Pando: Extremely Scalable BFT Based on Committee Sampling

Xin Wang

Shandong University

Haochen Wang

Tsinghua University

Haibin Zhang

Yangtze Delta Region Institute of Tsinghua University

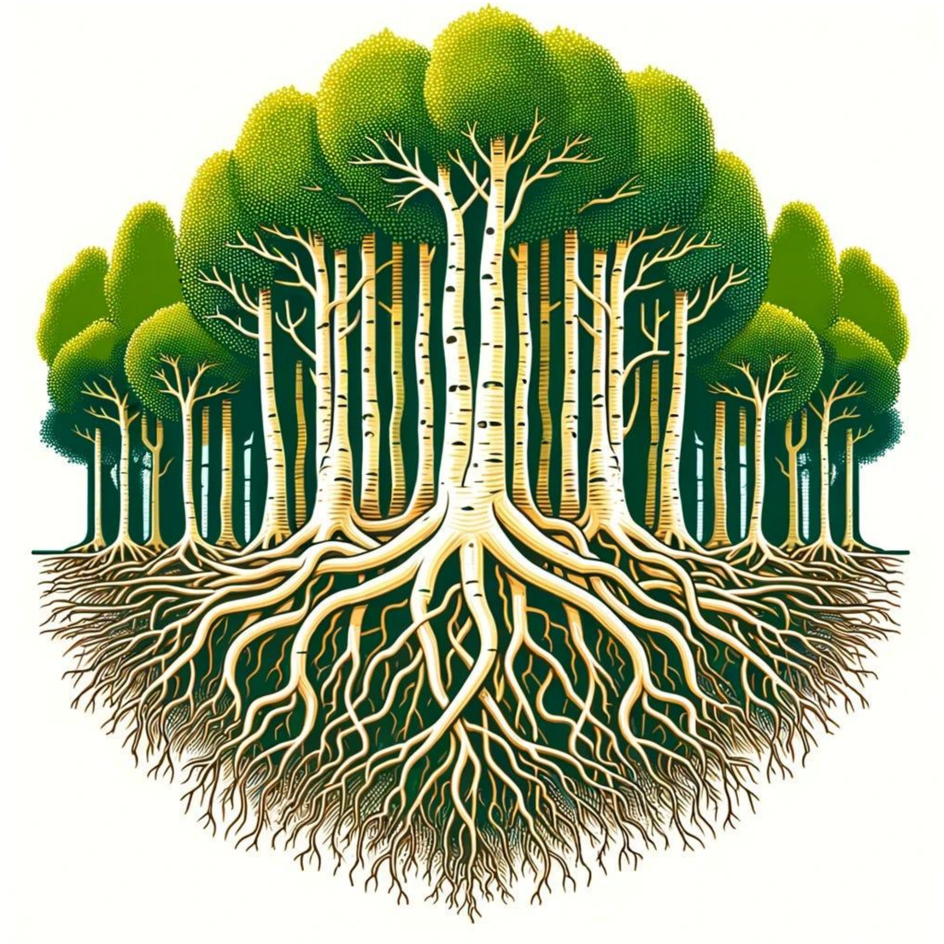
Sisi Duan

Tsinghua University

NDSS 2026

Pando is the world's largest tree...

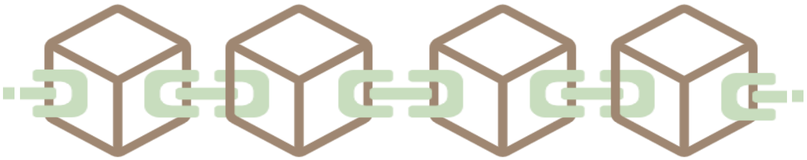
- An ancient aspen, found in Utah
- All trunks grow from one enormous root system
- Whose name means “I spread out”



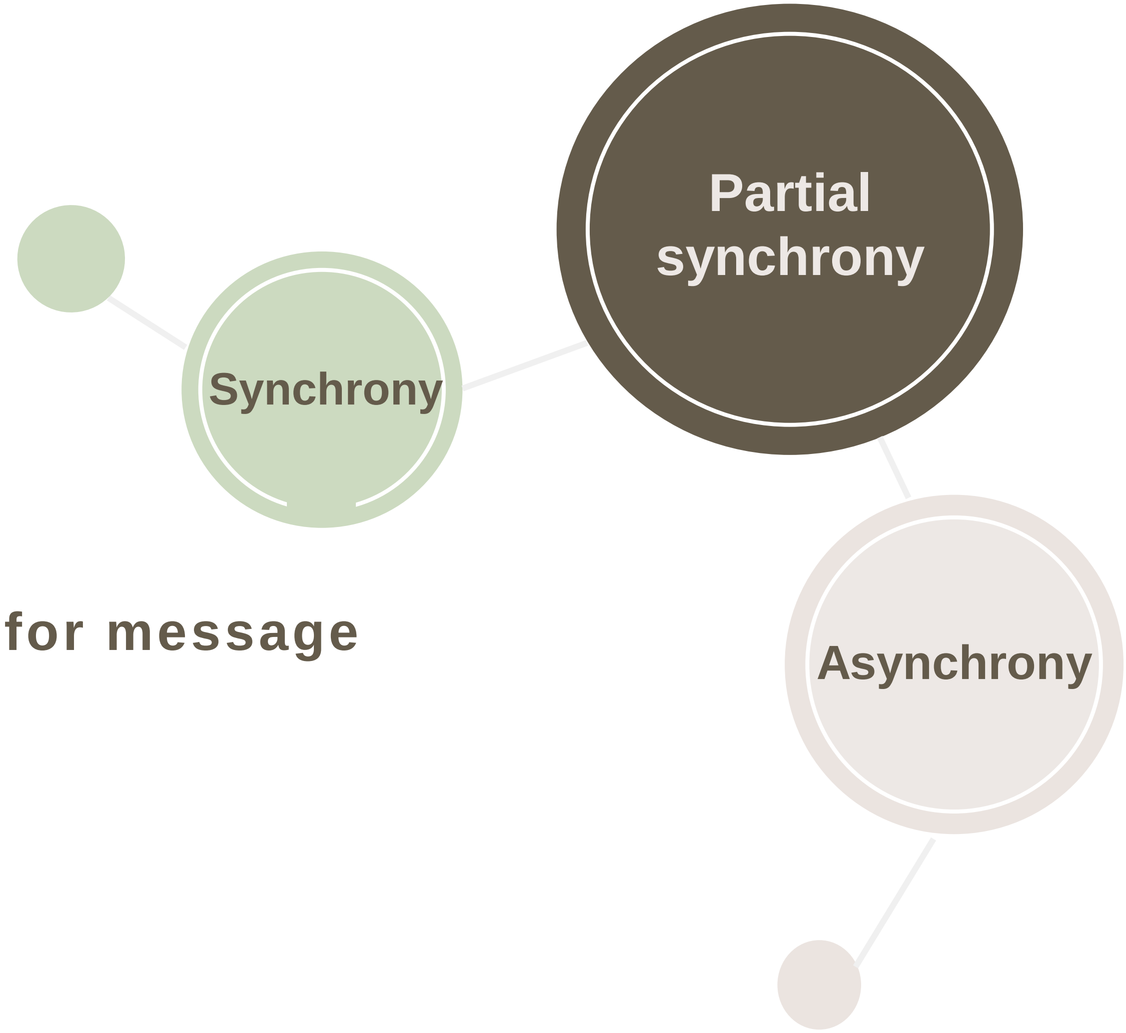
The world’s largest organism:
https://www.youtube.com/watch?v=zC8abuKnr90

Byzantine Fault Tolerance (BFT)

- Building block for blockchains

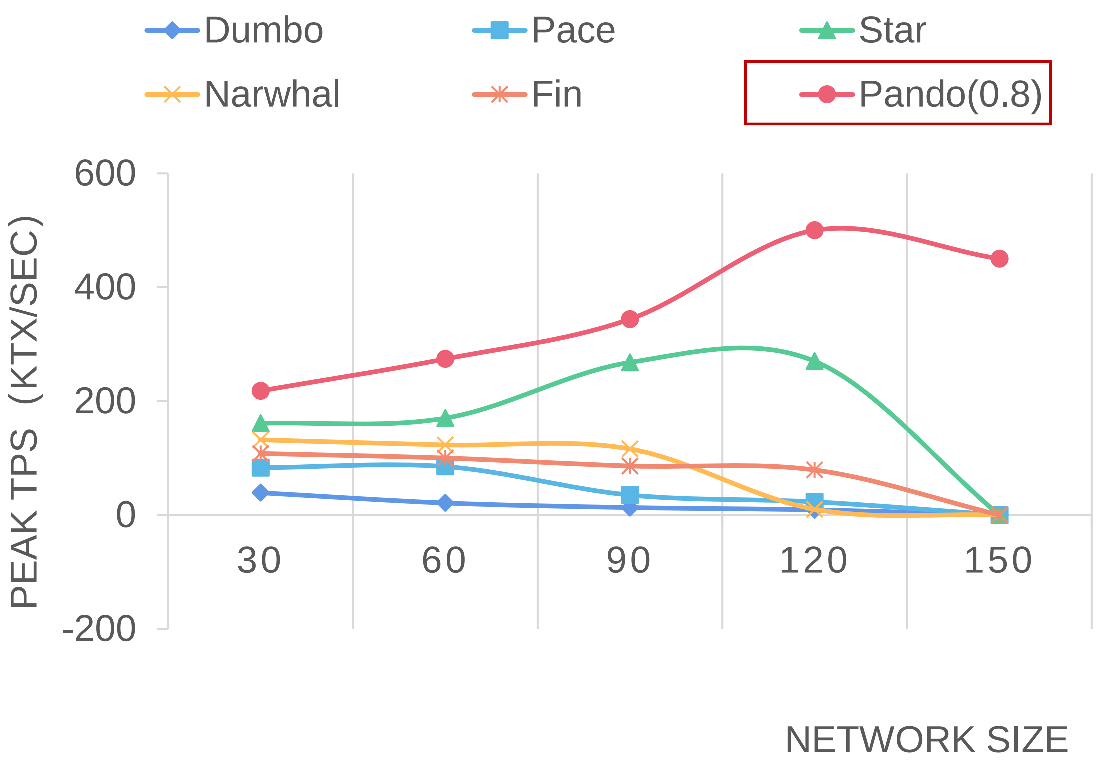


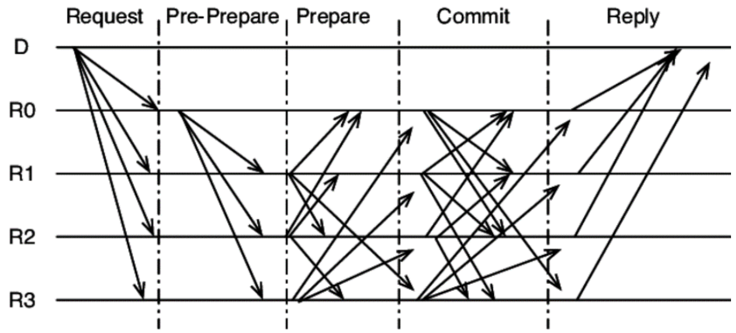
- Timing assumptions
  - Known/Unknown/No upper bound for message transmission
- Atomic broadcast (ABC)
  - No clients



Blockchain Scalability

- Communication / computational overhead
  - *N*-to-*n* communication
  - Quorum certificates (a set of *O(n)* signatures)
- Performance degrades significantly as the number of nodes grows





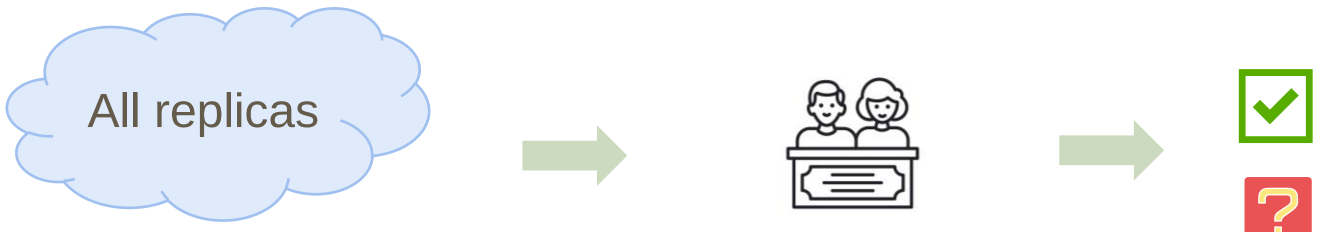
Protocols	Max network size evaluated
Narwhal (Eurosys 2022)	50
Star (Eurosys 2024)	90
FIN (CCS 2023)	160
Red Belly (S&P 2021)	240
Pando (this work)	**1000**

BFT from committee sampling

- ***Decouple*** **block transmission from consensus**

Transmission bulk data	Agreement on the order
Transmission in fully parallelize	Agree on hashes or digital signatures

- **Communication-efficient**



Blockchain network
(*O*(*n*) communication)

Committee
(*O*(*κ*) communication)

Static security model

Adaptive security model

Challenge for committee sampling-based BFT

- **Partially synchronous BFT**
  - $f < n/3$ (i.e., $n \geq 3f + 1$)
- **Committee sampling-based BFT**
  - $f < (1/3 - \epsilon)n$, $\epsilon \in (0,1/3)$
  - $\epsilon \to 0$, **near-optimal resilience** claim

Protocols	Max network size
Narwhal (Eurosys 2022)	50
Star (Eurosys 2024)	90
FIN (CCS 2023)	160
Red Belly (S&P 2021)	240
Pando (this work)	**1000**

- **Failure rate**
  - With 10^{-9}, Algorand (SOSP 2017) needs **$n \geq 2000$!**

Can we build a scalable BFT protocol from committee sampling by supporting small committee sizes while achieving a practically low failure rate?

Our Approach

Decoupling + Committee sample

Low communication and computational overhead

protocols	resilience	transmission	consensus	timing
Narwhal [11]/Bullshark [12]	$f < n/3$	$O(Ln^2 + \kappa n^4)$	$O(\kappa n^3)$	partial sync.
Tusk [11]	$f < n/3$	$O(Ln^2 + \kappa n^4)$	$O(\kappa n^3)$	async.
Dumbo-NG [16]	$f < n/3$	$O(Ln^2 + \kappa n^3)$	$O(\kappa n^3)$	async.
Star [14]	$f < n/3$	$O(Ln^2 + \kappa n^3)$	$O(\kappa n^3)$	partial sync.
Pando (this work)	$f < (1/3 - \epsilon)n$	$O(Ln^2 + \kappa^2 n^2)$	$O(\kappa^2 n^2)$	partial sync.

- Building blocks:
  - consistent broadcast
  - committee sampling function (*ComProve()*/*ComVerify()* oracle)

A scalable BFT: Pando

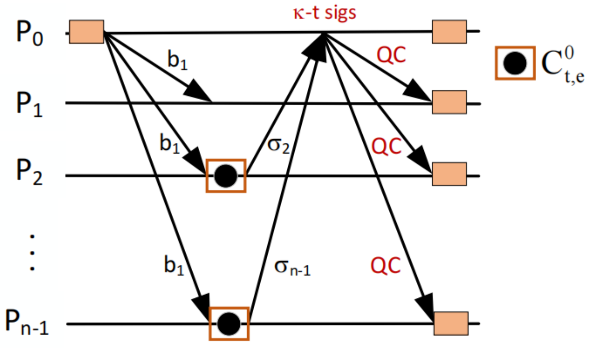
Low failure rate with smaller committee size

- Partially synchronous BFT
- Weakly adaptive adversary
- *O(κ)* communication
- Near-optimal resilience
  - 10^{-9} failure rate, only 200 committee size!
- Pando performance
  - easily scale to 1000, with ~70,000 TPS

Scalable CBC for transmission

- **κ term communication**
  - **Communication cost does not grow as *n* grows**

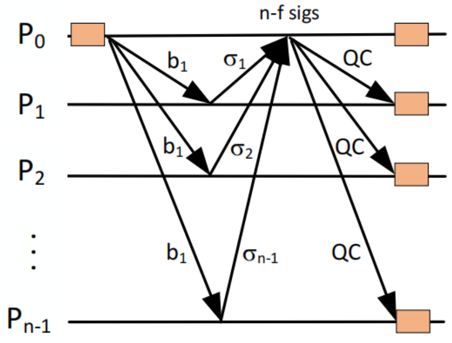
$O(\kappa)$ signatures as a QC



(b) Our scalable CBC approach.

vs.

n digital signatures / threshold signatures



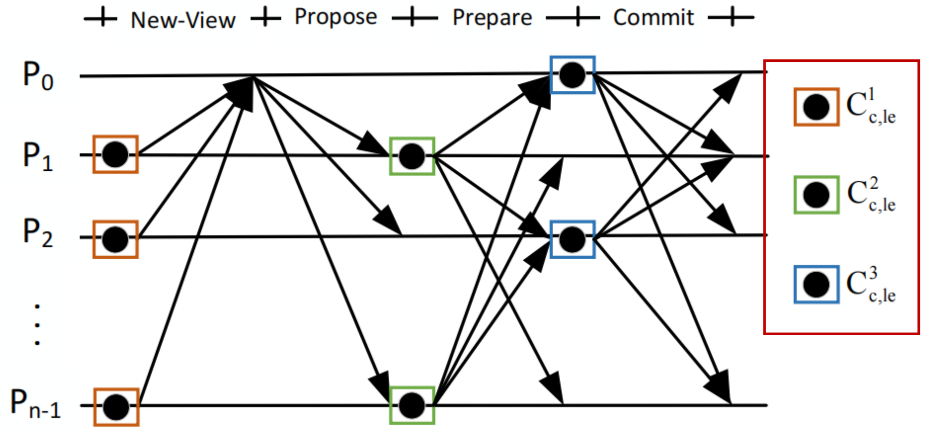
(a) Conventional consistent broadcast (CBC) protocol.

- **The fraction of Byzantine replicas in the committee ≈ that in the entire system**
  - Chernoff Upper Tail Bound:

$$\Pr\left(X \geq (1+\tau)E(X)\right) \leq \exp\left(-\frac{\tau \cdot \min\{\tau, 1\} \cdot E(X)}{3}\right)$$

ABC at scale for consensus

- **Sample 3 committees, twice in each epoch**
  - only committee members send votes
- **$O(\delta^2)$ failure rate**
  - δ: each committee has more than 1/3 of faulty relicas
- **Communication-efficient**
  - **input message M: $O(\kappa^2 n)$ -> $O(\kappa n^2)$**
  - **κ-to-all communication: $O(|M|n+\kappa^2 n)$ -> $O(|M|n+\kappa n^2)$**



(c) Our scalable atomic broadcast protocol.

Probability of safety and liveness violation

f < (1/3 - ϵ)n, $\epsilon \in$ (0,1/3)

- **Based on our proof:**
  - committee size $\lambda = \frac{3\alpha}{\epsilon^2} \ln \frac{1}{\delta}$, with probability *1-negl(κ)*
  - faulty replicas in the committee $\leq \frac{\lambda}{3}$
  - δ: failure rate, α: small constant

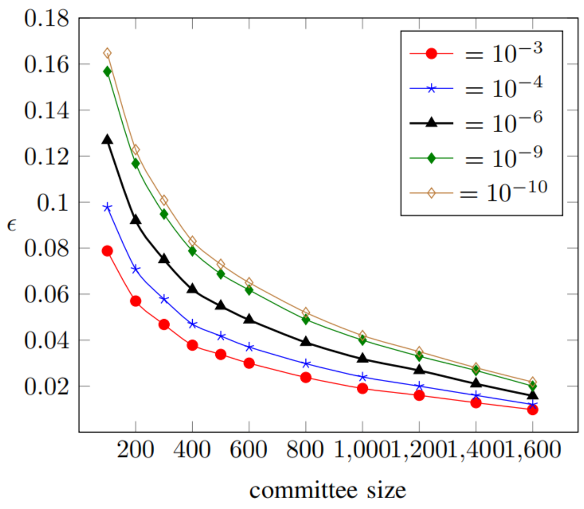


Fig. 3: Committee size vs. ϵ ($n = 2{,}000$) to limit the probability of violating safety and liveness to 10^{-3}, 10^{-4}, 10^{-6}, 10^{-9}, and 10^{-10} respectively.

- **Relationship between committee size and ϵ**

TABLE II: The value of ϵ for the system to achieve safety and liveness with a probability of at least $1 - 10^{-4}$. The system requires $f \in [0, \frac{1}{5}n)$, $f \in [\frac{1}{5}n, \frac{1}{4}n)$, and $f \in [\frac{1}{4}n, \frac{1}{3}n)$ for dark gray cells, gray cells, and white cells, respectively.

$n =$	100	200	300	400	500	1000
Pando (0.2)	0.193	0.133	0.123	0.103	0.091	0.067
Pando (0.4)	0.123	0.093	0.076	0.063	0.059	0.041
Pando (0.6)	0.093	0.063	0.053	0.046	0.041	0.029
Pando (0.8)	0.053	0.038	0.033	0.028	0.023	0.017

TABLE III: The value of ϵ for the system to achieve safety and liveness with a probability of at least $1 - 10^{-8}$. Hyphen means no ϵ value can make the desirable probability at least $1 - 10^{-8}$.

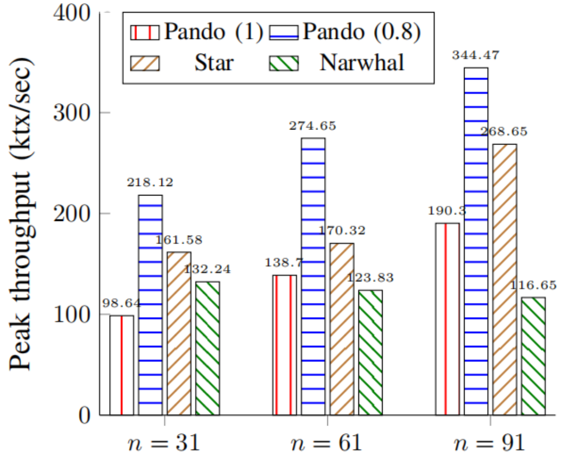
$n =$	100	200	300	400	500	1000
Pando (0.2)	0.253	0.198	0.177	0.153	0.137	0.102
Pando (0.4)	0.173	0.133	0.113	0.098	0.089	0.064
Pando (0.6)	0.123	0.093	0.08	0.068	0.061	0.044
Pando (0.8)	–	0.053	0.05	0.041	0.037	0.027

Evaluation

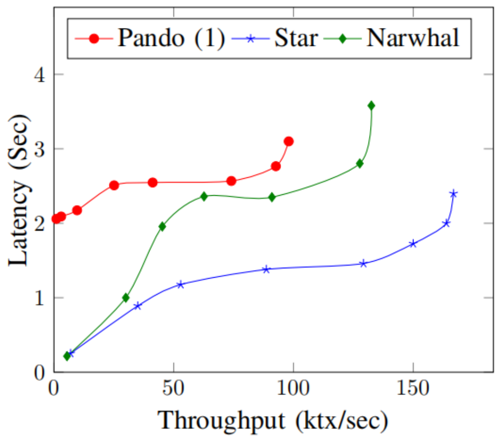
- **Golang**
- **10,000 LOC for protocols, 1,000 LOC for evaluation**
- **Evaluated 3 protocols in total**
  - Pando (Our approach), Star (Eurosys'24), Narwhal (Eurosys'22)
- **AWS instance with *m5.xlarge* (4 vCPU, 16GB memory)**
- **Up to 500 VMs and up to 1,000 replicas**

Results

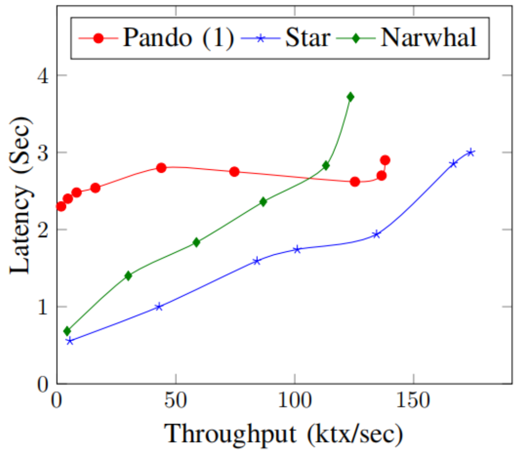
- Pando(x) -> *xn* committee members
- **Pando vs. Star vs. Narwhal**
  - **Pando(1)**:
    - consistently outperforms Narwhal
    - marginally lower than Star
  - When *n* = 91, the peak TPS of **Pando(0.8)**:
    - 28.22% that of Star
    - 81.01% that of Pando(1)



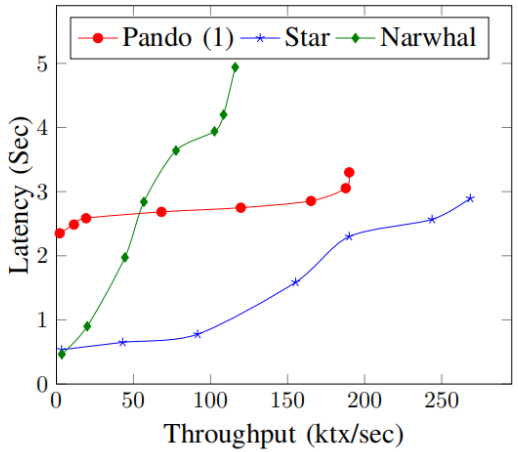
(a) Peak throughput of Star, Narwhal and Pando as f grows.



(b) Latency vs. throughput in WAN for $n = 31$.



(c) Latency vs. throughput in WAN for $n = 61$.



(d) Latency vs. throughput in WAN for $n = 91$.

Results

- **Low-end VMs**
  - When the **CPU and memory** become lower, the throughput decreases
  - For VMs with the same CPU and memory, the throughput becomes lower on the **bandwidth**-restricted VMs

instance	vCPU	memory (GiB)	bandwidth (Gbps)	batch size	peak tps (ktx/sec)
m5.xlarge	4	16	up to 10	100,000	2947.43
m5.large	2	8	up to 10	100,000	2443.05
m4.xlarge	4	16	0.75	100,000	1316.31
t2.micro[7]	1	1	up to 0.72	5,000	95.37

TABLE VI: Peak throughput of Pando (0.2) under different low-bandwidth and on-premise cluster.

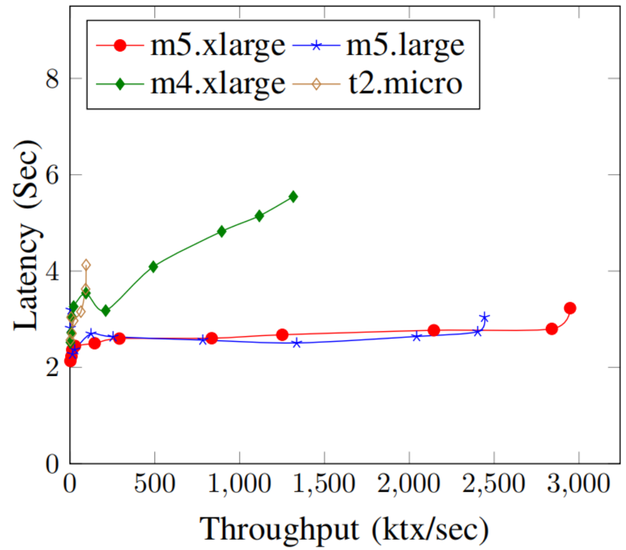
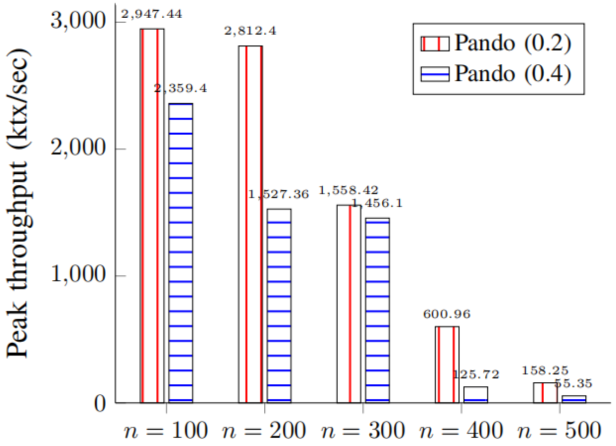


Fig. 5: Latency vs. throughput of Pando (0.2) for $n = 100$ under different low-bandwidth and on-premise cluster.

Results

- **Pando scalability**
  - For n = 100 to n = 500, TPS degrades significantly as n grows
- **Pando with 1,000 replicas!**
  - The **network bandwidth** is the bottleneck
  - With better configuration but lower bandwidth, Pando only gets 1,600 TPS
  - **up to 73,200 TPS!**



(k) Peak throughput of Pando as n grows.

instance	vCPU	memory (GiB)	bandwidth (Gbps)	batch size	peak tps (ktx/sec)
m5.2	8	32	up to 10	-	-
m5n.2	8	32	up to 25	5,000	62.57
m5.4	16	64	up to 10	100	1.22
c5.4	16	32	up to 10	100	1.6

(o) Peak throughput of Pando for $n = 1,000$ using different instance types.

instance	# VM	bandwidth (Gbps)	batch size	peak tps (ktx/sec)
m5.2	200	up to 10	-	-
m5.2	500	up to 10	-	-
m5n.2	200	up to 25	5,000	62.57
m5n.2	500	up to 25	5,000	73.2

TABLE VII: Peak throughput of Pando for $n = 1,000$ using different number of VMs.

Pando: Extremely Scalable BFT Based on Committee Sampling

Xin Wang, Haochen Wang, Haibin Zhang, Sisi Duan

- An adaptively secure and scalable BFT design from committee sampling
- Decouple block transmission from consensus on the order
- Communication efficient and computation-efficient

Xin Wang
Shandong University
wangxin87@sdu.edu.cn

NDSS 2026

Open source:
https://github.com/DSSLab-Tsinghua/Pando