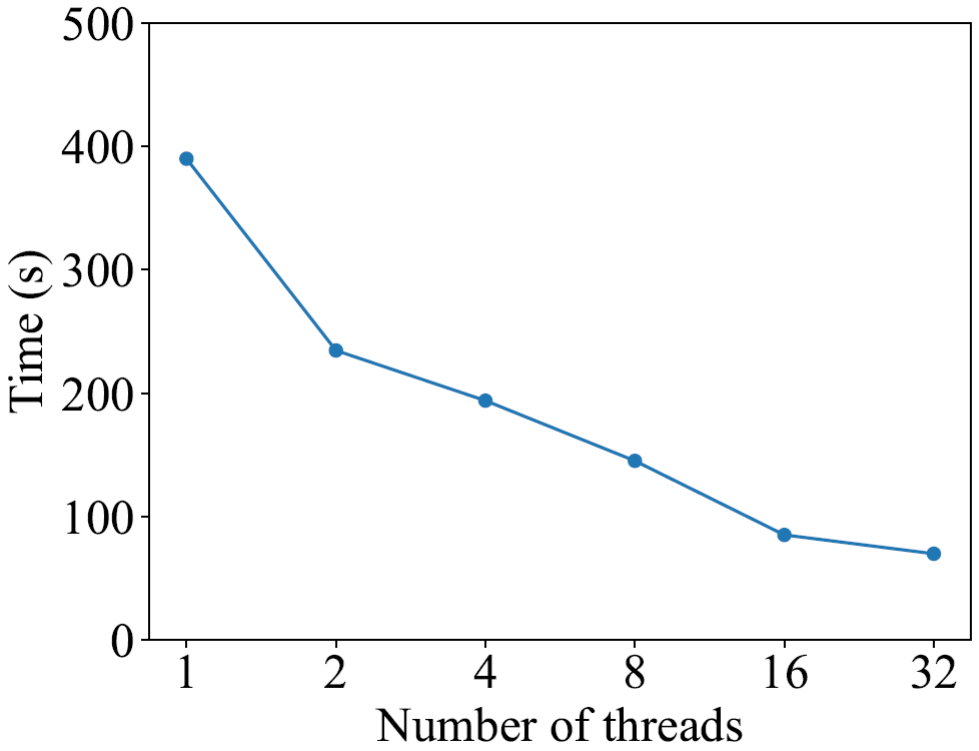


Sans outperforms

- HT of list by 1.1×
- HT of HT by 1.6×
- CSR by a million times

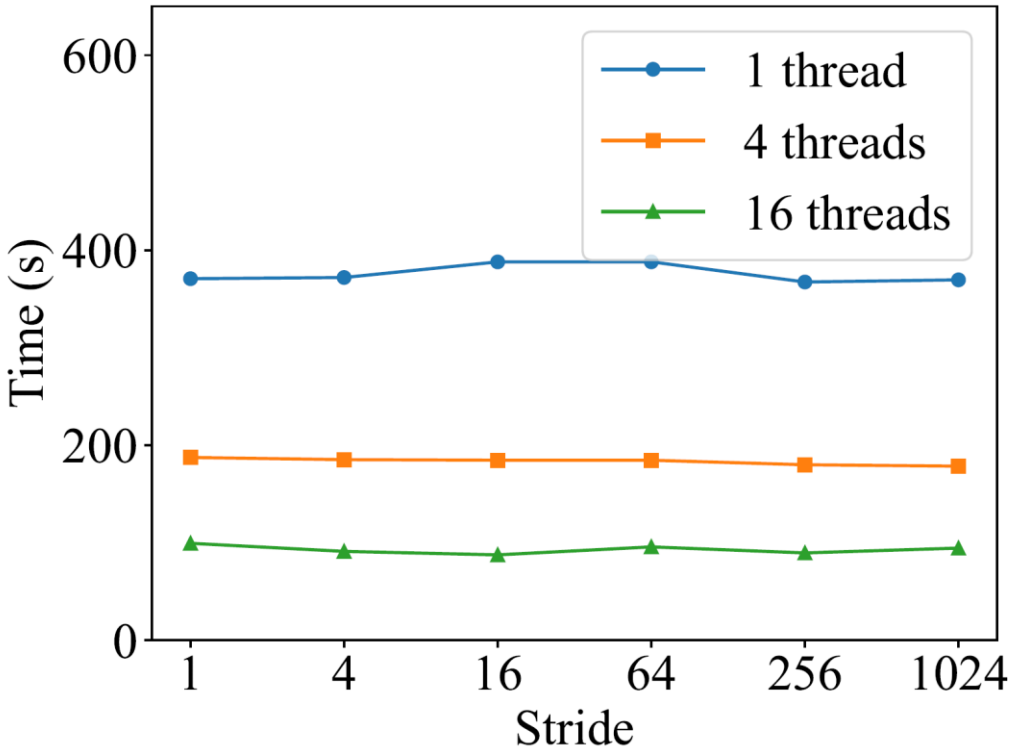


Experiment 2: Scalability



7.8× speedup from 1 to 32 threads

Experiment 3: Varying Stride



Our incremental analysis algorithm avoids redundant computation:

Only 5% faster when stride=1024 compared with stride=1

Conclusion

- Propose Sans, streaming anonymized network sensing system
  - Dynamic, efficient, and compressed data structure
  - Incremental calibration algorithm
  - Parallelization
- Outperforms all SOTA and has good scalability

THANK YOU

yuhangzhou@smail.nju.edu.cn