Mode-Suppression: A Simple and Provably Stable Chunk-Sharing Algorithm for P2P Networks

Vamseedhar Reddyvari
Texas A&M University
College Station, TX-77801
vamseedhar.reddyvari@gmail.com

Parimal Parag
Indian Institute of Science
Bengaluru, KA 560012, India
parimal@iisc.ac.in

Srinivas Shakkottai
Texas A&M University
College Station, TX-77801
sshakkot@tamu.edu

***Abstract*—The ability of a P2P network to scale its throughput up in proportion to the arrival rate of peers has recently been shown to be crucially dependent on the chunk sharing policy employed. Some policies can result in low frequencies of a particular chunk, known as the missing chunk syndrome, which can dramatically reduce throughput and lead to instability of the system. For instance, commonly used policies that nominally "boost" the sharing of infrequent chunks such as the well-known rarest-first algorithm have been shown to be unstable. Recent efforts have largely focused on the careful design of boosting policies to mitigate this issue. We take a complementary viewpoint, and instead consider a policy that simply prevents the sharing of the most frequent chunk(s). Following terminology from statistics wherein the most frequent value in a data set is called the mode, we refer to this policy as mode suppression. We prove the stability of this algorithm using Lyapunov techniques. We also design a distributed version that suppresses the mode via an estimate obtained by sampling three randomly selected peers. We show numerically that both algorithms perform well at minimizing total download times, with distributed mode suppression outperforming all others that we tested against.**

I. Introduction

Peer-to-Peer (P2P) file sharing networks such as BitTorrent [1] have been studied intensely in recent years, using analytical models, simulation studies, and large scale field experiments. This interest partly stems from the dominance of P2P as a source of Internet traffic in past years. Even today, although the traffic fraction has reduced to around 3-4% in North America, P2P sharing still occupies a significant fraction of about 30% of traffic in the Asia-Pacific region [2]. Interest also stems from a desire to understand the thought-provoking phenomenon of apparent scaling up of the throughput of a P2P network as the number of peers grows, which enables them to effectively distribute content with low file-download times during high demand situations called *flash-crowds*.

In a P2P network, a file is divided into fixed-size chunks, and a peer possessing a set of chunks can upload those chunks to other peers that need them. Once a peer has downloaded all chunks, it could continue to serve other peers or leave the system. A so-called *seed server* that possesses all chunks and never leaves is often used to ensure that no particular chunk ever goes missing. It is the feature of integrating the upload capacity of each peer into the system that is supposed to enable system-wide throughput scaling up with the number of peers. However, since peers can only share chunks that they possess, it is crucial to ensure the wide availability of all chunks to enable maximum usage of available upload capacity with each peer.

The problem of ensuring that all chunks are easily obtainable—ideally by engendering equal numbers of copies of each chunk over the network—was considered by the original designers of P2P networks. For example, BitTorrent, which is the most popular P2P network protocol, uses an algorithm called *rarest-first* (RF) to try to achieve this goal [1]. Here, the idea is to keep a running estimate of the frequency of all chunks in the system. When a peer has a chance to download a chunk, it chooses the least frequent (i.e., the "rarest") among all the chunks that it needs. In practice, peers keep track of the frequency of chunks in local subsets. Intuition suggests that such "boosting" of rare chunks might ensure a near-uniform empirical distribution of chunks.

Recent work has postulated that under some conditions, the rarest-first policy used by BitTorrent actually does not achieve its goal, and can actually be harmful to system performance. In particular, [3] studied a chunk-level model of P2P sharing under which new peers that do not possess any chunks arrive into the system at some rate, contacts between peers happen at random, and at each contact a chunk is transferred to a requesting peer under a given policy. Peers depart immediately after completing the file download. The objective was to determine if the system is *stable* under a given policy, i.e., at any time is the number of peers that have not yet received the whole file finite or is it exploding to infinity? The result was that under several policies including rarest-first and random chunk selection, a particular chunk can become very rare across the network—a phenomenon referred to as the *missing chunk syndrome*. This causes the creation of a large set of peers that are missing only that one chunk, referred to as the *one club*. In turn, the seed server must serve the missing chunk to almost all peers (which then depart), which means that the system is unstable unless the upload capacity of the seed server is of the order of the arrival rate of peers into the system. Thus, the phenomenon largely negates the value of the P2P system.

More recently, experimental studies have revealed that the

Research was funded in part by NSF grants CNS-1149458, AST 1443891, and Science & Engineering Research Board (SERB) grant number DSTO-1677. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

missing piece syndrome is an observable phenomenon occurring in BitTorrent networks [4]. The results show that when the seed server has low or intermittent upload capacity, the throughput of the system saturates as the number of peers grows. In turn, this causes lengthened stay of peers in the system between arrival and completion, where an increasingly large number of peers are waiting to obtain the final chunk before leaving. In other words, designing policies that can ensure stability of a P2P network under a fixed seed server capacity for all peer arrival rates is practically relevant.

A. Related Work on Stable Algorithms

There has been extensive work on P2P networks, and we refer here only to those directly relating to the scaling properties of a single swarm. A large system assumption was made in [5]–[7], and the evolution of peers and seeds is described using a system of differential equations. While [5], [6] study the stationary regime and indicate the stability of BitTorrent-like systems for all arrival rates, [7] considers the transient regime and studies how much seed server capacity is needed to attain a target sojourn time (the time between the arrival of a peer and its completing the file download). Results on stability and scaling here require that at least a fixed fraction of the peers' upload capacity can always be utilized—an implicit assumption of chunk availability. As shown in [3], this assumption need not hold for all chunk selection policies, and a chunk-level model is needed for accurate analysis.

Chunk-level models have considered the missing chunk problem from two angles. The first method is to explicitly insist that peers that have completed the download should stay in the system as servers for some period of time. For example, [8] presents results on fairness vs. system performance based on how long peers stay after completion. In a more recent work [9], it was analytically shown that the system is stable as long as peers stay long enough to serve of the order of one additional chunk after completion. Indeed, in the original BitTorrent implementation this often happened naturally, since most users manually stopped participation at some point after download was completed. However, current implementations allow for the peer to depart immediately after completion, which can lead to the instability observed in [4].

The second method is to assume that peers would leave immediately after completion, and to design the chunk sharing policy such that the missing chunk syndrome is avoided. Some algorithms of this nature are "boosting" policies that can be thought of as modified versions of rarest-first. For example, the *rare chunk* (RC) algorithm studied in [10]–[12] picks three peers at random and chooses a chunk that is available with exactly one of the selected peers (called a "rare" chunk). Also studied in [12] is a variant of this algorithm called the *common chunk* (CC) algorithm, which proceeds as in the RC algorithm when the peer has no chunks, then follows a policy of sampling a single peer with random selection among its required chunks until it only needs one more chunk, and then proceeds by sampling three peers and only downloading a chunk if every chunk with it appears at least twice with the sampled peers. However, although stable, these algorithms appear to have long sojourn times in some settings [13].

More recent work on chunk sharing policies [13] describes an algorithm called *group suppression* (GS), which is based on observations made in [3]. The policy is based on computing the empirical distribution of the states in the system, where a state of a peer is the set of chunks available with that peer. Peers that belong to the state with highest frequency are not allowed to upload chunks to peers that have fewer chunks than themselves, thus suppressing entry into the highest frequency group. Although this policy appears to have low sojourn times in simulations, it is somewhat complex since it requires the knowledge of the entire empirical state distribution. Furthermore, the authors are only able to prove stability in a P2P network with exactly 2 chunks, while the stability of the general case is left as a conjecture.

B. Main Results

The nominal objective of Rarest-First is to ensure a uniform chunk distribution across the network, which it actually does not achieve in all cases, causing instability as shown in [3]. Our intuition is that rather than following a policy of boosting low-frequency chunks as rarest-first does, simply preventing the most frequent chunk(s) from being shared would allow less frequent chunks to catch up, and drive the empirical distribution of chunks towards the desired uniform distribution. Implicitly, this would also remove a small fraction of the upload capacity, keeping peers in the system a little longer, and enabling them to share more copies of rare chunks.

Following this intuition, we propose a policy that we call *mode suppression* (MS), which is based on terminology used in statistics in which the *mode* is the most frequent value(s) in a data set. Thus, we keep track of the frequency of chunks in the system, and when a peer contacts another peer, it is allowed to download any chunk except the one(s) belonging to the mode. Any chunk may be downloaded if all chunks are equally frequent (i.e., if all chunks belong to the mode). The policy is simple to implement, since all that is needed is the chunk frequency (which is already a part of BitTorrent).

We consider model similar to [3], [12], [13] in which peers that have no chunks enter the system according to a Poisson process with a certain arrival rate. There is a seed server that has an independent Poisson clock of a fixed rate, and at each clock tick, it contacts a single peer and uploads a chunk to it following a given policy. Each peer also has an independent Poisson clock of a fixed rate, and at each clock tick, the peer contacts a randomly selected peer and uploads a chunk to it following the same policy. Our main analytical result is to show using a Lyapunov drift analysis that mode suppression is *stable under all peer arrival rates* in a system in which the *file is divided into any number of chunks*.

We also construct two variants on the idea that only depend on a much smaller set of sample statistics. The first varient is mode-suppression that samples only one peer at a time and uses an exponentially weighted moving average estimate of chunk frequency (MS-EWMA), while the second variant,

TABLE I
COMPARISON OF CHUNK SELECTION POLICIES

Policy	$\mathbf{m=2}$	$\mathbf{m>2}$	Information	Sojourn time	
				Download from 1 Peer	*Download from 3 Peers*
Random	Unstable	Unstable	None	N/A (unstable)	N/A (unstable)
Rarest-First (RF)	Unstable	Unstable	Chunk Frequency	N/A (unstable)	N/A (unstable)
Rare Chunk (RC)	Stable	Stable	3 Peers	Bad	Good
Common Chunk (CC)	Stable	Stable	3 Peers	Good	Bad
Group Suppression (GS)	Stable	Unknown	Complete Distribution	Good	Better
Mode Suppression (MS)	Stable	Stable	Chunk Frequency	Good	Better
EWMA Mode Suppression (MS-EWMA)	Unknown	Unknown	1 Peer	Better	Best
Distributed Mode Suppression (DMS)	Stable	Unknown	3 Peers	Best	Best

distributed mode suppression (DMS) samples 3 peers at a time, and uses a (noisy) mode constructed from only those samples.

We simulate all the algorithms by starting the system in a corner case where one of the chunks is available only at the seed server, and observe the evolution of the system afterwards. A comparison is presented in Table I, where m is the number of chunks that the file is divided into. An additional dimension that we explore is the impact on chunk diversity engendered by being able to pick a chunk from the set possessed across multiple peers, i.e., choosing one chunk from one randomly chosen peer, versus choosing one chunk from the chunk-set of three randomly chosen peers. We observe through simulations that mode suppression actually does come very close to attaining a uniform distribution, and has a comparable sojourn time to group suppression. The two variants of MS preformed the best overall, with the version of DMS that can download a chunk from the chunk-set of 3 random peers being near-optimal in terms of sojourn time.

II. SYSTEM MODEL

We consider a P2P file sharing system for a single file divided into m chunks. This file sharing system has a unique seed that has all m chunks, and the seed stays in the system indefinitely. Peers arrive according to a Poisson process with rate λ. Each incoming peer arrives without any chunks and stays in the system till it obtains all m chunks of the file. In this model, a peer leaves as soon as it has all m chunks of the file. The peers can receive the chunks in two ways, either directly from the seed or from other peers.

Whenever the seed or a peer contact another peer, it is deemed as a contact. Therefore, each peer and the seed have individual contact processes corresponding to the sequence of contact instants. Upon contact the seed or the peer transfer a missing chunk to the contacted peer, according to a *chunk selection policy*. When chunk selection policy depends solely on the current state of the system, it is called a Markov chunk selection policy.

A. Contact Processes

The time interval between two contacts are assumed to be random, independent, and identically exponentially distributed, i.e. all contact processes are assumed to be independent and Poisson. The Poisson contact rate for the seed is assumed to be U, and each peer is assumed to have a common contact rate of μ.

B. State space

At any time t, the number of peers in the system with a proper subset of chunks $S \subset [m]$ is denoted by $X_S(t) \in \mathbb{N}_0 \triangleq \{0, 1, \dots\}$. The system at time t can be represented by the state

$$X(t) = (X_S(t) : S \subset [m]).$$

The total number of peers at any time t is denoted by

$$|X(t)| = \sum_{S \subset [m]} X_S(t).$$

For any Markov chunk selection policy, the continuous time process $\{X(t), t \geqslant 0\}$ is Markov with countable state space $\mathcal{X} \triangleq \mathbb{N}_0^{\mathcal{P}([m]) \setminus [m]}$. The *stability region* is defined as the set of arrival rates λ, for which the continuous time Markov chain $X(t)$ is positive recurrent.

C. State transitions

The generator matrix for the process $X(t)$ is denoted by Q. For this continuous time Markov chain, there can only be a single transition in an infinitesimal time. We denote the system state as $x \in \mathcal{X}$ just before any transition, and let e_S be the unit vector in the dimension corresponding to a proper subset $S \subset [m]$.

There are three types of possible transitions. First type of state transition is the arrival of a new peer, that leads to an increase in the number of peers with no chunks. The corresponding transition rate is denoted by

$$Q(x, x + e_\emptyset) = \lambda.$$

Second and third type of transitions occur, when a peer with $S \subset [m]$ chunks receives a chunk $j \notin S$ from the contacting seed/peer. In both these cases, the next state is denoted by $\mathcal{T}_{S,j}(x)$. Second type of state transition occurs when the reception of new chunks doesn't lead to a departure. This transition is denoted by

$$\mathcal{T}_{S,j}(x) \triangleq x - e_S + e_{S \cup \{j\}}, \qquad x_S > 0, |S| < m - 1.$$

Third type of state transition occurs for a peer with $m-1$ chunks, which departs the system after getting the last chunk upon contact. This transition is denoted by

$$\mathcal{T}_{S,j}(x) \triangleq x - e_S, \qquad x_S > 0, |S| = m - 1.$$

At a system state x, if the contacting source has T chunks and the contacted receiving peer has S chunks, then the set of available chunks that can be transferred is $T \setminus S$. Selection of which chunk to transfer is called the *chunk selection policy*, that governs the evolution of the process $X(t)$. In particular, the last two transition rates $Q(x, \mathcal{T}_{S,j}(x))$ can only be computed for a specific Markov chunk selection policy. We describe the proposed chunk selection policy and the corresponding transition rates in the following section.

III. Mode Suppression policy

In this section, we describe the mode suppression policy and provide its rate transition matrix. First, let us establish some notation. The set of allowable transfers from a peer with set of chunks T to a peer with set of chunks S, is denoted by $A(x, T, S) \subseteq T \setminus S$, and the cardinality of this set is denoted by $h(x, T, S)$, that takes integral values between 0 and m. Recall that the seed has all the chunks, and hence the set of allowable chunk transfers by the seed is $A(x, [m], S)$. Below, we describe the specifics of selecting the set of allowable transfers.

If there are no peers in the system, there is no need for chunk transfer. Hence without any loss of generality, we consider the mode suppression policy when there exist peers in the system, or $|x| > 0$. Here, we assume that each peer has the knowledge of all chunk frequencies in the system. Frequency of the jth chunk is

$$\pi_j(x) \triangleq \frac{\sum_{j \in S} x_S}{|x|}.$$

The chunks that attain the highest frequency $\arg\max\{\pi_j(x) : j \in [m]\}$ are called the modes of the chunk frequencies. The set of modes is defined as

$$I(x) \triangleq \{i \in [m] : \pi_i(x) \geq \pi_j(x), \forall j \neq i\}.$$

The mode suppression policy restricts transmission of chunks that belong to the set of modes. Specifically, when the index set $I(x)$ is a strict subset of all chunks, the contacting source excludes the most popular chunk(s) (i.e., the modes) from the set of allowable transfers. Otherwise, when all chunks are equally popular, the source allows all possible transfers. Mathematically, one can write the allowable transfer set for mode suppression policy as

$$A(x, T, S) = \begin{cases} T \setminus (S \cup I(x)), & I(x) \subset [m], \\ T \setminus S, & I(x) = [m]. \end{cases}$$

From the superposition of independent Poisson contact processes, the rate at which either the seed or one of the peers with chunk j contact any peer is also Poisson with the aggregate rate

$$R_j(x) \triangleq U + \mu \sum_{T : j \in T} x_T = U + \mu |x| \pi_j(x).$$

The probability of the source contact process contacting a peer with chunk subset S is $\frac{x_S}{|x|}$. If the contacting source has T chunks, then it can transfer one out of $h(x, T, S)$ available chunks to the contacted peer with S chunks. The transition of type $\mathcal{T}_{S,j}$ occurs when either the seed or one of the peers with chunk $j \notin S$ contact a peer with chunks S, and transfer chunk j among all the possible choices. From the thinning and superposition of independent Poisson processes, we can write for $j \notin S$ and $x_S > 0$

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

IV. Stability Region of Mode Suppression Policy

In this section we characterize the stability region of mode suppression policy.

Theorem 1. *The stability region of the mode suppression policy is $\lambda \geq 0$ for file-sharing systems with at least two chunks, and positive contact rates U, μ.*

Proof: To prove the positive recurrence of the continuous time Markov chain $X(t)$, we employ Foster-Lyapunov criteria [14]. We consider the following Lyapunov function,

$$V(x) = \sum_{i=1}^{m} \Big((\overline{\pi} - \pi_i)|x| \Big)^2 + C_1 \big((1 - \overline{\pi}) \big) |x| + C_2 \Big(M - \sum_{i=1}^{m} \pi_i |x| \Big)^+, \tag{1}$$

where, C_1, C_2 and M are positive constants that depend on m, λ, U, μ, and $\overline{\pi} = \max_i \pi_i$. Note that the explicit dependency of $\pi(x)$ on x is not shown for simplicity.

The intuition behind this Lyapunov function is as follows. Since the nominal objective is to attain a uniform distribution, we should expect that the policy should promote negative Lyapunov drift whenever the current state differs from uniformity. Hence, our Lyapunov function is designed to penalize for the cases where chunks have differing frequency, where some might have zero frequency, and where all have zero frequency.

The expected rate of change of potential function for a Markov process $X(t)$ from state x is called the mean drift from this state, and is given by

$$\sum_{y} Q(x, y)(V(y) - V(x)) = QV(x).$$

Mean drift from a state x for the Markov process $X(t)$ in terms of its generator matrix Q can be written as

$$QV(x) = Q(x, x + \emptyset)(V(x + \emptyset) - V(x)) + \sum_{j \in [m]} Q(x, \mathcal{T}_{S,j}(x))(V(\mathcal{T}_{S,j}(x)) - V(x)).$$

First, we compute the mean drift corresponding to a new peer arrival. The arrival of a new peer does not change the number of peers with chunk $j \in [m]$. However, it does lead to a unit increase in the number of peers in the system. That is,

$$Q(x, x + e_\emptyset)(V(x + e_\emptyset) - V(x)) = \lambda C_1.$$

The rest of the proof proceeds as follows. We divide the states into two cases when the chunk frequency is (i) non-uniform

and (ii) uniform, and in each case we show that the drift is negative.

Case 1: $I(x) \subsetneq [m]$**:** In this case, no popular content is allowed to be transferred. Hence, any transition of type $\mathcal{T}_{S,j}(x)$ occurs only for $j \notin I(x)$. When $S \cup \{j\} \subsetneq [m]$, this transition leads to unit increase in the number of peers with chunk j, and no change in the number of peers with other chunks. The corresponding change in potential function for $S \cup \{j\} \subsetneq [m]$ and $M \in \mathbb{Z}_+$ equals

$$V(\mathcal{T}_{S,j}(x)) - V(x) = 1 - 2(\overline{\pi} - \pi_j)|x| - C_2 1_{\{M > \sum_i \pi_i |x|\}}.$$

Since the number of popular chunks has to be at least unity, this difference is strictly negative for all non-zero states x. For this transition, we can trivially bound the cardinality of the allowable transfers by $\sup_T h(x, T, S) \leq |S^c|$. This provides a lower bound on the transition rate

$$Q(x, \mathcal{T}_{S,j}(x)) \geq \frac{x_S}{|S^c||x|} R_j.$$

When $S = \{j\}^c$, it is clear that the set of allowable transfer is $\{j\}$ for the contacting sources. Hence, $h(x, T, S) = |S^c| = 1$ and the transition rate is

$$Q(x, \mathcal{T}_{S,j}(x)) = \frac{x_S}{|x|} R_j.$$

Further, the transition $\mathcal{T}_{S,j}(x)$ leads to a departure from the system of peer with $S = \{j\}^c$ chunks. That is, this transition leads to a unit decrease in number of peers with chunks other than j. The change in potential function $V(\mathcal{T}_{S,j}(x)) - V(x)$ for the transition from state x to state $\mathcal{T}_{S,j}(x)$, for $S \cup \{j\} = [m]$ and $M \in \mathbb{Z}_+$, is upper bounded by

$$1 - 2(\overline{\pi} - \pi_j)|x| + C_2(m-1)1_{\{M+m-1 > \sum_i \pi_i |x|\}}.$$

The fraction of users that have all the pieces except jth piece is denoted by $\gamma_j(x) \triangleq \frac{x_{\{j\}^c}}{|x|}$, and the aggregate number of chunks in the system at all peers is denoted by $r \triangleq \sum_{S \subseteq [m]} |S| x_S = \sum_{i \in [m]} \pi_i |x|$. Aggregating all the above results and notations, and observing that $|S^c| \leq m$, we can find an upper bound on the mean drift from state x as

$$C_1\lambda - \sum_{j \notin I(x)} \frac{R_j}{m} \Big[(2(\overline{\pi} - \pi_j)|x| - 1)(1 - \pi_j) + C_2(1 - \pi_j - \gamma_j) 1_{\{M > r\}} - \gamma_j C_2 m(m-1) 1_{\{M+m-1 > r\}} \Big]. \tag{2}$$

We will divide the state space in to three regions and show that in each region the drift is negative. The details are given in the Appendix A. Now, we consider the uniform chunk frequency case.

Case 2: $I(x) = [m]$**:** In this case, the chunk frequencies are identical, that is $\overline{\pi} = \pi_i$ for each chunk $i \in [m]$, and any chunk j can be transferred. This also implies that the contact rate $R_j = U + \mu \pi_j |x|$ is uniform for all chunks j, and can be denoted by $R = U + \mu \overline{\pi} |x|$. For $S \subsetneq \{j\}^c$, a transition of type $\mathcal{T}_{S,j}(x)$ doesn't lead to any departure from the system. The number of peers with chunk j has a unit increase by one, and chunk j becomes the popular chunk. There is no change in the number of peers for other chunks. Hence the potential change, due to this transition, is

$$V(\mathcal{T}_{S,j}(x)) - V(x) = m - 1 - C_1 - C_2 1_{\{M > r\}}.$$

For $S = \{j\}^c$, a transition of type $\mathcal{T}_{S,j}(x)$ leads to the departure of the receiving peer from the system. In this case, the number of peers with chunk j remains same, the number of peers having other chunks has a unit decrease. The potential change due to this transition, is

$$V(\mathcal{T}_{S,j}(x)) - V(x) \leq m - 1 - C_1 - C_2(m-1) 1_{\{M+m-1 > r\}}.$$

Using the same techniques as in Case 1, we can upper bound the drift of state x by,

$$C_1\lambda - R\Big[(C_1 - m + 1)(1 - \overline{\pi}) + C_2(1 - \overline{\pi} - \gamma_j) 1_{\{M > r\}} - \gamma_j C_2 m(m-1) 1_{\{M+m-1 > r\}} \Big]. \tag{3}$$

Similar to Case 1, we will divide the state space in to three regions and show that in each region the drift is negative. The details are given in the Appendix A. ■

V. Distributed Policies

Mode suppression requires global information of the chunk frequencies. We propose two policies which circumvent this requirement, and study their performance through simulations

A. EWMA Mode Suppression:

Under this policy, each peer calculates the empirical marginal chunk frequencies based only on the chunks possessed by all peers that it has met until (and including) the current time. The marginal chunk frequency is calculated using an Exponentially Weighted Moving Average (EWMA) to take into account both history and present, and the mode of this estimate is suppressed.

B. Distributed Mode Suppression Policy:

Under *distributed mode suppression* (DMS), a peer contacts three other peers at random, and among the chunks available with more than one peer, we define the *local mode* to be the chunk(s) with greatest frequency. The peer is allowed to download any chunk that is not part of the local mode. Any chunk may be downloaded if all chunks are equally frequent. The proof of stability of DMS for the case of a file with $m = 2$ chunks is similar to the proof of Rare Chunk policy given in [12], and is omitted. Stability for the case $m > 2$ chunks is left as conjecture.

VI. Simulation Results

In this section, we show the results from numerical simulations that illustrate the performance of different chunk selection policies. Recall that our candidate policies are (i) random chunk selection, (ii) rarest-first, (iii) rare chunk, (iv) common chunk, (v) group suppression, (vi) mode suppression, (vii) mode suppression-EWMA, and (viii) distributed mode

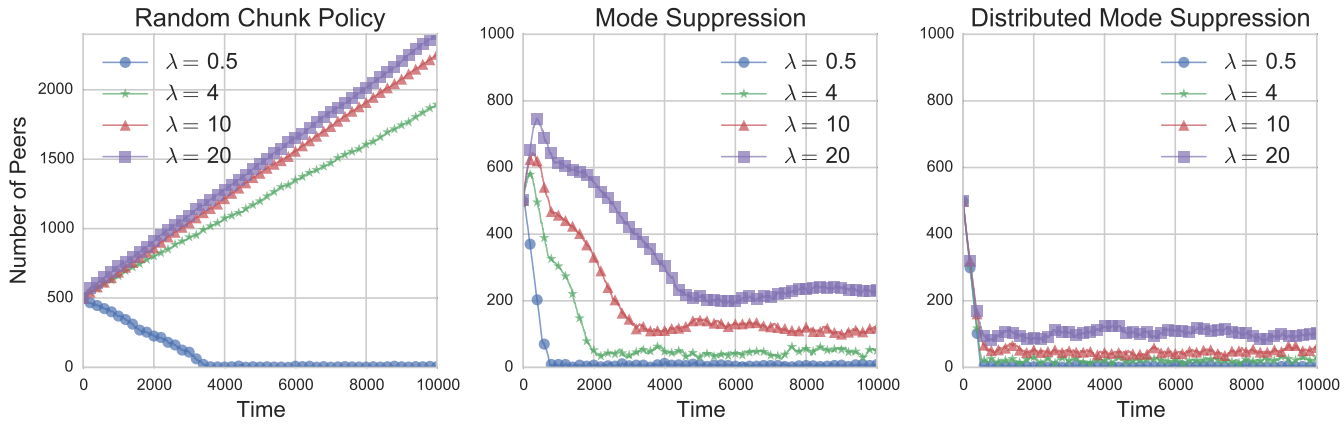

Fig. 1. Number of peers in the system when $m = 5$, $U = 1$ and $\mu = 1$. Random becomes unstable in some cases, whereas MS and DMS are always stable.

suppression. A description of these policies can be found in Section I and Section V. For all the simulations, we kept the peer contact rate and seed contact rate to 1. To simulate a Poisson process, we make use of the fact that inter arrival times of a Poisson process follow an exponential distribution. Each peer in the system, including the seed, generates an exponential random variable with mean $\frac{1}{\mu} = \frac{1}{U} = 1$, and the peer or the seed with the smallest value gets a chance to contact another peer. After the contact, a chunk transfer takes place instantaneously according to the chosen chunk selection policy.

A. Stability of Mode Suppression Policy:

We begin the simulation with 500 empty peers. Whenever a peer receives all the chunks, it immediately leaves the system. In Figure 1, we plot the number of peers in the system as time progresses for three different polices, namely (i) random chunk selection, (ii) mode suppression, and (iii) distributed mode suppression. The purpose of simulating the random chunk selection policy, which is known to be unstable, is to provide a visual representation of what an unstable regime appears like, in order to compare with stable policies. In this simulation, the number of chunks is taken as 5, and the peer arrival rate (λ) is varied. We observe that when the peer arrival rate is less than seed rate ($\lambda = 0.5 < 1 = U$), the random chunk selection policy is stable and in all other cases $\lambda > U$, the number of peers grows large and the system is unstable. However, in case of mode suppression and distributed mode suppression, the system is stable for all arrival rates.

B. Chunk Frequency Evolution:

A stable chunk selection policy has to be robust to the one-club state. In other words, a stable policy should be able to boost the frequency of a rare chunk. To see how different policies handle the one-club situation, we start the system with 500 peers that have all the chunks except first chunk (i.e., all peers are part of the one-club). In Figure 2, we plot the evolution of the chunk frequency for different policies under this initial condition. We see that when using the rarest-first policy, the rare chunk remains rare and abundant chunks remain abundant—a clear sign of instability. In all stabilizing policies, the rare chunk is made available by giving priority to that chunk in some way. For instance, in case of mode suppression, no other chunk will be transmitted until the frequency of the rare chunk is equal to the frequency of all other chunks. Once this happens, the frequencies of the different chunks remain almost same, and hence we only see a thin spread across the frequencies. Other policies also manage to bring the rare chunk back into circulation and the corresponding statistics become similar to all other chunks. We also observe that the *stabilization time* to increase the frequency of rare chunk to the same level as that of other chunk frequencies, is shorter for MS and DMS when compared to other algorithms.

C. Sojourn times:

In addition to stability, an important performance metric is the sojourn time of a peer, which is defined as the amount of time a peer spends in the system collecting all chunks before leaving. In Figure 3, the peer arrival rate is fixed at $\lambda = 30$ and we plot the mean stationary sojourn times of the peers for different policies, for different values of file chunks m. The stationary sojourn times are obtained by running the system for a long period of time and ignoring the first 2000 peers that left the system. Our goal is to evaluate how effectively the algorithms use their information on chunk statistics. Further, we also wish to study the effect of chunk diversity provided through being able to choose a chunk from 1 versus 3 peers. Thus, we have two versions of each algorithm that both use identical chunk statistics obtained through sampling all or some peers as per the algorithm. However, the first version (indicated using A-1, where A is the algorithm) can obtain any one chunk from those possessed by 1 randomly selected peer, while the second (indicated by A-3, where A is the algorithm) can pick any one chunk from the set of chunks possessed by 3 randomly selected peers. We see that GS and MS have

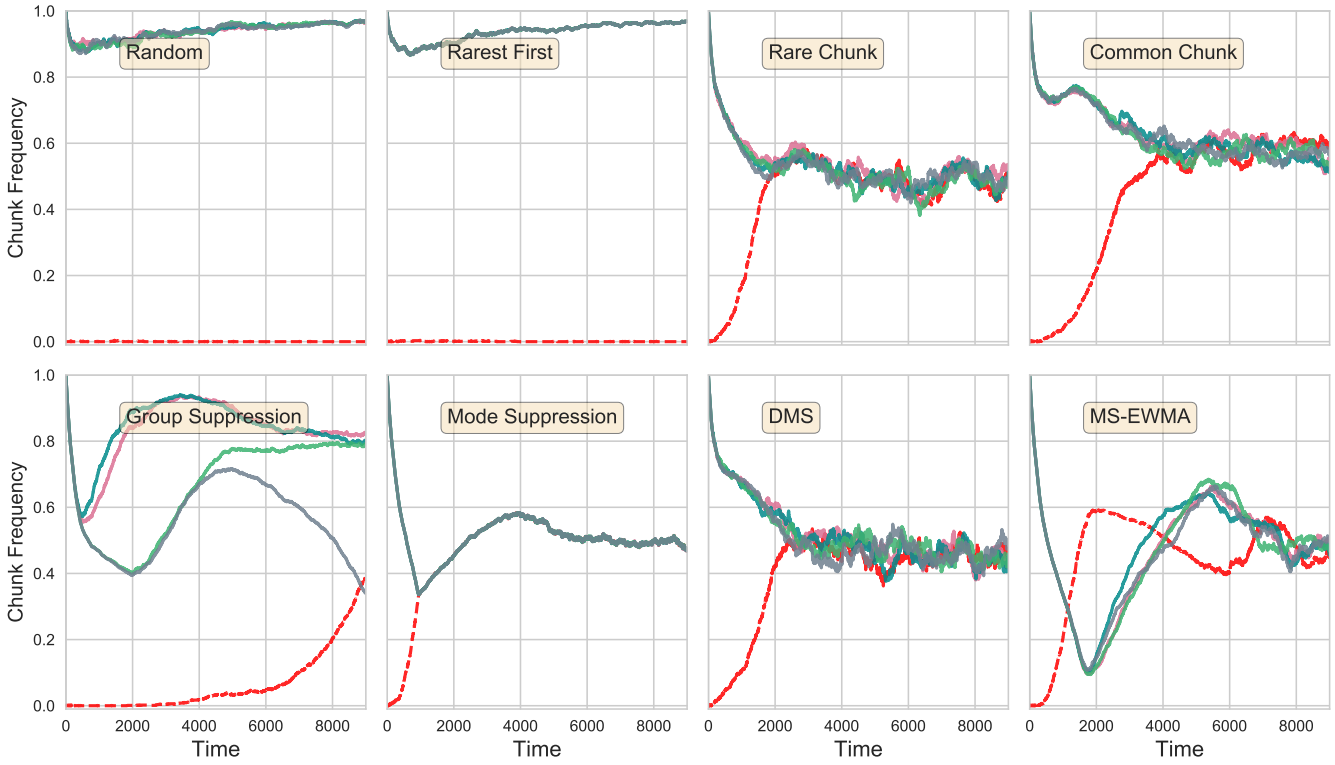

Fig. 2. Chunk frequency evolution in a system with $m = 5$ chunks under different policies when starting from the state of a "missing-chunk" (whose frequency is indicated by a red/dashed line). Rarest-first is clearly unstable, since it cannot recover, whereas the other protocols manage to bring the chunk back into peer circulation and stabilize the system.

comparable perfromances, while DMS has the least average sojourn times among all policies in both scenarios. Also note that on average, the stationary sojourn time for DMS-3 is essentially the same as the number of chunks m, i.e., peers collect close to 1 chunk per unit time on average. Since the rate of peer contact is 1, this fact indicates that among the algorithms compared, DMS-3 attains the best possible trade-off between suppression (to keep peers in the system) and sharing (to enable peers to gather chunks).

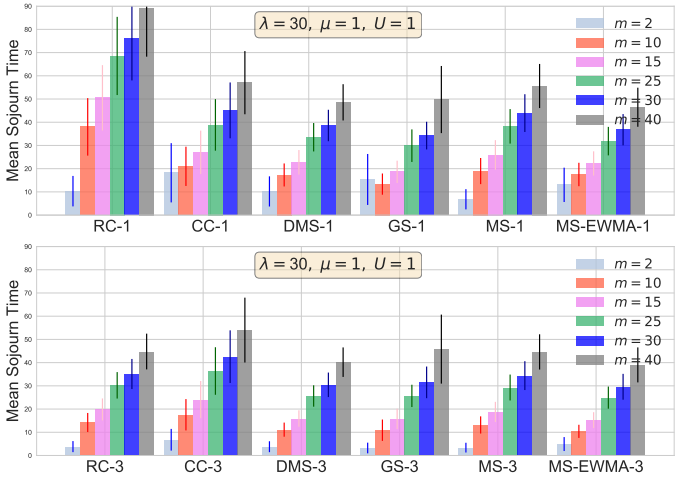

Fig. 3. Stationary mean sojourn times of stable policies for different values of m.

VII. Conclusion

In this work, we analyzed the scaling behavior of a P2P swarm with reference to its stability when subjected to an arbitrary arrival rate of peers. It has been shown earlier that not all chunk sharing policies are stable in such a regime, and our goal was to design a simple and stable policy that yields low sojourn times. Our main observation was that, contrary to the traditional approach of boosting the availability of rare chunks, preventing the spread of the most frequent chunk(s) yields a simple and stable policy that we entitled mode suppression (MS). We analytically proved its stability, and also described version of the policy entitled distributed mode suppression (DMS) that works on the same principle. DMS only uses locally sampled statistics using three randomly selected peers, and yields low (near-optimal) sojourn times in numerical studies. An additional observation is that DMS-3 attains this performance, i.e., it appears that the chunk diversity provided by choosing a chunk from the set possessed by three randomly selected peers is sufficient for optimality.

Our results indicate that there is a delicate trade-off between sharing (i.e., uploading a useful chunk if at all possible) and suppression (i.e., trying to reduce chunk transfers to keep peers in the system so that they can help others). The chunk selection policy has a fundamental impact on this trade-off. On one hand, by suppressing some chunk sharing (as in the GS, MS or DMS algorithms), we can ensure peers stay longer at the expense of increasing sojourn time, with too much suppression

leading to instability. On the other hand, trying too hard to be work conserving (maximizing sharing as in random or RF) with the idea of reducing sojourn times can lead to instability due to chunk starvation. Our future work will be to obtain a deeper understanding of the trade-off between suppression and sharing to minimize sojourn time for stable protocols.

REFERENCES

[1] B. Cohen, "Incentives build robustness in BitTorrent," in *Workshop on Economics of Peer-to-Peer systems*, vol. 6, 2003, pp. 68–72.

[2] SANDVINE, "Global Internet phenomena report. 2016," *URL: https://www.sandvine.com/trends/global-internet-phenomena/*, 2016.

[3] 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.

[4] 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.

[5] D. Qiu and R. Srikant, "Modeling and performance analysis of BitTorrent-like peer-to-peer networks," in *ACM SIGCOMM computer communication review*, vol. 34, no. 4. ACM, 2004, pp. 367–378.

[6] X. Yang and G. De Veciana, "Performance of peer-to-peer networks: Service capacity and role of resource sharing policies," *Performance evaluation*, vol. 63, no. 3, pp. 175–194, 2006.

[7] S. Shakkottai and R. Johari, "Demand Aware Content Distribution on the Internet," *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, April 2010.

[8] B. Fan, J. Lui, and D.-M. Chiu, "The design trade-offs of BitTorrent-like file sharing protocols," *IEEE/ACM Transactions on Networking (TON)*, vol. 17, no. 2, pp. 365–376, 2009.

[9] J. Zhu and B. Hajek, "Stability of a peer-to-peer communication system," *IEEE Transactions on Information Theory*, vol. 58, no. 7, pp. 4693–4713, 2012.

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

[11] 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.

[12] 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.

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

[14] S. P. Meyn and R. L. Tweedie, *Markov chains and stochastic stability*. Springer Science & Business Media, 2012.

APPENDIX

AUXILLIARY RESULTS

Lemma 2. *For each state x, the fraction of peers with least popular chunk is upper bounded by $\frac{m-1}{m}$.*

Proof: Any peer in the system can have at most $m-1$ pieces, or else it would leave the system. The result follows from bounding the total number of pieces in the system as

$$m\underline{\pi}|x| \leq \sum_{i\in[m]} \pi_i|x| = \sum_{S\subsetneq[m]:S\neq\emptyset} |S|x_S \leq (m-1)|x|.$$

■

Recall $\overline{\pi}$ and $\underline{\pi}$ respectively denote the fraction of peers that have the most and least popular chunks. When all chunks are equally popular, then $\underline{\pi} = \overline{\pi} = \pi_j$ for each chunk j.

When the set of most popular chunks $I(x) \subsetneq [m]$, the least popular chunk is denoted by $\underline{j} \notin I(x)$, and $\underline{\pi} = \pi_{\underline{j}}$. In this case, the least popular chunks are possessed by at least one less peer than the corresponding number for other chunks. That is, when $\pi_i > \underline{\pi}$, we have $\pi_i|x| - \underline{\pi}|x| \geqslant 1$. Specifically, $2(\overline{\pi} - \underline{\pi})|x| - 1 \geqslant 1$.

Lemma 3. *Let $K_1 > 0, K_2 < 2$ be constants. For each $\epsilon > 0$ there exists an $N(K_1, K_2, \epsilon) \in \mathbb{R}^+$, such that if $\overline{\pi}|x| \geq N$, then for $I(x) \subsetneq [m]$, we have*

$$C_1\lambda - K_1 \sum_{j\notin I(x)} R_j(1-\pi_j)(2(\overline{\pi}-\pi_j)|x| - K_2) < -\epsilon.$$

Proof: Lower bounding the summation over $[m] \setminus I(x)$ by a single term corresponding to the least popular chunk $\underline{j}$, and lower bounding $1-\underline{\pi}$ by $\frac{1}{m}$ from Lemma 2, we can upper bound the LHS of the above equation by

$$C_1\lambda - \frac{K_1}{m} R_{\underline{j}}(2(\overline{\pi} - \pi_{\underline{j}})|x| - K_2).$$

To upper bound the above equation, we define η as the ratio of number of peers with the least and the most popular chunks. That is, $\underline{\pi} = \eta\overline{\pi}$ and $\eta \in \left[0, 1 - \frac{1}{\overline{\pi}|x|}\right]$, and we can write

$$R_{\underline{j}}(2(\overline{\pi} - \pi_{\underline{j}})|x| - K_2) = (U + \eta\overline{\pi}\mu|x|)(2\overline{\pi}(1-\eta)|x| - K_2)$$
$$= -K_2U + 2U\overline{\pi}|x|(1-\eta) - K_2\eta\overline{\pi}|x|\mu + 2\overline{\pi}^2|x|^2\mu\eta(1-\eta).$$

Let us denote the above quadratic expression in η by $g(\eta)$. We can check that $g''(\eta) = -4\overline{\pi}^2|x|^2\mu < 0$. Hence, the function $g(\eta)$ is strictly concave and quadratic in η, with a unique maximum. This function attains minimum at the boundary values of η, and we can lower bound $g(\eta)$ as

$$g(\eta) \geq \min\{g(\eta) : \eta \in [0, 1 - \frac{1}{\overline{\pi}|x|}]\} = g(0) \wedge g(1 - \frac{1}{\overline{\pi}|x|})$$
$$= \frac{K_1}{m}\left[U(2\overline{\pi}|x| - K_2) \wedge (2 - K_2)(U + \mu(\overline{\pi}|x| - 1)\right].$$

The result follows since $C_1\lambda - \frac{K_1}{m}g(\eta) < -\epsilon$ if $\overline{\pi}|x| > N$, where we can choose N to be

$$\max\left\{\frac{1}{2}\left(\frac{C_1\lambda+\epsilon}{\frac{K_1}{m}U} + K_2\right), \left(\frac{C_1\lambda+\epsilon}{\frac{K_1}{m}(2-K_2)\mu} - \frac{U}{\mu} + 1\right)\right\}.$$

■

Corollary 4. *Let $K_1 > 0, K_2 < 2$ be constants, $\overline{\pi}(x) \geq \delta$, and $I(x) \subsetneq [m]$. Then, for each $\epsilon > 0$, we can find an L such that when $|x| > L$,*

$$C_1\lambda - K_1 \sum_{j\notin I(x)} R_j(1-\pi_j)(2(\overline{\pi}-\pi_j)|x| - K_2) < -\epsilon.$$

Proof: Fix $\epsilon > 0$, we choose the N from Lemma 3 and $L = \frac{N}{\delta}$. Then $\overline{\pi}|x| \geq N$, and the inequality holds. ■

DETAILS OF THEOREM 1

For any $\delta \in (0,1)$, we can partition the state space into following three regions,

$$\mathcal{R}_1 = \{\overline{\pi} \geq \delta\}, \mathcal{R}_2 = \mathcal{R}_1^c \cap \{\overline{\pi}|x| \geq \frac{M}{m}\}, \text{ and}$$
$$\mathcal{R}_3 = \mathcal{R}_1^c \cap \{\overline{\pi}|x| < \frac{M}{m}\}.$$

For each $i \in [3]$, we can further subdivide each region R_i into

$$\mathcal{R}_{i1} = \{x \in \mathcal{R}_i, I(x) \subsetneq [m]\}, \quad \mathcal{R}_{i2} = \{x \in \mathcal{R}_i, I(x) = [m]\}.$$

All these regions have countable number of states. We will prove that in each region $\mathcal{R}_{ij}$ where $i \in \{1,2,3\}$ and $j \in \{1,2\}$, the mean drift $QV(x) < -\epsilon$ for all states $x \in R_{ij} \setminus F_{ij}$ for some finite set F_{ij} dependent on ϵ. We fix $\epsilon > 0$, and choose $N(K_1, K_2, \epsilon)$ from Lemma 3.

Lemma 5. *For states in the region $\mathcal{R}_1$, the total number of chunks is lower bounded by $\delta|x|$.*

Proof: The total number of chunks in the system r is lower bounded by the number of most popular chunk, i.e.

$$r = \sum_{i \in [m]} \pi_i |x| \geq \overline{\pi}|x|.$$

The result follows since $\overline{\pi} \geq \delta$ in the region $\mathcal{R}_1$. ∎

Lemma 6. *For states in the region $\mathcal{R}_2 \cup \mathcal{R}_3$, the fraction of peers γ_j with the set of chunks $\{j\}^c$ is upper bounded by δ.*

Proof: We can upper bound the number of peers with the set of chunks $\{j\}^c$ as

$$x_{\{j\}^c} \leq \sum_{S: i \in S, i \neq j} x_S = |x| \pi_i 1_{\{i \neq j\}} \leq |x| \overline{\pi}.$$

The result follows since $\overline{\pi} < \delta$ in region $\mathcal{R}_2 \cup \mathcal{R}_3$. ∎

Region $\mathcal{R}_{11}$: For this region, we choose $N_{11} \triangleq N(\frac{1}{m}, 1, \epsilon)$ from Lemma 3, to define the finite set

$$F_{11} \triangleq \{\delta|x| \leq (M + m - 1) \vee N_{11}\}.$$

From Lemma 5, it follows that for the states $x \in \mathcal{R}_{11} \cap F_{11}^c$, the indicator corresponding to the event $\{M + m - 1 > r\}$ is zero in the mean drift of (2). Therefore, Corollary 4 implies that the mean drift in (2) is upper bounded by $-\epsilon$.

Region $\mathcal{R}_{12}$: For this region, we choose $C_1 > (m-1)$ and

$$N_{12} \triangleq \frac{m(C_1\lambda + \epsilon)}{\mu\delta(C_1 - m + 1)},$$

to define the finite set $F_{12} \triangleq \{\delta|x| \leq (M + m - 1) \vee N_{12}\}$. Let $x \in \mathcal{R}_{12} \cap F_{12}^c$. In this region, the indicator corresponding to the event $\{M + m - 1 > r\}$ is zero in the mean drift of (3). By choosing a lower bound on common contact rate $R \geq \mu\overline{\pi}|x|$, complement of the frequency $1 - \overline{\pi} \geq \frac{1}{m}$ from Lemma 2, and on the frequency $\overline{\pi} \geq \delta$ since $x \in R_1$, we can bound the mean drift in equation (3) by $-\epsilon$.

Region $\mathcal{R}_{21}$: We can upper bound the fraction of peers $\gamma_j < \delta$ by Lemma 6, and upper bound $1 < m(1 - \pi_j)$ from Lemma 2. Thus, we can upper bound

$$\gamma_j C_2 m(m-1) 1_{\{M+m-1>r\}} \leq \delta C_2 (1 - \pi_j) m^2 (m-1).$$

Hence, we can upper bound the mean drift in (2) with

$$C_1\lambda - \sum_{j \notin I(x)} \frac{R_j(1 - \pi_j)}{m} \left(2(\overline{\pi} - \pi_j)|x| - 1 - \delta C_2 m^2 (m-1)\right).$$

When $\delta C_2 m^2 (m-1) < 1$, we can choose from Lemma 3

$$N_{21} \triangleq N\left(\frac{1}{m}, \delta C_2 m^2 (m-1) + 1, \epsilon\right).$$

For each $x \in \mathcal{R}_{21}, \overline{\pi}|x| \geq \frac{M}{m}$, and hence by selecting $\frac{M}{m} > N_{12}$, we ensure that the mean drift in (2) is bounded above by $-\epsilon$ in this region.

Region $\mathcal{R}_{22}$: In this region, $\underline{\pi}|x| \geq \frac{M}{m}$, and hence the total number of chunks $r = \sum_i \pi_i |x| \geq M$. We again choose $C_1 > m - 1$ and bound the common contact rate $R = U + \underline{\pi}\mu|x| \geq \frac{M}{m}\mu$. We also use the upper bound for fraction of peers $\gamma_j < \delta$ from Lemma 6, and the lower bound $1 - \underline{\pi} \geq \frac{1}{m}$ from Lemma 2, to upper bound the mean drift from equation (3) with

$$C_1\lambda - \frac{M}{m^2}\mu\left(C_1 - m + 1 - \delta C_2 m^2 (m-1)\right).$$

Hence, the mean drift for all states $x \in \mathcal{R}_{22}$ is bounded above by $-\epsilon$, if $\delta C_2 m^2 (m-1) < C_1 - m + 1$ and

$$\frac{M}{m} > N_{22} \triangleq \frac{\left(C_1\lambda + \epsilon\right)m}{\mu\left(C_1 - m + 1 - \delta C_2 m^2 (m-1)\right)}.$$

Region $\mathcal{R}_{31}$: In this region, $m\overline{\pi}|x| < M$, and hence the total number of chunks $r = \sum_i \pi_i |x| \leq m\overline{\pi}|x| < M$. Therefore both the indicator functions associated with the events $\{M > r\}$ and $\{M + m - 1 > r\}$ equal unity in the equation (2). Recall that $\gamma_j < \delta$ by Lemma 6, $1 - \underline{\pi} > \frac{1}{m}$ by Lemma 2, $R_j \geq U$, and $2(\overline{\pi} - \underline{\pi})|x| - 1 > 0$ for state such that $I(x) \subsetneq [m]$. Summarizing all these results, and lower bounding the summation over $I^c(x)$ by the least popular term, we can upper bound the mean drift with

$$C_1\lambda - C_2\frac{U}{m}\left(\frac{1}{m} - \delta(m(m-1)+1)\right).$$

By choosing $\delta < \frac{1}{2m(m(m-1)+1)}$ and $C_2 = \frac{2m^2(C_1\lambda+\epsilon)}{U}$ we can bound the mean drift from state $x \in \mathcal{R}_{31}$ with $-\epsilon$.

Region $\mathcal{R}_{32}$: Similar to region $\mathcal{R}_{31}$ both the indicator functions in equation (2) will be equal to unity. In this region, $\overline{\pi} = \underline{\pi}$. Using the bounds $\gamma_j < \delta, 1 - \underline{\pi} > \frac{1}{m}, R \geq U$, and $C_1 > m - 1$, we can upper bound the mean drift in (3) for $x \in \mathcal{R}_{32}$ with

$$C_1\lambda - UC_2\left(\frac{1}{m} - \delta(m(m-1)+1)\right)$$

By choosing $\delta < \frac{2m-1}{2m^2(m(m-1)+1)}$ and $C_2 = \frac{2m^2(C_1\lambda+\epsilon)}{U}$ we can bound the drift with $-\epsilon$.

Choosing Parameters:

Following choice of C_1, C_2, M satisfy all the constraints,

$$C_1 > m - 1, \quad C_2 = \frac{2m^2(C_1\lambda + \epsilon)}{U}, \quad M > m\max\{N_{21}, N_{22}\},$$

where $m^2(m-1)\delta$ equals

$$\min\left\{\frac{m - \frac{1}{2}}{m + \frac{1}{m-1}}, \frac{\frac{m}{2}}{m + \frac{1}{m-1}}, \frac{1}{C_2}, \frac{C_1 - m + 1}{C_2}\right\}.$$