Minimum Cost Multiple Multicast Network Coding with Quantized Rates

M. A. Raayatpanah, H. Salehi Fathabadi, B. H. Khalaj and S. Khodayifar

1School of Mathematics, Statistics and Computer Science, University of Tehran, Tehran, Iran.
2Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran.

Abstract

In this paper, we consider multiple multicast sessions with intra-session network coding where rates over all links are integer multiples of a basic rate. Although having quantized rates over communication links is quite common, conventional minimum cost network coding problem cannot generally result in quantized solutions. In this research, the problem of finding minimum cost transmission for multiple multicast sessions with network coding is addressed. It is assumed that the rate of coded packet injection at every link of each session takes quantized values. First, this problem is formulated as a mixed integer linear programming problem, and then it is proved that this problem is strongly NP-hard on general graphs. In order to obtain an exact solution for the problem, an effective and efficient scheme based on Benders decomposition is developed. Using this scheme the problem is decomposed into a master integer programming problem and several linear programming sub-problems. The efficiency of the proposed scheme is subsequently evaluated by numerical results on random networks.

Keywords: Network Coding, Multicast Networks, Decomposition Algorithm.

1 Introduction

In communication networks, routing has long been an important technique for optimizing data transmissions from sources to destinations. In conventional optimal routing problems, each node is capable of forwarding the incoming packets without making any changes in them. Network coding generalizes traditional routing paradigm by allowing nodes to perform arbitrary operations on packets received and generate new output packets. The idea of network coding has its origin in the work of Ahlswede et al. [1], where it was shown that

*Corresponding author: Mohammad Ali Raayatpanah, E-mail address:raayatpanah@Khayam.ut.ac.ir; Tel.:+989132627161
network coding is capable of achieving the maximum-flow minimum-cut bound on the multicast capacity. Li et al. [2] showed that linear network codes are sufficient for the multicast capacity and Jaggi et al. [3] presented a polynomial-time algorithm for constructing such linear codes. The problem of finding a minimum cost multicast scheme in networks using a coded packet approach was addressed by Lun et al. [4]. It was shown that the solution of this problem can be decomposed into two parts: i) finding the minimum cost subgraph, and ii) determining a code to use over the optimal subgraph. A distributed solution for the second part was provided in [5]. To solve the first part, Lun et al. [4, 6] proposed a linear optimization formulation and presented a distributed algorithm using the dual subgradient method to obtain an optimal subgraph. Ghasvari et al. [7] considered the problem of finding a minimum cost multicast subgraph based on network coding, where delay values associated with each link, limited buffer-size of intermediate nodes and link capacity variations over time were taken into account. They proposed a decentralised algorithm using an auxiliary time-expanded network. Wu et al. [8] worked on utility maximization problems, where they developed a primal-subgradient type distributed algorithm based on finding the critical cut. In a similar fashion, there are other algorithms with the aim of finding flow subgraphs that guarantee a minimum rate and maximize a utility function [9, 10]. However, the algorithms proposed so far in the literature cannot derive quantized (multiples of a basic rate $\mu$) rates over communication links. Koetter et al. [11] pointed out that fractional rates can be well approximated by choosing the time units large enough. But, such capacity scaling will increase the encoding and decoding complexity and introduce large delays at receivers. The delay at receivers will limit the application of network coding in delay sensitive or computational resources limited network applications. Thus, it is necessary that the rates on links to be constrained in order to take quantized values. Cui et al. [12] presented a method for finding a minimum cost subgraph in a single multicast connection with network coding on directed graphs, where packet transmission rates on each link are constrained to be integral. In fact, link rates are constrained to integer multiples of the basic rate, $\mu = 1$. They proposed a greedy algorithm and an LP rounding algorithm, which have approximation ratios $k$ and $2k$, respectively, where $k$ is the number of receivers. In general communication networks, it is not necessary that link rates be integer values, but they generally can only accept quantized values. Moreover, in many applications, multiple multicast sessions governed by their own network constraints share the same network. As the flows do share common links, the single multicast session problems are not independent. In order to find an optimal flow, it is therefore needed to solve the problems in conjunction with each other. Hence, the problem of specifying whether or not a set of multicast connections is feasible becomes more complicated than an equivalent problem having only a single multicast connection. As a result, more effective techniques should be devised to examine the feasibility of a multicast connections set.

In this paper, we consider multiple multicast sessions with intra-session network coding and assume that, instead of one source process at a single node, $s$, there are $M$ single-rate multicast source processes at nodes $s_1, s_2, \ldots, s_M$, where all receivers in a session, $m$, receive services at the same rate, $R_m$. Furthermore, we assume that link rates are constrained
to be integer multiples of a positive basic rate $\mu$. Here, the aim is to find minimum cost solutions for the case of general connections, where the rate of the coded packet transmission on each link of every multicast connection is constrained to quantized values. To do so, we first formulate this problem as a mixed integer linear programming (MILP) problem and show that this problem is strongly NP-hard on general graphs, even on series-parallel graphs. Then, we develop a solution based on the Benders decomposition for this problem and implement the proposed approach. The Benders decomposition is an efficient method for solving large-scale MILP problems. Instead of considering all decision variables and constraints of a large-scale problem simultaneously, this approach partitions the original problem into a master integer programming problem and several linear programming sub-problems [13, 14]. Since the computational difficulty of MILP optimization problems significantly increases as the number of variables and constraints rises, solving smaller problems with less variables and constraints iteratively can be more efficient than solving a single large problem. Each iteration of the Benders decomposition algorithm acquires a lower bound for the objective value of the original problem. Hence we describe a heuristic procedure to obtain a tight bound in a reasonable computational time. The ability of the proposed approach in reaching the optimal solution is shown through simulations on random graphs.

The remaining of this paper is structured as follows: Section 2 presents a mixed-integer formulation of the proposed problem and its complexity. In section 3, the proposed problem is solved using the Benders decomposition algorithm. Section 4 introduces valid inequalities in order to reduce the number of iterations in the Benders decomposition algorithm. In Section 5, a heuristic method is used to obtain an upper bound on the optimal value of the objective function. Sections 6 and 7 are allocated to the computational results and conclusion, respectively.

## 2 Problem Definition and Complexity

### 2.1 Network Model and Notations

A communication network consists of a set of directed links connecting transmitters, switches and receivers. It can be represented by a directed graph $G = (V, A)$, where $V$ is a set of nodes and $A$ is a set of links. We denote a link either by a single index, $e$, or by the directed pair $(i, j)$ of nodes, which represents a lossless point-to-point link from node $i$ to node $j$. For link $e = (i, j)$, we write $\text{head}(e) = j$, and $\text{tail}(e) = i$. Each link, $e$, is associated with two parameters: a non-negative cost, $a_e$, that denotes the cost per the unit rate of sending coded packets over link $e \in A$, and a non-negative integer capacity, $u_e$, that denotes the maximum number of packets that can be sent over link $e$ in one time unit.

For node $i \in V$, two link adjacency lists $\delta_i^+$ and $\delta_i^-$ are defined as the set of links leaving node $i$ ($\text{tail}(e) = i$) and entering node $i$ ($\text{head}(e) = i$), respectively.

We now consider a set of multicast sessions, $M$, in network $G$. Subsequently, instead of one source process at a single node, $s$, we consider that there are $|M|$ source processes
at nodes $s_m$, $m \in M$. Each session $m \in M$ is identified by the source-destination pair $(s_m, T_m)$, where $s_m$ is the source node and $T_m$ is the set of receivers of session $m$. We consider a single-rate multicast, where all receivers in session $m$ receive service at the same rate, $R_m$, from $s_m$. Moreover, it is assumed that the data of different sessions are coded independently. Thus, network coding is applied to individual sessions.

We assume all link rates are constrained to be multiples of a positive basic rate $\mu$. The flow rate toward the receiver, $r$, which belongs to session $m$ and is passing through link $e$ is indicated by $\mu x_e^{m,r}$. As mentioned earlier, network coding generalizes the traditional routing paradigm. In fact, by applying network coding, multicast connections can be established with a significantly lower bandwidth requirement than that of traditional Steiner-tree-based multicast connections [1, 6]. Similar to the case of Steiner tree multicast, the consumed bandwidth of each link, $e$, for the multicast connection, $m$, is the $\max_{r \in T_m} (\mu x_e^{m,r})$ [6, 15]. In a network with limited-capacity links, the problem of determining feasibility of a set of multicast connections with network coding is more difficult than the equivalent single multicast connection problem. In the case of a single multicast connection, we have the following capacity constraint:

$$\max_{r \in T_m} (\mu x_e^{m,r}) \leq u_e.$$  

However, in the case of multiple multicast, the individual single multicast sessions are not independent and share common link capacities. In other words, each link, $e$, has a capacity, $u_e$, that restricts the total flow on that link. Consequently, the following inequality can be derived:

$$\sum_{m \in M} \max_{r \in T_m} (\mu x_e^{m,r}) \leq u_e \quad \forall e \in A. \tag{2.1}$$

In addition, for the multicast session, $m$, the flow rate $\mu x_e^{m,r}$ will be feasible if it satisfies the following flow conservation constraints [1, 16]:

$$\sum_{e \in \delta_i^+} \mu x_e^{m,r} - \sum_{e \in \delta_i^-} \mu x_e^{m,r} = \begin{cases} R_m, & i=s_m; \\ -R_m, & i=r; \\ 0, & \text{otherwise.} \end{cases} \quad \forall i \in V, \forall m \in M, r \in T_m. \tag{2.2}$$

In contrast, it is obvious that for a single multicast session problem only one set of flow conservation equations exists.

### 2.2 Problem Formulation

Here, we first describe the problem of finding the minimum cost transmission scheme for multiple multicast sessions with network coding such that link rates are constrained to
arbitrary multiples of a positive basic rate $\mu$. The problem can be formulated as follows:

\[
\begin{align*}
\min & \sum_{e \in A} \sum_{m \in M} a_e \max_{r \in T_m} (\mu x_e^{m,r}) \\
\text{subject to} & \sum_{m \in M} \max_{r \in T_m} (\mu x_e^{m,r}) \leq u_e & \forall e \in A, \\
& \sum_{e \in \delta_i^+} \mu x_e^{m,r} - \sum_{e \in \delta_i^-} \mu x_e^{m,r} = \sigma_i^{m,r} & \forall i \in V, \forall m \in M, r \in T_m, \\
x_e^{m,r} \geq 0 & \forall e \in A, m \in M, r \in T_m,
\end{align*}
\]

where

\[
\sigma_i^{m,r} = \begin{cases} 
R_m, & i = s_m; \\
-R_m, & i = r, r \in T_m; \\
0, & \text{otherwise.}
\end{cases}
\]

The objective function (2.3) minimizes the total cost of the problem and constraint (2.3b) is the capacity constraint that guarantees the total coded packet rate on link $e$ be less than its capacity. Equation (2.3c) formulates the flow conservation constraint of each session. Finally, Equation (2.3d) states the non-negative constraint for the variable $x_e^{m,r}$.

A common assumption in the existing network coding literature is that the capacity of any link is assumed to be unity or, more generally, an integer $[1, 2, 5, 11]$. It is expected that the solution to (2.3) may not generally result in quantized solutions. Indeed, $\max_{r \in T_m} (\mu x_e^{m,r})$ are not necessary to be integer multiples of a basic rate $\mu$. In this case, we may apply a rounding method in order to obtain an integer approximate solution. The resulting quantization error is then dependent on the values of the link capacities, as expected. In fact, in the case of large link capacities, although the relative quantization error would be smaller than the case of small link capacity values, still the rounded solution may not be optimal. In the network with limited-capacity, we can also use capacity scaling techniques to achieve quantized link flows. In this approach, fractional capacities can be approximated by choosing a time unit large enough [11]. If we choose the time unit $n$ times larger, the rate of coded packets on link $e$ and the multicast rate for session $m$ become $n \max_{r \in T_m} (\mu x_e^{m,r})$ and $nR_m$, respectively. If $n$ becomes sufficiently large, we can assume that all $n \max_{r \in T_m} (x_e^{m,r})$, $e \in A$ are integer multiples of a positive basic rate $\mu$. However, such capacity scaling scheme increases the encoding and decoding complexity and leads to a large delay at the receiver. If random network coding presented by Ho et al. [5] is used for intra-session network coding, then the encoding complexity at each node and the decoding complexity at each receiver increase by $n$ folds. Moreover, the receiver needs to find a matrix inverse and applies the inverse to the received coded packets in order to recover the input packets. In addition, the receiver needs to receive at least $nR_m$ coded packets to decode the original packets for each session [5], which combined with decoding delay may induce large latency at the receivers. Hence, the capacity scaling technique

5
is not a suitable method in general, and another problem formulation as an alternative to rounding or scaling schemes (2.3) is therefore needed.

Now, we consider the problem (2.3) such that link rates are constrained to be integer multiples of a positive basic rate \( \mu \). This problem is called minimum cost multiple multicast network coding with quantized rates (MCQ) and can be formulated as follows:

\[
\min \sum_{e \in A} \sum_{m \in M} a_e \max_{r \in T_m} (\mu x_e^{m,r}) \tag{2.4a}
\]

subject to constraints (2.3b), (2.3c), (2.3d) and

\[
\max_{r \in T_m} (x_e^{m,r}) \in \mathbb{Z} \quad \forall e \in A, m \in M, \tag{2.4b}
\]

where \( \mathbb{Z} \) denotes the integer set. Constraint (2.4b) ensures that the rate of injected coded packet on each link in each session to be integer multiples of a positive basic rate \( \mu \). Since variables \( x_e^{m,r} \) are continuous variables and \( \max_{r \in T_m} (x_e^{m,r}) \) is an integer variable, the problem (2.4) can be assumed as a mixed-integer problem. It should be noted that if we assume \( \mu = 1 \), then links take integer-valued rates. It can be easily verified that the problem presented by Cui et al. [12] is a special case of the problem (2.4), because they assumed only a single multicast session with \( \mu = 1 \). However, the approach taken by [12] is not applicable to scenarios in which multiple multicast sessions interact with each other and share common link capacities. In this paper, we, therefore, address such a general model where multiple multicast sessions pass through the network and link rates are integer multiples of a positive basic rate \( \mu \). To solve the problem, the proposed approach is to develop a decomposition algorithm. Also, in order to simplify the implementation of the optimization algorithm, we modify the problem by introducing auxiliary variables \( z_e^m = \max_{r \in T_m} (x_e^{m,r}) \). The variable, \( z_e^m \), can then be interpreted as the coded packet which is injected into link \( e \) in session \( m \). In addition, as seen from the problem formulation (2.4), link rates are normalized so that one unit is equal to the rate \( \mu \), implying that it can then be removed from the problem formulation through normalization. Therefore, the modified problem can be rewritten as follows:

\[
\min \sum_{e \in A} \sum_{m \in M} a_e z_e^m \quad \text{(MCQ)} \tag{2.5a}
\]

subject to

\[
x_e^{m,r} \leq z_e^m \quad \forall e \in A, m \in M, r \in T_m, \tag{2.5b}
\]

\[
\sum_{m \in M} z_e^m \leq u_e \quad \forall e \in A, \tag{2.5c}
\]

\[
\sum_{e \in \delta_i^+} x_e^{m,r} - \sum_{e \in \delta_i^-} x_e^{m,r} = a_i^{m,r} \quad \forall i \in V, \forall m \in M, r \in T_m, \tag{2.5d}
\]

\[
z_e^m \in \mathbb{Z} \quad \forall e \in A, m \in M. \tag{2.5e}
\]

\[
x_e^{m,r} \geq 0 \quad \forall e \in A, m \in M, r \in T_m. \tag{2.5f}
\]
It should be noted that the MCQ problem with convex cost can be transformed into an equivalent MCQ problem with linear cost in an expanded graph. This transformation is undertaken by defining multiple links with unit capacity instead of a normal link. The reader is referred to [17] for further details.

2.3 Problem Complexity

The MCQ problem with a general graph is strongly NP-hard since it contains the directed Steiner-tree problem as a special case. Here, it will be shown that the MCQ problem on series-parallel graphs is as hard as the knapsack problem.

Lemma 1: The minimum cost multiple multicast network coding with quantized rates is NP-hard on series-parallel graphs.

Proof. Let the knapsack problem as

\[
\max \{ \sum_{i=1}^{n} v_i x_i \} \text{ s.t. } \sum_{i=1}^{n} w_i x_i \leq W, x_i \in \{0, 1\},
\]

where \(w_i < 1\) for all \(i\) and \(W < 1\). Construct the directed graph \(G' = (V', A')\) where \(V' = \{s, 1, 2, \ldots, n, n+1\}\) and \(A' = \{(s, i) | i = 1, 2, \ldots, n+1\} \cup \{(n+1, i) | i = 1, 2, \ldots, n\}\). Set the cost of each link \(a_{si} = v_i\) and \(a_{n+1} = 0\) for \(i = 1, 2, \ldots, n\) and \(a_{sn+1} = \omega\), where \(\omega\) is a number greater than \(\sum_{i=1}^{n} v_i\). Moreover, let the capacity of each link be equal to one. We consider \(n + 1\) multicast sessions, \((s_m, T_m)\), where \(s_m = s\) and \(T_m = \{m\}\) for \(m = 1, 2, \ldots, n + 1\). Furthermore, assume \(R_m = w_m\) for \(m = 1, 2, \ldots, n\), and \(R_{n+1} = 1 - W\). The resulting network is shown in Figure 1.

Note that there is only one path from source \(s\) to node \(n + 1\). Since \(R_{n+1} \leq 1\) and satisfy the integrality property of coded packet rate on each link then the optimal value of variable \(z_{n+1}\) must equal one. After sending \(R_{n+1} = 1 - W\) units designated for node \(n + 1\), there are still \(W\) units of residual capacity left on link \((s, n + 1)\). For every node \(m \in \{1, \ldots, n\}\), there are two paths from \(s\) to \(m\); one using links \((s, n + 1)\) and \((n + 1, m)\) and the other directing link \((s, m)\). Given the optimal solution to this problem, one path obtains the optimal solution by setting \(x_m = 1\) in the knapsack problem if \(R_m\) is sent through link \((s, n + 1)\). \(\square\)

The remaining of the paper focuses on solving the MCQ problem using a decomposition algorithm. By solving this problem, a subgraph is obtained satisfying the specified constraints and has the minimum cost among all feasible answers. Prior to the algorithm, notations \(Z, Z^m, X, X^m\) and \(X^{m,r}\) are introduced which are:

\[
Z^m = (z_{e}^m)_{e \in A}, \quad Z = (Z^m)_{m \in M},
\]

\[
X^{m,r} = (x^{m,r}_{e})_{e \in A}, \quad X^m = (X^{m,r})_{r \in T_m}, \quad X = (X^m)_{m \in M}.
\]

3 Decomposition Algorithm

In order to obtain an optimal solution for the MCQ problem, a decomposition approach in which the problem is decomposed into several subproblems is used. The MCQ prob-
lem is a large-scale MILP problem with $O(|A||M|)$ integer variables, $O(|A|\sum_{m \in M} |T_m|)$ continuous variables, and $O(|N|\sum_{m \in M} |T_m| + |A|\sum_{m \in M} |T_m| + |A|)$ constraints. Hence, the application of current classical methods for the MILP problem (such as branch and bound method) is not suitable for solving the MCQ problem as these methods become computationally expensive and impractical as the size of the problem increases. A decomposition technique should therefore be employed to separate the original model into independent subproblems to find the optimal solution. The MCQ problem can be separated into $\sum_{m \in M} |T_m|$ independent problems when integer values of vector $Z$ are given. Thus, a Benders decomposition approach may be employed to solve the MCQ problem, in which the $Z$-variables are derived from an integer master problem and the $X$-variables are determined via a series of linear programming subproblems.

To apply the Benders decomposition approach for solving the MCQ problem, we first define the projection of the MCQ problem onto the space of the integer variables, $Z$, as follows:

$$\min \sum_{e \in A} \sum_{m \in M} a_e z_e^m$$

subject to

$$\sum_{m \in M} z_e^m \leq u_e \quad \forall e \in A,$$  \quad (3.6a)

$$z_e^m \in \mathbb{Z} \quad \forall e \in A, m \in M,$$  \quad (3.6b)

$$Z^m \in \Gamma^m \quad \forall m \in M,$$  \quad (3.6c)

where

$$\Gamma^m = \{Z^m \in \mathbb{R}^{|A|} | F_{m,r}^m \neq \emptyset, \forall r \in T_m\},$$

and

$$F_{m,r}^m = \{x_{m,r}^e | 0 \leq x_{m,r}^e \leq z_e^m \quad \forall e \in A, \sum_{e \in A^+} x_{e}^{m,r} - \sum_{e \in A^-} x_{e}^{m,r} = \sigma_i^{m,r}, \forall i \in V\}.$$  \quad (3.7a)

A vector $Z^m$ that satisfies constraints (3.6b) and (3.6c) provides a feasible flow for receiver $r \in T_m$ if and only if $F_{m,r}^m \neq \emptyset$, i.e. there exists a solution to the following system of equations:

$$-x_{e}^{m,r} \geq -z_e^m \quad \forall e \in A,$$  \quad (3.7a)

$$\sum_{e \in A^+} x_{e}^{m,r} - \sum_{e \in A^-} x_{e}^{m,r} = \sigma_i^{m,r} \quad \forall i \in V,$$  \quad (3.7b)

$$x_{e}^{m,r} \geq 0 \quad \forall e \in A.$$  \quad (3.7c)
By Farkas Lemma, system (I) has a solution if and only if the following system is infeasible:

\[
- \sum_{e \in A} \gamma_{e}^{m,r} z_{e}^{m} + (\pi_{s_m}^{m,r} - \pi_{r}^{m,r}) R_{m} > 0, \quad (3.8a)
\]

\[
(II) \quad - \gamma_{e}^{m,r} + \pi_{i}^{m,r} - \pi_{j}^{m,r} \leq 0 \quad \forall e = (i, j) \in A, \quad (3.8b)
\]

\[
\gamma_{e}^{m,r} \geq 0 \quad \forall e \in A, \quad (3.8c)
\]

where vectors \( \gamma \) and \( \pi \) are the dual variables corresponding to the constraints (3.7a) and (3.7b), respectively. The following linear programming is introduced in order to check the feasibility of system (II).

\[
SP(Z^{m}, m, r) :
\]

\[
z_{sp}^{m,r} = \min \sum_{e \in A} \gamma_{e}^{m,r} z_{e}^{m} - (\pi_{s_m}^{m,r} - \pi_{r}^{m,r}) R_{m}
\]

\[
s.t.
\]

\[
\gamma_{e}^{m,r} - \pi_{i}^{m,r} + \pi_{j}^{m,r} \geq 0 \quad \forall e = (i, j) \in A, \\
\gamma_{e}^{m,r} \geq 0 \quad \forall e \in A.
\]

**Lemma 2:** Optimal objective value of Model (3.9) is less than or equal zero for every optimal solution.

**Proof:** Model (3.9) has a feasible solution \( \gamma_{e}^{m,r} = 0, \forall e \in A, \pi_{i}^{m,r} = 0, \forall i \in N \) with the objective function equal zero. Hence, the optimal value of the objective function is less than or equal to zero. □

Lemma 2 concludes that if vector \( Z^{m} \) admits a feasible flow in system (I) using Farkas Lemma, system (II) is infeasible; then the optimal value of subproblem (3.9) is zero. Otherwise, the following result is obtained.

**Lemma 3:** If \( Z^{m} \) admits no feasible flow in system (I), there exists an extreme direction \( (\gamma_{e}^{m,r}, \pi_{r}^{m,r}) \) for subproblem (3.9) that \( \sum_{e \in A} \gamma_{e}^{m,r} z_{e}^{m} - (\pi_{s_m}^{m,r} - \pi_{r}^{m,r}) R_{m} < 0. \) □

Consequently, when system (I) is infeasible, the following inequality is obtained.

\[
\sum_{e \in A} \gamma_{e}^{m,r} z_{e}^{m} \geq (\pi_{s_m}^{m,r} - \pi_{r}^{m,r}) R_{m} \quad (3.10)
\]

Inequality (3.10) is known as a valid inequality for session \( m \) and receiver \( r \), which is called a Benders cut. The above mentioned results are used to specify the following master problem (MP), which is equivalent to Model (3.6):

\[
9
\]
\[
\min \sum_{e \in E} \sum_{m \in M} a_e z_{e}^m \\
\text{(MP)}
\]

subject to
\[
\sum_{m \in M} z_{e}^m \leq u_e \quad \forall e \in E, \\
\sum_{m \in M} z_{e}^m \in \mathbb{Z} \quad \forall e \in E, m \in M, \\
\sum_{e \in E} \gamma_{e}^{s_m,r} z_{e}^m \geq \left(\pi_{s_m}^{s_m,r} - \pi_{r}^{s_m,r}\right)R_m \\
\forall \left(\gamma_{e}^{s_m,r}, \pi_{e}^{s_m,r}\right) \text{ satisfying the constraints of problem (3.9), } \forall m \in M, r \in T_m
\]

Finding a direct solution to the master problem is nontrivial. This is due to the fact that there are typically an exponential number of extreme directions of subproblem (3.9), and generating all constraints of type (3.11d) is not practical. Hence, in this circumstance, we can adopt a relaxation strategy and start with only a subset of such constraints. The relaxed master programs are then built by adding the Benders cuts iteratively, corresponding to finding a solution to subproblem (3.9). Therefore, the relaxed master problem (RMP) takes the following form:

\[
\min \sum_{e \in E} \sum_{m \in M} a_e z_{e}^m \\
\text{(RMP)}
\]

subject to
\[
\sum_{m \in M} z_{e}^m \leq u_e \quad \forall e \in E, \\
\sum_{m \in M} z_{e}^m \in \mathbb{Z} \quad \forall e \in E, m \in M, \\
\sum_{e \in E} \gamma_{e}^{s_m,r} z_{e}^m \geq \left(\pi_{s_m}^{s_m,r} - \pi_{r}^{s_m,r}\right)R_m \quad \forall \left(\gamma_{e}^{s_m,r}, \pi_{e}^{s_m,r}\right) \in B,
\]

where \(B\) is initially null and is iteratively generated to be the set of extreme directions generated by solving (3.9) for \(m \in M, r \in T_m\).

Let \(\tilde{Z}^m, \forall m \in M\) be an optimal solution for the RMP. We then solve each subproblem \(SP(\tilde{Z}^m, m, r), \forall m \in M, r \in T_m\). It follows from Lemma 2 that \(z_{sp}^{m,r} \leq 0, \forall m \in M, r \in T_m\). If \(z_{sp}^{m,r} = 0, \forall m \in M, r \in T_m\), then it can be concluded that \(\tilde{Z}^m\) satisfies constraints (3.7). Thus, using the Benders decomposition, we will have \(\tilde{Z}^m, \forall m \in M\), which is an optimal solution of the MCQ problem and the algorithm stops. Otherwise, for some subproblem \(SP(\tilde{Z}^m, m, r)\), there is an extreme direction \((\hat{\gamma}_{e}^{m,r}, \hat{\pi}_{e}^{m,r})\) such that \(z_{sp}^{m,r} < 0\). Then, a violated inequality of type (3.10),

\[
\sum_{e \in E} \hat{\gamma}_{e}^{m,r} z_{e}^m \geq \left(\hat{\pi}_{s_m}^{m,r} - \hat{\pi}_{r}^{m,r}\right)R_m,
\]

(3.12)
can be identified in a polynomial time by solving Model (3.9). Hence, the extreme direction 
\((\hat{z}^m,r, \hat{\pi}^m,r)\) will be added to \(B\) to specify a new relaxation of the MP. Subsequently, the 
relaxed master problem is re-solved. Note that one constraint of form (3.10) is added to the 
relaxed master problem for each \(r\) and \(m\) in which \(zsp^m,r < 0\).

The RMP includes all constraints of the original problem involving integer variables \(z^m_e\) 
plus some of the Benders cuts. Since a smaller number of extreme directions is included in 
the RMP, the optimal value of the objective function of the RMP provides a lower bound for 
the original formulation. We have used the commercial software Cplex to solve the RMP 
[18]. Each time that a new constraint is added to the master problem, the optimal value 
of its objective function can only increase or remain at the same value. The procedure of 
adding new violated inequalities and solving the resulting RMP is repeated until no more 
violated inequalities are found. Since the number of constraints generated could be large, 
inactive constraints are deleted at the end of each iteration.

In the next section, several valid inequalities in the RMP are introduced to increase the 
convergence speed in the Benders decomposition algorithm.

4 Adding Valid Inequalities to The Master Problem

The computational efficiency of the above mentioned algorithm depends mainly on two 
issues including (i) the time needed to solve the RMP and (ii) the number of iterations 
required to obtain an optimal solution. In order to reduce the number of iterations in the 
Benders decomposition algorithm, Geoffrion et al. [14], and Magnanti et al. [19] sug-
gested to enrich the initial master program with valid inequalities as much as possible. An 
inequality is said to be valid for a set if it is satisfied by every point in this set. To increase 
the convergence speed, the following valid inequalities in the initial RMP are introduced.

4.1 Initial Inequality

At a basic level, in order to strengthen the RMP, we can set initial inequalities such that 
sufficient capacity for exiting the source nodes and entering the receiver nodes for each 
session is guaranteed. The inequalities can guarantee that the total coded packet rate of 
links coming out of source nodes for each session must not be less than the required rate 
to that session. According to equation (2.5b), the following inequalities are obtained for 
session \(m\).

\[ x_e^{m,r} \leq z_e^m \quad \forall r \in T_m, \quad e \in \delta_{s_m}^+ . \]

By taking sum over the flow of leaving links of source \(s_m\), we have:

\[ \sum_{e \in \delta_{s_m}^+} x_e^{m,r} \leq \sum_{e \in \delta_{s_m}^+} z_e^m \quad \forall r \in T_m. \]
Since $\sum_{e \in \delta^+_m} x^{m,r}_e = R_m$, due to the flow conservation constrain, the following inequalities are valid inequalities for session $m$.

$$
\sum_{e \in \delta^+(s_m)} z^m_e \geq R_m \quad \forall r \in T_m.
$$

(4.13)

In addition to inequalities derived from source node for session $m$, a similar set of inequalities can be generated from receiver nodes in the same session. Consider Equation (2.5b) with respect to the set of links entering to receiver $r \in T_m$ as follows:

$$
x^{m,r}_e \leq z^m_e \quad e \in \delta^-_r.
$$

In a similar manner, the following valid inequality is obtained for session $m$ and receiver $r \in T_m$.

$$
\sum_{e \in \delta^-(r)} z^m_e \geq R_m.
$$

(4.14)

It can be noted that the above mentioned constraints can be written as an aggregated constraint for session $m$ as follows:

$$
\sum_{r \in T_m} \sum_{e \in \delta^-(r)} z^m_e \geq R_m |T_m|.
$$

Suppose that the RMP is solved by adding the aggregated constraints. Since there is no restriction on the rate entering each receiver in the RMP, then it is possible that the rate entering some receivers become lower than the required rate, leading to infeasibility. Consequently, it is more suitable to solve the RMP with disaggregated constraints despite the fact that it contains a larger number of constraints.

### 4.2 Connectivity Inequality

Another set of valid inequalities deals with the condition that the RMP constructs a connected subgraph for multiple multicast connections, which is called connectivity inequality. If there is a positive flow on link $e$ that enters some node $i$ in session $m$ (i.e., $z^m_e > 0$), then a link must be constructed such that it exits this node. Similarly, a link that exits node $i$ should only be constructed if at least one link enters node $i$. Hence, the following valid inequalities can be added to the RMP for each session $m$ and for each $i \in N - \{s_m\} \cup T_m$.

$$
z^m_{ij} \leq \sum_{e \in \delta^-_i} z^m_e, \quad \forall (i, j) \in \delta^+_i
$$

$$
z^m_{ji} \leq \sum_{e \in \delta^+_i} z^m_e, \quad \forall (j, i) \in \delta^-_i.
$$
5 Heuristic Generating Upper Bound Procedure

The objective value of the RMP is a lower bound to the objective value of the MCQ problem. Moreover, adding the valid inequalities to the RMP improves the quality of lower bounds. In the Benders decomposition technique, for problems involving costs on subproblem variables, upper bounds on the optimal value of the objective function can be obtained. The bound can be found by adding up to the value of each subproblem objective, and adding them to the master problem objective, assuming that each subproblem is feasible [14]. But in the MCQ problem, according to the Lemmas 2 and 3, each subproblem (3.9) either is infeasible or has an objective value equal to zero. So, in such cases, a heuristic method can be employed to obtain the upper bound. Finding an upper bound may allow a decision maker to terminate the procedure before optimality is proved. This happens when the relative gap between upper and lower bounds is sufficiently small.

Hence, we develop a heuristic procedure based on the linear programming relaxation and the cutting-plane technique. A formal description of this heuristic procedure is given as follows:

**Step 0:** Begin with some $\tilde{Z}$, perhaps given by the solution of the current RMP, as the initial heuristic solution. Set iteration counter $l = 0$ and initialize the heuristic solution $Z^l = \tilde{Z}$. Also, set the heuristic objective $F^l = \sum_{e \in A} \sum_{m \in M} a_e z^m_e$.

**Step 1:** Solve subproblem (3.9) for each $m$ and $r$. If the optimal value of all the subproblems is equal to zero, the algorithm terminates with upper bound $F^l$. Otherwise, generate a Benders cut (3.10) for some $m$ and $r$ that their corresponding subproblems are infeasible. Label the set of Benders cuts generated at this iteration $BC^l$, and proceed to Step 2.

**Step 2:** Solve the following linear programming:

\[
F^{l+1} = \min \sum_{e \in A} \sum_{m \in M} a_e z^m_e
\]

s.t.

\[
\text{Benders cuts given by } BC^l,
\]

\[
\sum_{m \in M} z^m_e \leq u_e \quad \forall e \in A,
\]

\[
z^m_e \geq \tilde{z}^{1m}_e \quad \forall m \in M, \ r \in T_m.
\]

If problem (5.15) is feasible, an optimal solution $\tilde{Z}$ can be obtained from it. If vector $\tilde{Z}$ is an integer, increment $l$ by one unit and return to Step 1. Otherwise, the lower bound of the variables $z^m_e$, inequalities (5.15d), is updated as follows:

Fix the lower bounds as $z^m_e \geq \tilde{z}^m_e$ for all $\tilde{z}^m_e$ that are integral and, among the fractional solutions, select the variable with the smallest value of $a_e(\lceil \tilde{z}^m_e \rceil - \tilde{z}^m_e)$. The lower bound for this variable is fixed with $\lceil \tilde{z}^m_e \rceil$ subject to constraint (5.15c) not to be violated in respect to the lower bounds and then return to Step 2.

Otherwise, in a case that Model (5.15) is not feasible, terminate the algorithm with no upper bound obtained.
Note that this heuristic method can be repeated only in a limited number of times since the lower bound of the variables, inequalities (5.15d), strictly increases at each iteration (That is, $Z_{l+1}^t \geq Z_l^t$ and strict inequality must be held for at least one component since constraints (5.15b) are infeasible with respect to $Z_l^t$). Additionally, we solve problem (5.15) only with constraints in $BC_l^t$, due to a strict increase of the lower bound of variables. Inequalities (5.15d) guarantee that constraints in $BC_0^l, ..., BC_l^{l-1}$ are satisfied. So, using a similar argument, we do not need to re-examine subproblems that were shown to be feasible with respect to a prior heuristic solution.

6 Simulation Results

In this section, the computational results of simulations undertaken are presented in order to evaluate the performance of the proposed technique. To do this, we initially simulated the proposed algorithm using some trivial cases. Then, we ran the simulations for simpler and more complex cases. In all cases, the rate of coded packet injection at each link of each session was assumed to be a multiple of the basic rate $\mu$. Three different scenarios were considered for these simulations. In the first scenario, the rates on links were assumed to have a quantized value. This scenario is the main model considered in this study, which is solved using the algorithm presented in Section 3. In the second scenario, link rates were constrained to arbitrary multiples of a positive basic rate $\mu$. Note that in this scenario, the obtained solution is not necessarily an integer value of multiple $\mu$. In the third scenario, we chose the closest integer from the solutions obtained in Scenario 2.

All scenarios were solved in the Windows environment using GAMS 23.6 Integrated Development Environment with Cplex 12 on a PC with processor Intel(R) Atom(TM) N270 with 1.6 GHz and 1GB of RAM.

6.1 Simulation using the trivial case

Consider a network with two multicast sessions where each link has a cost, as shown in Figure 2. Each session is identified by the source-destinations pair of $(s1, \{t1, t2\})$ and $(s2, \{t2, t3\})$. Furthermore, assume that each link has a capacity of one unit, and the requested rate of each session is 1. The rate of coded packet injection at each link of every session was assumed to be a multiple of basic rate $\mu = \frac{1}{3}$.

Figures 3(a), 3(b) and 3(c) show the coded packet rates injected on the links of each session in Scenarios 1, 2 and 3, respectively. As a result of Figure 3, the algorithm has to choose a quantized value in spite of its higher cost. However, the cost of a quantized value is lower than the closest integer to the non-quantized value. This result may explain the advantage of the proposed method.
6.2 Simulation using more complex topologies

In this section, we assess the proposed method on random networks. Firstly, we describe how to generate the random test instances of the problem, and then the computational experiments and associated results are presented. In this scenario, we assumed that the basic rate is $\mu = \frac{1}{2}$. The networks were randomly generated using two methods including (i) the random directed graph (RDG) and (ii) random geometric graph (RGG). The random directed graphs were generated using the methodology proposed by Erdos and Renyi [20]. In this case, it was assumed that there is a link from node $i$ to node $j$ with probability 0.5. Then, for each link, a uniform random number was generated on an interval, say $[a, b]$, in order to represent the link cost. In experiments conducted in this case, we selected $[a, b]$ equal to $[5, 7]$. The random geometric graphs, RGGs, were generated using the methodology described in [21]. Using this methodology, the nodes are randomly placed in a $10 \times 10$ units square, according to a uniform distribution. Then, a link from node $i$ to node $j$ is constructed when the Euclidean distance between the nodes $i$ and $j$ is less than $\rho_i$, where $\rho_i$ is a uniform distribution number on an interval, which are considered as $[2, 5]$ in this case. Finally, the cost of link $(i, j)$ is set to $d^2$, where $d$ is the Euclidean distance from node $i$ to node $j$.

In order to complete the instance networks, we considered the capacity of each link to be equal one unit and the transmit rate, $R_m = 1$, for each session. The test instances were generated according to the RDG and RGG methodologies and named $\text{Rand-x}$ and $\text{Geo-x}$, respectively, where $x$ is the number of the instance.

The obtained results for all three scenarios are shown in Table 1. Since the main goal of this study is to achieve a multiple of the basic rate, a comparison was undertaken between the obtained costs in Scenarios 1 and 3. It can be seen that depending on the network and size of the multicast group, the total cost reduction ranges between 23 and 58% in Scenarios 1 and 3, respectively. However, in Scenario 2 the total costs is lower than in Scenario 1. This is due to the fact that in this scenario, the feasible region is expanded.

As mentioned earlier, the direct solution of the proposed problem using the existing classical methods is expensive and impractical. In contrast, a crucial point in most implementations of the Benders decomposition is the application of the feasibility cuts effectively for either a priori or iteratively generated cuts. We also tested the following strategies by augmenting the master problem with specific cuts, as described in Section 4.

1. Direct solution of the proposed problem via the existing classical methods (Dir)
2. The application of the Benders decomposition with no augmentation (BD)
3. The application of the Benders decomposition with initial and connectivity inequalities (Ini-Con)

Note that these strategies are only a subset of all possible combinations. To measure the performance of different strategies, the program execution time (CPU time) and required number of iterations were used.

The results presented in Table 2 show the performance of the proposed technique under different strategies. As expected, the program execution time and the number of iterations
were reduced when cuts were added to the RMP. Some of these instances run faster than other strategies, which reveal the fact that the efficacy of some of these approaches highly depend on the network topology and/or instance data. However, note that the Ini-Con strategy had the least average number of iterations, which is due to the choice of a suitable initial solution for the Benders decomposition technique. The results also indicated that the Dir strategy performs poorly for these instances and cannot find an optimal solution within the time taken by the BD strategy.

Figure 4 shows the behavior of the Benders decomposition with no augmentation when it was used to obtain an optimal solution for instances in Table 2. The number of iterations was limited to 15 and the gap between the obtained solutions after 15 iterations and the optimal solution for instances presented in Table 2 are displayed in Table 3. As seen in Table 3, the gap between the obtained and optimal solutions using the Ini-Con strategy is less than the Dir strategy.

Finally, the procedure of generating the upper bound was tested. This procedure was run on the instances presented in Table 2. The obtained solutions under the Ini-Con strategy, which are presented in Table 3, were used as the initial heuristic solutions for generating the upper bound procedure. Table 4 shows the obtained upper bound using this procedure after 10, 20, 30, and 40 iterations. It reports “Fail” if the algorithm terminates with no upper bound obtained. Note that the heuristic provides a poor feasible solution after 10 iterations, but as the Benders cuts are added to the master problem, the obtained upper bound becomes tighter very quickly. In Figure 5, the behavior of the upper and the lower bounds applied to test instance Geo_2, as presented in Table 2, is shown. The optimal cost of test instance Geo_2 was obtained by directly solving Model (2.5). It can be seen that in less than 18 iterations, the obtained cost from the Benders decomposition using the Ini-Con strategy is within 5% of the optimal value. Moreover, in less than 25 iterations, the cost using the heuristic generating upper bound procedure is within 10% of the optimal value. Thus, the upper and lower bounds yield an interval that the optimal solution is within it, and a near-optimal solution of any required accuracy can always be found within a finite number of iterations. As seen in Table 4, this procedure cannot, however, obtain an upper bound for general networks, and may fail in some networks. This issue is shown in Table 4 for test instance Geo_1. In any scenario, if the heuristic method does not fail, there is a possibility that the upper and lower bounds converge to the same value.

7 Conclusion

In this paper, the problem of the minimum cost transmission scheme for a set of multicast connections with network coding was examined, where the injection rate of the coded packets on each link of every session was constrained to quantized values. The problem was formulated using an MIP, where it was shown that the problem is NP-hard. An effective algorithm was developed to solve it in an acceptable amount of time by employing a standard Benders decomposition approach. The results showed that this approach sig-
nificantly improve the solution of the problem instances, although it still takes too long to solve more difficult instances. In order to improve the convergence speed, some augmentations were then proposed, which tend to reduce the number of iterations of the master program. Each iteration of the Benders decomposition algorithm acquired a lower bound for the objective value of the MCQ problem. Therefore, a heuristic method was used to obtain a tight bound in a reasonable amount of computational time. The upper and lower bounds yielded an interval containing the optimal solution. Also, a near-optimal solution of any required accuracy could always be found within a finite number of iterations. In addition, the results showed that the proposed algorithm has cost lower than the solution obtained by finding the closest integer to the non-quantized value. In general, there was no guarantee that the closest integer solution would be feasible, and that the proposed scheme could lead to optimal subgraphs when such solutions are unavailable using the traditional schemes by which quantized values cannot be directly found.
References


