Stochastic Optimal Multirate Multicast in Socially Selfish Wireless Networks

Hongxing Li*, Chuan Wu*, Zongpeng Li†, Wei Huang*, Francis C.M. Lau*
*Department of Computer Science, The University of Hong Kong, {hxli,cwu,whuang,fcmlau}@cs.hku.hk
†Department of Computer Science, University of Calgary, zongpeng@ucalgary.ca

***Abstract*—Multicast supporting non-uniform receiving rates is an effective means of data dissemination to receivers with diversified bandwidth availability. Designing efficient rate control, routing and capacity allocation to achieve optimal multirate multicast has been a difficult problem in fixed wireline networks, let alone wireless networks with random channel fading and volatile node mobility. The challenge escalates if we consider also the selfishness of users who prefer to relay data for others with strong social ties. Such *social selfishness* of users is a new constraint in network protocol design. Its impact on efficient multicast in wireless networks has yet to be explored especially when multiple receiving rates are allowed. In this paper, we design an efficient, social-aware multirate multicast scheme that can maximize the overall utility of socially selfish users in a wireless network, and its distributed implementation. We model social preferences of users as differentiated costs for packet relay, which are weighted by the strength of social tie between the relay and the destination. Stochastic Lyapunov optimization techniques are utilized to design optimal scheduling of multicast transmissions, which are combined with multi-resolution coding and random linear network coding. With rigorous theoretical analysis, we study the optimality, stability, and complexity of our algorithm, as well as the impact of social preferences. Empirical studies further confirm the superiority of our algorithm under different social selfishness patterns.**

I. Introduction

Multicasting with non-uniform receiving rates is highly desirable for data streaming in heterogeneous networks where the destinations in a multicast session may have diversified bandwidth availability. As compared to the single-rate version, *multirate multicast* allows full utilization of the network capacity. High-bandwidth receivers can enjoy higher quality streaming, while low-bandwidth receivers streaming at lower quality would not be excluded from the multicast service.

A typical method to implement multirate multicast is to divide the data flow into layers based on multi-resolution coding (MRC), *e.g.*, H.264/SVC [16] and MPEG-4 [12], and to allocate different numbers of layers to different receivers.[1] In multi-resolution coding, a base layer consists of the most important and basic information of the flow, *e.g.*, video/audio tracks decodable with a basic quality, and is intended to be received by every destination. Several enhancement layers progressively provide incremental details of the flow, and can be optionally obtained to augment the receiver's utility, *e.g.*, for better video/audio playback quality. An enhancement layer is useful only if all lower layers are also correctly received.

This project is supported by Hong Kong RGC GRF grants No. 714009 and No. 714311.

[1]An alternative is multiple description coding (MDC), which is however not widely adopted due to its high coding overhead [9].

To achieve optimal multirate multicast in a network, *i.e.*, to maximize the aggregated throughput utility of all users, a cross-layer algorithm that handles end-to-end rate control, routing, and capacity allocation is typically needed. End-to-end rate control decides the number of layers to send to each destination, and the data rate of each layer; the routing scheme finds the multicast paths for each layer from the source to the destinations; the capacity allocation module schedules packet transmissions along each link. Even in fixed wireline networks, such an optimal multicast solution is non-trivial. Existing literature mostly assumes either known receiving rates at the destinations or that the routing topologies of the layers are given, while addressing other complimentary parts of the problem [2], [6], [7], [17], [19].

The challenge escalates in wireless networks with random channel fading and volatile node mobility, and even more so when a distributed solution is demanded. In a mobile ad-hoc network consisting of users with mobile wireless devices, the available capacity between each pair of adjacent nodes is time varying due to the nature of wireless communication. Multirate multicast can potentially maximize the receiver utilities in such wireless networks. The essential questions are: how does the source calibrate the number of layers each receiver should take, and how does each relay make its packet forwarding decisions such that the aggregated receiver utility is maximized over time? Little existing results exist on this problem. Seemingly a stochastic optimal solution is needed, which can be quite complex.

We bring in another dimension to the problem: social relationships. Network users in the real world are often *socially selfish*; they may be connected with *social ties* of various strengths [4], [14]. Naturally, a user prefers helping others (in data relay) with stronger social ties. Such social selfishness adds to the complexity of designing efficient multirate multicast protocols, especially when dealing with routing and capacity allocation decisions. To illustrate, a short path with high-capacity links may not be desirable unless the nodes along the way are highly willing to help the receiver.

In this work, we design an efficient, social-aware multirate multicast scheme in wireless networks, which can maximize the overall utility of all destinations allowable by their social relationships and available bandwidths. We model social preferences of users as differentiated costs for packet relay, which are weighted by the strength of social ties between the relays and the destinations. Combined with random linear network coding (NC) [3] at each layer for better multicast throughput, stochastic Lyapunov optimization [15] is utilized to design an

optimal joint end-to-end rate control, routing, and capacity allocation mechanism. A distributed implementation of the algorithm is proposed, by which each node needs only to make its own transmission decisions based on local information. With rigorous theoretical analysis, we study the optimality, stability, and complexity of our algorithm, as well as the impact of social preferences.

The contribution of this work can be summarized as follows:

First, as the first effort in the related literature, we investigate optimal multirate multicast in wireless networks where the network topology and link capacities are time-varying due to user mobility and channel fading. We exploit stochastic optimization techniques in the solution design.

Second, we model social selfishness of users as differentiated costs for their packet relay, which are weighted by the strength of the social tie between a relay and the destination, in a Lyapunov optimization framework for achieving receiver utility maximization. To the best of our knowledge, this is the first work investigating the impact of social selfishness on multicast protocol design in wireless networks.

Third, we design a joint end-to-end rate control, routing, and capacity allocation scheme, which is novelly combined with multi-resolution coding and random linear network coding to achieve social-aware utility-maximizing multirate multicast. A distributed implementation is further proposed.

Finally, through rigorous theoretical analysis, we show that the overall achieved utility can be arbitrarily close to the ultimate optimum, and that the transmission queues in the network have guaranteed stability. Decodability of network codes inside each layer and successful recovery of multiple layers at each receiver are also carefully proved. We explore impact of social selfishness on receiver utility under different social selfishness patterns, using both case studies and empirical studies. We observe that destinations having larger social tie strengths with the rest of network do not necessarily achieve lower throughput utility as compared with destinations with smaller social tie strengths, in both networks with uniformly distributed social ties and clustered social relationships.

The remainder of the paper is organized as follows. We discuss related work in Sec. II and present the problem model in Sec. III. Detailed protocol design and performance analysis are presented in Sec. IV and Sec. V, respectively. The protocol performance is evaluated via an empirical study in Sec. VI. We conclude the paper in Sec. VII.

II. Related Work

A. Multirate Multicast with Network Coding

Random linear network coding in multicast networks is introduced by Ho *et al.* [3]. Based on [3], Yan *et al.* [20] propose a dynamic intra-session network coding mechanism for single-rate multicast in time-varying wireless networks, and demonstrate that generation-based random network coding is sufficient to achieve performance optimality. To our best knowledge, [20] is the only existing work that applies Lyapunov optimization for multicast. Different from [20], our paper here tackles the multirate (instead of single-rate) multicast by applying multi-resolution coding in the Lyapunov optimization framework [15].

Sundaram *et al.* [19] tackle multirate multicast with layered flows and intra-layer network coding, by assigning different numbers of layers to each destination and constructing a multicast subgraph for each layer. A similar approach is proposed by Shao *et al.* [17]. Inter-layer network coding has also been considered and jointly applied with intra-layer network coding for multirate multicast in either centralized [2] or distributed manners [6]. However, inter-layer network coding requires each intermediate node to know the network topology and the rates of down-stream destinations, which is less practical in dynamic networks with time-varying topologies.

Much of existing literature assume a static network topology and fixed multicast paths that are either given or computed *a priori*. Apparently, only one work [7] addresses multirate multicast with dynamic routing decisions. Nevertheless, the number of supportable layers at each destination needs to be known *a priori*, which is not feasible in dynamic environments. No existing work has solved the optimal multirate multicast problem in networks with both topology and channel capacity changes; here, we are able to arrive at a solution using Lyapunov optimization.

B. Social Selfishness in Network Protocol Design

For this aspect, there can be two extremes: full node collaboration, or all the network users are completely selfish. Under the latter assumption, existing work have been focusing on incentive design, *e.g.*, [5], [21]. In comparison, we consider a new assumption of social selfishness, where users are not polarized to be completely selfish or altruistic, but prefer helping their social ties. This better captures user preferences in many practical networks [4], [14].

Few work exist on designing network protocols for socially selfish users. Li *et al.* [11] investigate routing design in socially selfish delay tolerant networks, where a node has differentiated probabilities in forwarding traffic. In this paper, we study a joint rate control, routing, and capacity allocation scheme to achieve optimal multirate multicast in dynamic wireless networks, which addresses social selfishness of users by differentiating relay costs towards different destinations.

III. Problem Model

We now present the multicast model, the modules of protocol design using Lyapunov optimization, and our social selfishness model in the framework.

A. Wireless Multicast with Socially Selfish Users

We consider a multicast session in a multi-hop wireless ad-hoc network, where $\mathcal{N}$ is the set of wireless nodes sharing a common available channel, $s \in \mathcal{N}$ is the multicast source, and $\mathcal{D} \subset \mathcal{N}$ is the set of destinations. The system runs in a time-slotted fashion. A generic node mobility model is considered, where the location of a node changes dynamically following an ergodic process but only at the beginning of each time slot.

The broadcast nature of wireless communication is exploited in our design, for maximum utilization of network capacity. Let h_{iJ} denote a directed hyperarc from node i to

node set J, where all nodes in J are within the transmission range of i. $\mathcal{H}(t)$ is the set of hyperarcs in the network in time slot t.[2] Let $c_{ij}(t)$ denote the maximum number of packets i can deliver to j during time slot t. Due to channel fading and node mobility, $c_{ij}(t)$ may change from one time slot to another in the range of $[0, c^{max}]$, following an ergodic process.[3] It remains constant within one time slot. The capacity $c_{iJ}(t)$ of hyperarc h_{iJ} in time slot t, *i.e.*, the maximum number of packets i can broadcast to nodes in J in t, is calculated as the minimum of the maximum numbers each node can receive, *i.e.*, $c_{iJ}(t) = \min_{j\in J}\{c_{ij}(t)\}$.

A generic interference model is employed to characterize interferences among transmissions along the hyperarcs. Let $\mathcal{I}(t)$ be the set of interference relations among potential hyperarc transmissions in time slot t, where $(h_{iJ}, h_{uZ}) \in \mathcal{I}(t)$ denotes that transmission along hyperarc h_{iJ} cannot be scheduled concurrently with that along hyperarc h_{uZ} during time slot t. In addition to interferences captured by $\mathcal{I}(t)$, *primary interference* is assumed at each node, *i.e.*, a node cannot transmit and receive simultaneously, and cannot transmit or receive over multiple hyperarcs at the same time.

In the multicast session, the source node has an infinitely backlogged stream to send. The stream is encoded with multi-resolution coding (MRC) into L layers with base layer 1, and a number of enhancement layers numbered $l > 1$. The maximum data rate of each layer is R packets per time slot. In each layer, the sub-stream is divided into consecutive *generations*, each including M packets, which are further encoded into M coded packets with random linear network coding (NC) [3]. The reason for using network coding in each layer is to increase the diversity of packets in the network, each being equivalently useful, and to avoid reception of duplicated packets arising from multi-path routing, in order to boost throughput [3]. Also, generation-based network coding is practical in cases of long flows [1], for reducing decoding complexity and delay. At a receiver, a generation in a layer is NC decodable if sufficient NC-coded packets of the generation are received, rendering a full-rank coefficient matrix; a NC decoded generation in enhancement layer l can be MRC decoded, if corresponding generations in lower layers have been received and recovered, as enabled by H.264/SVC like MRC techniques [16].

The strength of social tie between each pair of nodes i and d is characterized by a rational number $\rho_{id} \in [0, 1]$, where $\rho_{id} = 1$ is strongest and $\rho_{id} = 0$ means no tie at all. Such social ties will be elaborated on in connection with packet routing in later design.

Table I summarizes the notations for ease of reference.

B. Problem Model for Three Protocol Modules

We next model the problems involved in the three modules of the multirate multicast protocol.

End-to-End Rate Control: Starting concurrently for all layers, source s injects NC coded packets into the network, one generation after another, in each layer (see Fig. 1 for an example with consecutive generations $G1$, $G2$, $G3$, ... on three layers). Let $r_{dkl}(t) \in [0, R]$ be admissible end-to-end data rate, in terms of the number of packets admitted to the network (*i.e.*, $Q_s^{dkl}(t)$ to be introduced shortly), for generation k in layer l towards destination $d \in D$ in time slot t, such that network stability, to be defined in Sec. III-D, is achieved. A counter $P_{dkl}(t)$ is maintained at the source for not-yet-admitted packets at the beginning of t, for each generation k in each layer l towards each destination d. Each counter is initialized as $P_{dkl}(0) = M$ and updated with queueing law,

$$P_{dkl}(t+1) = \max\{P_{dkl}(t) - r_{dkl}(t), 0\}, \quad \forall d \in D, l \in [1, L], k \geq 1, t \geq 0. \quad (1)$$

When all packets for generation k on layer l towards destination d have been admitted to the network, *i.e.*, $P_{dkl}(t) = 0$ at some time slot t, the source starts to inject the next generation, $k + 1$, on that layer towards that destination. Thus, at any time slot, the source s deals with packets for one generation only, instead of all $k \geq 1$ ones, on each layer towards each destination. Note that counting $r_{dkl}(t)$ and $P_{dkl}(t)$ for different destinations $d \in D$ does not conflict with the multicast nature of our transmissions to be scheduled: we will show that when one coded packet in generation k of layer l is delivered to the next hop, the counters $r_{dkl}(t)$ and $P_{dkl}(t)$, $\forall d \in D$, will be increased and decreased, respectively.

To ensure the generations received on an enhancement layer can be MRC decoded, we need to guarantee that the average admitted data rate for a generation on a lower layer is no lower than that on a higher layer, *i.e.*,

$$\bar{r}_{dkl} \geq \bar{r}_{dkl^+}, \ \forall d \in D, l \in [1, L-1], k \geq 1, \quad (2)$$

where $l^+ = l + 1$, and $\bar{r}_{dkl} = \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}(r_{dkl}(t))$ is the time-averaged data rate of generation k in layer l for destination d. Here, $\mathbb{E}(\cdot)$ denotes the expectation.

Routing: Each node $i \in \mathcal{N}$ may receive data for different generations in multiple layers, and makes routing decisions

$\mathcal{N}$	Node set	$\mathcal{H}(t)$	Hyperarc set at slot t
h_{iJ}	Hyperarc $i \to J$	$c_{iJ}(t)$	Capacity of h_{iJ} at slot t
s	Source node	$\mathcal{D}$	Destination set
$U_l(\cdot)$	Utility function for layer l	$\mathbb{E}(\cdot)$	The expectation
ρ_{id}	Social tie between i and d	$\xi(\cdot)$	Social price function
$c_{ij}(t)$	Capacity between node i and j at slot t		
c^{max}	Maximum transmission capacity between any two nodes		
$\mathcal{I}(t)$	Set of interference relations at slot t		
L	Maximum number of multicast layers		
R	Maximum data rate of one layer		
M	Number of packets per generation		
$P_{dkl}(t)$	Backlog counter for destination d, generation k, layer l at slot t		
$Q_i^{dkl}(t)$	Queue on node i for destination d, generation k, layer l at slot t		
$Y_{dkl}(t)$	Virtual queue for destination d, generation k, layer l at slot t		
$G_{dkl}(t)$	Virtual queue for constraint (2)		
$r_{dkl}(t)$	Admitted packets to $Q_s^{dkl}(t)$ at slot t		
$\gamma_{dkl}(t)$	Auxiliary variable for $r_{dkl}(t)$		
$\mu_{ijJ}^{dkl}(t)$	Routed packets over h_{iJ} from Q_i^{dkl} to Q_j^{dkl} at slot t		
$g_{iJ}^{kl}(t)$	Physical flow over h_{iJ} for generation k and layer l at slot t		
$\alpha_{iJ}(t)$	Resource allocation variable for h_{iJ} at slot t		
$\mathcal{D}_{iJ}(t)$	Set of destinations, towards which the packets are transmitted over h_{iJ} at slot t		
$p_{iJ}(t)$	Social cost of scheduling h_{iJ} at slot t		

TABLE I
NOTATION.

[2]Assuming there are n nodes in node i's transmission range, up to $2^n - 1$ hyperarcs can be defined with i as the sender.

[3]Arbitrary channel fading or node mobility may make the problem intractable.

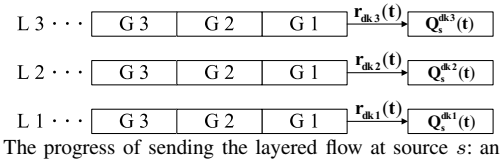


Fig. 1. The progress of sending the layered flow at source s: an example.

on further forwarding. Let $g_{iJ}^{kl}(t)$ denote the rate of the actual physical flow of packets in generation k of layer l, delivered over hyperarc h_{iJ} in time slot t. Since network coding is employed in each generation of a layer, we may also consider multiple virtual flows, each towards a different destination, inside this physical flow. Suppose node i maintains virtual packet queue Q_i^{dkl}, buffering packets in each generation k of each layer l destined to each destination node $d \in D$, except when node $i = d$ (packets delivered directly to application layer). The queues are virtual in the sense that pointers instead of true copies of packets are buffered, since the same packet may be enqueued in multiple queues for different destinations.

Let $\mu_{ijJ}^{dkl}(t)$ denote the rate of the virtual packet flow from node $i(\neq d)$'s queue $Q_i^{dkl}(t)$ to node j over hyperarc h_{iJ}, where $j \in J$, in time slot t. Meanwhile, $\mu_{uiJ}^{dkl}(t)$ is the virtual incoming rate from node u to i's queue $Q_i^{dkl}(t)$ over hyperarc h_{uJ} with $i \in J$, in t. The details on network coding and enqueueing/dequeueing of these virtual queues will be given when we discuss the algorithm in Sec. IV. The queueing law on the size of queue Q_i^{dkl} at node $i \in \mathcal{N}$ $(i \neq d)$ is as follows:

$$Q_i^{dkl}(t+1) = \max\{Q_i^{dkl}(t) - \sum_{j:j\in J, h_{iJ}\in\mathcal{H}(t)} \mu_{ijJ}^{dkl}(t), 0\} + \sum_{u:h_{uJ}\in\mathcal{H}(t), i\in J} \mu_{uiJ}^{dkl}(t) + \mathbf{1}_{\{i=s\}} r_{dkl}(t), \quad \forall d \in D, l \in [1, L], k \geq 1. \tag{3}$$

Here, $\mathbf{1}_{\{i=s\}}$ is an indicator function, which is equal to 1 if $i = s$ (source node), and 0 otherwise.

Let $\alpha_{iJ}(t) \in \{0, 1\}$ denote the capacity allocation decision for hyperarc h_{iJ} in time slot t, where $\alpha_{iJ}(t)$ is 1 if h_{iJ} is scheduled for transmission in time slot t, and 0 otherwise. We further have the following conditions at node i:

$$\sum_{j\in J} \mu_{ijJ}^{dkl}(t) \leq g_{iJ}^{kl}(t), \ \forall d \in \mathcal{D}, k \geq 1, l \in [1, L], h_{iJ} \in \mathcal{H}(t), \tag{4}$$

$$\sum_{l\in[1,L]} \sum_{k\geq 1} g_{iJ}^{kl}(t) \leq \alpha_{iJ}(t) c_{iJ}(t), \ \forall h_{iJ} \in \mathcal{H}(t), \tag{5}$$

$$\mu_{ijJ}^{dkl}(t) \geq 0, \ \forall h_{iJ} \in \mathcal{H}(t), j \in J, d \in D, k \geq 1, l \in [1, L]. \tag{6}$$

(4) states that the virtual flow rate of generation k in layer l for each destination d over a hyperarc should be no larger than the rate of the corresponding physical flow. (5) is the capacity constraint on hyperarc h_{iJ}.

Capacity Allocation: A capacity allocation and hyperarc scheduling scheme is needed in the MAC layer for achieving collision-free transmissions in the network. The following constraints guarantee a feasible capacity allocation scheme, where (7) guarantees that no interfering hyperarcs in $\mathcal{I}(t)$ would be scheduled for transmission simultaneously, and (8) ensures that each node i cannot transmit or receive data over multiple hyperarcs at the same time:

$$\alpha_{iJ}(t) + \sum_{h_{uZ}:(h_{iJ},h_{uZ})\in\mathcal{I}(t)} \alpha_{uZ}(t) \leq 1, \ \forall h_{iJ} \in \mathcal{H}(t), \tag{7}$$

$$\sum_{h_{iJ}\in\mathcal{H}(t)} \alpha_{iJ}(t) + \sum_{h_{uZ}:h_{uZ}\in\mathcal{H}(t), i\in Z} \alpha_{uZ}(t) \leq 1, \ \forall i \in \mathcal{N}. \tag{8}$$

C. Social Preference in Routing

A socially selfish node i may differentiate its capacity allocation when routing data to different destinations. Let $\xi(\rho_{id})$ be a non-negative non-increasing function on ρ_{id}, the strength of the social tie between i and d, which represents the unit cost of sending one unit of data destined to d at i, *e.g.*, one packet from queue $Q_i^{dkl}(t)$ for some generation k of layer l. Such a unit cost can be understood as the energy consumed for transmitting one unit of data, δ, biased by the social relationship between i and d: if ρ_{id} is larger (strong social tie), the cost is smaller, and vice versa. An example form $\xi(\rho_{id}) = \delta(1 - \rho_{id})$ is used in our simulation in Sec. VI.

Consider the multicast characteristic of our design and the broadcast nature of wireless transmissions. Sending a packet from queue $Q_i^{dkl}(t)$ destined to destination d is just virtual—the actual packet transmission is one over hyperarc h_{iJ}, which can be received by multiple nodes in J and enqueued in multiple destination queues at each node. Therefore, the actual overall cost $p_{iJ}(t)$ involved for transmitting actual packets for different generations and layers over hyperarc h_{iJ} in time slot t is related with social ties between node i and each of the intended destinations $d \in \mathcal{D}_{iJ}(t)$, as well as the data rates, *i.e.*, $\alpha_{iJ}(t) \cdot c_{iJ}(t)$. Here, $\mathcal{D}_{iJ}(t) \subseteq \mathcal{D}$ is the set of destinations, towards which the packets are transmitted over hyperarc h_{iJ} in time slot t, *i.e.*, $\{d | d \in \mathcal{D}, \exists k \geq 1, l \in [1, L], j \in J, \text{ such that } \mu_{ijJ}^{dkl} > 0\}$. We define $p_{iJ}(t)$ to be the maximum cost to route for any individual destination $d \in \mathcal{D}_{iJ}(t)$ as follows:[4]

$$p_{iJ}(t) = \alpha_{iJ}(t) \cdot c_{iJ}(t) \cdot \max_{d\in\mathcal{D}_{iJ}(t)} \{\xi(\rho_{id})\}. \tag{9}$$

In our optimal joint algorithm design with Lyapunov optimization, this cost will be included in the optimization objective, which aims to maximize the overall net utility subtracting cost.

D. Network Stability

Some important definitions and theorems are borrowed from [15] for use in our protocol design and analysis.

Definition 1 (Queue and Network Stability [15]): A queue Q is *strongly stable* (or *stable* for short) if and only if

$$\lim_{T\to\infty} \sup \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}(Q(t)) < \infty,$$

where $Q(t)$ is the queue size at time slot t and $\mathbb{E}(\cdot)$ is the expectation. A network is *strongly stable* (or *stable* for short) if and only if all queues in the network are strongly stable.

Theorem 1 (Necessity for Queue Stability [15]): For any queue $Q(t)$ with the following queuing law,

$$Q(t+1) = \max\{Q(t) - b(t)\} + a(t),$$

where $a(t)$ and $b(t)$ are the queue incoming rate and outgoing rate at time slot t, respectively. If queue $Q(t)$ is strongly stable, then its average incoming rate $\bar{a} =$

[4] $p_{iJ}(t)$ is defined as the maximum cost to route for any individual destination $d \in \mathcal{D}_{iJ}(t)$, since nodes are socially selfish and reluctant to provide free rides to destinations with low social ties.

$\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}(a(t))$ is no larger than the average outgoing rate $\bar{b}=\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}(b(t))$.

IV. Stochastic Optimal Algorithm

In this section, we present the utility maximization problem for social-aware multirate multicast, and design a dynamic algorithm based on Lyapunov optimization theory.

A. Utility Maximization Problem

Let $\bar{r}_{dkl}$ denote the average end-to-end admissible data rate of generation k on layer l for destination d, such that $\bar{r}_{dkl}=\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}(r_{dkl}(t))$. Let the vector of average end-to-end admissible rates, $\bar{\mathbf{r}}=(\bar{r}_{dkl},d\in\mathcal{D},l\in[1,L],k\geq 1)$, denote the throughput of the network.

Let $U_l(\cdot)$ be a concave, differentiable, and non-decreasing utility function on throughput $\sum_{k\geq 1}\bar{r}_{dkl}$ of layer l, received at destination d. $\bar{p}_{iJ}=\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}(p_{iJ}(t))$ is the time averaged cost of transmitting packets over hyperarc h_{iJ} in a time slot. Our objective is to maximize the overall *net utility* (utility-minus-cost) with guarantee on network stability.

$$\begin{aligned}\max\quad & \sum_{d\in D}\sum_{l\in[1,L]}U_l(\sum_{k\geq 1}\bar{r}_{dkl})-\sum_{\exists t,h_{iJ}\in\mathcal{H}(t)}\bar{p}_{iJ} \qquad (10)\\ s.t.\quad & \bar{\mathbf{r}}\in\Lambda.\\ & 0\leq r_{dkl}(t)\leq R,\forall d\in\mathcal{D},l\in[1,L],k\geq 1,t=0,1,2,\ldots\end{aligned}$$

Λ is the *capacity region* of the network. $\bar{\mathbf{r}}\in\Lambda$ guarantees that the derived average admissible data rates in $\bar{\mathbf{r}}$ can achieve network stability, *i.e.*, there exists a routing and capacity allocation protocol that decides a set of feasible admissible data rates $r_{dkl}(t),\forall d\in\mathcal{D},l\in[1,L],k\geq 1$, in each time slot t (*i.e.*, those satisfying constraints (1)- (9)), such that all queues are strongly stable in the network (Definition 1).

B. Virtual Queues

We exploit Lyapunov optimization techniques to make decisions in each time slot, *i.e.*, $r_{dkl}(t)$ and $p_{iJ}(t)$, while guaranteeing their time averages, *i.e.*, $\bar{r}_{dkl}$ and $\bar{p}_{iJ}$, maximize the net utility. Two types of virtual queues are introduced to facilitate the design of a dynamic algorithm.

Virtual queue $Y_{dkl}(t)$: According to Lyapunov optimization theory [15], if the utility functions $U_l(\cdot),\forall l\in[1,L]$, are non-linear, a virtual queue $Y_{dkl}(t)$ and an auxiliary variable $\gamma_{dkl}(t)$ should be introduced for each rate control variable $r_{dkl}(t)$, $\forall d\in D,l\in[1,L],k\geq 1$, with the following queueing law:

$$Y_{dkl}(t+1)=\max\{Y_{dkl}(t)-r_{dkl}(t),0\}+\gamma_{dkl}(t), \qquad (11)$$

under the constraint that

$$0\leq\gamma_{dkl}(t)\leq R, \qquad (12)$$

such that guaranteeing the stability of this queue will ensure that our dynamic algorithm can get a lower bound for the optimal net utility in (10).

Virtual queue $G_{dkl}(t)$: To ensure (2) by controlling rates in each time slot, we introduce another virtual queue $G_{dkl}(t)$ at source s, for each generation $k\geq 1$ of layer $l\in[1,L-1]$ destined towards each destination $d\in\mathcal{D}$ with queueing law:

$$\begin{aligned}G_{dkl}(t+1)=&\max\{G_{dkl}(t)-r_{dkl}(t),0\}+r_{dkl^+}(t),\\ &\forall d\in D,l\in[1,L-1],k\geq 1,\ \text{with } l^+=l+1. \qquad (13)\end{aligned}$$

By Theorem 1, if each virtual queue $G_{dkl}(t)$ is made stable, then the average end-to-end rate of layer l, $\bar{r}_{dkl}$, is no less than that of layer $l+1$, $\bar{r}_{dkl^+}$, *i.e.*, constraint (2) is satisfied.

C. Dynamic Algorithm

In summary, in our dynamic algorithm that solves the utility maximization problem, three types of queues are needed. Let $\Theta(t)=(\mathbf{Q},\mathbf{Y},\mathbf{G})$ be the vector of all queues in the system in time slot t, with $\mathbf{Q}=(Q_i^{dkl}(t),d\in\mathcal{D},i\neq d,l\in[1,L],k\geq 1)$, $\mathbf{Y}=(Y_{dkl}(t),d\in\mathcal{D},l\in[1,L],k\geq 1)$, and $\mathbf{G}=(G_{dkl}(t),d\in\mathcal{D},l\in[1,L-1],k\geq 1)$. Define the Lyapunov function as

$$\begin{aligned}L(\Theta(t))=&\frac{1}{2}\sum_{d\in D}\sum_{k\geq 1}[\sum_{l\in[1,L]}[(Y_{dkl}(t))^2+\sum_{i\in\mathcal{N},i\neq d}(Q_i^{dkl}(t))^2]\\ &+\sum_{l\in[1,L-1]}(G_{dkl}(t))^2].\end{aligned}$$

The conditional one-slot Lyapunov drift is

$$\Delta(\Theta(t))=L(\Theta(t+1))-L(\Theta(t)).$$

Squaring the queuing laws (3), (11) and (13), we derive the following inequality (detailed derivations are in the technical report [10]):

$$\begin{aligned}&\Delta(\Theta(t))-V\cdot(\sum_{d\in D}\sum_{l\in[1,L]}U_l(\sum_{k\geq 1}\gamma_{dkl}(t))-\sum_{h_{iJ}\in\mathcal{H}(t)}p_{iJ}(t))\\ \leq&B-\Phi(t)-\Psi(t)-\Omega(t), \qquad (14)\end{aligned}$$

where $B=\frac{|\mathcal{D}|}{2}[(4L-2)\cdot R^2+L\cdot(c^{max}+R)^2+2L\cdot(|\mathcal{N}|-1)\cdot(c^{max})^2]$ is a constant value, and V is a user-defined constant that can be understood as the weight of the net utility. $\Phi(t)$, $\Psi(t)$, and $\Omega(t)$ are as follows.

- Terms related to auxiliary variables $\gamma_{dkl}(t)$:

$$\Phi(t)=V\cdot\sum_{d\in D}\sum_{l\in[1,L]}U_l(\sum_{k\geq 1}\gamma_{dkl}(t))-\sum_{d\in D}\sum_{l\in[1,L]}\sum_{k\geq 1}Y_{dkl}(t)\cdot\gamma_{dkl}(t).$$

- Terms related to end-to-end rate control variables $r_{dkl}(t)$:

$$\begin{aligned}\Psi(t)=&\sum_{d\in D}\sum_{l\in[1,L]}\sum_{k\geq 1}[Y_{dkl}(t)-Q_s^{dkl}(t)+\mathbf{1}_{\{l<L\}}\cdot G_{dkl}(t)\\ &-\mathbf{1}_{\{l>1\}}\cdot G_{dkl^-}(t)]\cdot r_{dkl}(t).\end{aligned}$$

  Here, $l^-=l-1$. $\mathbf{1}_{\{l<L\}}$ and $\mathbf{1}_{\{l>1\}}$ are two indicator functions:

$$\mathbf{1}_{\{l<L\}}=\begin{cases}1 & l<L\\ 0 & \text{otherwise,}\end{cases}\qquad\mathbf{1}_{\{l>1\}}=\begin{cases}1 & l>1\\ 0 & \text{otherwise.}\end{cases}$$

- Terms related to routing variables $\mu_{ijJ}^{dkl}(t)$ and capacity allocation variables $\alpha_{iJ}(t)$:

$$\begin{aligned}\Omega(t)=&\sum_{d,l,k,i\neq d,j}\sum_{h_{iJ}\in\mathcal{H}(t)}\mu_{ijJ}^{dkl}(t)\cdot(Q_i^{dkl}(t)\\ &-Q_j^{dkl}(t))-V\sum_{h_{iJ}\in\mathcal{H}(t)}p_{iJ}(t).\end{aligned}$$

Here, $p_{iJ}(t)$ can be determined by capacity allocation variable $\alpha_{iJ}(t)$ according to Eqn. (9).

Applying the *drift-plus-penalty* framework in Lyapunov optimization [15], we derive the following dynamic algorithm that observes queues $\Theta(t)$ at each time slot t and makes control decisions that maximize $\Phi(t)$, $\Psi(t)$ and $\Omega(t)$, such that the lower bound for the net utility in (10) is maximized [15].

End-to-end Rate Control: At the beginning of each time slot t, source s decides the auxiliary variables $\gamma_{dkl}(t)$ and end-to-end rates $r_{dkl}(t)$, for the specific generation k it is currently

sending in each layer l towards each destination d, by solving the following optimization problems, respectively.

$$\begin{aligned} \max \quad & V \cdot U_l(\gamma_{dkl}(t)) - Y_{dkl}(t)\gamma_{dkl}(t) \\ s.t. \quad & 0 \le \gamma_{dkl}(t) \le R, \end{aligned}$$

and

$$\begin{aligned} \max \quad & [Y_{dkl}(t) - Q_s^{dkl}(t) + \mathbf{1}_{\{l<L\}} \cdot G_{dkl}(t) \\ & - \mathbf{1}_{\{l>1\}} \cdot G_{dkl^-}(t)] \cdot r_{dkl}(t) \\ s.t. \quad & 0 \le r_{dkl}(t) \le \min\{P_{dkl}(t), R\}. \end{aligned}$$

The above two problems are convex with linear constraints, whose solutions can be given as follows, $\forall d \in D, l \in [1, L]$:

$$\gamma_{dkl}(t) = \max\{\min\{U_l^{'-1}(\frac{Y_{dkl}(t)}{V}), R\}, 0\}, \tag{15}$$

$$r_{dkl}(t) = \begin{cases} \min\{P_{dkl}(t), R\} & \text{if } Y_{dkl}(t) + \mathbf{1}_{\{l<L\}} \cdot G_{dkl}(t) \\ & > Q_s^{dkl}(t) + \mathbf{1}_{\{l>1\}} \cdot G_{dkl^-}(t) \\ 0 & \text{otherwise} \end{cases}, \tag{16}$$

where $U_l^{'-1}(\cdot)$ is the inverse function of $U_l'(\cdot)$, the first order derivative of $U_l(\cdot)$. The above solutions are only related to local information at source s, *e.g.*, $Y_{dkl}(t)$, $P_{dkl}(t)$, $G_{dkl^-}(t)$, $Q_s^{dkl}(t)$, and can thus be derived in a fully distributed fashion.

Joint Routing and Capacity Allocation: At the beginning of each time slot t, routing variables $\mu_{ijJ}^{dkl}(t)$ and capacity allocation variables $\alpha_{iJ}(t)$ in the network ($\forall h_{iJ} \in \mathcal{H}(t), j \in J, d \in D, l \in [1, L], k \ge 1$), can be jointly decided by solving the following optimization problem.

$$\max \ \ \Omega(t), \ s.t. \ \text{Constraints (4) - (9).} \tag{17}$$

We next simplify problem (17) as a pure capacity allocation problem (24), related only to variables $\alpha_{iJ}(t), \forall h_{iJ} \in \mathcal{H}(t)$.

We start by reducing the number of variables in the optimization, by analyzing the structure of the objective function and constraints. Considering constraint (4) and the first half of the objective function where $\mu_{ijJ}^{dkl}(t)$'s appear, we can conclude that for given d, l, k and h_{iJ}, only the routing variable $\mu_{ijJ}^{dkl}(t)$ associated with the largest weight $Q_i^{dkl}(t) - Q_j^{dkl}(t)$ in (17) needs to remain, while the rest can be safely set to zero, *i.e.*,

$$\mu_{ijJ}^{dkl}(t) = \begin{cases} g_{iJ}^{kl}(t) & \text{if } j = arg\max_{j \in J}(Q_i^{dkl}(t) - Q_j^{dkl}(t)) \\ 0 & \text{otherwise} \end{cases}. \tag{18}$$

Define

$$j_{iJdkl}^* = arg\max_{j \in J}(Q_i^{dkl}(t) - Q_j^{dkl}(t)). \tag{19}$$

Let $\mathcal{D}_{iJ}(t)$ be the set of destinations, towards which the packets over hyperarc h_{iJ} in time slot t are destined (we will discuss how the set is decided subsequently). (17) can be simplified to $\sum_{h_{iJ}} \sum_l \sum_k [g_{iJ}^{kl}(t) \cdot \sum_{d \in \mathcal{D}_{iJ}(t)} (Q_i^{dkl}(t) - Q_{j_{iJdkl}^*}^{dkl}(t))] - V \sum_{h_{iJ}} p_{iJ}(t)$. Define

$$W_{iJ}^{kl}(t) = \sum_{d \in \mathcal{D}_{iJ}(t)} (Q_i^{dkl}(t) - Q_{j_{iJdkl}^*}^{dkl}(t)). \tag{20}$$

Considering constraint (5), we infer that for given h_{iJ}, only the physical flow rate variable $g_{iJ}^{kl}(t)$ associated with the largest weight $W_{iJ}^{kl}(t)$ should remain, and the rest can be safely set to zero, *i.e.*,

$$g_{iJ}^{kl}(t) = \begin{cases} \alpha_{iJ}(t) \cdot c_{iJ}(t) & \text{if } (k, l) = arg\max_{l \in [1,L], k \ge 1}\{W_{iJ}^{kl}(t)\} \\ 0 & \text{otherwise} \end{cases}. \tag{21}$$

In this way, (17) can be further simplified to $\sum_{h_{iJ}} [\alpha_{iJ}(t) \cdot c_{iJ}(t) \cdot \max_{l \in [1,L], k \ge 1}\{W_{iJ}^{kl}(t)\}] - V \sum_{h_{iJ}} p_{iJ}(t)$. Substituting (9) into this function, we derive the new objective function as

$$\sum_{h_{iJ} \in \mathcal{H}(t)} [\alpha_{iJ}(t) \cdot c_{iJ}(t) \cdot (\max_{l \in [1,L], k \ge 1}\{W_{iJ}^{kl}(t)\} - V \max_{d \in \mathcal{D}_{iJ}(t)}\{\xi(\rho_{id})\})].$$

We next decide the $\mathcal{D}_{iJ}(t)$ as the subset of $\mathcal{D}$ which maximizes $(\max_{l \in [1,L], k \ge 1}\{W_{iJ}^{kl}(t)\} - V \max_{d \in \mathcal{D}_{iJ}(t)}\{\xi(\rho_{id})\})$ in the above objective function, *i.e.*,

$$\mathcal{D}_{iJ}(t) = arg \max_{\mathcal{D}' \subseteq \mathcal{D}}\{\max_{l \in [1,L], k \ge 1}\{W_{iJ}^{kl}(t)\} - V \max_{d \in \mathcal{D}'}\{\xi(\rho_{id})\}\}.$$

Define

$$W_{iJ}(t) = c_{iJ}(t) \cdot (\max_{\mathcal{D}' \subseteq \mathcal{D}}\{\max_{l \in [1,L], k \ge 1}\{W_{iJ}^{kl}(t)\} - V \cdot \max_{d \in \mathcal{D}'}\{\xi(\rho_{id})\}\}). \tag{22}$$

(23)

The joint routing and capacity allocation problem in (17) can be finally reduced to the following:

$$\max \sum_{h_{iJ} \in \mathcal{H}(t)} \alpha_{iJ}(t) \cdot W_{iJ}(t), \quad s.t. \ \text{Constraints (7), (8).} \tag{24}$$

After solving the above capacity allocation problem, routing decisions along each hyperarc $h_{iJ} \in \mathcal{H}(t)$ can be made as follows, according to Eqn. (18) and (21):

$$\mu_{ijJ}^{dkl}(t) = \begin{cases} \alpha_{iJ}(t) \cdot c_{iJ}(t) & \text{if } d \in \mathcal{D}_{iJ}(t),\ j = j_{iJdkl}^* \text{ and} \\ & (k, l) = arg\max_{l \in [1,L], k \ge 1}\{W_{iJ}^{kl}(t)\} \\ 0 & \text{otherwise.} \end{cases}. \tag{25}$$

The capacity allocation problem (24) is a 0-1 integer program. A centralized solution with $1 - \theta$ approximation ratio to the optimality can be obtained using the branch-and-bound method [8], where $\theta \in (0, 1)$ is the solution accuracy defined by the users. We also design a distributed algorithm to solve this problem, to be discussed in Sec. IV-D.

Packet Scheduling

The above calculation of routes and transmission rates leads to a detailed solution on network coding and routing. Consider hyperarc h_{iJ} scheduled for transmissions in t, *i.e.*, $\alpha_{iJ}(t) = 1$. The implication of Eqn. (21) is that the actual packets to deliver over the hyperarc in t should all be from one selected generation k^* of a selected layer l^*, where $(k^*, l^*) = arg\max_{l \in [1,L], k \ge 1}\{W_{iJ}^{kl}(t)\}$ corresponding to the largest differential queue backlog $W_{iJ}^{kl}(t)$, calculated with Eqn. (20). Eqn. (18) shows that inside the physical packet flow, only selected virtual flows are enclosed (corresponding to the non-zero $\mu_{ijJ}^{dkl}(t)$'s), which reveals how each network coded actual packet should be produced at i and delivered to the proper virtual queues on the next-hop nodes in J: each packet should be produced at node i using random linear network coding from the head-of-line packets in queues $Q_i^{dk^*l^*}(t)$, $\forall d \in \mathcal{D}_{iJ}(t)$ (Eqn. (22)), and dispatched to each destination d's queue $Q_{j_{iJdk^*l^*}^*}^{dk^*l^*}(t)$ at the corresponding next-hop node $j_{iJdk^*l^*}^*$ (decided by Eqn. (19) according to the largest differential queue backlog). After network coding, the head-of-line packets in i's queues $Q_i^{dk^*l^*}(t)$, $\forall d \in \mathcal{D}_{iJ}(t)$ will be removed, and the number of coded packets that are actually delivered over hyperarc h_{iJ} is $\max_{d \in \mathcal{D}_{iJ}(t)}\{\min\{Q_i^{dk^*l^*}(t), c_{iJ}(t)\}\}$, according to Eqn. (25) and considering the actual number of packets in each queue $Q_i^{dk^*l^*}(t)$.

The sketch of our dynamic algorithm is summarized in Algorithm 1. The implication of the joint routing and capacity allocation is to prioritize transmissions of more urgent generations/layers, *i.e.*, with larger differential queue backlogs, and to direct the packets to destinations with stronger social ties, *i.e.*, lower social cost, over a hyperarc with higher capacity.

Algorithm 1 Dynamic Net Utility Maximization Algorithm in Time Slot t

Input: $Q_i^{dkl}(t)$, $Y_{dkl}(t)$, $G_{dkl}(t)$, $P_{dkl}(t)$, ρ_{id}, $\mathcal{H}(t)$, $\mathcal{I}(t)$, $c_{iJ}(t)$, R, V, $(\forall d \in \mathcal{D}, l \in [1, L], k \geq 1, i \in \mathcal{N}, i \neq d, h_{iJ} \in \mathcal{H}(t))$.
Output: $\gamma_{dkl}(t)$, $r_{dkl}(t)$, $\alpha_{iJ}(t)$, $\mu_{ijJ}^{dkl}(t)$ $(\forall d \in \mathcal{D}, l \in [1, L], k \geq 1, i \in \mathcal{N}, i \neq d, h_{iJ} \in \mathcal{H}(t), j \in J)$.

1: **End-to-End Rate Control**: For each generation $k \geq 1$ on layer $l \in [1, L]$ for destination $d \in \mathcal{D}$, source s decides the end-to-end rate $r_{dkl}(t)$ and auxiliary variable $\gamma_{dkl}(t)$ by Eqn. (15) and (16).
2: **Joint Routing and Capacity Allocation**: For each hyperarc $h_{iJ} \in \mathcal{H}(t)$, its weight $W_{iJ}(t)$ is calculated with Eqn. (23).
- Derive capacity allocation variable $\alpha_{iJ}, \forall h_{iJ} \in \mathcal{H}(t)$, by solving (24) with the branch-and-bound algorithm [8] or our distributed algorithm, Algorithm 2.
- Routing decisions are made according to Eqn. (25).
3: **Packet Scheduling**: On each hyperarc h_{iJ} scheduled for transmission, node i transmits $\max_{d \in \mathcal{D}_{iJ}(t)}\{\min\{Q_i^{dk^*l^*}(t), c_{iJ}(t)\}\}$ network-coded packets:
- Take the head-of-line packets from queues $Q_i^{dk^*l^*}(t)$, $\forall d \in \mathcal{D}_{iJ}(t)$, do random linear combination of these packets to produce a coded packet.
- Dequeue all packets used for network coding from their respective queue $Q_i^{dk^*l^*}(t)$.
- The newly coded packet is delivered to each destination d's queue $Q_{j^*_{iJdk^*l^*}}^{dk^*l^*}(t)$ at the corresponding next-hop node $j^*_{iJdk^*l^*}$.
4: Update virtual queues $P_{dkl}(t+1)$, $Y_{dkl}(t+1)$ and $G_{dkl}(t+1)$ based on queuing law (1), (11) and (13), respectively.

Algorithm 2 Distributed Capacity Allocation Algorithm in Time Slot t

Input: $Q_i^{dkl}(t)$, ρ_{id}, $\mathcal{H}(t)$, $\mathcal{I}(t)$, $c_{iJ}(t)$, V, $(\forall d \in \mathcal{D}, l \in [1, L], k \geq 1, i \in \mathcal{N}, i \neq d, h_{iJ} \in \mathcal{H}(t))$.
Output: $\alpha_{iJ}(t)$ $(\forall h_{iJ} \in \mathcal{H}(t))$.

1: **Initialization**
- Initialize capacity allocation variable $\alpha_{iJ}(t) \leftarrow 0, \forall h_{iJ} \in \mathcal{H}$, and a candidate set of hyperarcs to schedule $\mathcal{L}_i \leftarrow \emptyset$;
- Exchange queue sizes $Q_i^{dkl}(t)$, $\forall d \in \mathcal{D}, l \in [1, L], k \geq 1$ with neighbors;
- Calculate and propagate weight $W_{iJ}(t)$ calculated using Eqn. (23) for each hyperarc $h_{iJ} \in \mathcal{H}(t)$;
2: **Capacity allocation**.
- For each hyperarc $h_{iJ} \in \mathcal{H}(t)$, do:
  if $W_{iJ} \geq \max_{h_{uZ} \in \mathcal{H}(t), (h_{iJ}, h_{uZ}) \in \mathcal{I}(t)}\{W_{uZ}\}$
    update candidate hyperarc set $\mathcal{L}_i \leftarrow \mathcal{L}_i \bigcup \{h_{iJ}\}$.
- Schedule hyperarc $h_{iJ} = \arg\max_{h_{iZ} \in \mathcal{L}_i}\{W_{iZ}\}$ for transmission by setting $\alpha_{iJ}(t) = 1$; inform each neighbor and senders of interfering hyperarcs about this allocation.
3: **Information update**: Upon receiving a capacity allocation notification, update candidate hyperarc set and inform the updates to sender of each interfering hyperarc.

D. Distributed Implementation

We next propose a distributed algorithm, Algorithm 2, to solve the capacity allocation problem in (24). Here we refer to node j with $h_{iJ} \in \mathcal{H}(t), j \in J$, as a "neighbor" of node i, and each hyperarc h_{iJ} as a "local hyperarc" of node i. The local network topology, *i.e.*, $\forall h_{iJ} \in \mathcal{H}(t)$ and the corresponding channel status, *i.e.*, $\forall (h_{iJ}, h_{uZ}) \in \mathcal{I}(t)$ and $c_{iJ}(t), \forall h_{iJ} \in \mathcal{H}(t)$, can be obtained by sending pilot bits by each node i.

The distributed algorithm executed at each node greedily schedules one local hyperarc for transmission in each time slot. Node i calculates weight $W_{iJ}(t)$ based on Eqn. (23) for each local hyperarc h_{iJ}, using necessary information from neighbors. A hyperarc h_{iJ} satisfying the following two conditions will be chosen for transmission: (i) $W_{iJ}(t)$ is the largest among the weights $W_{uZ}(t)$ on all its interfering hyperarcs $h_{uZ} \in \mathcal{H}(t)$, where $(h_{iJ}, h_{uZ}) \in \mathcal{I}(t)$; (ii) W_{iJ} is also the largest among the weights on all the local hyperarcs at node i, each of which has the largest weight among its respective interfering hyperarcs in the network.

We will show in Sec. VI that the performance of the distributed protocol is close to that achieved by the centralized branch-and-bound capacity allocation method.

V. Performance Analysis

We prove the utility optimality and network stability achieved by our Algorithm 1, and discuss the impact of social selfishness and the storage complexity.

A. Utility Optimality and Network Stability

Definition 2 ($\epsilon-$optimum): The ϵ-optimal solution $(\bar{\mathbf{r}}^\epsilon, \bar{\mathbf{p}}^\epsilon)$, with $\bar{\mathbf{r}}^\epsilon = (\bar{r}^\epsilon_{dkl}, \forall d \in D, l \in [1, L], k \geq 1)$ and $\bar{\mathbf{p}}^\epsilon = (\bar{p}^\epsilon_{iJ}, \exists t, h_{iJ} \in \mathcal{H}(t))$, is the optimal solution to the modified utility maximization problem from that in (10) by replacing constraint $\bar{\mathbf{r}} \in \Lambda$ by $\bar{\mathbf{r}} + \vec{\epsilon} \in \Lambda$ where $\vec{\epsilon} = (\epsilon, \ldots, \epsilon)$ and $\epsilon > 0$.

Theorem 2 (Utility Optimality and Network Stability): Suppose all network-coded packets can be successfully decoded at the destinations. The overall net utility achieved with Algorithm 1 is within a constant gap $\frac{B}{V}$ from the ϵ-optimum utility, *i.e.*,

$$\lim_{T\to\infty} \inf \frac{1}{T} \sum_{t=0}^{T-1} \left[\sum_{d\in\mathcal{D}} \sum_{l\in[1,L]} U_l(\sum_{k\geq 1} r_{dkl}(t)) - \sum_{h_{iJ}\in\mathcal{H}(t)} p_{iJ}(t) \right]$$
$$\geq \sum_{d\in\mathcal{D}} \sum_{l\in[1,L]} U_l(\sum_{k\geq 1} \bar{r}^\epsilon_{dkl}) - \sum_{\exists t, h_{iJ}\in\mathcal{H}(t)} \bar{p}^\epsilon_{iJ} - \frac{B}{V},$$

and the network is stable as

$$\lim_{T\to\infty} \sup \frac{1}{T} \sum_{t=0}^{T-1} \sum_{d\in\mathcal{D}} \sum_{k\geq 1} [\sum_{l\in[1,L]} [\sum_{i\in\mathcal{N}, i\neq d} \mathbb{E}(Q_i^{dkl}(t)) + \mathbb{E}(Y_{dkl}(t))]$$
$$+ \sum_{i\in[1,L-1]} \mathbb{E}(G_{dkl}(t))]$$
$$\leq \frac{L+1}{\epsilon} [B + V[\sum_{d\in\mathcal{D}} \sum_{l\in[1,L]} [U_l(R) - U_l(\sum_{k\geq 1} \bar{r}^\epsilon_{dkl})] + \sum_{\exists t, h_{iJ}\in\mathcal{H}(t)} \bar{p}^\epsilon_{iJ}]].$$

This theorem is proved in [10] using Lyapunov optimization theory. Since B is a constant independent of V, Theorem 2 shows that the overall utility achieved with Algorithm 1 can be arbitrarily close to the optimum utility of (10) when $\epsilon \to 0$ and $V \to \infty$.

Corollary 1: Suppose all network-coded packets can be successfully decoded at the destinations. If generation k of layer $l, \forall k \geq 1, l \in [1, L]$, is received at destination $d, \forall d \in \mathcal{D}$, then it can be successfully MRC decoded, *i.e.*, generation k in all lower layers $l' < l$ is successfully received at d as well.

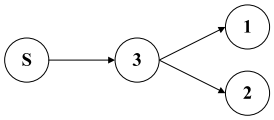


Fig. 2. A toy network with two destinations and one relay node.

This corollary is proved by induction in [10].

We next analyze the success probability of network-coded decoding, and show the achievable utility in this case.

Lemma 1: Suppose generation k of layer l is received by a set of destinations $\mathcal{D}_{kl}$, and p_{kl} is the probability that all destinations in $\mathcal{D}_{kl}$ can successfully decode this generation k of layer l. We have

$$p_{kl} \geq (1 - \frac{|\mathcal{D}_{kl}|}{q})^{B_2},$$

where $q > |\mathcal{D}_{kl}|$ is the finite field size in network coding, and $B_2 = \frac{|\mathcal{N}| \cdot M \cdot C^{max}}{2\bar{r}}$ with $\bar{r} = \sum_{l \in [1,L]} \sum_{k \geq 1} \max_{d \in \mathcal{D}} \{\bar{r}_{dkl}\}$.

This lemma is proved in [10] based on results of [3].

Theorem 3 (Utility Optimality with Imperfect NC *Decoding):* Under the decoding probability of network-coded packets given in Lemma 1, the overall net utility achieved with Algorithm 1 is within a constant gap B/V from the utility achieved with a throughput that is a constant fraction of the ϵ-optimal throughput, *i.e.*,

$$\lim_{T \to \infty} \inf \frac{1}{T} \sum_{t=0}^{T-1} \left[\sum_{d \in \mathcal{D}} \sum_{l \in [1,L]} U_l(\sum_{k \geq 1} r_{dkl}(t)) - \sum_{h_{iJ} \in \mathcal{H}(t)} p_{iJ}(t) \right]$$
$$\geq \sum_{d \in \mathcal{D}} \sum_{l \in [1,L]} U_l((1 - \frac{|\mathcal{D}|}{q})^{B_2} \sum_{k \geq 1} \bar{r}^{\epsilon}_{dkl}) - \sum_{\exists t, h_{iJ} \in \mathcal{H}(t)} \bar{p}^{\epsilon}_{iJ} - \frac{B}{V}.$$

This theorem is proved based on Theorem 2 and Lemma 1 in [10]. The network stability result in Theorem 2 still holds.

B. Impact of Social Selfishness

We next present a proposition on the impact of social selfishness on multicast performance as experienced by different destinations.

Proposition 1: For each destination $d \in \mathcal{D}$, suppose its social ties with all other nodes in the network are the same, *i.e.*, $\rho_{id} = \rho_d, \forall i \in \mathcal{N}, i \neq d$. Sort all destinations in descending order of their social ties, ρ_d, into a sequence $\{1, \ldots, |\mathcal{D}|\}$. Suppose $\rho_0 = 0$ and the network is homogeneous, *i.e.*, the number of hops from the source to each destination is identical while the link capacity on each hop is also the same. We derive the following cases (where $d^- = d - 1$ and $d^+ = d + 1$):

(I) If $\xi(\rho_{d^+}) + \xi(\rho_{d^-}) \geq 2\xi(\rho_d)$, $\forall d \in [1, |\mathcal{D}| - 1]$, the throughput $\sum_{l \in [1,L]} \sum_{k \geq 1} \bar{r}_{dkl}$ achieved by destinations with stronger social ties with other nodes, is no lower than that achieved by destinations with weaker social ties with others.

(II) If $\xi(\rho_{d^+}) + \xi(\rho_{d^-}) < 2\xi(\rho_d)$, $\forall d \in [1, |\mathcal{D}| - 1]$, the destinations can achieve the same throughput.

We briefly illustrate the proposition with a case study in Fig. 2, and a more detailed analysis can be found in [10]. There are four nodes: s is the source, 1 and 2 are two destinations, and 3 is the relay. Let $\rho_1 > \rho_2$. Only the base layer is considered for simplicity of illustration. Link capacities are identical over different links and time slots.

In case (I), given $\xi(\rho_2) \geq 2\xi(\rho_1)$ and that $U_l(\cdot)$ is concave, differentiable and non-decreasing, we derive $\min\{U_l^{'-1}(2\xi(\rho_1)), R\} \geq \min\{U_l^{'-1}(2(\xi(\rho_2) - \xi(\rho_1))), R\}$, where the former is throughput achieved by destination 1 and the later is throughput achieved by destination 2.

In case (II), the two destinations achieve the same throughput of $\min\{U_l^{'-1}(\xi(\rho_2)), R\}$.

C. Storage Complexity

In our algorithm, each node $i \in \mathcal{N}$ maintains at most $L \cdot |\mathcal{D}|$ queues of packets, *i.e.*, $Q_i^{dkl}(t), d \in \mathcal{D}, l \in [1, L]$ at any given time t. Since the generations are transmitted sequentially, at any time t, only one queue per layer per destination needs to be maintained at i, containing packets in the current generation being delivered. In addition, recall that one packet may be enqueued in multiple packet queues, and thus these queues may only cache pointers to the same copy. Source node s maintains additional $(2L-1) \cdot |\mathcal{D}|$ virtual queues, *i.e.*, $Y_{dkl}(t)$'s and $G_{dkl}(t)$'s, which can be implemented as counters only and consume negligible storage space.

VI. Empirical Study

We evaluate the performance of our dynamic algorithm and the impact of different social relationship patterns with discrete-event simulations under realistic settings. A wireless network is simulated with $|\mathcal{N}| = 50$ (or 100), where a source streams data to 15 (or 30) randomly chosen destinations. The stream is encoded into $L = 5$ layers with MRC. At time slot 0, each node is randomly positioned in a disk area with radius 1000. After each time slot, each node moves randomly to a position in a disk of radius 1 centered at its previous position. The transmission range of a node is 10. We adopt the graph interference model such that hyperarc h_{iJ} can be scheduled only if all the receivers in J are out of the transmission range of any other current transmitters. The primary interference is also avoided. Link capacity $c_{ij}(t)$ is randomly chosen from $[0, c^{max}]$ in a slot-by-slot fashion. The max data rate of each layer is $R = 10$ packets per time slot. In our random linear network coding, each generation contains $M = 500$ packets and the finite field size is $q = 2^{10}$. $U_l(x) = \lg(1 + x), \forall l$, $\xi(\rho_{id}) = 1 - \rho_{id}$, and $V = 5000$.

To add the social ties, we first construct a social graph following a power law distribution of node degree with shape parameter $k = 1.76$ [13], with average node degree of 5 (or 8) in a 50 (or 100) node network. Then we assign ρ_{id} to the links in the social graph in three ways:

1) **Uniform Distribution of Social Ties (UST)**: The social ties ρ_{id} are uniformly randomly assigned between $(0, 1]$.

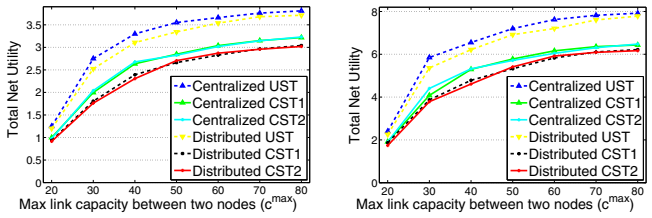
2) **Clustered Distribution of Social Ties 1 (CST1)**: We calculate device contact frequencies from traces in [18], normalize them to values within $(0, 1]$, and set social tie ρ_{id} between two nodes in the network following the distribution of normalized contact frequencies. From the traces, the social tie strength among most nodes is low, and only a few nodes have strong social ties with others, similar to a Pareto distribution and corresponding to *case I* in *Proposition 1*.

3) **Clustered Distribution of Social Ties 2 (CST2)**: We create a scenario where the social tie on each link in the social graph is $1 - \rho_{id}$, where ρ_{id} is the corresponding social tie value

in CST1. Therefore, most nodes have high average strength of social ties with others. It reflects *case II* in *Proposition 1*.

In all cases, nodes i and d without a direct link in the social graph are assigned a social tie $\rho_{id} = 0$.

A. Utility Optimality

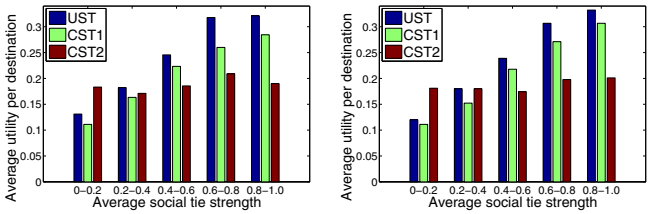


(a) a 50-node network (b) a 100-node network

Fig. 3. Centralized vs. distributed algorithm on total net utility.

We compare the total net utility achieved with Alg. 1, in cases that (24) is solved with the centralized decision in [8] and the distributed method in Alg. 2, respectively. The average end-to-end rates in utility calculation are computed after $t = 100,000$ rounds of execution. Fig. 3 shows that the total net utility achieved by the distributed algorithm is close to that of the centralized algorithm, under each social tie distribution.

B. Impact of Social Selfishness



(a) a 50-node network (b) a 100-node network

Fig. 4. Impact of different social tie strengths.

We calculate the average social tie strength for each destination node between itself and other nodes in the network, group the destinations based on the average tie strength, and compare the average utility per destination achieved among different groups in Fig. 4. Here $c^{max} = 60$. In cases of uniform distribution (UST) and clustered distribution of social ties case I (CST1), destinations with stronger social ties with relay nodes obtain larger utility. In case II of the clustered distribution of social ties (CST2), all destinations with different strengths of social ties turn out to achieve similar levels of utility. These results are consistent with the analysis in Sec.V.

VII. Conclusion

This paper investigates stochastic optimal multirate multicast in wireless networks with channel fading and node mobility, under the new constraint of socially selfish users. A joint end-to-end rate control, routing and capacity allocation algorithm, together with its distributed implementation, is proposed to achieve utility optimality with network stability guarantees, using novel combinations of Lyapunov optimization techniques, random linear network coding and multi-resolution coding. Social selfishness of users is novelly modeled as differentiated relay costs for forwarding traffic towards different destinations, which are decided by social tie strengths. Utility optimality and the impact of social selfishness are carefully studied with rigorous theoretical analysis and simulations. We have observed that multiple receiving rates are achieved at different destinations not only based on their bandwidth availability, but also depending on their social ties with relay nodes. Nevertheless, our algorithm can always find the close-to-optimal rates and routes that maximize the overall net utility to all destinations in the system. As future work, we will explore the optimal multirate multicast with QoS guarantees and a reduced number of queues on each node.

References

[1] P. A. Chou, Y. Wu, and K. Jain. Practical network coding. In *Proc. of IEEE Allerton'03*, 2003.

[2] S. Dumitrescu, M. Shao, and X. Wu. Layered multicast with interlayer network coding. In *Proc. of IEEE INFOCOM'09*, 2009.

[3] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong. A random linear network coding approach to multicast. *IEEE Transactions on Information Theory*, 52:4413 – 4430, 2006.

[4] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Growcroft, and C. Diot. Pocket switched networks and human mobility in conference environments. In *Proc. of ACM SIGCOMM'05 Workshops*, 2005.

[5] J. J. Jaramillo and R. Srikant. Darwin: Distributed and adaptive reputation mechanism for wireless ad-hoc networks. In *Proc. of ACM MOBICOM'07*, 2007.

[6] M. Kim, D. Lucani, X. shi, F. Zhao, and M. Médard. Network coding for multi-resolution multicast. In *Proc. of IEEE INFOCOM'10*, 2010.

[7] S. Lakshminarayana and A. Eryilmaz. Multi-rate multicasting with network coding. In *Proc. of IEEE WICON'08*, 2008.

[8] E. L. Lawler and D. E. Wood. Branch-and-bound methods: A survey. *Operations Research*, 14:699–719, 1966.

[9] B. Li and J. Liu. Multirate video multicast over the internet: An overview. *IEEE Network*, 17:24–29, 2003.

[10] H. Li, C. Wu, Z. Li, W. Huang, and F. C. M. Lau. Stochastic optimal multirate multicast in socially selfish wireless networks. Technical report, `http://i.cs.hku.hk/~hxli/socialmulticast.pdf`.

[11] Q. Li, S. Zhu, and G. Cao. Routing in Socially Selfish Delay Tolerant Networks. In *Proc. of IEEE INFOCOM'10*, 2010.

[12] W. Li. Overview of the fine granularity scalability in mpeg-4 video standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 11:301–317, 2001.

[13] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and Analysis of Online Social Networks. In *Proc. of IMC*, 2007.

[14] M. Motani, V. Srinivasan, and P. Nuggehalli. Peoplenet: Engineering a wirelessvirtual social network. In *Proc. of ACM MOBICOM'05*, 2005.

[15] M. J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan&Claypool Publishers, 2010.

[16] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the scalable video coding extension of the h.264/avc standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 17:1103 – 1120, 2007.

[17] M. Shao, X. Wu, and N. Sarshar. Rainbow network flow with network coding. In *Proc. of NetCod'08*, 2008.

[18] V. Srinivasan, A. Natarajan, and M. Motani. CRAWDAD data set nus/bluetooth (v. 2007-09-03). Downloaded from http://crawdad.cs.dartmouth.edu/nus/bluetooth, Sept. 2007.

[19] N. Sundaram, P. Ramanathan, and S. Banerjee. Multirate media stream using network coding. In *Proc. of IEEE Allerton'05*, 2005.

[20] X. Yan, M. J. Neely, and Z. Zhang. Multicasting in time-varying wireless networks: Cross-layer dynamic resource allocation. In *Proc. of IEEE ISIT'07*, 2007.

[21] S. Zhong, J. Chen, and Y. R. Yang. Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks. In *Proc. of IEEE INFOCOM'03*, 2003.