ICRA: Incremental Cycle Reduction Algorithm for optimizing multi-constrained multicast routing

Naouel Ben Ali
HANA Research Group, ENSI, University of Manouba, Tunisia
naouel.benali@ensi.rnu.tn

Miklos Molnar
INSA, University of Rennes 1, France
molnar@irisa.fr

Abdelfettah Belghith
HANA Research Group, ENSI, University of Manouba, Tunisia
abdelfattah.belghith@ensi.rnu.tn

Abstract

In future Internet, multimedia applications will be strongly present. When a group of users is concerned by the same traffic flow, the multicast communication can decrease considerably the network bandwidth utilization. The major part of this kind of multicast communication needs quality of service (QoS) specification. Often, the QoS is given as a set of QoS criteria and the computation of feasible or optimal routes corresponds to a multi-constrained optimization. Finding the multicast graph respecting the defined QoS requirements and minimizing network resources is a NP-complete optimization task. Exhaustive search algorithms are not supported in real networks. Greedy algorithms was proposed to find good multicast sub-graphs. In this paper, we propose a greedy algorithm ICRA to improve the multi-constrained multicast sub-graph computed by the already proposed algorithm Mamcra. We show through different examples that ICRA tackles all cases that Mamcra fails to deal with.

Keywords: multicast routing, Quality of Service, additive constraints, optimisation, greedy approach

1. INTRODUCTION

Recently, the Internet has shown a tremendous growth. Emergent multicast applications like audio/video conferencing, video on demand, IP-telephony, etc. . . usually have Quality Of Service (QoS) requirements, which include bandwidth, bounded delay, jitter and loss rate. Several IP multicasting techniques have been proposed to support point-to-multipoint communications by sharing link resources leading to a reduction in network resource consumption. All these techniques are based on IP multicast routing protocols which use shortest path tree algorithm based on one single metric, typically delay or hop count. Multicast applications today needs to optimize more than one metric that's why multi-constrained QoS routing is needed. QoS routing is a routing scheme under which paths for flows are attributed by taking into account flow requirements and are based on some knowledge of resource availability in the network. Multicast routing deployed in the Internet aims to use resources efficiently. In a point to multipoint session, p destinations will receive the same information. Sending p times over each shortest path to each individual multicast member is inefficient. Sending single packets through the shared links and duplicating them if it is necessary is more efficient. When we consider a single metric, multicast source routing can be achieved by forwarding packets over the shortest path tree for example. When the overall cost of the tree must be minimized, the problem must be tackled differently. Determining the minimal cost multicast tree for a multicast group corresponds to the Minimum Steiner Tree problem which is shown to be NP-complete [4]. An additional dimension to the multicast routing problem is to construct trees or sub-graphs that will satisfy multiple QoS requirements. Routing problem (in the rest of the paper, QoS routing refers to multi-constrained QoS Routing) even in the unicast case is known to be NP-complete problem and has been extensively studied by the research community. [7] gives an overview of the main proposed QoS routing algorithms which try to found a path between a source and a destination node that satisfies a set of constraints. For the multicast case, a number of QoS routing algorithms based on single, dual and multiple metrics have been proposed. Single metric QoS multicast routing algorithms have been proposed for cost [1, 9], delay

[2, 5]. Dual metric based routing algorithms have been proposed for the following combinations: cost-delay [3] and delay-jitter [9]. For the multi-constraint multicast routing problem which involves multiple QoS metrics, few algorithms have been proposed due to the complexity nature of this problem. Multicast Adaptive Multiple Constraints Routing Algorithm (Mamcra) [6] attempts to find multiple QoS constraint paths to the multicast members in an efficient but not always optimal manner. The main idea of Mamcra is to compute multi-constraint shortest paths from the source node to each destination using a unicast QoS routing algorithm, Samcra [8]. The set of obtained paths is then optimized to obtain a multicast sub-graph that uses as few links from the first paths set as possible. Mamcra proposes a greedy heuristic approach to solve this second problem. The quality of the approximation isn't proved and the shortcoming of the proposed greedy algorithm can be improved. This paper deals with multiple constraint QoS multicast routing problem which constitutes one of the most interesting problems of multi-objective optimization in network field. In this paper, we will focus, essentially, on optimizing the set of shortest paths to solve the multiple constraint QoS multicast routing problem. The set of paths can be obtained by Samcra or any multiple constraint QoS unicast routing algorithm. Considering the drawbacks of Mamcra's greedy algorithm, we propose ICRA, an improvement version of the greedy algorithm algorithm to provide a solution which can be close to the optimal solution.

This paper is organized as follows. Section 2 specifies the multiple constraint problems for multicast QoS routing and provide an overview of most proposed approaches to treat these problems. We describe also Mamcra algorithm proposed to solve multiple constraint multicast routing problem. This section emphasizes also Mamcra's weak points. Section 3 proposes a formulation of a new problem aiming to optimize the multicast sub-graph, the OMS problem. Section 4 investigates how the problem of optimizing multicast sub-graph must be tackled if incremental search algorithms are adopted and it proposes ICRA, an incremental algorithm to solve the OMS problem.

2. MULTI-CONSTRAINED MULTICAST ROUTING

The problem of QoS routing, even in the unicast case, is known to be NP-complete. Before presenting the mechanisms used to construct multi-constrained QoS multicast delivery structures, we need first to specify some hypotheses used to solve these problems and the notation used throughout this paper. Then we present existing approaches to achieve multicast QoS routing.

2.1. Hypotheses

QoS routing approaches assume that the network-state information is temporarily static and has been distributed throughout the network by any appropriate traffic engineering mechanism. The QoS metrics are categorized into additive (e.g., delay, jitter,..) and min (resp. max) metrics (e.g. bandwidth). The constraints on min (resp. max) QoS measures can be easily treated by pruning all links (and disconnected nodes) which do not satisfy the requested min (resp. max) QoS constraints. In contrast, constraints on additive QoS measures cause more difficulties. Hence, without loss of generality, all QoS measures are assumed to be additive.

A network topology is modeled as an undirected graph $G = (V, E)$, where V is the set of vertices representing the network nodes and E is the set of edges representing the network links. Each link is characterized by m additive QoS metrics. So, we associate to each link an m -dimensional link weight vector of m non-negative QoS weights (w_i^j , for $i = 1, 2, \dots, m$ and j is a link in E). The m QoS constraints (limits) L_i for $i = 1, 2, \dots, m$ are represented by the constraint vector $\vec{L}$. A multicast group g is composed of a source s and a set of members $D = \{d_1, d_2, \dots, d_p\}$ where p is the number of multicast members.

2.2. Multicast QoS routing problem specification

For multicast QoS routing, the constraint vector $\vec{L}$ represents the limits allowed for path weights from the source node to every member of the multicast group. The constraint vector $\vec{L}$ is assumed to be the same for all multicast members. In the following, we present firstly the exiting problems specified in the literature then we formulate a new problem dealing with multicast QoS routing.

2.2.1. Existing multicast QoS routing problems

Multicast QoS routing problem consists on finding a set of paths from a node source s to p destination nodes d_j ($j = 1, \dots, p$). In traditional multicast routing this set of paths corresponds to a tree but in multicast QoS routing it is not compulsorily a tree. In the general case, it corresponds to a sub-graph $M = (W, H)$, $M \subseteq G$. M is regarded as a set of paths from s to d_j . Under such hypothesis, three problems were formulated in [6].

Problem I: Multiple Constrained Multicast (MCM)

Given s and D , find $M(W, H)$ such that, for each path $P(s, d_j)$ from s to $d_j \in D (j = 1, \dots, p)$:

$$w_i(P) \leq L_i \text{ for } i = 1, \dots, m \quad (1)$$

Note that, if for a certain $d_j \in D$ no feasible path exists, the problem has no solution. To find a solution for the remaining sub-set of destinations, d_j should be removed from D .

Problem II: Multiple Parameter Steiner Tree (MPST)

Given s and D , find $M(W, H), \{s, D\} \subseteq W$ for which $l(M)$ is minimum where l is length fonction.

$$l(M) = \max_{i=1, \dots, m} \left(\frac{w_i(M)}{L_i} \right) \quad (2)$$

where $w_i(M) = \sum_{(u,v) \in H} w_i(u, v) \text{ (i=1, \dots, m)}$

Problem III: Multiple Constrained Minimum Weight Multicast (MCMWM)

Given s and D , find $M(W, H)$ such that, for each path $P(s, d_j)$ from s to $d_j \in D (j = 1, \dots, p)$:

$$w_i(P) \leq L_i \text{ for } i = 1, \dots, m \text{ and } l(M) \text{ is minimum.} \quad (3)$$

We introduce a new metric $cl(M)$, the critical length of the multicast sub-graph M . $cl(M)$ is given by:

$$cl(M) = \max_{d_j \in D} l(P(s, d_j)) \quad (4)$$

In the rest of papers, we will focus only on the problem MCM and eventually on the problem MCMWM. Using the metric $cl(M)$ defined in (4), the MCM problem can be formulated as the determination of a multicast sub-graph M satisfying (5)

$$cl(M) \leq 1. \quad (5)$$

Using the cl metric, the MCMWM problem corresponds to find $M(V, H)$ satisfying (6).

$$cl(M) \leq 1 \text{ and } l(M) \text{ is minimum.} \quad (6)$$

So, to summarize, we can deduce that solving the MCM problem results in satisfying the QoS requirement of multicast members. The MCMWM optimizes the total weight of the multicast sub-graph while satisfying QoS requirements.

2.3. Multi-constrained multicast routing proposals

For the multicast case, a number of QoS routing algorithms based on single, dual and multiple metrics have been proposed. Few works, however, have dealt with multiple QoS metrics. The Multicast Adaptive Multiple Constraints Routing Algorithm (Mamcra) proposed by [6] as the multicast extension of Samcra, a unicast QoS routing algorithm [8] is the main proposed algorithm. The Mamcra algorithm computes the solution of the multiple constrained multicast routing problem by following two steps. In the first step, the set S of shortest paths from the source node s to all p multicast destination members is computed using a lightly modified version of Samcra. The second step aims the reduction of the resulting sub-graph such that the length function is reduced without violating the constraints. The obtained sub-graph s , it may contain cycles. Mamcra adopts a greedy approach to eliminate as many cycles as possible. The resulting sub-graph may not be a tree and even if M corresponds to a tree, it is not necessary a minimal length tree. This is due to the greedy approach adopted by the Mamcra reduction step. In order to decrease the network ressources use, our proposition concerns the improvement of the multicast-graph reduction. Authors in [10] analyze this step and showed that it suffers from four problems. These problems are resumed as follows:

- The first problem is raised when cycles are removed. In Mamcra, only one possibility is checked to eliminate a cycle.
- The second problem is the eventual existence of intermediate destination nodes in a path. Generally, only a part of the sub paths composing a cycle can be removed not to affect the reception of the intermediate nodes. Multiple possibilities must be examined.
- The third problem is raised when some links are used by different paths in the opposite direction. The criterion choice of common nodes in the path reduction algorithm should be specified and this criterion influences the reduced structure.
- The last problem is the order adopted to remove cycles influences the number of cycles removed. That is due to the incremental approach of Mamcra used in removing cycles.

The first problem can be easily resolved by making little modification to Mamcra procedure without changing Mamcra principles. But to tackle the three last problems, the approach must be reviewed wholly and a global optimization procedure is needed. Before proposing algorithms improving MAMCRA reduction procedure, we must first formulate the problem of optimizing multicast sub-graph.

3. OPTIMIZING MULTICAST SUB-GRAPH (OMS) PROBLEM

Eliminating redundancies from the multi-constraint multicast routing structure which is achieved during the second step of Mamcra can be formulated as an optimization problem. In this section, we propose a formulation of the Optimal Multicast Sub-Graph Problem.

Generally, let us suppose that a set of feasible paths to the destinations of a multicast group is available (for example: the set of "shortest paths" computed by Samcra). Our goal is the global optimization of this set: remove the more redundancies without violating the QoS constraints.

Optimizing Multicast Sub-Graph (OMS) Problem:

Let us suppose that, for a source $s \in V_S$ and a set of destinations $D = \{d_i \in V_S, i = 1, \dots, g\}$, a graph $G_S = (V_S, E_S)$ is given such as the union of feasible paths from the source to each destination. The goal is to find M a sub-graph of G_S optimizing (minimizing) an objective function $f(M)$. This objective function can be either the length function $l(M)$ specified in the MCMWM or the number of hops $h(M)$.

To solve the OMS problem efficiently, some reductions are possible and we propose to adopt some definitions as follows.

A node $n \in G_S$ is a *significant node* if it is the source node or a destination node or a branching node having a degree more than 2.

A *segment* is a path connecting two neighboring significant nodes of G_S .

A segment is called *articulation segment*, if its deletion increase the number of connected components in the graph.

For example, we can consider the example in (Figure 1) as a graph in which only the significant nodes are represented. (s, x_1) and (x_1, x_2) are segments containing intermediate nodes or not. (x_0, d_3) , (d_5, d_2) and (x_3, d_5) are articulation segments in this graph.

Reduction 1: A link belongs to the optimal solution of the OMS problem if and only if the segment containing it is in the solution.

So the set S of paths that we aim to optimize can be considered as a set of segments. Thus, optimizing multicast sub-graph consists of determining the set of segments of G_S that should be removed. In this way, a (probably reduced) graph G'_S containing the segments of G_S can be examined to solve the problem.

Another reduction of the search space is possible by detecting the segments which belong to all solutions. These segments can never be removed. For example, if we refer to the topology of Figure 1, the segments (x_0, d_3) , (d_3, d_2) , (x_3, d_5) and (x_3, d_1) can not be removed.

Figure 1: Decomposition of a set of paths in segments

From the point of view of its complexity, the OMS problem is equivalent to the original MCMWM problem. find the optimal sub-graph of reduced graph G'_S (instead of S) spanning the multicast group with respect to the QoS constraints. In the worst case the sub-graph G'_S corresponds to the whole graph G and if reductions are not possible, $G'_S = G_S = G$. In this case OMS corresponds exactly to the original MCMWM problem which is NP-complete [6]. To find the optimal solution of the OMS problem, exhaustive search algorithms can be imagined (enumeration of each combination of redundant segments for example). Incremental greedy algorithms such as Mamcra can be adopted. In the following, we propose ICRA, a greedy based algorithm to solve OMS problem which improves Mamcra but optimality of the computed solutions is not guaranteed.

4. INCREMENTAL CYCLE REDUCTION

Solving the OMS problem consists on finding a set of links from the multicast subgraph G'_S to be omitted while fulfilling the members requirements. In this section, cycle reduction will be considered in an incremental and greedy manner by adding the feasible paths of S one after the other to the multicast routing structure M already examined.

4.1. Exploring issues raised by adding a new path to the routing structure

Contrarily to Mamcra algorithm, in our proposition the reduction is analyzed and executed regarding all the existing paths, with which the new path forms cycles. So, the reduction procedure is realized with a unique scan of the new path from the farthest common node to the one which is closest to the source. According to the presented reductions and to simplify the analysis, the paths in M and the new path denoted P_n are decomposed in segments. Each segment r is characterized by two informations: the weight $\vec{w}(r)$ and the number of links (the hop count) $h(r)$. The new added path P_n forms a cycle in M , if it exists a set of paths P_o^i in M having common nodes with P_n other then the source node. Let $X = \{x_i, i = 1, \dots\}$ be this set of common nodes between P_n and the paths already in M (other then the source node). To simplify the analysis of the elimination of a given cycle $P_n(s, x_i)P_o^i(x_i, s)$, we propose some contractions.

- All the segments of the new path following the node x_i can be contracted in a single segment $r_{P_n^{i+1}}$ and the weight vector and hop count metric of this contracted segment correspond to the sum of weight vectors and hop counts of the concerned segments.
- The sub-path of P_o^i rooted at x_i can be contracted in a single segment $r_{P_o^{i+1}}$ to analyze the possible reductions. The weight represent the weight of the critical path from x_i to the farthest destination on P_o^i and the hop count corresponds to the sum of hop counts in the sub-path.

Figure 2 illustrates the contraction in a simple case.

A new added path P_n may have common nodes with many old paths in M . To examine cycle elimination, we study here three cases. The first case occurs when the examined common node x_i is in P_n and in only one old path. The seconde case is when x_i is in P_n and in at least two other old paths. The last case is when 2 common nodes in P_n are within an old paths in inverted order. We will study each case separately and we will propose an approach involving all these cases.

Figure 2: Contraction of components in the general reduction case

Case 1: New Path P_n has a common node with only 1 old path P_o

This case corresponds to the more frequent case. Without loss of generality, let us suppose that the two paths P_n and P_o form a cycle between the source s and another common node x_i . Let n_o and n_n be the last significant node before x_i , respectively, in the old path P_o and in the new one P_n . Eliminating the cycle $P_n(s, x_i)P_o^i(s, x_i)$ can be achieved by deleting one of the two segments $P_n(n_n, x_i)$ or $P_o^i(n_o, x_i)$ which precedes x_i respectively on P_n and on P_o^i . The segment $P_n(n_n, x_i)$ can be eliminated, if

$$\vec{w}(P_o^i(s, x_i)) + \vec{w}(r_{P_n^{i+1}}) \leq^d \vec{L}$$

Similarly, the segment $P_o^i(n_o, x_i)$ can be eliminated, if

$$\vec{w}(P_n(s, x_i)) + \vec{w}(r_{P_o^{i+1}}) \leq^d \vec{L}$$

If the two conditions are verified simultaneously and the two segments are candidates to the elimination, then the segment corresponding to a higher gain (for example, the segment with higher hop count) should be deleted. To illustrate such a case, let's consider the example presented in Figure 3. If P_o should be rerouted through P_n to eliminate cycle in x_i , only the segment preceding x_i can be removed to ensure forwarding data to nodes n_1 , n_2 and n_3 . We recall that these nodes can not be disconnected as they are whether destination nodes or branching nodes which forward data to other destination nodes. In the other case, eliminating the whole sub-path (s, x_i) from P_n is allowed even if nodes such as n_4 and n_5 exist. In fact, if these nodes are destination nodes, paths for these destinations exist in G_S as removing such paths from the initial graph G_S occurs only after adding the new path P_n .

Figure 3: Cycle reduction when sub-paths contains intermediate destination(s)

Case 2: New Path P_n has a common node with a set of old paths

In a more general case, a cycle involves more than one old path. This case occurs when the common node is in the newly added path and in more than one old paths as it is depicted in Figure 4.

Figure 4: Contraction of components in the general reduction case

In this case, cycle reduction can be achieved if :

- rerouting the newly added path through one of the existing paths is possible due to a common node x_i : the sub-path $P_n(s, x_i)$ is eliminated and one path from the old paths can be used to forward data to the segment $r_{P_n}^{i+1}$.
- rerouting some of the old paths through the newly added path. In that case, some of the segments preceding x_i and belonging to a set of old paths can be omitted. In the best case, all the old paths can be rerouted through P_n .

We should notice that the already existing cycles formed by the old paths can not be eliminated without the adding of the new path. In fact, when adding these paths all elimination procedures have been tested and no elimination was possible as the cycles persist.

So, to achieve cycle reduction, whole sub-paths or a set of segments can be candidate to elimination. When more than one candidate to elimination exists, the candidate that optimizes an objective function g is deleted. Basically, for g specification, there are two extreme possibilities: minimize the number of links used for data forwarding or maximize the quality at the destination nodes. In order to minimize the number of used links, we propose the elimination of the segment with larger hop count. In the case of equality, the QoS criteria is discussed and the solution which guarantees the smaller critical value for QoS criteria is chosen. This decision can be seen as a multi-constrained optimization problem in the plan hop distance and critical length value. Other selection criteria can be also proposed but this is out of scope of our analysis.

Case 3: New Path P_n has 2 common nodes with at least one old path in inverted order

Another issue that must be not neglected is the order of considered common nodes in the old and in the new path. In some rare but possible cases, the order of common nodes in the old paths in M can be different from their order in the new path. It can be examined by detecting the position of the node x_i in the two paths compared to the preceding common nodes. In fact, two nodes may belong to two paths but they are used in inverted order. That is the case of nodes a and b in Figure 5: a and b are two common nodes of P_1 and P_2 but they are used from b to a in P_1 and from a to b in P_2 . So, when considering 2 common nodes x_i and x_{i-1} , the order adopted here is the order of these nodes in the new path P_n .

Elimination strategy must involve all the above presented cases. In the following, we propose a new algorithm enhancing Mamcra mechanisms and considering all topology cases.

4.2. Improved Cycle Reduction Algorithm

ICRA algorithm supposes that a feasible path is known from the source node to each destination of the multicast group. The set S of these paths forms the graph G_S which can contain cycles. ICRA algorithms aims to eliminate redundancies from this set of QoS constrained paths. The set of paths, result of the elimination procedure, must contain a feasible path for all destination nodes for which a feasible path exists in the initial set. In other words, the

Figure 5: Decomposition of a set of paths in segments

goal is to decrease the number of links in the resulting structure respecting the given constraints: find an approached solution of the MCMWM problem.

As for Mamcra algorithm, the inputs of the algorithm are the following. We suppose that the QoS constraints vector $\vec{L}$ is given and the m dimensional link value vectors are known for each link. Each path $P_k \in S$ is a feasible path from the source s to a destination $d_k \in D$ for $k = 1, \dots, d$.

The output of our algorithm is a set M containing paths routed at s . Each path covers a sub-set of destination nodes and the union of the paths covers the totality of destinations. The result contain only feasible paths from the source node to the covered destinations as it will be done by the following algorithm.

Similarly to Mamcra, the algorithm ICRA works in a greedy manner and proceeds incrementally in two phases: first, selecting a path form S then attempting to add it to the final routing structure while removing as much redundancies as possible. These phases are detailed in the following.

4.2.1. Path selection

To process the input set S of paths, we propose the same procedure of selection as it is defined in Mamcra. For each path P_k , the end to end QoS length $l(P_k)$ and the destination nodes on the path are known. The selection of the next path to treat is based on the number of multicast members present in the path. Paths with large number of multicast members are examined at first. This implies that the selected path will cover as many destinations as possible. In the case of equality on this metric, the algorithm chooses the path with smallest QoS length. So, at a first time the set of paths grows with shortest paths and a short "kernel" is created to facilitate future path additions. If there is an equality on the two mentioned metrics, we propose to select randomly one of the equal paths. Even if the selection influences the result of the reduction, examining all possible orders is expensive that's why we propose the random selection.

4.2.2. Add a path to the set of multicast paths

The principal steps of path addition to the multicast sub-graph are as follows:

- At first, the common nodes $X = \{x_i, i = 1, \dots\}$ between P_n and the paths already in M are determined, but the source node s is excluded from X .
- If there is no intersection, then the path P_n is added as a new path to the set M . In this case, the other paths from the source to the destination nodes in P_n can be deleted from S if they exist.
- If the path P_n forms cycles with some paths in M , then examining redundancies is achieved in a greedy manner. We propose a cycle reduction procedure based on selecting from a set of determined candidate segments which candidate to eliminate.

Computing the set of segments candidate to elimination. In Mamcra algorithm, removing cycles is processed by examining cycles formed between the new added path and older paths one at a time. For exhaustive examination, older paths are selected one after the other using the adding order. In ICRA algorithm, the reduction is analyzed and executed regarding all the existing paths, with which the new path forms cycles. Common nodes are scanned from the farthest common node to the one which is closest to the source according to the order in the new path. The first

step of the cycle reduction procedure is to determine the set of segments candidate to the elimination. Investigation made in the previous section leads us to state the following reduction rules that must be used to determine segments candidate to the elimination.

Reduction Rule 1: A sub-path or segment s_{p^i} can be eliminated from a path P_k existing between s and d_k and it can be replaced by a sub-path s_{p^j} if the replacement results a continual path from s to d_k and if

$$\vec{w}(P_k) - \vec{w}(s_{p^i}) + \vec{w}(s_{p^j}) \leq^d \vec{L}$$

Reduction Rule 2: When a sub-path from source node of an old path in M is candidate to elimination, only the segment preceding the currently examined common node can be removed.

This last reduction rule is used to preserve intermediate significant nodes from deletion. In fact, when the old path is subject to elimination, only the segment preceding the common nodes can be omitted not to disconnect intermediate destination and significant nodes.

Selection of segments that should be eliminated. In the following, each candidate to elimination is assimilated to a set of segments. If there is more than one candidate to elimination, the one that optimizes an objective function g is chosen. In ICRA algorithm, we propose to define the function g as the simple sum of the hop numbers of the eliminated segments. This sum corresponds to the gain of the elimination and our objective is to maximize it.

Definition : Given a new path P_n and an old path P_o having a set of common nodes $X = \{x_i, i = 1, \dots\}$, the elimination gain induced by the removal of a set of segments r_P^i from P_n or P_o when examining the common node x_i is

$$g_P = \begin{cases} g_{P_n} = h(r_P^i), & \text{if } r_P^i \text{ is in } P_n; \\ g_{P_o} & \text{if } r_P^i \text{ is in } P_o \end{cases}$$

and

$$g_{P_o} = \begin{cases} h(r_P^i), & \text{if } x_i \text{ and } x_{i-1} \text{ are in the same order in } P_o \text{ and } P_n; \\ h(r_P^i) + h(r_P^{i-1}) & \text{if } x_i \text{ and } x_{i-1} \text{ are in inverted order in } P_o \text{ and } P_n. \end{cases} \quad (7)$$

where r_P^j is the segment or the set of segments preceding x_j .

It is important to notice, that in (7) when r_P^j corresponds to a set of segments (case of elimination of a set of segments from P_o), $h(r_P^j)$ corresponds to the sum of the number of hops of each segments of the considered set of segments.

To better understand the gain definition, let us consider Figure 6 where Case A shows that eliminating $r_{P_n}^i$ from P_n saves the use of $h(r_{P_n}^i)$ hops. In the same way, eliminating $r_{P_o}^i$ from P_o induces a gain of $h(r_{P_o}^i)$ hops, if the common nodes in P_n and in P_o have the same order in the two paths as it is in the Case B of the figure. Let us notice that the elimination extends on the whole sub-path in the new path, from the source to the last common node where the two reduction rules are respected, while the elimination concerns only the last segment (if the condition is verified for this elimination) on the old path. The reason is simple: the intermediate destination nodes on the new path can be reached using their own existing feasible paths in S . These paths will be examined later in the procedure. Regarding the intermediate destination present on the old path, there is no more path in S to reach them. So, only the last segment before the examined common node candidates to be eliminated.

The last case occurs when a segment $r_{P_o}^i$ of the old path P_o is candidate to elimination and when P_o and P_n have at least two common nodes (for example: x_i and x_{i-1}) in inverted order as depicted in Case C of Figure (6). Elimination of segments $r_{P_o}^i$ implies the elimination of the last segment $r_{P_o}^{i-1}$ preceding the node x_{i-1} as traffic to node d_{p_o} will be rerouted through P_n . In this case, chosen to eliminate the segments on the old path saves $h(r_{P_o}^i) + h(r_{P_o}^{i-1})$ hops. In general case, several common nodes can be in inverse order on the two paths. It is easy to show that the elimination of the last segment on the old path at a common node x_i implicates the elimination of all the last segments which are before the other common nodes going from x_i to the last destination on the old path.

Figure 6: Computing gain associated to segment elimination

Finally, let us consider the general case where the examined common node x_i belongs to a set of old path $PO = P_o^i, i = 1, \dots$. This set can be partitioned in 2 sets PO_{in} and PO_{out} . PO_{in} contains old paths having common nodes in the same order as in the new path. Unlike this, PO_{out} contains paths having some common nodes in inverted order compared to their order in P_n . The gain associated to the different elimination possibilities can be computed as follows:

- If rerouting P_n through one of the existing paths in PO is feasible, then the gain corresponds to the total length of the sub-path of P_n from the source to x_i and it is given by

$$g_{P_n} = h(r_{P_n,i})$$

- If rerouting a set of old paths PR through P_n which is conserved, the gain is

$$g_{PR} = \sum_{P \in (PR \cap PO_{in})} g_P + \sum_{P \in (PR \cap PO_{out})} g_P$$

Reduction rule 3: To decide whether candidate to eliminate, compute the gain g associated to each candidate then chose the elimination that maximizes the gain.

4.2.3. Formal description of ICRA algorithm

Using reduction rules, definitions and the algorithm given previously, ICRA mechanisms can be summarized as follows. When adding a new path to the multicast graph M , ICRA algorithm determines the set $X = \{x_i, x_i \neq s, i = 1, \dots\}$ of common node that are in the new path and in other old paths. These common nodes are sorted in ascendant order of their distance from the source node in the new path. They are scanned node by node from the farthest one. Examining a node consists on determining the sub-paths and the segments that can be removed to eliminate redundancies in this node. Sub-paths are candidate to elimination when reduction rule 1 is respected to fulfill destination requirements. If reduction rule 2 is not respected, the examined sub-path must be reduced to the

last segment preceding the examined node, as it is described above, to avoid the disconnection of intermediate destination nodes. When the set of segments candidate to elimination is not empty, ICRA chooses the optimal set of segments. Paths subject of segment removals must be updated to ensure forwarding multicast traffic to the corresponding destination nodes.

5. CONCLUSION

In this paper, we adopt an incremental approach to solve multi-constrained multicast routing. It is proposed to go beyond Mamcra problems that we presented above. After showing how Mamcra fails in solving some possible cases, we describe how ICRA tackles the optimization of the multicast subgraph by taking into account different topology cases and by examining different elimination possibilities. We show in this paper that ICRA overcomes the different problems that Mamcra does not handle. For the general case which is handled by both ICRA and Mamcra, we plan to perform simulations to compare performances of ICRA and Mamcra and this issue will be investigated in our future work. And as ICRA is also a greedy method, it is interesting to see how metaheuristics can be adopted to solve this problem and how it behaves compared to incremental solutions such as Mamcra and ICRA. This issue will be investigated in further work.

Bibliography

- [1] S. Chen, K. Nahrstedt, An Overview of Quality-of-Service Routing for the Next Generation High-Speed Networks: Problems and Solutions, in *IEEE Network*, Special Issue on Transmission and Distribution of digital Video, pp. 64-79, (1998).
- [2] M. Faloutsos, A. Banerjee, R. Pankaj, QoSMIC: Quality of Service sensitive Multicast Internet protoCol, in *ACM SIGCOMM*, pp. 144-153, (1998).
- [3] L. Henrique, M. K. Costa, S. Fdida, O. Carlos, M. B. Duarte, Hop-by-Hop Multicast Routing Protocol, In *ACM SIGCOMM*, pp. 249-259, (2001).
- [4] F. Hwang, D. Richards, Steiner Tree Problem, *Networks*, vol. 22. no. 1, pp.55-89, (1992).
- [5] X. Jia, Y. Zhang, N. Pissinou, K. Makki, A distributed multicast routing protocol for real-time multicast applications, in *Computer Networks* 31, pp.101-110, (1999).
- [6] F. Kuipers, P. V. Mieghem, Mamcra: A Constrained-Based Multicast Routing Algorithm, In *Computer Communications*, Vol. 25, No. 8, pp. 801-810, (2002).
- [7] P. Van Mieghem (ed.), F.A. Kuipers, T. Korkmaz, M. Krunz, M. Curado, E. Monteiro, X. Masip-Bruin, J. Solé-Pareta, S. Sánchez-López, Quality of Service Routing, Chapter 3 in *Quality of Future Internet Services*, EU-COST 263 Final Report, edited by Smirnov et al. in *Springer LNCS 2856*, pp. 80-117, (2003).
- [8] P. Van Mieghem, H. De Neve, F.A. Kuipers, Hop-by-Hop Quality of Service Routing, *Computer Networks*, Vol. 37, No. 3-4, pp. 407-423, (2001).
- [9] B. Wang, J. C. Hou, Multicast Routing and its QoS Extension: Problems, Algorithms and Protocols, in *IEEE Network*, pp. 22-36, 2000.
- [10] N. Ben Ali, M. Molnar and A. Belghith, Multi-constrained QoS Multicast Routing Optimization, Technical Report 1882, Irisa, february 2008.