Reliability-based Rate Allocation in Wireless Inter-session Network Coding Systems

Keivan Ronasi, A. Hamed Mohsenian-Rad, Vincent W. S. Wong, Sathish Gopalakrishnan, and Robert Schober
Department of Electrical and Computer Engineering
The University of British Columbia, Vancouver, Canada
e-mail: {keivanr, hamed, vincentw, sathish, rschober}@ece.ubc.ca

***Abstract*— Network coding has recently received increasing attention to improve performance and increase capacity in both wired and wireless communication networks. In this paper, we focus on *inter-session* network coding, where multiple unicast sessions jointly participate in network coding. Wireless links are often *unreliable* because of varying channel conditions. We consider multi-hop unicast sessions over unreliable links and propose a distributed end-to-end transmission rate adjustment mechanism to maximize the *aggregate network utility* by taking into account the wireless link reliability information. This includes an elaborate modeling of *end-to-end reliability*. Simulation results show that by taking into account the reliability information, we can increase the network throughput by up to 100% for some network topologies. We can also increase the aggregate network utility significantly for various choices of utility functions.**

I. Introduction

Following the seminal paper by Ahlswede *et al.* [1], a rich body of work has focused on developing techniques to improve network performance using network coding. Network coding can be performed by *jointly* encoding multiple packets either from the *same* source or from *different* sources. The former is called *intra-session* network coding [1] while the latter is called *inter-session* network coding [2]. Network coding has been shown as a promising approach in communication networking, particularly for maximizing capacity in wireless networks [3]. Other network coding design objectives include maximizing network lifetime [4], minimizing energy consumption [5], and maximizing the network throughput [6].

Wireless networks are usually unreliable (i.e., data flows may experience significant packet losses) due to channel imperfection, noise, and interference. It is crucial to develop strategies to improve reliability in wireless networks. Ghaderi *et al.* [7] determined the reliability gain of intra-session network coding for multicast flows. In their work, the expected number of retransmissions per packet is used as the performance metric for reliability. Reliability in intra-session network coding is also studied by Lun *et al.* in [8]. Lee *et al.* [9] proposed a method for rate allocation with reliability considerations but *without* network coding.

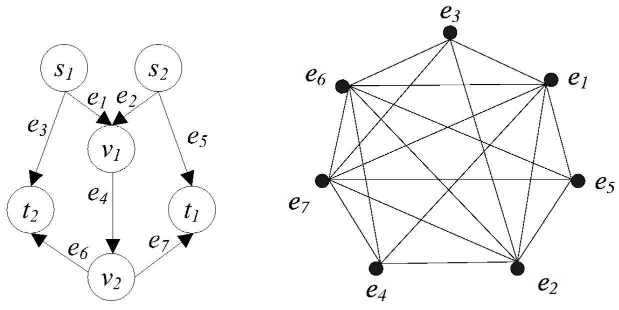
In this paper, we focus on inter-session network coding for multiple unicast sessions in a wireless network. Our objective is to increase the network utility and the *end-to-end reliability* of data transmissions (e.g., HTTP, FTP, P2P traffic) by proper allocation of routing and inter-session network coding rates for each data source in the network. For this purpose, we use the failure probability of intermediate links to calculate the reliability (i.e., the probability that data is successfully received) of various routing and network coding paths. Given the calculated reliability information, we maximize the effective aggregate network throughput by choosing the optimal rate allocation for network coding paths. We use the *network utility maximization* framework developed by Kelly *et al.* in [10]. To the best of our knowledge, there has been no prior work on improving the *end-to-end reliability* in an inter-session network coding system among unicast sessions.

The contributions of this paper are as follows:

- We develop a recursive algorithm to calculate end-to-end reliability, i.e., the probability of correctly delivering a packet, over each routing and network coding path. This allows us to mathematically model the effective throughput for each unicast session in the network.
- We formulate a network utility maximization problem for unreliable inter-session network coding systems. This problem formulation takes into account the network topology, mutual interference among wireless links, session utility functions, and link reliability information.
- We propose a distributed algorithm to solve the formulated network utility maximization problem using the *proximal method* and *gradient projection*.
- Simulation results show that by taking into account the reliability information, the aggregate network throughput can be increased by up to 100% while the aggregate network utility is also improved significantly.

Unlike intra-session network coding, there is no dominant coding scheme for inter-session network coding. Our inter-session network coding scheme is similar to the scheme by Khreishah *et al.* in [6] for wired networks. However, we extend that model in [6] to the *wireless* networking case by representing wireless capacity using the *contention graph*. Furthermore, we take reliability information into consideration. In this regard, the system model in [6] can be considered as a special case of our inter-session network coding model. Our proposed approach is based on cooperation among all network users. Game theoretic analysis of inter-session network coding with non-cooperative users is also studied in [11].

The rest of this paper is organized as follows. The system model is described in Section II. Our algorithm to calculate end-to-end reliability is presented in Section III. We solve the considered network utility maximization problem in Section IV using a distributed algorithm. Simulation results are presented in Section V. Conclusions are given in Section VI.



(a) (b)

Fig. 1. (a) An example directed acyclic network graph with 6 nodes and 7 links. One data session exists between s_1 and t_1 and another one exists between s_2 and t_2. (b) The corresponding (undirected) contention graph.

II. System Model

Consider a wireless network modeled as a directed acyclic graph $G(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of all nodes in the network and $\mathcal{E}$ is the set of all wireless links. We denote $e = (u, v) \in \mathcal{E}$ if and only if $d(u, v) \leq d_T$, where $d(u, v)$ is the Euclidean distance between nodes u and v, and d_T denotes the maximum transmission range. Let $\mathcal{S} = \{1, 2, \ldots, S\}$ denote the set of all unicast sessions in the network. Each session $i \in \mathcal{S}$ is denoted by a tuple (s_i, t_i), where s_i and t_i denote the source node and the destination node of session i, respectively.

A. Pair-wise Inter-session Network Coding

Following the inter-session network coding model in [2], we model the network graph $G(\mathcal{V}, \mathcal{E})$ as a *superposition* of S *routing subgraphs* $G_r^{(i)}(\mathcal{V}_r^{(i)}, \mathcal{E}_r^{(i)})$ for all $i \in \mathcal{S}$ and $S(S-1)$ *pairwise inter-session network coding subgraphs* $G_{nc}^{(i,j)}(\mathcal{V}_{nc}^{(i,j)}, \mathcal{E}_{nc}^{(i,j)})$ for all $i, j \in \mathcal{S}$ such that $i \neq j$. For each session $i \in \mathcal{S}$, routing subgraph $G_r^{(i)}$ includes all routing paths from source node s_i to destination node t_i. We define $\mathcal{P}_r^{(i)}$ as the set of *all* routing paths $P_{s_i t_i}$ in graph $G_r^{(i)}$, where $P_{s_i t_i}$ denotes a path from source s_i to destination t_i. We also define $N_r^{(i)} = |\mathcal{P}_r^{(i)}|$. Furthermore, for each $k = 1, \ldots, N_r^{(i)}$, we define $\epsilon_i^{ek} = 1$ if link e belongs to the k^{th} routing path of session i; otherwise, $\epsilon_i^{ek} = 0$.

For any two sessions $i, j \in \mathcal{S}$, we define $\mathcal{P}_{nc}^{(i,j)}$ as the set of *all* triples $\{P_{s_i t_i}, P_{s_j t_j}, P_{s_j t_i}\}$ in graph $G_{nc}^{(i,j)}$ such that *at most* two of the three paths $P_{s_i t_i}$, $P_{s_j t_j}$, and $P_{s_j t_i}$ share the same link. We define $N_{nc}^{(i,j)} = |\mathcal{P}_{nc}^{(i,j)}|$. For a pair of sessions i and j, we construct subgraph $G_{ij}^{lm}(\mathcal{V}_{ij}^{lm}, \mathcal{E}_{ij}^{lm})$ by the union of the l^{th} triple path in $\mathcal{P}_{nc}^{(i,j)}$ and the m^{th} triple path in $\mathcal{P}_{nc}^{(j,i)}$. The graphs G_{ij}^{lm} can be used to implement inter-session network coding (cf. [2], [6]) as we explain in the following example.

Consider the network in Fig. 1(a), where $\mathcal{V} = \{s_1, s_2, v_1, v_2, t_1, t_2\}$ and $\mathcal{E} = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$. This network topology is sometimes referred to as the *butterfly* network [2]–[6]. We can see that there is only one routing path from s_1 to t_1 in Fig. 1(a), denoted by $(e_1 e_4 e_7)$. Similarly, there is only one routing path from s_2 to t_2, denoted by $(e_2 e_4 e_6)$. Therefore, sets $P_r^{(1)}$ and $P_r^{(2)}$ have only one member and $N_r^{(1)} = N_r^{(2)} = 1$. On the other hand, set $\mathcal{P}_{nc}^{(1,2)}$ has one member, denoted by triple $\{e_1 e_4 e_7, e_2 e_4 e_6, e_5\}$. Similarly, set $\mathcal{P}_{nc}^{(2,1)}$ has one member, denoted by triple $\{e_2 e_4 e_6, e_1 e_4 e_7, e_3\}$. Therefore, $N_{nc}^{(1,2)} = N_{nc}^{(2,1)} = 1$. Furthermore, graph $G_{12}^{11} = G_{21}^{11} = G$. To implement network coding, node v_1 jointly encodes packets it receives from source nodes s_1 and s_2. The encoded packets are then transmitted to receiver nodes t_1 and t_2 via node v_2. Node t_1 can decode the received packets using the *remedy* data it receives from node s_2 over side link e_5. Similarly, node t_2 can decode the received packets using the remedy data it receives from node s_1 over side link e_3. Notice that encoding packets at node v_1 *reduces* the required bandwidth on links e_4, e_6 and e_7, leading to an increase in network throughput.

In a general network, the network coding scheme can be constructed by using the *add-up and reset* scheme [2]. Here, we assume that the network coding graphs are predetermined.

B. Rate Allocation

For each session $i \in \mathcal{S}$, let α_i^k denote the data rate of source s_i on the k^{th} routing path in subgraph $G_r^{(i)}$ for $k = 1, \ldots, N_r^{(i)}$. Also let α_{ij}^{lm} denote the rate of source s_i on network coding subgraph G_{ij}^{lm} for $l = 1, \ldots, N_{nc}^{(i,j)}$ and $m = 1, \ldots, N_{nc}^{(j,i)}$, where $j \in \mathcal{S} \backslash \{i\}$. Clearly, we must have $\alpha_{ij}^{lm} = \alpha_{ji}^{ml}$. The aggregate *sending* rate of source s_i is obtained as

$$\sum_{k=1}^{N_r^{(i)}} \alpha_i^k + \sum_{j \in \mathcal{S}, j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \alpha_{ij}^{lm}. \tag{1}$$

Notice that since the wireless links are prone to error, the effective *receiving* rate at the destination node t_i can be different from the above sending rate at source node s_i. We will investigate this issue in detail in Sections II-D and III.

Given the sending rates, we can also model the aggregate traffic load on each wireless link $e \in \mathcal{E}$ as [6]:

$$u_e = \sum_{i \in \mathcal{S}} \left(\sum_{k=1}^{N_r^{(i)}} \epsilon_i^{ek} \alpha_i^k + \sum_{j > i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \alpha_{ij}^{lm} \max\{H_{ij}^{el}, H_{ji}^{em}\} \right), \tag{2}$$

where $H_{ij}^{el} = 1$ if link e belongs to at least one path in the l^{th} triple $\{P_{s_i t_i}, P_{s_j t_j}, P_{s_j t_i}\}$ of set $\mathcal{P}_{nc}^{(i,j)}$; otherwise, $H_{ij}^{el} = 0$. For the network in Fig. 1(a), $u_{e_1} = u_{e_7} = \alpha_1^1 + \alpha_{12}^{11}$, $u_{e_2} = u_{e_6} = \alpha_2^1 + \alpha_{12}^{11}$, $u_{e_3} = u_{e_5} = \alpha_{12}^{11}$, and $u_{e_4} = \alpha_1^1 + \alpha_2^1 + \alpha_{12}^{11}$.

C. Interference

In a wireless network, where some of the links can interfere with each other, mutual interference can be modeled using a *contention graph* $G'(\mathcal{V}', \mathcal{E}')$. In a contention graph G', the set of vertices $\mathcal{V}'$ represents the set of all wireless links $\mathcal{E}$ in the network graph G. There exists an edge between any two vertices in set $\mathcal{V}'$ if the wireless links corresponding to the two vertices mutually interfere with each other. That is, if the receiver of one link is within the interference range of the sender of the other link. Given the contention graph, each *complete* subgraph, i.e., a subgraph in which all vertices are connected, is called a *clique*. A *maximal clique* is defined as a clique which is *not* a subgraph of any other clique. We denote the set of all maximal cliques in contention graph G' by $\mathcal{Q}$. Each maximal clique is denoted by $Q_n \in \mathcal{Q}$ for $n = 1, \ldots, |\mathcal{Q}|$. Only one link among all the links corresponding to the vertices of a maximal clique can be active at a time.

Let c_e denote the *nominal* data rate of link $e \in \mathcal{E}$. The ratio $\frac{u_e}{c_e}$ denotes the portion of time that each link $e \in \mathcal{E}$ would be active. It is required that

$$\sum_{e \in Q_n} \frac{u_e}{c_e} \leq \gamma, \qquad \forall Q_n \in \mathcal{Q}, \tag{3}$$

where $\gamma \in (0,1]$ is called the clique capacity. It is common practice to select $\gamma = \frac{2}{3}$ [12]. For the network in Fig. 1(a), the corresponding contention graph is shown in Fig. 1(b). We can see that the contention graph includes three maximal cliques. They impose three constraints. For instance, clique $Q_1 = \{e_1, e_2, e_4, e_6, e_7\}$ requires that $\frac{u_1}{c_1} + \frac{u_2}{c_2} + \frac{u_4}{c_4} + \frac{u_6}{c_6} + \frac{u_7}{c_7} \leq \gamma$.

D. Link Failure

In practice, since the wireless channel between any two neighboring nodes u and v is *not* perfect due to environmental obstacles and background noise, each link $e = (u, v)$ may have a probability of failure $p_e \in [0, 1]$ with which *packets* sent by node u are corrupted and *not* received by receiver node v correctly. We model the wireless channels as binary erasure channels (BEC) [13, p. 187] and assume that data packets transmitted on link e are received successfully at the receiver node with probability $1 - p_e$. The *failure probability vector* of all links in the network $\mathbf{p} = (p_e,\ e \in \mathcal{E})$ is assumed to be known through measurements (e.g., by probe transmissions).

Given the link failure probability vector $\mathbf{p}$, we can further obtain the *end-to-end* failure probability for each routed or network coded packet. For each session $i \in \mathcal{S}$, let P_i^k denote the end-to-end reliability (i.e., 1 minus failure probability) for a packet that is *routed* over the k^{th} routing path, where $k = 1, \ldots, N_r^{(i)}$. Furthermore, for each pair of sessions $i, j \in \mathcal{S}$, let P_{ij}^{lm} denote the end-to-end reliability for a *network coded* packet that is transmitted over network coding subgraph G_{ij}^{lm}. The aggregate *receiving* rate at destination t_i is obtained as

$$\sum_{k=1}^{N_r^{(i)}} \alpha_i^k P_i^k + \sum_{j \in \mathcal{S}, j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \alpha_{ij}^{lm} P_{ij}^{lm}. \tag{4}$$

We will discuss in detail how the end-to-end reliability can be calculated in Section III.

E. Network Utility Maximization Formulation

Let us concatenate all sending rates α_i^k and α_{ij}^{lm} for all $i, j \in \mathcal{S}$, all $k = 1, \ldots, N_r^{(i)}$, all $l = 1, \ldots, N_{nc}^{(i,j)}$, and all $m = 1, \ldots, N_{nc}^{(j,i)}$ and denote the resulting vector as $\boldsymbol{\alpha}$. The network utility maximization problem for unreliable wireless networks with inter-session network coding among multiple unicast sessions can be formulated as

$$\begin{aligned} \max_{\boldsymbol{\alpha} \succeq 0} \ & \sum_{i \in \mathcal{S}} U_i \left(\sum_{k=1}^{N_r^{(i)}} \alpha_i^k P_i^k + \sum_{j \in \mathcal{S}, j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \alpha_{ij}^{lm} P_{ij}^{lm} \right) \\ \text{s. t.} \ & \sum_{e \in Q_n} \frac{u_e}{c_e} \leq \gamma, \ \forall Q_n \in \mathcal{Q} \\ & \alpha_{ij}^{lm} = \alpha_{ji}^{ml}, \quad \begin{array}{l} \forall i, j \in \mathcal{S}, i \neq j \\ l = 1, \ldots, N_{nc}^{(i,j)}, \ m = 1, \ldots, N_{nc}^{(j,i)}, \end{array} \end{aligned} \tag{5}$$

where $\succeq$ denotes coordinate-wise inequality, u_e for each $e \in \mathcal{E}$ is as in (2), and for each session $i \in \mathcal{S}$, $U_i(\cdot)$ denotes a *strictly concave* and *increasing* utility function. For example, utility functions can be logarithmic. In that case, the utility maximization problem (5) becomes a *proportionally fair* resource allocation problem. Also notice that if $U_i(x) = x$, then problem (5) reduces to a *throughput maximization* problem.

III. End-to-End Reliability

Recall from Section II-D that the aggregate receiving rate of data at receiver node t_i for each $i \in \mathcal{S}$ is as in (4), where P_i^k and P_{ij}^{lm} are end-to-end reliabilities for all routing and network coding paths for session i. In this section, we develop an algorithm to obtain these end-to-end reliability measures.

For each session $i \in \mathcal{S}$, consider the k^{th} routing path in graph $G_r^{(i)}$ for $k = 1, \ldots, N_r^{(i)}$. The probability that data is transmitted successfully along this path can be obtained as

$$P_i^k = \prod_{e \in \mathcal{E}_r} \epsilon_i^{ek} (1 - p_e). \tag{6}$$

For the network in Fig. 1(a), we have $P_1^1 = (1 - p_{e_1})(1 - p_{e_4})(1 - p_{e_7})$ and $P_2^1 = (1 - p_{e_2})(1 - p_{e_4})(1 - p_{e_6})$.

Obtaining the end-to-end reliability of pairwise inter-session network coding paths is more complex due to the overlapping among different paths and the fact that an encoded packet is successfully received only if the corresponding remedy data is also received successfully. To explain our model, let us consider the example network in Fig. 1(a). Node t_1 can successfully receive some data in a network coding setting if and only if *all* of the following three conditions hold: (a) Intermediate node v_1 successfully receives the data packets from both source nodes s_1 and s_2 over links e_1 and e_2, respectively. This happens with probability $(1 - p_{e_1})(1 - p_{e_2})$. (b) The encoded packet is successfully received by node t_1 over links e_4 and e_7. This happens with probability $(1 - p_{e_4})(1 - p_{e_7})$. (c) The remedy packet, corresponding to the data packet, is successfully received by node t_1 over link e_5. This happens with probability $(1 - p_{e_5})$. Therefore, we have

$$P_{12}^{11} = (1 - p_{e_5})(1 - p_{e_4})(1 - p_{e_7})(1 - p_{e_1})(1 - p_{e_2}). \tag{7}$$

Similarly, we can show that

$$P_{21}^{11} = (1 - p_{e_3})(1 - p_{e_4})(1 - p_{e_6})(1 - p_{e_1})(1 - p_{e_2}). \tag{8}$$

To generalize the idea to an arbitrary network, recall that for each pair $i, j \in \mathcal{S}$, any $m = 1, \ldots, N_{nc}^{(i,j)}$, and any $l = 1, \ldots, N_{nc}^{(j,i)}$, inter-session network coding subgraph G_{ij}^{lm} is constructed as the union of the m^{th} triple path in $\mathcal{P}_{nc}^{(i,j)}$ and the l^{th} triple path in $\mathcal{P}_{nc}^{(j,i)}$. Given G_{ij}^{lm}, let φ_v denote the probability that node $v \in \mathcal{V}_{ij}^{lm}$ receives an original/encoded/remedy packet *correctly*. For the simplicity of exposition, we define $\varphi_{s_i} = \varphi_{s_j} = 1$, since the source nodes s_i and s_j have the correct data with probability one. For the network in Fig. 1(a), $\varphi_{v_1} = (1 - p_{e_1}) \times \varphi_{s_1}(1 - p_{e_2})\varphi_{s_2}$, $\varphi_{v_2} = (1 - p_{e_4})\varphi_{v_1}$, and $\varphi_{t_1} = (1 - p_{e_7})\varphi_{v_2} \times (1 - p_{e_5})\varphi_{s_2}$. We can see that there is a *recursive* relationship on failure probabilities of neighboring nodes. This observation motivates our *end-to-end reliability calculation for inter-session network coding* algorithm in Algorithm 1. For each node $v \in \mathcal{V}_{ij}^{lm}$, let $in(v)$ denote the set of in-neighbors (i.e., neighbors with incoming links) of node v in graph G_{ij}^{lm}. Notice that $in(v) \subseteq \mathcal{V}_{ij}^{lm} \backslash \{v\}$. For each node $u \in in(v)$, the directed wireless link from node u to node v is denoted by $e = (u, v)$. In this case,

$$\varphi_v = \prod_{u \in in(v)} (1 - p_{e=(u,v)}) \varphi_u. \tag{9}$$

Algorithm 1 can be used to obtain end-to-end reliability for *any* arbitrary inter-session network coding subgraph.

Algorithm 1 End-to-end reliability calculation for inter-session network coding between sessions $i \in \mathcal{S}$ and $j \in \mathcal{S} \backslash \{i\}$, executed for any $l = 1, \ldots, N_{nc}^{(i,j)}$ and $m = 1, \ldots, N_{nc}^{(j,i)}$.

```
Set φ_{s_i} = φ_{s_j} = 1, φ_v = −1 for all v ∈ V_{ij}^{lm} \ {s_i, s_j}.
while φ_{t_i} = −1 do
    Find node v ∈ V_{ij}^{lm} \ {s_i, s_j} such that φ_v = −1 and
    φ_u ≠ −1 for all neighboring nodes u ∈ in(v).
    Set φ_v = ∏_{u∈in(v)} (1 − p_{e=(u,v)}) φ_u.
end while
P_{ij}^{lm} = φ_{t_i}
```

IV. Reliability-based Rate Allocation

In this section, we propose a distributed rate allocation algorithm to solve the network utility maximization problem in (5). For notational simplicity, we define $F_{ij}^{elm} = \frac{1}{2}\max\{H_{ij}^{el}, H_{ji}^{em}\}$. In that case, for each clique $Q_n \in \mathcal{Q}$, where $n = 1, \ldots, |\mathcal{Q}|$, the clique constraint in (3) becomes

$$\sum_{e \in Q_n} \sum_{i \in \mathcal{S}} \left(\sum_{k=1}^{N_r^{(i)}} \frac{\epsilon_i^{ek}}{c_e} \alpha_i^k + \sum_{j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \frac{F_{ij}^{elm}}{c_e} \alpha_{ij}^{lm} \right) \leq \gamma. \tag{10}$$

Next, we notice that although the objective function in problem (5) is concave, it is not *strictly* concave due to the linear terms inside each utility function. Thus, problem (5) may have *multiple* optimal solutions. This can pose some difficulties if we require a *distributed* scheme to solve the optimization problem. To overcome this problem, we use the *proximal method* [14]: we add some extra terms to the objective function to make it *strictly* concave. Then, problem (5) becomes

$$\begin{aligned} \max_{\boldsymbol{\alpha}, \boldsymbol{\beta} \succeq \mathbf{0}} \quad & \sum_{i \in \mathcal{S}} U_i \left(\sum_{k=1}^{N_r^{(i)}} \alpha_i^k P_i^k + \sum_{j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \alpha_{ij}^{lm} P_{ij}^{lm} \right) \\ & - \sum_{i \in \mathcal{S}} \sum_{k=1}^{N_r^{(i)}} \frac{a_i}{2} (\alpha_i^k - \beta_i^k)^2 \\ & - \sum_{i \in \mathcal{S}} \sum_{j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \frac{b_i}{2} (\alpha_{ij}^{lm} - \beta_{ij}^{lm})^2 \\ \text{s.t.} \quad & \sum_{e \in Q_n} \sum_{i \in \mathcal{S}} \left(\sum_{k=1}^{N_r^{(i)}} \frac{\epsilon_i^{ek}}{c_e} \alpha_i^k \right. \\ & \left. + \sum_{j > i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \frac{F_{ij}^{elm}}{c_e} \alpha_{ij}^{lm} \right) \leq \gamma, \quad \forall Q_n \in \mathcal{Q} \\ & \alpha_{ij}^{lm} = \alpha_{ji}^{ml}, \quad \begin{array}{l} \forall i, j \in \mathcal{S}, i \neq j \\ l = 1, \ldots, N_{nc}^{(i,j)}, \ m = 1, \ldots, N_{nc}^{(j,i)}, \end{array} \end{aligned} \tag{11}$$

where β_i^k and β_{ij}^{lm} are *auxiliary* variables introduced for all $i, j \in \mathcal{S}$, $l = 1, \ldots, N_{nc}^{(i,j)}$, and any $m = 1, \ldots, N_{nc}^{(j,i)}$. On the other hand, a_i and b_i are arbitrary positive coefficients. Notice that if $\beta_i^k = \alpha_i^k$ and $\beta_{ij}^{lm} = \alpha_{ij}^{lm}$, then the objective function in problem (11) reduces to the original objective function in problem (5). For notational simplicity, we concatenate all β_i^k and β_{ij}^{lm} for all $i, j \in \mathcal{S}$, $k = 1, \ldots, N_r^{(i)}$, $l = 1, \ldots, N_{nc}^{(i,j)}$, and $m = 1, \ldots, N_{nc}^{(j,i)}$, and denote the resulting vector by $\boldsymbol{\beta}$.

To solve the *modified* problem (11) via a *distributed* scheme, we first obtain the dual Lagrangian function [15]

$$L(\boldsymbol{\alpha}, \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\beta}) = \sum_{i \in \mathcal{S}} B_i(\boldsymbol{\alpha}, \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\beta}) + \sum_{Q_n \in \mathcal{Q}} \lambda_n \gamma, \tag{12}$$

where $B_i(\boldsymbol{\alpha}, \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\beta})$

$$\begin{aligned} = \ & U_i \left(\sum_{k=1}^{N_r^{(i)}} \alpha_i^k P_i^k + \sum_{j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \alpha_{ij}^{lm} P_{ij}^{lm} \right) \\ & - \frac{a_i}{2} \sum_{k=1}^{N_r^{(i)}} (\alpha_i^k - \beta_i^k)^2 - \frac{b_i}{2} \sum_{j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} (\alpha_{ij}^{lm} - \beta_{ij}^{lm})^2 \\ & - \sum_{Q_n \in \mathcal{Q}} \sum_{e \in Q_n} \sum_{k=1}^{N_r^{(i)}} \frac{\epsilon_i^{ek}}{c_e} \alpha_i^k \lambda_n \\ & - \sum_{Q_n \in \mathcal{Q}} \sum_{e \in Q_n} \sum_{j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \alpha_{ij}^{lm} \frac{F_{ij}^{elm}}{c_e} \lambda_n \\ & - \sum_{j > i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \nu_{ij}^{lm} \alpha_{ij}^{lm} + \sum_{j < i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \nu_{ji}^{ml} \alpha_{ij}^{lm}, \end{aligned}$$

and $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ are vectors of Lagrange multipliers corresponding to clique capacity constraints and equality constraints, respectively. Notice that for each clique constraint $\sum_{e \in Q_n} \frac{u_e}{c_e} \leq \gamma$, where $n = 1, \ldots, |\mathcal{Q}|$, the corresponding Lagrange multiplier is denoted by λ_n. For each equality constraint $\alpha_{ij}^{lm} = \alpha_{ji}^{ml}$, where $i, j \in \mathcal{S}$, $l = 1, \ldots, N_{nc}^{(i,j)}$, and $m = 1, \ldots, N_{nc}^{(j,i)}$, the corresponding Lagrange multiplier is denoted by ν_{ij}^{lm}. The dual problem of the primal problem in (11) can be obtained as

$$\min_{\boldsymbol{\lambda} \succeq \mathbf{0}, \ \boldsymbol{\nu}} \Gamma(\boldsymbol{\lambda}, \boldsymbol{\nu}), \tag{13}$$

where $\Gamma(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \max_{\boldsymbol{\alpha} \succeq 0} L(\boldsymbol{\alpha}, \boldsymbol{\lambda}, \boldsymbol{\nu}, \boldsymbol{\beta})$. In dual problem (13), the variables are the Lagrange multipliers. Problem (13) can be solved using the *gradient projection method* [15]. Notice that since primal problem (11) is convex and its constraints are linear, *strong duality* holds [15, p. 226]. Thus, by solving the dual problem (13), the optimal solution of the primal problem (11) is readily obtained. However, we want to solve the original network utility maximization problem (5). We now explain how the solution to the problem (5) can be obtained.

Our proposed distributed algorithm to identify the optimal data rates includes *two* sub-algorithms which are executed *iteratively* and *alternatively*. The first iterative sub-algorithm is based on the *gradient method* which is executed in *shorter* intervals. This sub-algorithm is used to update the *dual* variables $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ and the *primal* variables $\boldsymbol{\alpha}$. On the other hand, the second iterative sub-algorithm is based on the *proximal method* which is executed in *larger* intervals. This sub-algorithm is used to update the *auxiliary* variables $\boldsymbol{\beta}$. In our iterative distributed algorithm, the first sub-algorithm forms the *inner loop*, while the second sub-algorithm forms the *outer loop*.

1) First Sub-algorithm: At each time $t = 1, \ldots, T$, where T denotes the network operation time, source node s_i for each session $i \in \mathcal{S}$ updates its data rates individually as

$$\boldsymbol{\alpha}_i(t+1) = \arg\max_{\boldsymbol{\alpha}_i \succeq \mathbf{0}} \ B_i(\boldsymbol{\alpha}_i, \boldsymbol{\lambda}(t), \boldsymbol{\nu}(t), \boldsymbol{\beta}_i(t)) \tag{14}$$

where $\boldsymbol{\alpha}_i$ denotes the vector of all transmission rates of session i. Given the new data rates, then for each n such that $Q_n \in \mathcal{Q}$, for $n = 1, \ldots, |\mathcal{Q}|$, *dual* variable λ_n is updated as

$$\lambda_n(t+1) = \left[\lambda_n(t) + \delta_n \left(\sum_{i \in \mathcal{S}} \sum_{e \in Q_n} \left(\sum_{k=1}^{N_r^{(i)}} \frac{\epsilon_i^{ek}}{c_e} \alpha_i^k(t+1) + \sum_{j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} \alpha_{ij}^{lm}(t+1) \frac{F_{ij}^{elm}}{c_e}\right) - \gamma\right)\right]^+, \tag{15}$$

where δ_n is a small constant step size. Furthermore, for each $i, j \in \mathcal{S}$, such that $j > i$, and for any $l = 1, \ldots, N_{nc}^{(i,j)}$ and any $m = 1, \ldots, N_{nc}^{(j,i)}$, the *dual* variable ν_{ij}^{lm} is updated as

$$\nu_{ij}^{lm}(t+1) = \nu_{ij}^{lm}(t) + \sigma_{ij}^{lm}(\alpha_{ij}^{lm}(t+1) - \alpha_{ji}^{ml}(t+1)), \tag{16}$$

where σ_{ij}^{lm} is a small constant step size. Notice that the update equations in (15) and (16) are based on applying the gradient method to convex problem (13). The convergence of (14) in our first sub-algorithm follows from the *descent lemma* [16, p. 639]. In particular, we can show that the first sub-algorithm converges if the following *sufficient* condition holds:

$$\left[\sum_{n: Q_n \in \mathcal{Q}} \sum_{e \in Q_n} \left(\sum_{i \in \mathcal{S}} \sum_{k=1}^{N_r^{(i)}} \epsilon_i^{ek} + \sum_{i \in \mathcal{S}} \sum_{j \neq i} \sum_{l=1}^{N_{nc}^{(i,j)}} \sum_{m=1}^{N_{nc}^{(j,i)}} (F_{ij}^{elm})^2\right)\right] \max_{n: Q_n \in \mathcal{Q}} \delta_n + \max_{i,j>i,l,m} \sigma_{ij}^{lm} < 2 \min\{a_1, \ldots, a_S, b_1, \ldots b_S\}. \tag{17}$$

The key idea is to show that our proposed (joint) distributed algorithm forms a *pseudo-contraction mapping* [16, p. 182]. Details are omitted here for brevity. We notice that for *any* arbitrary network topology and *any* arbitrary choice of system parameters, we can *always* select step sizes δ_n and σ_{ij}^{lm} small enough such that the strict inequality in (17) holds.

2) Second Sub-algorithm: At larger intervals, i.e., large enough such that the first sub-algorithm converges within each interval, the second sub-algorithm simply sets

$$\boldsymbol{\beta}(t+1) = \boldsymbol{\alpha}(t). \tag{18}$$

After that, the first sub-algorithm is triggered again based on the new values of $\boldsymbol{\beta}$. We continue alternate operation of the two sub-algorithms until the joint algorithm converges. The convergence is always guaranteed [16, p.232].

Next, we show that the joint algorithm formed by update equations (14)-(18) results in *optimal* rate allocation. In this regard, we notice that after convergence of the algorithm, we would have $\boldsymbol{\beta} = \boldsymbol{\alpha}$. In that case, the objective value in problem (11) reduces to the objective value in problem (5) as all the additional terms will become zero. Thus, the obtained data rates, which form the optimal solution to the primal problem (11), are also the optimal data rates with respect to the original network utility maximization problem (5). Similar to some other distributed algorithms to solve a convex problem via decomposition, our approach also needs message passing among different nodes.

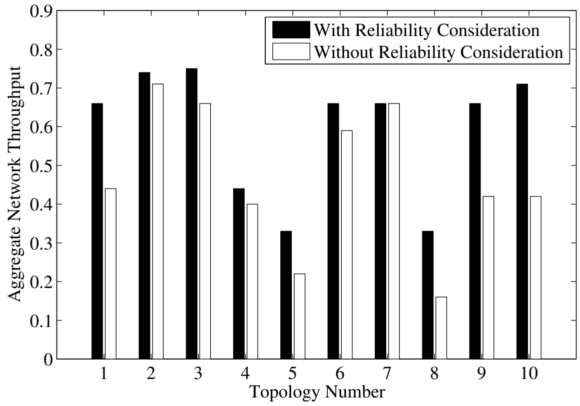


Fig. 2. Aggregate network throughput in all 10 simulated topologies *with* and *without* reliability consideration when the utility functions are selected as $U_i(x) = x$ for all $i \in \mathcal{S}$. Taking reliability information into account increases the throughput by 36.2% on average, among all ten scenarios.

V. Performance Evaluation

In this section, we assess the performance of our proposed reliability-based rate allocation algorithm and compare it with the rate allocation algorithm in [6] which does *not* consider link reliability. In particular, we evaluate how the performance improves if we take into account reliability information. In this regard, we simulate ten different randomly generated wireless topologies. Each topology is randomly selected to include between 10 to 15 wireless nodes.

In the first experiment, we assume that the utility functions are selected to maximize the network throughput. That is, $U_i(x) = x, \; \forall \; i \in \mathcal{S}$. In each topology, one or two links are selected randomly as *unreliable links*. The failure probability of unreliable links is chosen to be 0.5. This implies that *half* of the packets transmitted over the unreliable links experience transmission errors and are not received correctly. Simulation results are shown in Fig. 2. We can see that reliability considerations can significantly increase the network throughput. Notice that the exact performance gain differs among the topologies. This particularly depends on which link is selected as unreliable link in each topology. For example, the performance is improved by more than 100% for the case of topology 8 while the performance gain is negligible for topology 7. For the latter case, the low performance gain is due to the fact that the selected unreliable link does *not* carry any significant traffic load in an optimal design, even if it is *assumed* to be reliable. Therefore, its unreliability does not affect the network performance significantly. On average, among all 10 simulated topologies, accounting for reliability information increases the throughput by 36.2%.

Next, we study the impact of changes in the link failure probability. We assume that the failure probability of the unreliable links vary from 0 to 0.5. The former case indicates having reliable links, while the latter case indicates having unreliable links which lose *half* of the packets. Regarding the choice of utility functions, we consider the two important cases of maximizing the throughput and achieving proportional fairness. For the latter case, the utility functions are selected to be *logarithmic*. Simulation results for the case of topology number 1 are shown in Fig. 3. Results for other topologies are

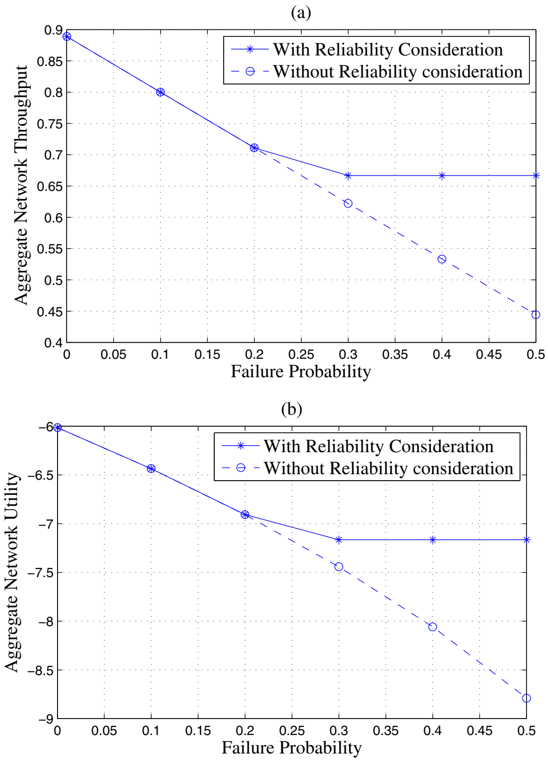


Fig. 3. Comparison between our algorithm (with reliability consideration) and the rate allocation algorithm in [6] which does *not* use link reliability information: (a) Maximizing the aggregate network throughput. (b) Maximizing the aggregate network utility when the utility functions are logarithmic.

similar. As the link failure probability increases, it becomes more important to take into account the reliability information. From Fig. 3, if the link failure probability is 0.5, then reliability consideration can increase the throughput and network utility by 50% and 18.4%, respectively.

Finally, we assess the impact of the choice of utility function on the achieved performance gain. To this end, we consider the utility function $U_i(x)$ to be equal to $\frac{x^{1-\psi}}{1-\psi}$ if $\psi \neq 1$, and $\log x$ if $\psi = 1$, where $\psi > 0$ is a utility parameter. Simulation results for the case of the first topology when ψ varies from 0 to 10, and the link failure probability is chosen to be 0.5, are shown in Fig. 4. Results for the remaining topologies are similar. We can see that reliability consideration always improves the performance for *any* choice of utility parameter ψ.

VI. Conclusions

In this paper, we considered the problem of allocating data rates for all sources in a wireless inter-session network coding system. We focused on a scenario where some of the wireless links are *not* reliable, i.e., have *non-zero* failure probability. Using a simple algorithm, we calculated the *end-to-end* reliability measures of all network coding paths in the network. We then formulated a network utility maximization problem, where the objective function is the sum of the utility functions of all data sessions. We also proposed a distributed iterative algorithm to solve the formulated optimization problem. Simulation results show that it is essential to consider link reliability to achieve high throughput. In our evaluations, the network throughput increased by 36.2% on average with up to 100% performance gain for some scenarios. The proposed

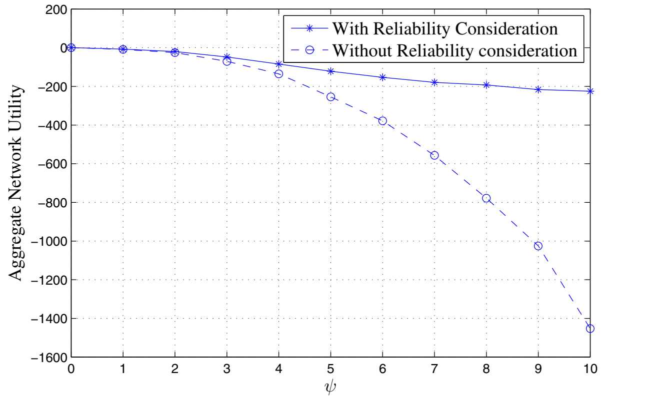


Fig. 4. The impact of changing the utility parameter ψ on the achieved aggregate network utility *with* and *without* reliability consideration.

scheme also significantly improves aggregate network utility for various choices of utility functions.

Acknowledgement

This research is supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada under grant number STPSC 356767-07.

References

[1] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. on Information Theory*, vol. 46, pp. 1204–1216, July 2000.
[2] C. C. Wang and N. B. Shroff, "Beyond the butterfly - A graph-theoretic characterization of the feasibility of network coding with two simple unicast sessions," in *Proc. of IEEE ISIT*, Nice, France, June 2007.
[3] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: practical wireless network coding," in *Proc. of ACM SIGCOMM*, Pisa, Italy, Sept. 2006.
[4] V. Shah-Mansouri and V. Wong, "Maximum-lifetime coding subgraph for multicast traffic in wireless sensor networks," in *Proc. of IEEE Globecom*, New Orleans, LA, Dec. 2008.
[5] T. Cui, L. Chen, and T. Ho, "Energy efficient opportunistic network coding for wireless networks," in *Proc. of IEEE Infocom*, Phoenix, AZ, Apr. 2008.
[6] A. Khreishah, C. C. Wang, and N. B. Shroff, "Optimization based rate control for communication networks with inter-session network coding," in *Proc. of IEEE Infocom*, Phoenix, AZ, Apr. 2008.
[7] M. Ghaderi, D. Towsley, and J. Kurose, "Reliability gain of network coding in lossy wireless networks," in *Proc. of IEEE Infocom*, Phoenix, AZ, Apr. 2008.
[8] D. Lun, N. Ratnakar, M. Medard, R. Koetter, D. Karger, T. Ho, E. Ahmed, and F. Zhao, "Minimum-cost multicast over coded packet networks," *IEEE Trans. on Information Theory*, vol. 52, pp. 2608–2623, June 2006.
[9] J. Lee, M. Chiang, and A. Calderbank, "Price-based distributed algorithms for rate-reliability based tradeoff in network utility maximization," *IEEE J. Select. Areas Commun.*, vol. 24, pp. 962–976, May 2006.
[10] F. P. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *Journal of Operations Research Society*, vol. 49, pp. 237–252, Mar. 1998.
[11] A. H. Mohsenian-Rad, J. Huang, V. W. S. Wong, S. Jaggi, and R. Schober, "Game-theoretical analysis of inter-session network coding," in *Proc. of IEEE ICC*, Dresden, Germany, June 2009.
[12] C. Shannon, "A theorem on coloring the lines in the network," *J. Math. Phys.*, vol. 28, pp. 148–151, Sept. 1949.
[13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 1999.
[14] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.
[15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
[16] D. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.