Mode-suppression

A simple and provably stable chunk sharing algorithm for P2P networks

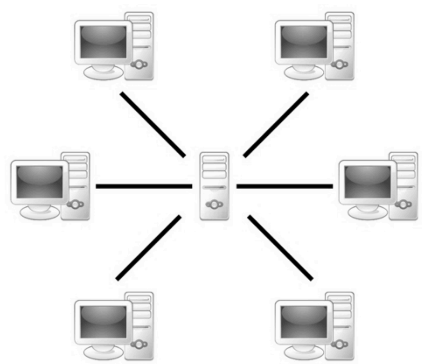
Srinivas Shakkottai
Texas A&M University

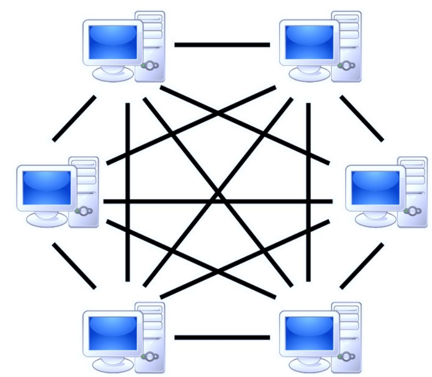
Joint work with
Vamseedhar Reddyvari (TAMU) and
Parimal Parag (IISc)

COMPUTER ENGINEERING & SYSTEMS GROUP

Peer to Peer Network

- P2P network offers many advantages over Client-Server approach
  - Scalability
  - Decrease the cost of distribution
  - Build robustness
- 30% of P2P traffic in Asia-Pacific region in 2016

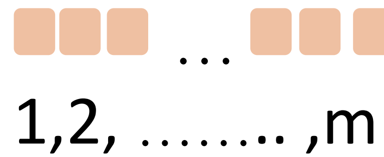


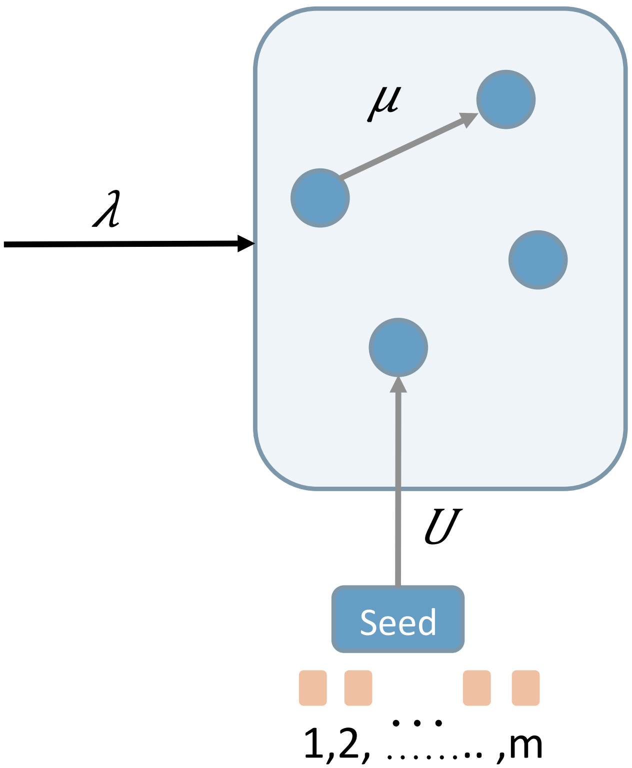


- BitTorrent is the popular P2P application used for file sharing
- Spotify uses a combination of client server and P2P for music streaming and downloads
- Microsoft is using P2P for distributing Windows 10 updates

P2P File sharing: System Model

- File is divided into m chunks
- New peers enter the system with no chunks
- Arrival process is $\text{Poisson}(\lambda)$
- Peer leaves the system as soon as it receives all the chunks





- There always exists a seed that posseses all the chunks
- Seed contacts a peer according to a $\text{Poisson}(U)$ contact process
- Every peer contacts another peer(s) according to a $\text{Poisson}(\mu)$ contact process
- Chunks are transmitted according to given chunk selection policy

Hajek's – (In) Stability Result

- The system is said to be stable for a given λ if the Markov Chain is positive recurrent
- In general if a peer selection is random then any "work conserving" policy will be unstable if $\lambda > U$

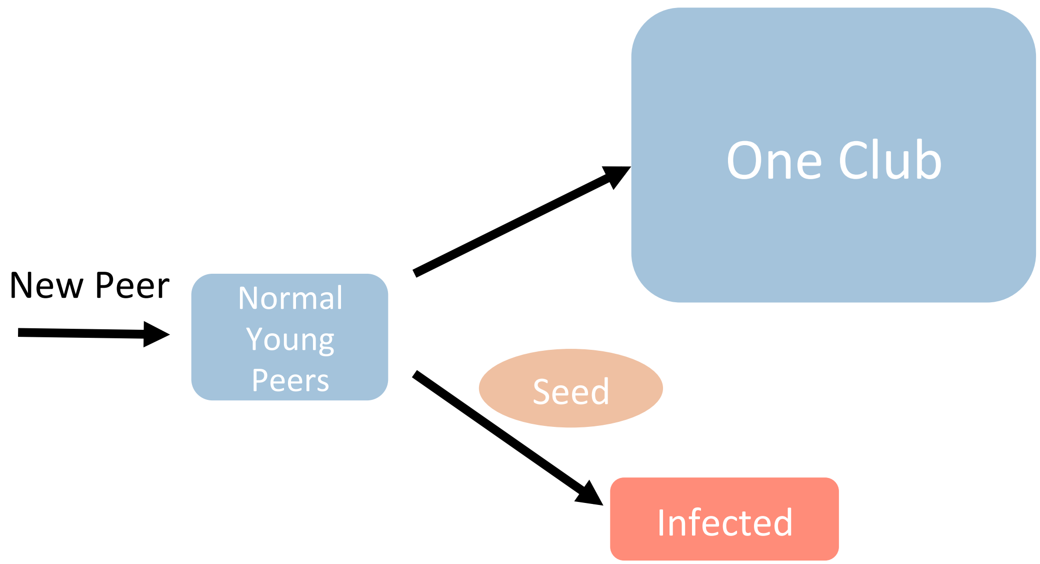
Rarest First Policy: Find the list of useful chunks and among them select the chunk with least marginal chunk frequency

- Mendes, Towsley et.al observed that P2P networks following the BitTorrent protocol show unstable behavior due to the formation of large One clubs.

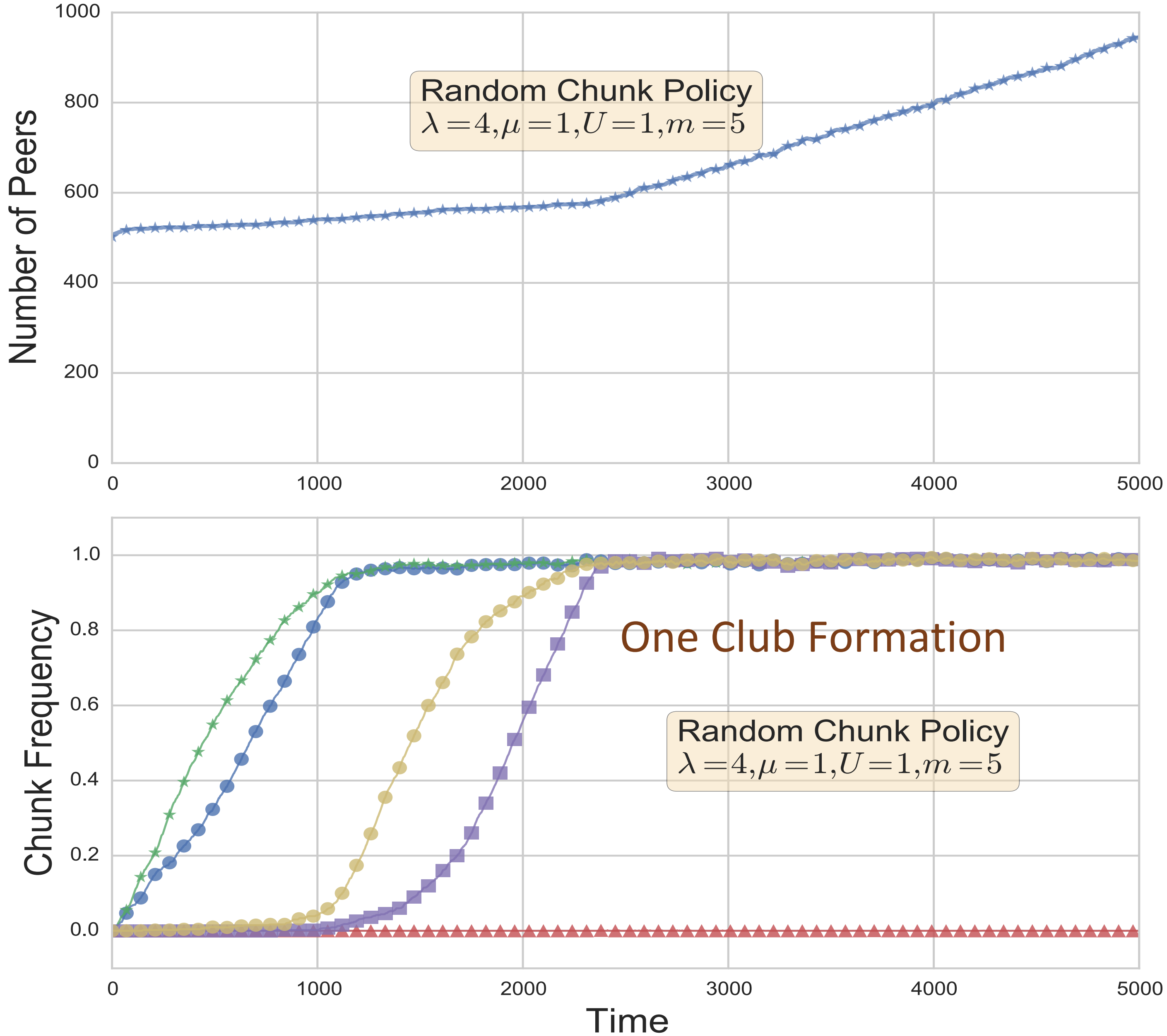
Why Unstable? Missing Chunk Syndrome

One Club: Group of peers which have all the chunks except one particular chunk
Infected Peers: Peers that posses the chunk that one club peers are missing

- An entering peer is likely to join the One club
- If a One club peer samples an infected peer, it will leave the system: the infected peers will not grow and One club keeps growing.
- System becomes unstable



Missing Chunk Syndrome in Simulations



[1] B. Hajek and J. Zhu, "The missing piece syndrome in peer-to-peer communication," in *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on*. IEEE, 2010, pp. 1748–1752.

[2] B. Oguz, V. Anantharam, and I. Norros, "Stable distributed p2p protocols based on random peer sampling," *IEEE/ACM Transactions on Networking (TON)*, vol. 23, no. 5, pp. 1444–1456, 2015.

[3] O. Bilgen and A. Wagner, "A new stable peer-to-peer protocol with non-persistent peers," in *Proceedings of INFOCOM*, 2017, pp. 1783–1790.

[4] H. Reittu, "A stable random-contact algorithm for peer-to-peer file sharing." in *IWSOS*. Springer, 2009, pp. 185–192.

[5] I. Norros, H. Reittu, and T. Eirola, "On the stability of two-chunk file-sharing systems," *Queueing Systems*, vol. 67, no. 3, pp. 183–206, 2011.

[6] D. X. Mendes, E. d. S. e Silva, D. Menasche, R. Leao, and D. Towsley, "An experimental reality check on the scaling laws of swarming systems," in *Proceedings of INFOCOM*, 2017, pp. 1647–1655.

Group Suppression

- A peer from the largest group should not give a chunk to any peer possessing fewer chunks than itself
- A seed should give chunks only to the most deprived peers

- The idea behind the Group Suppression is to avoid the growth of One club peers
- In particular the peers in One club will not transfer chunks to new young peers if One club is large

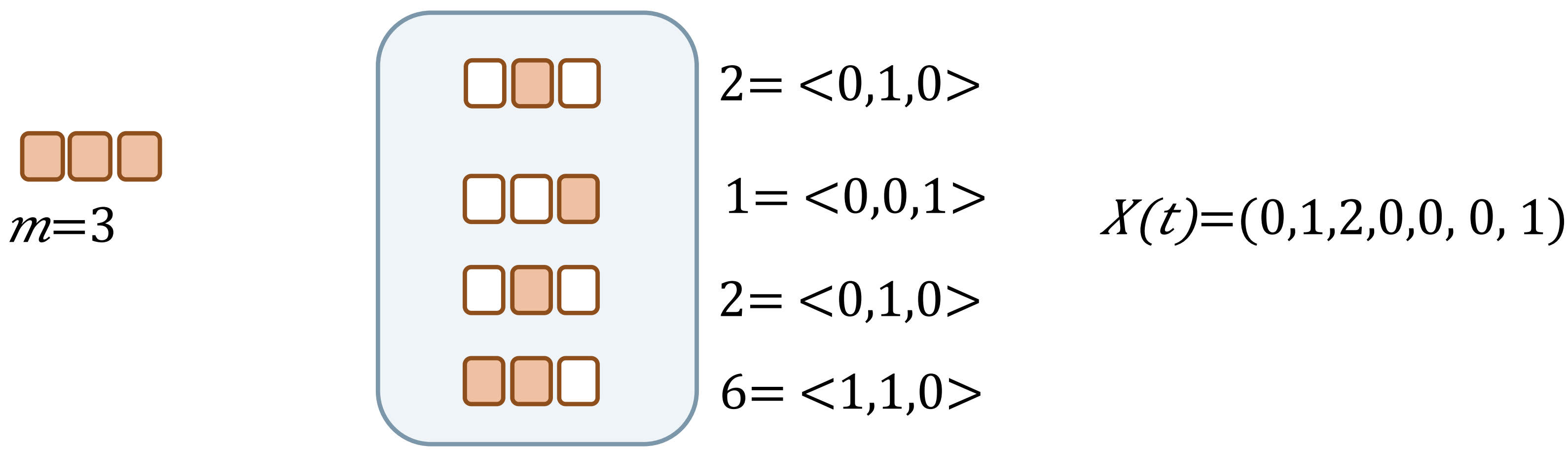
Group Suppression is stable for $\lambda > 0$ if $m = 2$

- Has good sojourn time in general (although variable).

Sojourn Time : Sojourn time is the amount of time a peer spends in the system before leaving the system by receiving all the chunks

System Model

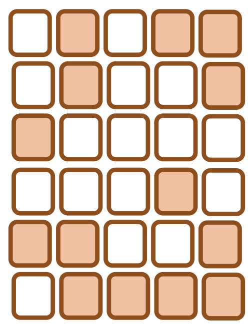
- $X_S(t)$ denote number of peers with chunk profile S
- State of the System at time t is denoted by $X(t)$ and is a vector of length $2^m - 1$ elements
- The element at index i is the number of peers with chunk profile corresponding to i.
- $X(t)[i] = X_S(t)$ where $<S> = i$
- Total number of peers in the system is $|X(t)| = \sum_{s \subset [m]} X_S(t)$



Mode Suppression

COMPUTER ENGINEERING & SYSTEMS GROUP

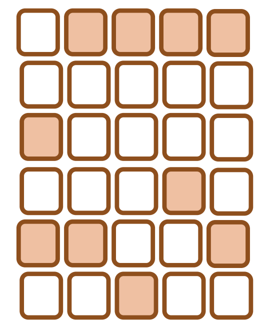
Suppress chunks which are in the mode except when all the indices are in the mode



2 4 1 3 4

$\mathcal{M}(x)=\{2,5\}$

$\mathrm{D}(x)=\{2,5\}$



2 2 2 2 2

$\mathcal{M}(x)=\{1,2,3,4,5\}$

$\mathrm{D}(x)=\phi$

- In the first case if a peer with $(0,0,0,0,0)$ *meets* $(0,1,0,0,1)$ then no chunk transferred

Mode Suppression – Q Matrix

COMPUTER ENGINEERING & SYSTEMS GROUP

- Let $A(x,B,S)$ be the set of allowed chunks from B to S, then

$$A(x,B,S) = B \backslash (S \cup D(x))$$

$$\forall S : j \notin S; j \notin D(x),$$

$$Q(x, \mathcal{T}_{S,j}(x)) = \frac{x_S}{|x|}\left(\frac{U}{|A(x,[m],S)|} + \mu \sum_{T:j \in T} \frac{x_T}{|A(x,T,S)|}\right)$$

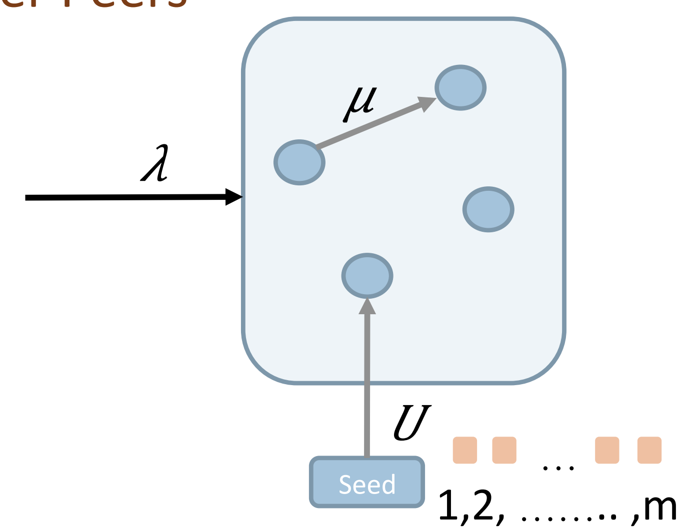
$$Q(x, x + e_\phi) = \lambda$$

Chunk from Seed

Chunk from other Peers

Where,

$$\mathcal{T}_{S,j}(x) = x - e_{<S>} \text{ if } S \cup j = [m]$$
$$= x - e_{<S>} + e_{<S \cup j>} \text{ o.w}$$



Stability of Mode Suppression

Theorem 1 *The stability region of the Mode Suppression Policy is $\lambda > 0$ if $m \geq 2, U > 0$ and $\mu > 0$.*

- To prove the stability we need to prove that the Markov Chain is positive recurrent
- Proving positive recurrence directly is difficult in this case
- So, we employ Foster-Lyapunov criteria and come up with a Lyapunov function for which the drift is negative

(Foster-Lyapunov Criteria:) *Suppose $X(t)$ is irreducible and if there exists a function $V : \mathcal{S} \to \mathbb{R}^+$ such that*

1. $\sum_{y \neq x} Q(x,y)(V(y) - V(x)) \leq -\epsilon$ *if* $x \notin \mathcal{F}$, *and*
2. $\sum_{y \neq x} Q(x,y)(V(y) - V(x)) \leq K$ *if* $x \in \mathcal{F}$,

for some $\epsilon > 0, K < \infty$ *and a bounded set* $\mathcal{F}$, *then* $X(t)$ *is positive recurrent.*

Foster - Lyapunov Criteria

- The following Lyapunov Function will satisfy the Foster-Lyapunov criteria

$$V(x) = \underbrace{\sum_{i=1}^{m}\Big((\overline{\pi} - \pi_i)|x|\Big)^2}_{L_1} + \underbrace{C_1\big((1-\overline{\pi})\big)|x|}_{L_2} + \underbrace{C_2\Big(M - \sum_{i=1}^{m}\pi_i|x|\Big)^+}_{L_3}$$

- All are functions of state
- We need to show that that the drift is negative except in some finite set

Intuition

COMPUTER ENGINEERING & SYSTEMS GROUP

- L_1 is the sum of differences in marginal chunk frequencies.
- Mode suppression increases π_i but not $\bar{\pi}$

if $\pi_i < \pi$

$$\underbrace{\sum_{i=1}^{m}\left((\bar{\pi}-\pi_i)|x|\right)^2}_{L_1}$$

$$\underbrace{C_1\left((1-\bar{\pi})\right)|x|}_{L_2}$$

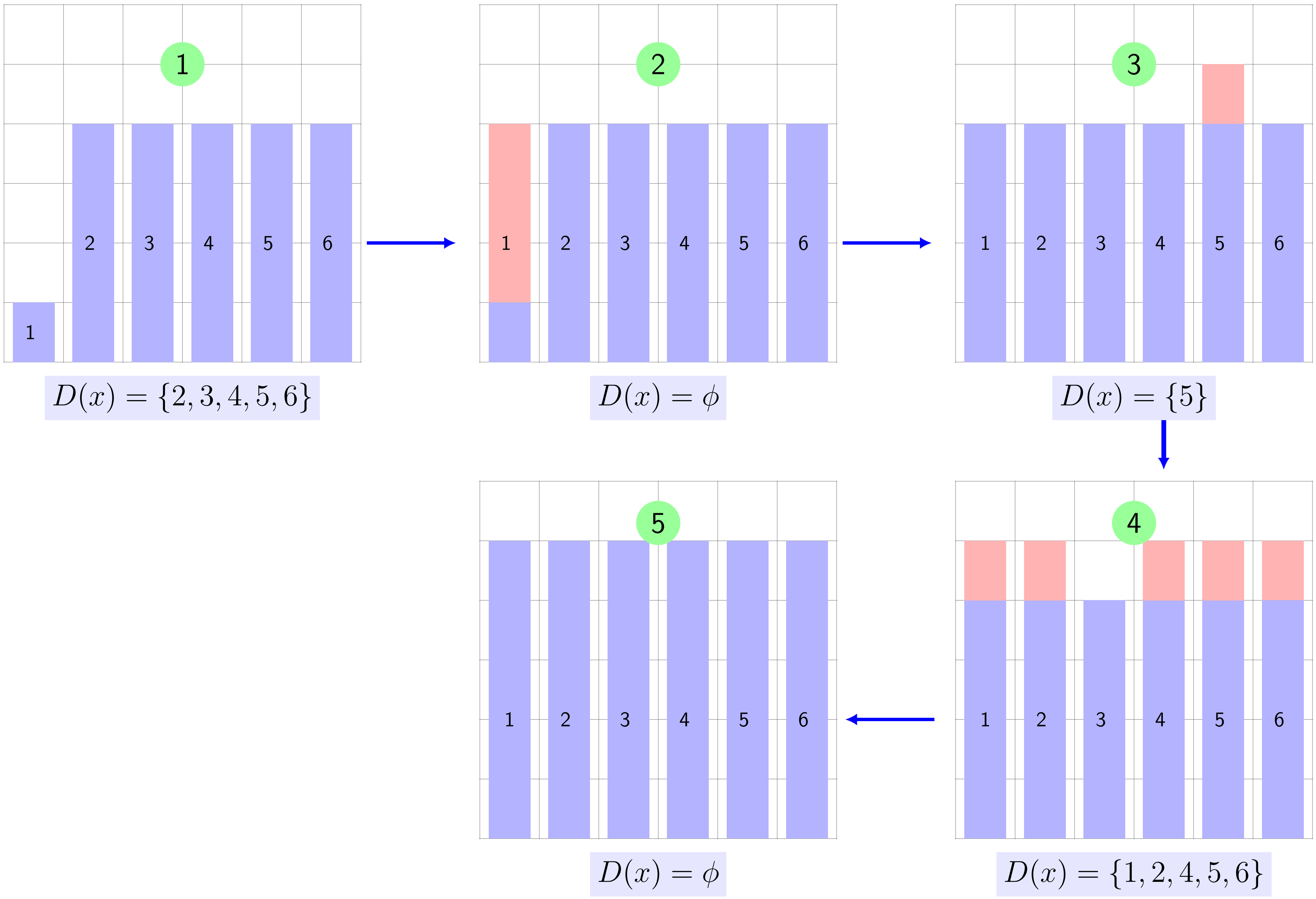
L_2 decreases if $\bar{\pi}$ increases.

In Mode Suppression this will happen only when the marginal chunk frequencies is uniform

- Number of chunks in the system increases

$$\underbrace{C_2\left(M-\sum_{i=1}^{m}\pi_i|x|\right)^+}_{L_3}$$

Progression of Mode Suppression



Threshold Mode Suppression

- In MS any slight deviation from uniform marginal chunk frequency will result in suppression
- Though this is favorable for stability, this will not result in best sojourn times

- Is there a way to reduce the suppression of MS without compromising stability?
- Use a "noisy" mode estimate: Threshold Mode Suppression(TMS)

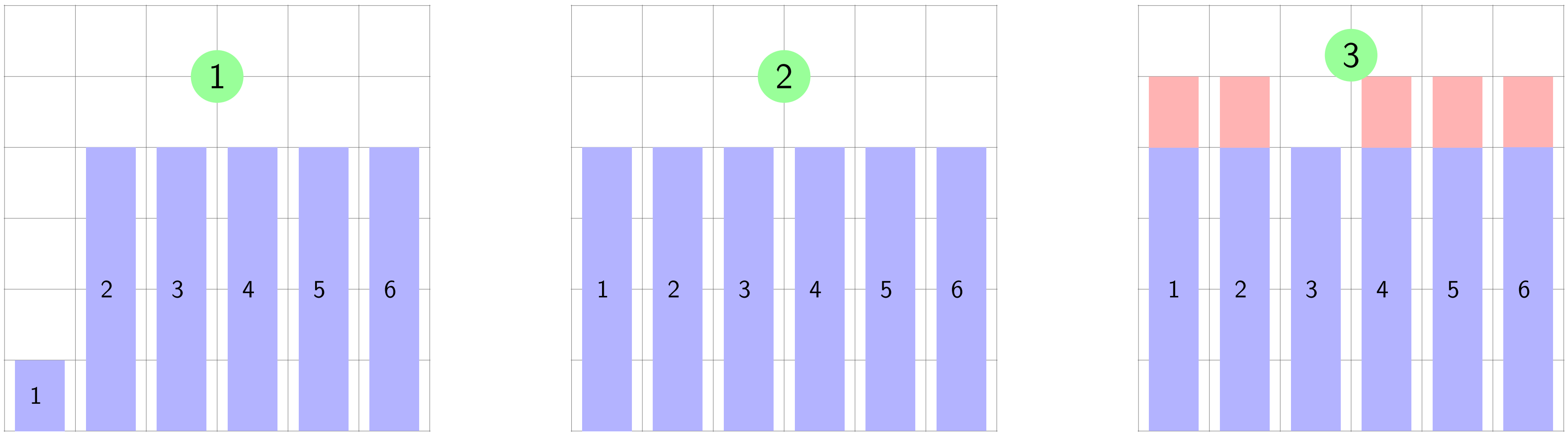
- The idea of TMS is to suppress the modes only if they are abundant compared to least frequency chunks
- The set of indices suppressed in Threshold Mode Suppression are

$$D_T(x) = \left\{ k | \pi_k(x) = \overline{\pi}(x), \overline{\pi}(x)|x| \geq \underline{\pi}(x)|x| + T \right\}$$

Threshold Mode Suppression

COMPUTER ENGINEERING & SYSTEMS GROUP

- Example when T=2



$D(x) = \{2,3,4,5,6\}$
$D_T(x) = \{2,3,4,5,6\}$

$D(x) = \phi$
$D_T(x) = \phi$

$D(x) = \{1,2,4,5,6\}$
$D_T(x) = \phi$

- When T=1, TMS = Mode Suppression
- When $T \to \infty$, TMS → Random Chunk because there won't be any suppression

TMS - Stability

Theorem 2 *The stability region of Threshold Mode Suppression (TMS) is* $\lambda > 0$ *for any finite threshold* $T < \infty$*, if* $m \geq 2, \mu > 0$ *and* $U > 0$.

- Proof is using Foster-Lyapunov Criteria
- Same Lyapunov function works
- With the constant $C_1 > (2T-1)(m-1)$

$$V(x) = \underbrace{\sum_{i=1}^{m} \Big((\overline{\pi} - \pi_i)|x|\Big)^2}_{L_1} + \underbrace{C_1\big((1-\overline{\pi})\big)|x|}_{L_2} + \underbrace{C_2\Big(M - \sum_{i=1}^{m} \pi_i |x|\Big)^+}_{L_3}$$

Scaling of Sojourn Time with λ

Theorem 3 *In Threshold Mode Suppression policy, as $\lambda \to \infty$,*

1. **(Scaling of Number of Peers)** *the average number of peers (L) scales linearly with λ.*

2. **(Scaling of Sojourn time)** *the average sojourn time of the peers (W) remains bounded and doesn't scale with λ.*

- To prove this we use a Kingman moment bound

(Kingman Moment bound) Suppose V, f, and g are nonnegative functions of S, and suppose $QV(i) \leq -f(i) + g(i)$ for all $i \in S$. In addition, suppose X is positive recurrent, so that the means, $\bar{f} = \pi f$ and $\bar{g} = \pi g$ are well defined. Then $\bar{f} \leq \bar{g}$.

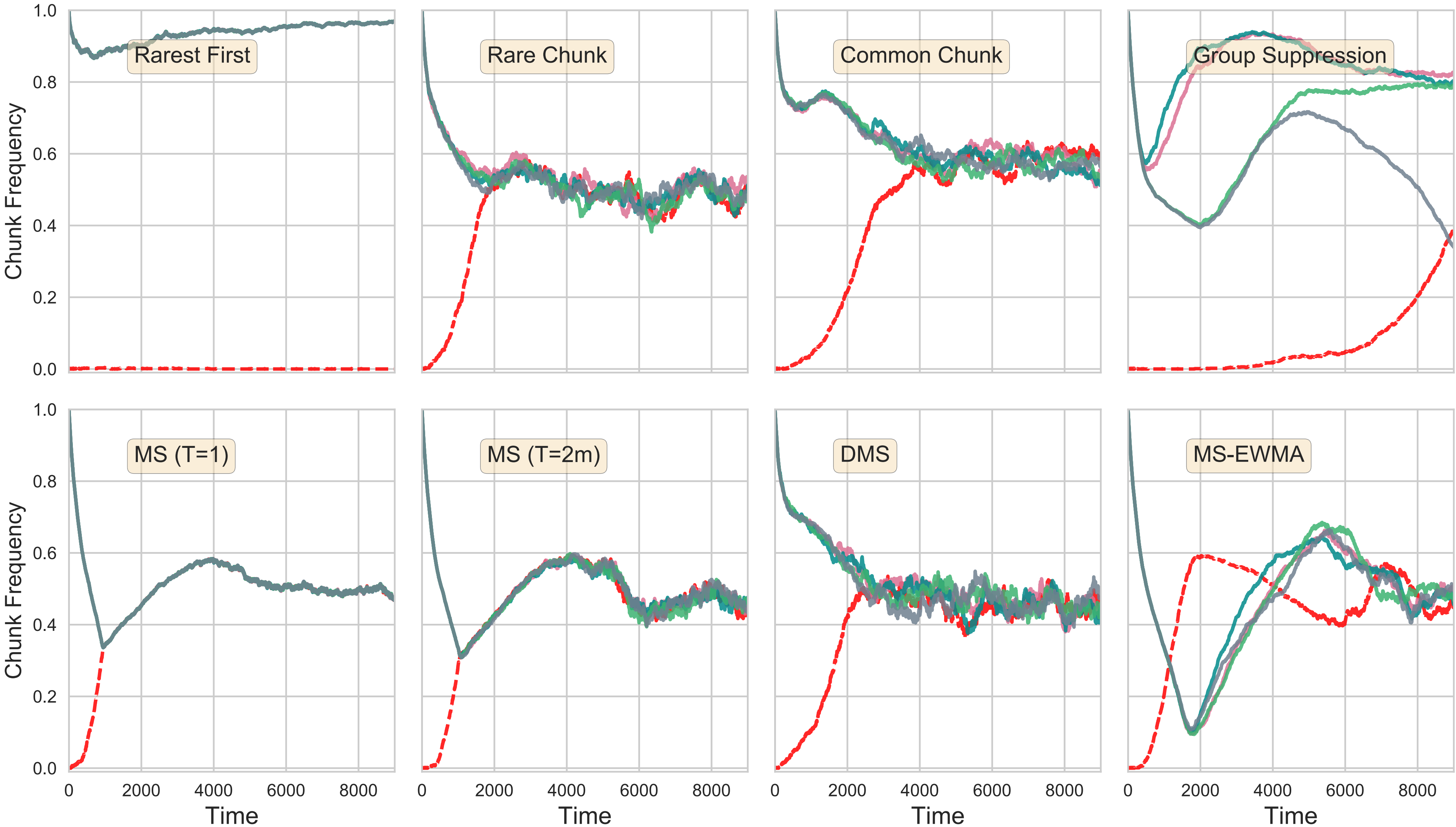
Heuristic Variants

- Noise in calculating the mode is acceptable within limits (threshold mode suppression).

- Distributed Mode Suppression: Sample three peers and calculate mode using those three. Suppress mode as long as it appears in more than one peer.

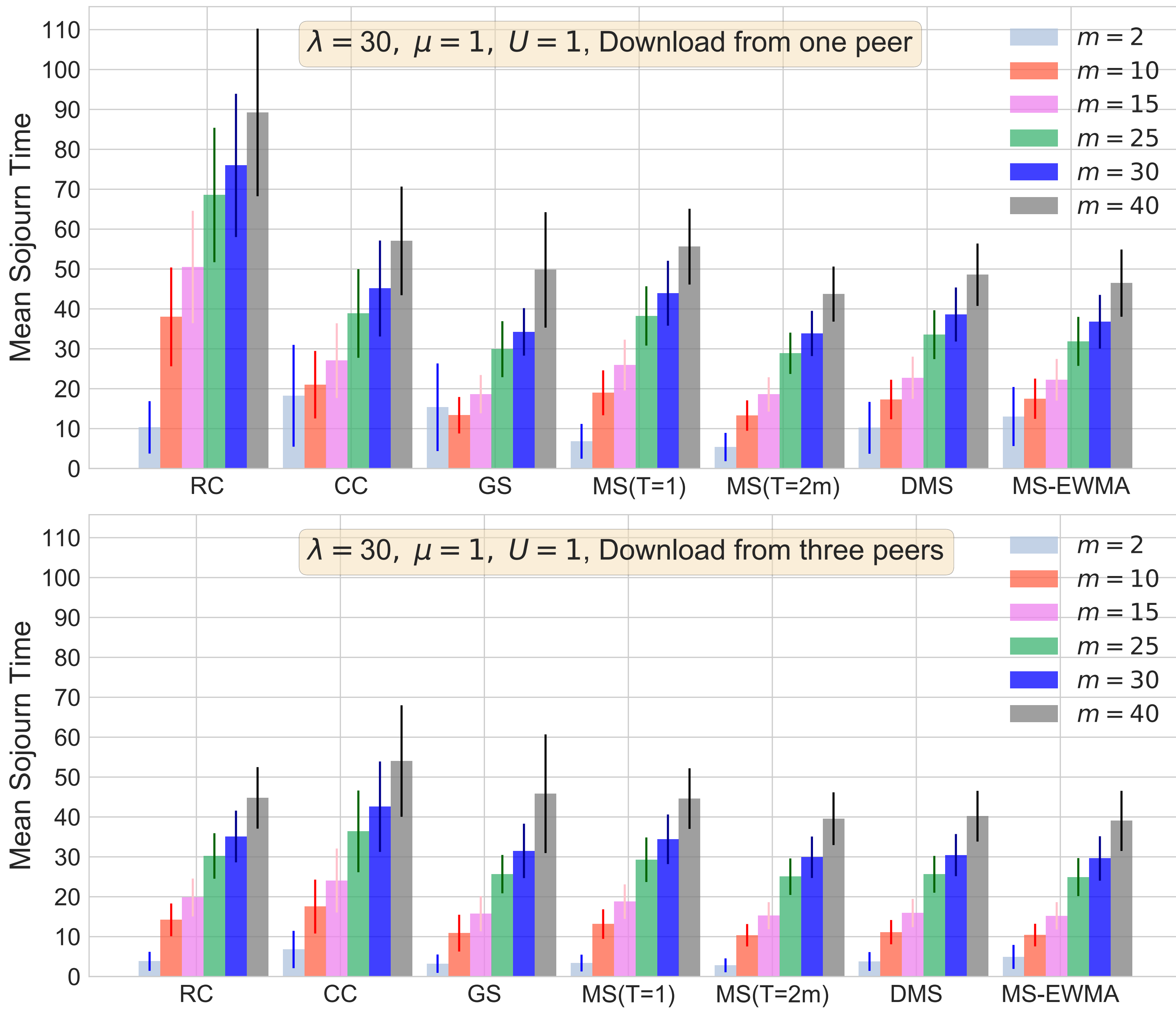
- Exponentially Weighted Moving Average Mode Suppression: Sample only one peer each time. Build up a mode estimate by using historical chunk frequency information from each contacted peer with diminishing weights.

Stability


Rarest First
Rare Chunk
Common Chunk
Group Suppression
MS (T=1)
MS (T=2m)
DMS
MS-EWMA
Chunk Frequency
Time
0.0
0.2
0.4
0.6
0.8
1.0
0
2000
4000
6000
8000

Performance

COMPUTER ENGINEERING & SYSTEMS GROUP



Conclusions

COMPUTER ENGINEERING & SYSTEMS GROUP

- Random Chunk & Rarest first policies are not stable at higher peer arrival rate and we need suppression for stability
- Came up with provably stable policies MS & TMS
- TMS has the right amount of suppression to provide stability and good sojourn times
- Developed a distributed version of MS and proved its stability when m=2
- Proved that in TMS waiting time remains bounded even when $\lambda \rightarrow \infty$

Thank You
Questions?