Topological Properties Affect the Power of Network Coding in Decentralized Broadcast

Di Niu, Baochun Li
Department of Electrical and Computer Engineering
University of Toronto
{*dniu, bli*}*@eecg.toronto.edu*

***Abstract*—There exists a certain level of ambiguity regarding whether network coding can further improve download performance in P2P content distribution systems, as compared to commonly applied heuristics such as rarest first protocols. In this paper, we revisit the problem of broadcasting multiple data blocks from a single source in an overlay network using gossip-like protocols. Our new finding reveals that the marginal benefit of network coding critically depends on the dynamics of network topologies. We show that although network coding is optimal as a block selection mechanism, simple non-coding protocols are close to optimal in complete and random graphs, leading to marginal benefits of network coding. However, network coding demonstrates salient benefits in clustered and time-varying topologies, which are common in real-world systems with ISP-locality mechanisms implemented. Through both theoretical analysis and simulation results, we unveil the underlying reasons behind discrepancies in the power of network coding under different scenarios.**

I. Introduction

Peer-to-peer (P2P) systems, or application-layer overlay networks, have emerged as a powerful tool for broadcasting bulk content in today's Internet. Many practical deployments (e.g. BitTorrent) of such systems are built on a simple design philosophy: each peer connects to a random set of other peers to form a mesh-like overlay network. Gossip-like algorithms are then applied on top of it to disseminate content blocks.

Known as a powerful tool to achieve multicast capacities in directed acyclic graphs (DAGs) [1], [2], *randomized network coding* [2] has been introduced into P2P content dissemination systems (Avalanche [3]). With network coding, each peer is able to encode the blocks it has obtained with a random linear code and transmit the encoded block. Although it has been experimentally shown that network coding reduces download times [3] in BitTorrent-like systems, the subsequent related literature has raised doubts regarding the benefits of network coding in P2P networks.

On one side, network coding has been proved to be optimal in a time-synchronized gossiping model [4], and greatly outperforms a naive sequential dissemination in complete graphs [5]. However, assuming P2P networks as complete graphs, it is shown that network coding cannot offer further benefits over centralized scheduling without coding [6]. Such a view is strengthened by the empirical observation [7] that BitTorrent's rarest first algorithm, as a decentralized protocol, guarantees close-to-ideal diversity of the blocks among peers, and thus applying network coding in such systems cannot be justified.

In this paper, we consider the fundamental problem of distributing multiple data blocks from one or a few sources to all the other nodes in a randomly connected network, with upload bandwidth constraints at peers. The performance metric is the broadcast delay needed for all the peers to obtain all the blocks. We consider classic gossiping algorithms, in which each peer randomly chooses a neighboring peer to upload to during each transmission opportunity, and focus on the fundamental problem of which block to choose for transmission when multiple blocks contend for limited bandwidth resources.

Motivated by the aforementioned debate, we attempt to demystify the power of network coding in P2P networks by asking the following questions: 1) Does randomized network coding achieve the optimal broadcast delay as a block selection protocol? 2) Even if network coding achieves the optimal delay, how much benefit can it bring over reasonably good non-coding protocols such as rarest first, which is also decentralized and requires much lower computational complexity? and 3) Are there any factors that critically affect the marginal benefit of network coding, so much so that such benefit is only substantial under certain circumstances? Note that all these questions require a careful reinvestigation in gossip-based overlay broadcast, as it features a distinctly different model from the DAG, heavily studied in information theory.

Seeking answers to these questions, we first show the delay optimality of network coding in a continuous time model where random transmission delays and arbitrary network topologies are allowed, extending Yeung's optimality result [4] for a discrete time model. We further give a theoretical lower bound on the delay of any gossip algorithms that use random neighbor selection with arbitrary block selection schemes in complete graphs. The theory of approximating Markov population processes with ODEs is used to derive the bound. We numerically show that both network coding and non-coding protocols (rarest first policies) can achieve performance very close to the theoretical limits in complete and random graphs. This means the marginal benefit of network coding is trivial in these graphs.

Motivated by the third question, we proceed to study the impact of network topology on the power of network coding. Due to the ever-increasing burden P2P applications put on network providers [8], a large number of application-level traffic control schemes have been proposed to constrain cross-ISP traffic, reducing the costs to ISPs [8], [9]. However, traffic

This work is supported by NSERC Discovery, CRD and Strategic Grants (RGPIN 238994-06, CRDPJ 379623-08, STPGP 364910-08).

locality induces ISP-based topological clustering, the impact of which on algorithm performance remains to be explored. Time-varying topologies are another phenomenon inherent to P2P networks: peers may actively alter their neighbors to discover better connections or passively do so due to peer dynamics.

While most prior theoretical work analyzes P2P algorithms on *static complete graphs* [5], [6], [10], [11], the impact of *topological clustering and time-varying properties* on algorithm performance remains largely unexplored. To our knowledge, this paper is the first attempt to analyze the impact of these important topological dynamics on the power of network coding in gossip-based overlay broadcast.

We leverage an epidemic spreading model [12] to explicitly bound from below the delay gap between randomized network coding and the local rarest first [7] policy, when the network consists of peer clusters with random global links, and when links may vary over time. The model solves a basic case when two blocks are to be broadcast, due to the significant difficulty in treating inter-dependent random processes. However, what outweighs its direct result is the model's value to shed light on the root causes of performance discrepancy of different algorithms with the presence of these topological dynamics. We extend the results to more general scenarios, including synthetic BitTorrent workloads, through extensive simulations.

We find clustering (traffic locality) and time-varying topologies are two major factors that determine the benefit of network coding in gossip-based overlay broadcast:

- In static clustered topologies, unlike in complete and random graphs, network coding demonstrates significant lower broadcast delays than non-coding protocols. The marginal benefit of network coding exhibits a threshold behavior, depending on certain clustering metrics.
- Time-varying topologies reduce the broadcast delay of network coding while adversely affecting the performance of local rarest first — which critically depends on the accessibility of unbiased global views — amplifying network coding's benefit.
- When time-varying topologies and clustering are considered together, network coding's benefit exhibits complex manners depending on the topological dynamics.

The remainder of this paper is organized as follows. The related work is reviewed in Sec. II. We formulate the problem in Sec. III and prove the delay-optimality of network coding in arbitrary graphs with a continuous-time model in Sec. IV. In Sec. V, we give a lower bound on the broadcast delay of any gossip algorithms that fall into a certain class in complete graphs, and show that both coding and non-coding protocols can achieve performance close to the theoretical limits in complete and random graphs. In Sec. VI, we model traffic locality and time-varying topologies via epidemic spreading models and quantify the benefit of network coding in these cases. Sec. VII presents extensive simulation results under a wide range of settings, including synthetic BitTorrent workloads. Sec. VIII concludes the paper.

II. Related Work

The pioneering work by Ahlswede *et al.* [1] has proved that network coding can achieve multicast capacity in directed networks from a network-flow perspective. Ho *et al.* [2] proposed randomized network coding, which was subsequently applied to BitTorrent-like P2P content distribution by Gkantsidis *et al.* [3], who show network coding can speed up downloads over random block selection by 2-3 times in their simulations. However, Legout *et al.* [7] find in their experiments that the rarest first algorithm of BitTorrent guarantees close-to-ideal diversity of the blocks among peers, and using network coding in such systems cannot be justified.

Such a confusion is largely due to the lack of theoretical understanding of network coding's benefit in P2P networks, which are better modeled by gossip-based overlay broadcast, instead of the widely explored directed network model in information theory. Yeung [4] shows in a time-synchronized model that network coding achieves the optimal delay performance for any transmission schedules in P2P networks. Deb *et al.* [5] shows in a time-synchronized model that network coding can achieve a shorter broadcast delay of k blocks in complete graphs, as compared to a naive sequential dissemination. Mosk-Aoyama *et al.* [13] further analyzes the broadcast delay using network coding in arbitrary graphs and shows its correlation with the spectral properties of the graph. Sanghavi *et al.* [11] considers the problem of broadcasting multiple blocks in P2P networks, and proposes a decentralized block exchange algorithm based on push and pull that has a close-to-optimal performance.

Despite these efforts, there exists a major gap in understanding the benefit of network coding over *state-of-the-art protocols* such as rarest first policies in practical P2P systems, where transmissions are not synchronized and may incur random delays. Motivated by this, we not only prove the optimality of network coding in a continuous-time gossiping model, but more importantly, we focus on the marginal benefits of network coding over reasonably good non-coding block selection policies. While most prior work assumes complete graph as the underlying network, our new finding reveals that topological dynamics serve as a critical factor that impacts the marginal benefits of network coding in P2P networks.

III. Problem Formulation

In this paper, we model the P2P network as a graph $G_t = (V, E_t)$ with $|V| = N$ nodes (peers) and edge set E_t that may change over time. Each node i has an average upload bandwidth μ_i and sufficiently large download bandwidth. To accommodate random transmission delays, we assume the time it takes for node i to transmit a block follows a certain distribution with mean $1/\mu_i$, whereas the size of each block is assumed to be 1.

An edge between two peers represents a data connection between them. A node maintains connections with a subset of all the other peers, which form its *neighborhood*. Inspired by gossip-based overlay broadcast systems, we study the problem of delivering k data blocks $\{b_1, b_2, \ldots, b_k\}$ that are initially possessed by N_0 source nodes to all the other nodes in the network. We are concerned with the *broadcast delay* $T(G_t, k)$, defined as the time needed to disseminate all k blocks to all the nodes in G_t. We are also interested in the ϵ-broadcast delay $T^{(\epsilon)}(G_t, k)$, at which $1 - \epsilon$ of all the peers finish downloading, e.g., $\epsilon = 5\%$ gives a 95th percentile value in the CDF of individual peer download times.

We consider a class of ***Gossip Algorithms*** that conform to the following rules. For each node $i \in V$, at rate μ_i, it

a) randomly chooses one of its neighbors to serve, and
b) transmits one or a linear combination (in Galois field) of blocks it has obtained.

Criterion a) is the random target peer selection originated from the classical gossiping problem [14], [15] and has recently been applied to the analysis of information dissemination [5], [11], [13], [16]. Criterion b) concerns with the block selection. We consider the following block selection/encoding algorithms:

- **Random Useful Block (RUB).** Among the blocks needed by the target peer, the sender transmits a random block.
- **Local Rarest First (LRF).** Among the blocks needed by the target peer, the sender transmits a random block with the smallest number of copies in the neighborhood.
- **Global Rarest First (GRF).** Among the blocks needed by the target peer, the sender transmits a random block with the smallest number of copies in the network.
- **Randomized Network Coding (NC).** The sender linearly encodes all the (coded) blocks it has obtained using random coefficients in Galois field $GF(2^q)$ and uploads the encoded block to the target peer [17], [18]. A peer has finished downloading when it has obtained k linearly independent coded blocks.

Note that RUB, LRF and NC can all be implemented in a decentralized way. RUB requires data reconciliation between the sender and receiver. LRF requires each peer to be aware of the block distribution in its neighborhood. In contrast, NC does not involve any control overhead. Since GRF requires global views at the peers, it is impractical to implement and only serves as a reference algorithm.

IV. On the Optimality of Network Coding

First, it is necessary to point out that applying a random linear code at each transmitting node can achieve the optimal broadcast delay, regardless of the network topology and target peer selection schedule (transmission schedule). Yeung *et al.* [4] have shown its optimality in a discrete-time model where transmissions are synchronized. Here we extend [4] to our continuous-time model mentioned in Sec. III, where random transmission delays are allowed.

We model the block transmission using a continuous-time trellis $G^* = (V^*, E^*)$ constructed in the following way. To model block transmissions, if a block is sent from node u at time t_1 and is received by node v at time t_2, we introduce vertices $u_{t_1} \in V^*$ and $v_{t_2} \in V^*$ to represent node u at time t_1 and node v at time t_2, respectively. A directed edge of capacity 1 from u_{t_1} to v_{t_2} is also introduced. Denote the source node at time 0 by s_0. Note that all the "transmission edges" are determined by transmission schedules.

To model the process of information accumulation at nodes over time, for each node $u \in V$, we then connect the introduced vertices u_{t_i} along the time line with edges of infinite capacity. In other words, for any two consecutive vertices over time u_{t_1} and u_{t_2} ($t_1 < t_2$), there is an edge of infinite capacity from u_{t_1} to u_{t_2}. These "memory edges" model the fact that the blocks, once possessed by a node, are retained in that node indefinitely over time. Without loss of generality, we may assume that all the blocks possessed by nodes u_{t_1} are transmitted uncoded on the edge from u_{t_1} to u_{t_2} ($t_1 < t_2$).

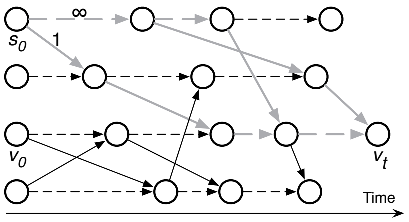

Fig. 1. Continuous-time trellis for a network where transmissions are subject to random delays. $\text{maxflow}(v_t) = 3$ at time t as shown by the grey edges.

Denote the value of a max-flow from node s_0 to a node $v_t \in V^*$ by $\text{maxflow}(v_t)$. Note that G^* is an acyclic graph since each edge in E^* goes from a node at an earlier time to a node at a later time. According to the well-known theorem on multicast in acyclic graphs [1], [19], those nodes v_t with $\text{maxflow}(v_t) \geq k$ can receive all k blocks. Thus, given a transmission schedule (V^*, E^*), the minimum possible time $t^*(v)$ it takes a node $v \in V$ to receive all k blocks is

$$t^*(v) = \inf\{t : \text{maxflow}(v_t) \geq k\}.$$

By [2], when the field size q is large enough, this lower bound is achievable with high probability by applying a random linear code at each node $v \in V$. Therefore, we have obtained the following optimality result of network coding in gossip-based overlay broadcast.

Proposition 1: Randomized Network Coding achieves the minimum possible broadcast delay for any topology and any transmission schedule $G^* = (V^*, E^*)$ with high probability.

V. Complete and Random Graphs

In this section, we first derive a theoretical lower bound on the broadcast delay of any "gossip algorithm" in complete graphs. We then compare different algorithms against the derived lower bound numerically. We find in complete and random graphs, rarest first algorithms are enough to achieve close-to-optimal performance, and the further improvement brought by network coding is trivial.

A. Performance Bounds

Assume that the network is a complete graph of N nodes ($G_t \equiv K_N,\ \forall t \geq 0$). We call a node a type i-node if it holds i blocks. Let $X_i(t) \in \mathbb{Z}^+$ ($i = 0, 1, \ldots, k$) denote the number of type i-nodes. Then the process $\boldsymbol{X}(t) := \{X_0(t), X_1(t), \ldots, X_k(t)\}$ characterizes network states. Define the normalized process of $\boldsymbol{X}(t)$ as $\boldsymbol{x}^{(N)}(t) := \boldsymbol{X}(t)/N$. For tractability, we assume the time for each peer to upload a block follows an exponential distribution with rate $\mu = 1$.

We denote the probability that a type-i node can update a type-j node with useful blocks at time t by $\sigma_{ij}(t)$. With any gossip algorithm defined in Sec. III, $\sigma_{ij}(t) \equiv 1$ if $i > j$. We define the **ideal gossip algorithm (IDEAL)** as the one with $\sigma_{ij}(t) \equiv 1$ for all $i \neq 0, j \neq k$. This means *any non-empty node can update any of its non-full neighbors with useful blocks*. It is clear that the broadcast delay of IDEAL gives a lower bound on the broadcast delay of any gossip algorithm. Based on this, we can obtain the following proposition:

Proposition 2: Assume $G_t \equiv K_N$, $\forall t \geq 0$, and initially $N_0 = N\epsilon$ nodes each hold k blocks, where $\epsilon \in (0, 1)$ is a small constant. Let $\{x_i(t) : t \geq 0\}$ be determined by the ODEs

$$\begin{cases} \dot{x}_0 = -(1 - x_0)x_0, \\ \dot{x}_i = (1 - x_0)(x_{i-1} - x_i), \quad i = 1, 2, \ldots, k-1, \\ \dot{x}_k = (1 - x_0)x_{k-1}, \end{cases} \tag{1}$$

with $\sum_{i=0}^{k} x_i(t) = 1$, $x_0(0) = 1-\epsilon$, $x_k(0) = \epsilon$, and $x_i(0) = 0$ for $i = 1, \ldots, k-1$. As $N \to \infty$, the ϵ-broadcast delay of any gossip algorithm $T^{(\epsilon)}$ has a lower bound:

$$T^{(\epsilon)} \geq \inf\{t : x_k(t) \geq 1 - \epsilon\}. \tag{2}$$

If $k = 2$, such a lower bound is explicitly given by

$$T^{(\epsilon)} \geq 2\ln(\frac{1}{\epsilon} - 1) + \ln\ln(\frac{1}{\epsilon} - 1). \tag{3}$$

Sketch of Proof: Let us first analyze the case of $k = 2$. Consider the delay of IDEAL. When $\sigma_{ij}(t) \equiv 1$ for all $i \neq 0$, $j \neq 2$, $\boldsymbol{X}(t)$ is a Markov process with transitions $l_1 = (-1, 1, 0)$ and $l_2 = (0, -1, 1)$, and their corresponding intensities $q^{(N)}_{X,X+l_i}$ $(X \in {\mathbb{Z}_+}^3)$:

$$\boldsymbol{X} \to \boldsymbol{X} + l_1, \qquad q^{(N)}_{X,X+l_1} = (N - X_0)\mu \cdot \tfrac{X_0}{N}$$
$$\boldsymbol{X} \to \boldsymbol{X} + l_2, \qquad q^{(N)}_{X,X+l_2} = X_2\mu \cdot \tfrac{X_1}{N} + X_1\mu \cdot \tfrac{X_1}{N}.$$

This is because when a non-empty node uploads to an empty node (which happens at rate $(N - X_0)\mu \cdot \frac{X_0}{N}$), X_0 decreases by 1 and X_1 increases by 1, and the second transition occurs when a type 2 or type 1-node updates another type 1-node.

Considering the normalized process $\boldsymbol{x}^{(N)}(t) := \boldsymbol{X}(t)/N$, the above intensities can be rewritten as $q^{(N)}_{X,X+l} = N\beta_l(\frac{X}{N})$ $(X \in {\mathbb{Z}_+}^3)$, where

$$\beta_{l_1}(\frac{X}{N}) = \mu(1 - \frac{X_0}{N}) \cdot \frac{X_0}{N}$$
$$\beta_{l_2}(\frac{X}{N}) = \mu \cdot (\frac{X_2}{N} \cdot \frac{X_1}{N} + (\frac{X_1}{N})^2).$$

Hence, $\boldsymbol{x}^{(N)}(t)$ is a *density dependent jump Markov process* (see [20], pp. 51). We set $F(x) = \sum_l l\beta_l(x)$ and note that $\boldsymbol{x}^{(N)}(0) = \boldsymbol{x}(0) = (1 - \epsilon, 0, \epsilon)$ does not depend on N. By Kurtz Theorem (Theorem 8.1 in [20]), under the conditions easily verified here (boundedness and Lipschitz continuity of $F(x)$), as $N \to \infty$, $\boldsymbol{x}^{(N)}(t)$ converges *almost surely* to the deterministic fluid $\boldsymbol{x}(t) = \{x_0, x_1, x_2\}$:

$$\boldsymbol{x}(t) = \boldsymbol{x}(0) + \int_0^t F(\boldsymbol{x}(u))du, \quad t \geq 0,$$

which can be rewritten as

$$\begin{cases} \dot{x}_0 = -(1 - x_0)x_0, \\ \dot{x}_1 = (1 - x_0)x_0 - x_1x_2 - x_1^2. \end{cases} \tag{4}$$

with $x_0(0) = 1 - \epsilon$, $x_2(0) = \epsilon$. If $k > 2$, a similar argument can be used to derive (1). Now it is clear that $T^{(\epsilon)}$ for IDEAL equals to the smallest t such that $x_k(t) \geq 1 - \epsilon$, proving (2). When $k = 2$, (1) can be solved analytically to give an explicit lower bound (3). □

It turns out in simulation that the above bound also provides a good approximation for random graphs if the average node degree is large enough.

B. Performance of Different Algorithms

We now evaluate NC, RUB, LRF and GRF against the derived delay lower bound in complete graphs, and against IDEAL in random graphs through simulations. We utilize hardware accelerated network coding [21] implemented with SSE2 SIMD vector instructions on x86 processors to scale to a large number of data blocks simulated. Coding operations are performed in $GF(2^8)$. For each set of parameters, 30 independent experiments are conducted to take the average.

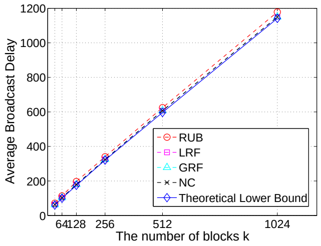

Fig. 2. Average broadcast delays in a complete graph $N = 1000$, $k = 32 \sim 1024$.

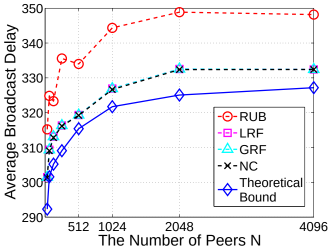

Fig. 3. Average broadcast delays in complete graphs as N varies. $k = 256$, $N = 32 \sim 4096$.

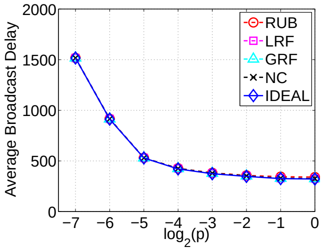

Fig. 4. Average broadcast delays as random graph parameter p varies. $N = 1000$, $k = 256$.

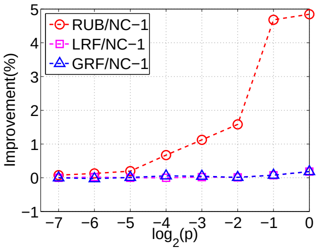

Fig. 5. The delay improvements of NC over other algorithms as p varies. $N = 1000$, $k = 256$.

Fig. 2 and Fig. 3 show the average broadcast delays in complete graphs as k and N vary. The theoretical lower bound is obtained from Proposition 2 by letting $\epsilon = 1/N$. We can see that in complete graphs, GRF and LRF can result in almost *exactly* the same delay as NC can. All these algorithms achieve performance very close to the lower bound, with RUB being slightly inferior (by less than 5% though). We also see the expected broadcast delay grows linearly in trend as k increases, and nearly logarithmically as N grows (although not exactly logarithmically by (3)).

We have also compared the algorithm performance in Erdos-Renyi random graphs [22] with parameter p, where each pair of peers are connected with probability p. Fig. 4 shows that as p decreases from 1 to 2^{-7}, all the algorithms are always close-to-optimal in that their performance approaches the ideal gossip algorithm. To take a closer view, we plot the delay improvements of NC over RUB, LRF and GRF in Fig. 5. What's interesting is that NC's marginal benefits become even more trivial as the graph becomes sparser (with a smaller p). Even for complete graphs, NC can improve at most 5% over RUB, and at most 0.18% over GRF and LRF.

From our analysis and simulation in this section, we find that network coding is not necessarily needed to achieve close-to-optimal broadcast delay in complete or random graphs. Local rarest first (LRF) as a decentralized algorithm can achieve almost exactly the same performance as its global counterpart GRF and network coding can. When the graph is sparse ($p < 2^{-4}$), even RUB can be close to optimal.

VI. Clustered and Time-Varying Topologies

To study the impact of clustering (traffic-locality) and time-varying topologies on gossip algorithms, we develop an epidemic model [12] in this section to quantify the performance gap between NC and LRF, LRF being the decentralized algorithm that achieves very close-to-optimal performance in random graphs. GRF will be used only as a reference algorithm in simulations as it assumes centralized knowledge and is impractical to be implemented in reality.

A. Network Model

To model ISP-aware traffic locality, we consider a network composed of clusters with random global links across clusters. Such a topology is a natural abstraction of those networks where peers prefer connections within the same ISP.

Define $G_t(m, n)$ as a graph of size $N = mn$ that consists of m clusters of peers: $K_n^1, K_n^2, \ldots, K_n^m$, each of which is a clique of size n, as shown in Fig. 6. We could view each cluster as a model of an ISP or a geographically clustered community. Each peer p in K_n^i also maintains global links with d_G ($d_G \ll n$) other nodes chosen u.a.r. from $\bigcup_{j=1,\ldots,m,j\neq i} K_n^j$.

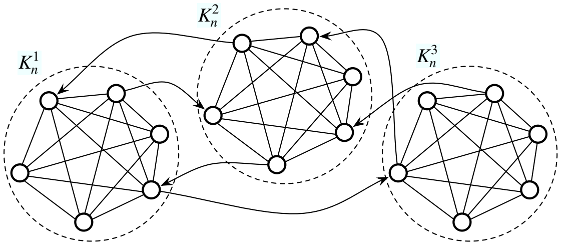

Fig. 6. A clustered topology composed of $m = 3$ clusters, each being a complete graph of n nodes.

The $n - 1 + d_G$ links from peer p are changing periodically with cycle δ, i.e., they are reselected by the above rules every time p has uploaded a multiple of δ blocks. In our analysis, we consider the extreme case of $\delta = 1$. This leads to the d_G global links being reselected every time before peer p uploads a block. By random target peer selection from the neighborhood, this essentially means that at the time of an upload, peer p will choose a random peer in its own cluster (a local neighbor) with probability $(n-1)/(n-1+d_G)$, or a random peer in other clusters (a global neighbor) with probability $d_G/(n-1+d_G)$. Since the upload process is Poisson with rate $\mu = 1$, each peer uploads to a random global neighbor at the points of a Poisson process with rate

$$\lambda_o = 1 \cdot d_G/(n - 1 + d_G) \ll 1. \qquad (5)$$

B. Intuitions on the Benefit of Network Coding

Intuitively, careful choices of blocks should be made when transmitting across clusters to optimally utilize the precious bandwidth between them. Unlike in complete or random graphs, *a peer's choice based on a rarest first policy heavily depends on whether it can obtain an unbiased global view*.

As a starting point, we analyze the case $m = 2$, $k = 2$ and bound from below the gap between the expected broadcast delays of NC and LRF. We will extend the results to more general cases using simulations.

Let $T_{K_n^i}$ $(i = 1, 2)$ denote the time at which all peers in K_n^i finish downloading. Assume the source peer is in K_n^1. It is easy to see $T(G_t(2, n), 2) = T_{K_n^2}$. Let Z_1 denote the first time that a block (say b_1) gets into K_n^2, and Z_2 the first time that the other block (say b_2) gets into K_n^2. We have

$$T(G_t(2, n), 2) = T_{K_n^2} = Z_1 + (Z_2 - Z_1) + (T_{K_n^2} - Z_2).$$

We can derive $T(G_t(2, n), 2)$ by deriving Z_1, $Z_2 - Z_1$ and $T_{K_n^2} - Z_2$.

First, denote by $S_1(t)$ the number of non-empty nodes in K_n^1. Z_1 is the first time that any of such nodes uploads to a node in K_n^2. The evolution of $S_1(t)$ in K_n^1 is approximately the same as if there were a single complete graph K_n, as $\lambda_o \ll 1$ and the uploads from K_n^2 can hardly affect data propagation in K_n^1. Since $S_1(t)$ evolves in roughly the same way for NC and LRF according to Sec. V-B, NC and LRF will have roughly the same Z_1. Once b_1 gets into K_n^2, any further injections of b_1 from K_n^1 will trivially affect its dissemination in K_n^2 as $\lambda_o \ll 1$.

However, NC and LRF have different values of $Z_2 - Z_1$. Consider the phase $Z_1 \le t < Z_2$. Denote by p_{ij} $(i = 0, 1, 2, j = 0, 1)$ the probability that a type i-node in K_n^1 can inject a useful block into K_n^2 when it's updating a type j-node in K_n^2. For NC, if the field size q is big enough and there is no linear dependency (like the ideal algorithm), we have in the second phase (as shown in Fig. 7):

$$p_{ij}^{(NC)} = 1, \quad \forall i \in \{0, 1, 2\},\ \forall j \in \{0, 1\}.$$

For LRF, consider the best scenario (to give a lower bound on LRF-NC gap) that a uniform distribution of different blocks is achieved in K_n^1. We have

$$p_{ij}^{(LRF)} = \begin{cases} 1, & \text{if } i = 2 \text{ and } j = 1, \\ \frac{1}{2}, & \text{otherwise.} \end{cases}$$

We have $p_{21} = 1$, because any node with 2 blocks can update a node with one block. However, $p_{20} = 1/2$, because when a type 2-node in K_n^1 updates an empty node in K_n^2, the view of the sender is dominated by the block distribution in K_n^1 as $d_G \ll n-1$, and thus b_1 and b_2 will be chosen equally likely. Moreover, $p_{11} = 1/2$, $p_{10} = 1/2$, since when a type 1-node in K_n^1 is uploading to any node in K_n^2, a new block, say b_2, can be injected into K_n^2 only if the sender holds b_2. Because NC has greater p_{ij}, it has a smaller $Z_2 - Z_1$.

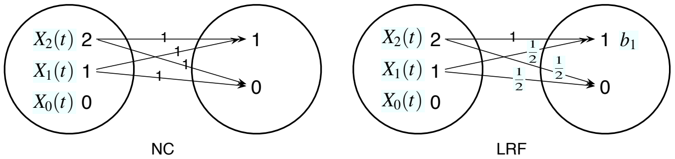

Fig. 7. The probability that a type i-node in K_n^1 can inject a new block into K_n^2 when updating a type j-node in K_n^2 during the phase $Z_1 \leq t < Z_2$.

The time needed for the third phase $T_{K_n^2} - Z_2$ is also the same for NC and LRF. For LRF, once b_2 gets into K_n^2, further injections of b_2 from K_n^1 will have a trivial impact as $\lambda_o \ll 1$. As b_1 gets into K_n^2 first, we can assume the number of b_2 is always less than that of b_1 in K_n^2. Thus, whenever a K_n^2-node that holds a b_2 is ready to upload, b_2 will be chosen by local rarest first. Thus, b_2 propagates in K_n^2 like there were no b_1 in K_n^2, which takes time $2\ln(n-1) + O(2)$ on expectation (easily following from an argument using the linearity of expectations). Thus, we have $\mathbf{E}[T_{K_n^2} - Z_2] \approx 2\ln(n-1) + O(2)$. It is not hard to verify when network coding is applied, $\mathbf{E}[T_{K_n^2} - Z_2]$ cannot be further reduced and is still $2\ln(n-1) + O(2)$.

Therefore, network coding is beneficial because it induces a smaller $Z_2 - Z_1$ due to better utilization of the bottleneck across clusters. In other words, the expected broadcast delay gap between NC and LRF satisfies:

$$\mathbf{E}[T_{LRF} - T_{NC}] \geq \mathbf{E}[Z_2^{LRF}] - \mathbf{E}[Z_2^{NC}].$$

C. Quantifying the Gap between NC and LRF

We now give a lower bound on $\mathbf{E}[Z_2^{LRF}] - \mathbf{E}[Z_2^{NC}]$. We need to derive $\mathbf{E}[Z_2]$ for both NC and LRF. Let $D_1 = Z_1$, and $D_2 = Z_2 - Z_1$. Then $\mathbf{E}[Z_2] = \mathbf{E}[D_1] + \mathbf{E}[D_2]$. Note that for large n, the evolution of the number of non-empty nodes in K_n^1 is

$$S_1(t) = n - X_0(t) = \frac{n}{(n-1)e^{-t} + 1} \approx e^t, \tag{6}$$

by the solution to (1) for $k = 2$. We first derive $\mathbf{E}[D_1]$ and then $\mathbf{E}[D_2]$ by conditioning on D_1, arriving at the following proposition:

Proposition 3: For large n, $\lambda_o \ll 1$, $m = 2$ and $k = 2$, the gap between the expected broadcast delays of NC and LRF has the lower bound:

$$\mathbf{E}[T_{LRF} - T_{NC}] \geq \mathbf{E}[Z_2^{LRF}] - \mathbf{E}[Z_2^{NC}] > \frac{1}{3}. \tag{7}$$

Proof: By the discussions above, $D_1 = Z_1$ is the same for NC and LRF. Consider a *non-homogeneous Poisson process* $\{N(t), t \geq 0\}$ [23] with intensity function $\lambda_o S_1(t) \approx \lambda_o e^t$. Then $D_1 \equiv \inf\{t : N(t) \geq 1\}$. Since the mean value function of $N(t)$ is

$$m_1(t) = \int_0^t \lambda_o e^s ds = \lambda_o e^t - \lambda_o,$$

we have

$$F_{D_1}(t) := \Pr(D_1 \leq t) = \Pr(N(t) \geq 1) = 1 - e^{-m_1(t)},$$

and thus

$$\mathbf{E}[D_1] = \int_0^\infty t d(1 - e^{-m_1(t)}) = e^{\lambda_o}\mathrm{E}_1(\lambda_o), \tag{8}$$

where $\mathrm{E}_1(\lambda_o) = \int_{\lambda_o}^\infty (e^{-y}/y)dy$ is the exponential integral (see [24], pp. 228), regardless of whether NC or LRF is applied.

The values of D_2 are different for NC and LRF. To derive $\mathbf{E}[D_2]$, we consider the process $\{N'(t) := N(t + D_1) - N(D_1), t \geq 0\}$ conditioning on D_1. Note that $N'(t)$ is another *non-homogeneous Poisson process* with different intensities for NC and LRF.

We first consider NC. Since $p_{ij}^{(NC)} = 1$ for all $i = 0, 1, 2, j = 0, 1$ in the second phase, which remain the same as in the first phase, for large n, the intensity function is

$$\lambda_1(t) = \lambda_o S_1(t + D_1) = \lambda_o e^{t + D_1}. \tag{9}$$

The mean value function of $N'(t)$ for NC is $m_2^{NC}(t) = \int_0^t \lambda_1(s)ds$. Given D_1, we have $D_2 \equiv \inf\{t : N'(t) \geq 1\}$. Hence, for NC, we have

$$\begin{aligned}\Pr(D_2 \leq t) &= \mathbf{E}_{D_1}[\Pr(D_2 \leq t | D_1)] = \mathbf{E}_{D_1}[\Pr(N'(t) \geq 1 | D_1)] \\ &= \int_0^\infty (1 - e^{-m_2^{NC}(t)})dF_{D_1}(s) = 1 - e^{\lambda_o - t - \lambda_o e^t},\end{aligned}$$

and thus

$$\begin{aligned}\mathbf{E}[D_2^{NC}] &= \int_0^\infty t d(1 - e^{\lambda_o - t - \lambda_o e^t}) = 1 - \lambda_o e^{\lambda_o}\mathrm{E}_1(\lambda_o) \\ &< 1 - \frac{1}{2}\lambda_o \ln(1 + \frac{2}{\lambda_o}) \to 1, \quad \text{as } \lambda_o \to 0, \end{aligned} \tag{10}$$

where the inequality holds because $e^z\mathrm{E}_1(z) > \frac{1}{2}\ln(1 + \frac{2}{z})$, for $z > 0$ (see [24], pp. 229, Eq. 5.1.20).

For LRF, since $p_{21} = 1$ and $p_{ij} = 1/2$ for all other i, j, the intensity function of $N'(t)$ is

$$\lambda_2(t) = \frac{1}{2}\lambda_o S_1(t + D_1) + \frac{1}{2}X_2(t + D_1)\lambda_o \cdot \frac{e^t}{n} \tag{11}$$

By the solution to (1) for $k = 2$ in Sec. V, we have

$$\begin{aligned}X_2(t) &= \frac{n(n-1) + ne^t}{(n^2 - 1) + (n-1)t + e^t} \leq \frac{n^2 - n + ne^t}{n^2 - 1 + nt - t + t + 1} \\ &= \frac{n + (e^t - 1)}{n + t} \leq \frac{e^t - 1}{t}.\end{aligned} \tag{12}$$

where the first inequality is due to $e^t \geq t + 1$, $\forall t$. Thus,

$$\lambda_2(t) \leq \frac{1}{2}\lambda_o S_1(t + D_1) + \frac{1}{2}\frac{e^{t+D_1} - 1}{t + D_1}\lambda_o \cdot \frac{e^t}{n} \to \frac{1}{2}\lambda_o e^{t + D_1},$$

as $n \to \infty$. Since $\lambda_2(t) \geq \frac{1}{2}\lambda_o e^{t+D_1}$, we have for large n, $\lambda_2(t) = \frac{1}{2}\lambda_o e^{t+D_1}$. The mean value function of $N'(t)$ for LRF is $m_2^{LRF}(t) = \int_0^t \lambda_2(s)ds$. Similarly, we have for LRF

$$\Pr(D_2 \le t) = \int_0^\infty (1 - e^{-m_2^{LRF}(t)})dF_{D_1}(s)$$

$$= - \int_{s=0}^\infty (1 - e^{-\frac{1}{2}\lambda_o(e^{t+s} - e^s)})de^{-\lambda_o e^s + \lambda_o} = 1 - \frac{2e^{-\frac{1}{2}\lambda_o(e^t - 1)}}{e^t + 1}$$

Hence, by letting $y = \frac{1}{2}\lambda_o e^t$, $u = \frac{4}{3\lambda_o}y + \frac{1}{3}$, we have

$$\begin{aligned}
&\mathbf{E}[D_2^{LRF}] \\
&= \int_{t=0}^\infty td\frac{-2e^{-\frac{1}{2}\lambda_o(e^t-1)}}{e^t+1} = \lambda_o e^{\frac{1}{2}\lambda_o} \int_{\frac{1}{2}\lambda_o}^\infty \frac{e^{-y}}{y^2 + \frac{1}{2}\lambda_o y}dy \\
&\ge \lambda_o e^{\frac{1}{2}\lambda_o} \int_{\frac{1}{2}\lambda_o}^\infty \frac{e^{-y}}{(y + \frac{\lambda_o}{4})^2}dy = \frac{4}{3}e^{\frac{3}{4}\lambda_o} \int_1^\infty \frac{e^{-\frac{3\lambda_o}{4}u}}{u^2}du \\
&= \frac{4}{3}e^{\frac{3}{4}\lambda_o}\mathrm{E}_2(\frac{3\lambda_o}{4}) \quad (\mathrm{E}_2(z) := \int_1^\infty \frac{e^{-zt}}{t^2}dt) \\
&= \frac{4}{3} - \lambda_o e^{\frac{3}{4}\lambda_o}\mathrm{E}_1(\frac{3\lambda_o}{4}) \quad (\mathrm{E}_2(z) = e^{-z} - z\mathrm{E}_1(z)) \\
&> \frac{4}{3} - \frac{4}{3}\ln(1 + \frac{4}{3\lambda_o})^{\frac{3\lambda_o}{4}} \to \frac{4}{3}, \quad \text{as } \lambda_o \to 0,
\end{aligned} \tag{13}$$

where $e^z\mathrm{E}_1(z) < \ln(1 + \frac{1}{z})$ [24]. From (8), (10) and (13), we have for large n and $\lambda_o \ll 1$, $\mathbf{E}[Z_2^{LRF}] - \mathbf{E}[Z_2^{NC}] = \mathbf{E}[D_2^{LRF}] - \mathbf{E}[D_2^{NC}] > \frac{1}{3}$. □

Implications. The above analysis implies that NC automatically makes better choices of blocks when transmitting across clusters. Such an implication can be generalized to the case of $k > 2$ blocks and $m > 2$ clusters. When a type i-node in one cluster transmits to a type j-node in another cluster, NC proves its benefits over LRF in two aspects.

First, if $i > j$, with the local rarest first, since the sender's view is dominated by its own cluster, it may not choose the rarest block that is most urgently needed by the receiver's cluster, while NC does not have this issue. Second, if $i \le j$, any non-coding protocol suffers from the curse of the coupon collector problem, as the sender can hold a subset of the blocks in the receiver or in the receiver's cluster. In contrast, NC alleviates redundancy by issuing more diverse coded blocks as long as the field size is large enough. Due to these reasons, we conjecture that as m and k increase, the marginal benefit of network coding will become more salient. While deriving analytical bounds for $k > 2$ and $m > 2$ is an open question, we resort to extensive simulations to study the general case.

VII. Experimental Studies

We now study the impact of topological dynamics on algorithm performance through simulation. Coding operations are done in $GF(2^8)$ and again implemented with SSE2 SIMD instructions [21] to achieve simulation scalability. For each parameter setting, 30 independent experiments are conducted to calculate the average.

A. Time-Varying and Clustered Topologies

First, we discuss the impact of time-varying property in non-clustered topologies. Fig. 8 shows when each peer reforms its neighborhood with cycle $\delta = 1$, the performance of NC, GRF, RUB are not affected by the sparsity (p) of the graph. Comparing with Fig. 4 for static topologies, we find even if the graph is sparse, the performance of NC, GRF and RUB can be maintained at the same level as that in dense graphs ($p = 1$) by letting peers reselect their neighbors once in a while. However, the performance of LRF suffers in time-varying sparse graphs due to a lack of accurate global knowledge.

Fig. 9 shows varying δ can hardly affect the performance of RUB, NC and GRF, with RUB being constantly inferior. Similarly, LRF suffers when the graph is changing at a higher frequency due to its inability to keep track of the information in a node's new neighborhood. Fig. 10 shows that the relation between broadcast delay and k is nearly linear even if the topology is changing. From Fig. 11, we see that as k increases, the benefit of NC over RUB becomes less significant, as RUB enjoys more diversity in choosing blocks in a time-varying topology. In contrast, NC's benefit over LRF becomes more salient, as the inaccuracy in predicting the locally rarest block in a time-varying graph is amplified when there are more blocks to be distributed.

In a nutshell, periodically changing the topology can prevent the performance of NC, RUB and GRF from degrading as the graph becomes sparser. NC can best utilize the block diversity offered by a time-varying topology, whereas LRF — the close-to-optimal algorithm in static random graphs — suffers from it dramatically.

Second, we consider the effect of clustering alone in static topologies ($\delta = \infty$). Fig. 12 and Fig. 13 show the broadcast delays and NC's improvements as the number of global neighbors of each peer d_G varies. As d_G decreases, the network has a higher degree of clustering. Clearly, the benefits of NC over all other algorithms exhibit a threshold behavior; they only increase dramatically when $d_G \le 1$. It's worth noting that NC's benefit over GRF also exceeds 20% when $d_G = 1$ even if GRF has the global knowledge. The benefit of NC becomes to drop again if d_G is too small.

We now consider the combined effect of clustering and time-varying topologies. When the topology is changing, the performance of RUB, NC and GRF is increased as diversity is introduced, shown in Fig. 16 as compared to Fig. 12, while LRF always performs poorly in clustered graphs due to a lack of global knowledge. This explains why we see a drastic increase in NC's benefit over LRF (its benefit over RUB and GRF also increases, but less dramatically) when $d_G \le 1$ (Fig. 14 and 15). Interestingly, there is a small bump around $d_G = 2^7 \sim 2^8$ (128-256 random global neighbors) in Fig. 14 and 15 regarding the benefit of network coding. This is because in this range, the graph exhibits behavior similar to that of time-varying random graphs (Fig. 8 and 11) instead of clustered graphs. Furthermore, NC's benefit becomes more salient for all d_G as δ decreases from ∞ (static) to 1, shown in Fig. 13, 14, and 15. Finally, as m increases, while N remains unchanged, the benefit of NC first increases as more cross-cluster bottlenecks appear, shown in Fig. 18, but then decreases as m further increases, since the network demonstrates more behavior of a random graph.

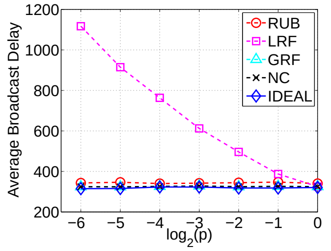

Fig. 8. Average broadcast delays in random graphs with time-varying topologies. $k = 256$, $N = 1000$, $\delta = 1$.

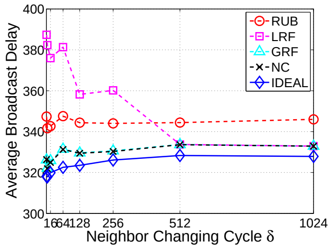

Fig. 9. Average broadcast delays in random graphs as the topology changes at different frequencies. $k = 256$, $N = 1000$, $p = 0.5$. $\delta \in \{1, 4, 16, 64, 128, 256, 512, 1024\}$.

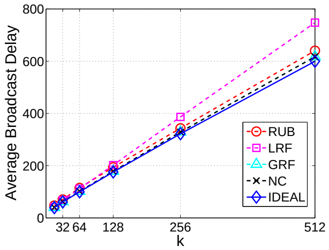

Fig. 10. Average broadcast delays in random graphs with time-varying topologies. $N = 1000$, $p = 0.5$, $\delta = 1$.

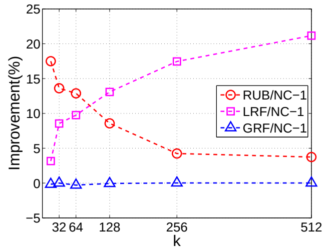

Fig. 11. Average improvements of NC in random graphs with time-varying topologies. $N = 1000$, $p = 0.5$, $\delta = 1$.

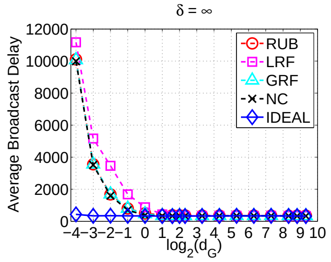

Fig. 12. Average broadcast delays in clustered topologies. $N = 1000$, $m = 10$, $k = 256$, $\delta = \infty$ (static topology).

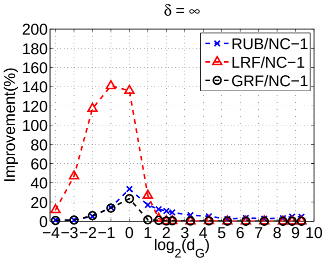

Fig. 13. The average improvements of NC over other algorithms in clustered topologies. $N = 1000$, $m = 10$, $k = 256$, $\delta = \infty$ (static topology).

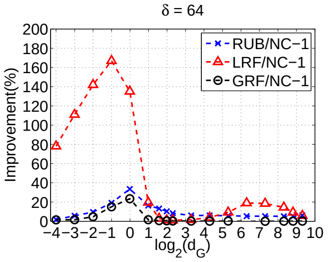

Fig. 14. The average improvements of NC in clustered graphs with time-varying global links. $k = 256$, $N = 1000$, $m = 10$, $\delta = 64$.

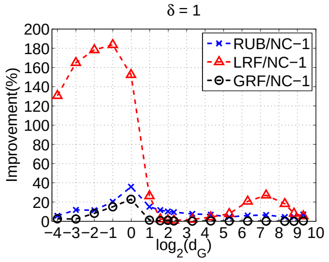

Fig. 15. The average improvements of NC in clustered graphs with time-varying global links. $k = 256$, $N = 1000$, $m = 10$, $\delta = 1$.

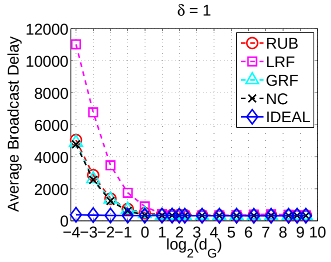

Fig. 16. Average broadcast delays in clustered graphs with time-varying global links. $k = 256$, $N = 1000$, $m = 10$, $\delta = 1$.

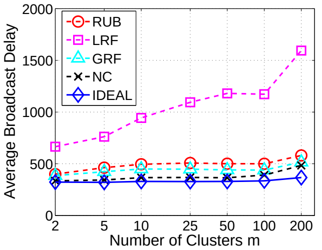

Fig. 17. Average broadcast delays in clustered graphs with time-varying topologies. $k = 256$, $N = 1000$, $d_G = 1$, $\delta = 1$.

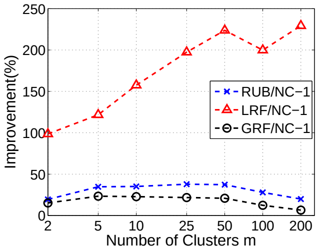

Fig. 18. Average improvements of NC in clustered graphs with time-varying topologies. $k = 256$, $N = 1000$, $d_G = 1$, $\delta = 1$.

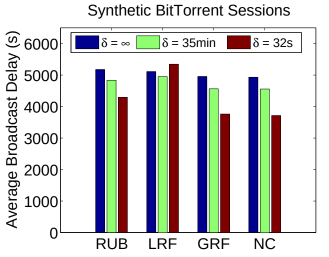

Fig. 19. Average broadcast delays in synthetic BitTorrent sessions. 996 Peers spread among 354 ASes. File size = 100MB, block size = 256KB.

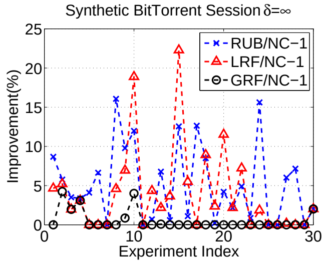

Fig. 20. NC's benefit in each experiment. 996 Peers spread among 354 ASes. File size = 100MB, block size = 256KB. $\delta = \infty$.

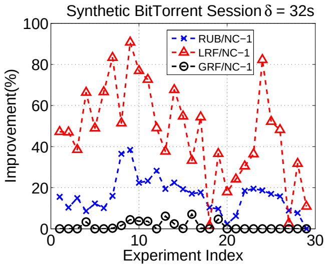

Fig. 21. NC's benefit in each experiment. 996 Peers spread among 354 ASes. File size = 100MB, block size = 256KB. $\delta = 32$s.

B. *Heterogeneous and ISP-aware BitTorrent Sessions*

To verify our findings under more realistic workloads, we synthesize a BitTorrent session with biased neighbor selection that limits cross-ISP traffic. ISP-awareness has been shown to bring savings to ISPs by ensuring traffic locality in content distribution sessions (e.g. [8], [9]). However, our focus here is to study the impact of such a mechanism on the performance of different gossip algorithms.

In the synthesized session, the ISP statistics are based on the torrent trace from a game reported in [9]. The network consists of 996 peers spread among 354 ASes, the largest AS (ISP 1) consisting of 31 peers. Since we lack the detailed information about ISP sizes, we let the size of ISP-i $(i > 1)$ be a random number between 1 and $31/\log(i + 1)$. Peer upload bandwidth values (kbps) are drawn from the distribution: 64 (2.8%), 256 (4.3%), 128 (14.3%), 384 (32.3%), 768 (46.3%), representing modem, ISDN, DSL, Cable and Ethernet connections, respectively [25]. A file of size 100MB broken into 400 blocks (each of size 256KB) is to be broadcast to all the peers. The seed (source) is a random peer in ISP 1. We consider ISP-aware neighbor selection [26]: each peer first selects as many neighbors as possible from its own ISP and then randomly chooses peers in other ISPs to maintain a total of 35 neighbors. Each peer reselects its neighbors every δ seconds to simulate time-varying topologies.

Fig. 19 plots the mean broadcast delays under different δ. Apparently, a time-varying topology ($\delta < \infty$) helps to reduce broadcast delay (except for LRF) by introducing topological diversity, with NC's delay decreasing the fastest as δ drops. In the static topology ($\delta = \infty$), NC's average benefits (5% against RUB, 3.66% against LRF, 0.5% against GRF) are less salient, although it is still superior (up to 22.4% against LRF, 16.1% against RUB) in many individual experiments due to its efficiency in clustered topologies, shown in Fig. 20. However, such benefits critically depend on the randomly formed topology in each individual experiment, making the mean benefits across all experiments low. As the topology becomes more dynamic, NC's benefit becomes much more significant (15.55% against RUB, 43.8% against LRF, 1.22% against GRF on average when $\delta = 32s$).

One interesting observation is that the performance of LRF actually degrades as the topology is changing more frequently, and becomes even worse than that of RUB shown in Fig. 19 and 21. This means LRF, while being close-to-optimal in static random graphs as shown in Sec. V-B, does not take advantage of the diversity offered by a time-varying topology and is not suitable for dynamic environments with ISP-aware neighbor selections, which are quite likely to happen in real content distribution sessions. Finally, although GRF is always close to NC in the synthetic session, it assumes the availability of global block statistics and is thus never feasible for real implementation.

VIII. Concluding Remarks

In this paper, we study the problem of broadcasting multiple data blocks in networks of certain topologies using gossip-like algorithms, focusing on analyzing the benefit of randomized network coding. Although network coding achieves the optimal delay in any topologies, non-coding protocols such as the local rarest first policy can achieve performance very close to the theoretical limits in complete and random graphs. As a result, the application of network coding in these graphs cannot be sufficiently justified.

We further demonstrate that clustering and time-varying topologies are two key factors that boost the benefit of network coding. In clustered graphs, randomized network coding behaves as if it has the global knowledge to make optimal decisions, while other decentralized block selection algorithms fail to do so. Time-varying topologies can reduce broadcast delays only for topology-oblivious protocols and will degrade the performance of topology-dependent protocols such as the local rarest first. Considering all these topological dynamics, network coding is resilient to traffic locality mechanisms that are common in ISP-aware P2P applications, and can take the best advantage of the path diversity, introduced by either passive or proactive topological changes, whereas other decentralized block selection schemes suffer from different degrees of insufficiency in these cases. One interesting direction for further investigation is to theoretically understand the complex behavior of network coding as compared to other gossiping algorithms in different kinds of random graphs.

References

[1] R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung, "Network Information Flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[2] T. Ho, R. Koetter, M. Medard, D. R. Karger, and M. Effros, "The Benefits of Coding over Routing in a Randomized Setting," in *Proc. of IEEE International Symposium on Information Theory*, 2003.

[3] C. Gkantsidis and P. Rodriguez, "Network Coding for Large Scale Content Distribution," in *Proc. of IEEE INFOCOM '05*, March 2005.

[4] R. W. Yeung, "Avalanche: A Network Coding Analysis," *Communications in Information and Systems*, vol. 7, no. 4, pp. 353–358, 2007.

[5] S. Deb, M. Médard, and C. Choute, "Algebraic Gossip: A Network Coding Approach to Optimal Multiple Rumor Mongering," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2486–2507, June 2006.

[6] D. M. Chiu, R. W. Yeung, J. Huang, and B. Fan, "Can Network Coding Help in P2P Networks?" in *Proc. of Netcod '06*, Boston, April 2006.

[7] A. Legout, G. Urvoy-Keller, and P. Michiardi, "Rarest First and Choke Algorithms Are Enough," in *Proc. of Internet Measurement Conference (IMC) 2006*, Rio de Janeiro, Brazil, October 2006.

[8] H. Xie, Y. R. Yang, A. Krishnamurthy, Y. Liu, and A. Silberschatz, "P4P: Provider Portal for Applications," in *Proc. of SIGCOMM'08*. Seattle, Washington, USA: ACM, August 17-22 2008.

[9] S. L. Blond, A. Legout, and W. Dabbous, "Pushing BitTorrent Locality to the Limit," INRIA, Tech. Rep., 2008.

[10] L. Massoulie, A. Twigg, C. Gkantsidis, and P. Rodriguez, "Randomized Decentralized Broadcasting Algorithms," in *Proc. of IEEE INFOCOM '07*, Anchorage, Alaska, USA, May 2007.

[11] S. Sanghavi, B. Hajek, and L. Massoulie, "Gossiping with Multiple Messages," in *Proc. of IEEE INFOCOM '07*, Anchorage, Alaska, 2007.

[12] F. Ball and P. Neal, "Network Epidemic Models with Two Levels of Mixing," *Mathematical Biosciences*, vol. 212, no. 1, March 2008.

[13] D.Mosk-Aoyama and D.Shah, "Information Dissemination via Network Coding," in *Proc. of IEEE International Symposium on Information Theory (ISIT'06)*, Seattle, WA, October 2006.

[14] B. Pittel, "On Spreading a Rumor," *SIAM Journal of Applied Mathematics*, vol. 47, no. 1, p. 213, 1987.

[15] R. Karp, C. Schindelhauer, S. Shenker, and B. Vocking, "Randomized Rumor Spreading," in *Proc. of the 41st Annual Symposium on Foundations of Computer Science (FOCS '00)*, Washington, DC, 2000.

[16] L. Massoulie and M. Vojnovic, "Coupon Replication Systems," in *Proc. of ACM SIGMETRICS '05*, Banff, Alberta, Canada, 2005.

[17] T. Ho, M. Médard, J. Shi, M. Effros, and D. Karger, "On Randomized Network Coding," in *Proc. of the 41st Annual Allerton Conference on Communication, Control, and Computing*, October 2003.

[18] P. A. Chou, Y. Wu, and K. Jain, "Practical Network Coding," in *Proc. of the 41st Annual Allerton Conference on Communication, Control and Computing*, October 2003.

[19] R. Koetter and M. Medard, "An Algebraic Approach to Network Coding," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 782–795, October 2003.

[20] T. G. Kurtz, "Approximation of Population Processes," *CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM*, 1981.

[21] H. Shojania and B. Li, "Parallelized Progressive Network Coding with Hardware Acceleration," in *Proc. of IWQoS'07*, Chicago, Illinois, 2007.

[22] B. Bollobas, *Random Graphs*. London: Academic Press, Inc., 1985.

[23] S. M. Ross, *Stochastic Processes*. Wiley, 1996.

[24] M. Abramowitz and I. A. Stegun, Eds., *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. U.S. Department of Commerce, National Bureau of Standards, Applied Mathematics Series 55, 1964.

[25] C. Huang, J. Li, and K. W. Ross, "Can Internet Video-on-Demand be Profitable?" in *Proc. of SIGCOMM'07*, Kyoto, Japan, 2007.

[26] R. Bindal, P. Cao, W. Chan, J. Medved, G. Suwala, T. Bates, and A. Zhang, "Improving Traffic Locality in BitTorrent via Biased Neighbor Selection," in *Proc. of ICDCS'06*, 2006.