Int. J. Electron. Commun. (AEÜ) 68 (2014) 561–564

Contents lists available at ScienceDirect

International Journal of Electronics and Communications (AEÜ)

journal homepage: www.elsevier.com/locate/aeue

Rendezvous point based approach to the multi-constrained multicast routing problem

Krzysztof Stachowiak*, Piotr Zwierzykowski

Communication and Computer Networks, Faculty of Electronics and Telecommunications, Poznan University of Technology, Poland

ARTICLE INFO

Article history:
Received 18 March 2013
Received in revised form
23 November 2013
Accepted 6 January 2014

Keywords:
Quality of Service
Multi-constrained routing

ABSTRACT

The Quality of Service (*QoS*) routing requires a special approach to graph algorithms modeling. One of the mathematical concepts that reflects this class of problems is the multi-constrained minimum Steiner tree problem (*MCMST*). In this article, the RDP (named after the concept of the *RenDezvouz Point*), a novel algorithm for solving the MCMST problem is presented.

1. Introduction

The MCMST concept can be used to reflect Quality of Service (*QoS*) routing problems in the group communication [1,2]. It reflects finding multicast communication trees that satisfy multiple constraints with regard to some of the link properties, while minimizing the cost associated with the utilization of the tree's resources.

There are different approaches to the multi-criteria optimization of routing problems, but the class of the multi-constrained algorithms is particularly suitable for this task. On the one hand, it enables the optimization against multiple criteria, but on the other hand it does not require minimizing all the metric associated with the results. This kind of a mathematical problems can be handled with some specialized techniques that are worth analysing within the context of the QoS routing because they may offer some simplifications with regard to the computational complexity and, at the same time, can solve problems in a model that reflects the modern routing problems. The complexity of the multicast optimization is high and therefore such a compromise solution is an attractive subject for deeper studies.

The article is divided into the following parts: Section 2 contains a brief overview of the past research in the MCMST optimization. In Section 3, the mathematical model of the network and the considered problem is presented. The proposed algorithm is introduced in detail in Section 4, and the simulation procedure and the results are discussed in Section 5. Section 6 concludes the article.

* Corresponding author. Tel.: +48 607247426.
E-mail address: krzysiek.stachowiak@gmail.com (K. Stachowiak).

1434-8411/$ – see front matter © 2014 Elsevier GmbH. All rights reserved.
http://dx.doi.org/10.1016/j.aeue.2014.01.002

2. Related work

The optimization of the multicast routing for more than a single criterion has been proven to be the $\mathcal{NP}$-complete [2]. Some of the earlier research has introduced several interesting solutions to this problem, for both single and multiple constraints, e.g. [3–6].

2.1. Two-criterial optimization

In [3] two complementary techniques based on the tabu search are presented: the short- and long-term variants. Both solutions consist in finding a feasible solution to the problem and improving its quality incrementally by traversing the multi-dimensional solution space around it. The difference between the two approaches lies in the complexity of the model of excluding the already visited solution space areas from the further analysis. In [4] a relatively simple, preprocessing approach is presented. The initial solution is built with an extended Prim's algorithm, that guarantees that the constraint has not been broken, but does not guarantee finding a complete solution. In the following steps of the algorithm the initial solution is refined based on the information gathered in the first phase.

2.2. Multicriterial optimization

An interesting variation of the *k-shortest* paths algorithm is presented in [5]. This algorithm utilizes the potential of the *k*-shortest paths algorithm to provide solutions of very high quality. By defining the optimization criteria properly, the authors managed to provide a guarantee of finding the optimal solution at the cost of the computation complexity. Another approach to the multi-criteria multicast optimization is the genetic algorithms that can be represented by Li et al. [6]. The genetic approach presents promising characteristics for solving problems of high complexity. In our view, however, the computational complexity and the necessity of fine-tuning makes them not the best suited for the routing problems, which allows for utilization of simpler and more specific techniques. Wan and Zhang [7] presents another variation of a genetic approach. The authors propose an immune system-based algorithm that is similar to the genetic approaches. The proposed algorithm optimizes a fixed number of criteria, but there are four of them, hence the algorithm's quality is comparable to those of the algorithms optimizing multiple criteria in general.

2.3. The result model

An interesting and important discussion of the multicast optimization result model has been conducted in [8]. In many cases the constrained multicast problem has an optimal solution that is neither a tree nor a set of paths. The tree is in some cases inappropriate as the optimal solutions may include loops. The set of paths cannot be considered in many cases, because some of the edges are shared between the routes to particular destinations and should not be considered multiple times in the evaluation of the solution. In our studies we use a similar approach that makes it possible to avoid the problems mentioned above.

3. Mathematical model

The network is modelled by an undirected graph $G(E, V)$, where V is a finite set of nodes and $E \subseteq (u, v) : u, v \in V$ is a set of edges that represent point-to-point links. Each of the edges is assigned M metrics, given by the functions: $(m_i : E \to \mathbb{R}^+, i = 0, 1, \ldots, M-1)$, that reflect additive costs of the according edges.

A path $p(s, d)$ from the node s to the node d, where $s, d \in V$, is defined as a sequence of non-repeated nodes $v_1, v_2, \ldots, v_k \in V$ such that for each $1 \leq i < k$ an edge $(v_i, v_{i+1}) \in E$ and $v_1 \equiv s$, $v_k \equiv d$. We define the accumulated metrics for the paths, so that the cost of a path p, based on the edges that form it $e \in p \subseteq E$, the $i-th$ is defined: $m_i(p) = \sum_{e \in p} m_i(e)$.

A rooted multicast tree $t(s, d_1, d_2, \ldots)$, connecting the source node $s \in V$ with the multiple destinations $d_1, d_2, \ldots \in D \subseteq V$, is defined as a tree in G, of which the only leaf nodes are the ones from the set $\{s\} \cup D$, with one of them, the node s, arbitrarily selected as the root. We define the accumulated cost of a tree t analogously to the accumulated path's cost as: $m_i(t) = \sum_{e \in t} m_i(e)$. Let $T(s, d_1, d_2, \ldots)$ define the set of all the trees spanning the nodes from the set $\{s\} \cup D$.

For a tree t we define a path $p_t(s, d_i)$ that is a path connecting the nodes s and d_i within the given tree.

We define the constraints set C as: $(c_i \in \mathbb{R}^+, i = 1, 2, \ldots, M-1)$. The constrains are associated with the metrics of the same indices.

The MCMST problem is defined as finding the tree t^* spanning the source node s and the destination nodes D that fulfils the following conditions:

(1) $\forall t \in T(s, d_1, d_2, \ldots) : m_0(t^*) \leq m_0(t)$,
(2) $\forall d_i \in D, c_j \in C : m_j(p_t(s, d_i)) \leq c_j$.

4. The RDP algorithm

4.1. Simulation semantics

In order to make the algorithm description clearer we will start with a rephrasing of Dijkstra's algorithm [9] into the simulation semantics in the similar fashion to what has been presented in [10]. In such a case, the algorithm is treated as a simulation of the propagation of a virtual signal through a graph. The selection of the nearest node is rephrased as the selection of the soonest event; the visit at a given node is interpreted as the handling of the arrival of the signal at a given node; the labelling of the neighbor nodes becomes the scheduling of the arrival of the signal at given nodes. The simulation time is an equivalent of the cost of reaching a given node, and the propagation times are determined based on the edges' costs.

4.2. Multi-source Dijkstra's algorithm

The RDP algorithm is based on simulating the multiple Dijkstra's algorithm instances (called the *convergence processes*) concurrently. The processes are conducted independently, except for the order in which they progress. The information about which nodes have been visited by particular processes is stored by the algorithm core. The selection of the next process to proceed is made based on the assumption that the labels assigned by the processes to the nodes may be interpreted as the time that has passed in a given process. Once a given node has been visited by all of the convergence processes, it may be considered the middle of a multicast tree – the *rendezvous point* or in short, the *RDP*. An attempt may be made at building the result starting from the RDP by using the predecessor maps of the convergence processes. There are two particular implementations of this general concept.

4.2.1. Quasi-exact approach

In this approach, presented in Algorithm 1, multiple RDP's are considered. The convergence processes only optimize the metric m_0, which guarantees that the RDPs will be visited in the order of the increasing sum of the individual m_0 metrics. Each of the RDPs is successively tested for fulfilling the constraints with regard to the metrics $m_1, m_2, \ldots, m_{M-1}$. Once the one that fulfils the constraints is found, the tree built from this RDP is considered the result.

Algorithm 1. RDP_QE

```
procedure RDP_QE(g, s, D, C)
  for d ∈ {s} ∪ D do
    init_cost_optimizing_process(d, g)
  end for
  while true do
    p ← select_soonest_process()
    n ← handle_soonest_event(p)
    visit_counter(n) := visit_counter(n) + 1
    if visit_counter(n) = |D| + 1 then
      t ← build_tree_from(n, g)
      if fulfills_constraints(t, C) then
        Return t
      end if
    end if
    if all_nodes_visited() then
      Return failure
    end if
  end while
end procedure
```

In a way, this procedure may be interpreted as scanning the space of the solutions, only considering some of them, but considering them in the order of the increasing cumulative m_0 metric. The name "Quasi-exact" has therefore been chosen because if all of the solutions have been scanned in this manner the exact solution could be obtained, i.e. the cheapest result that fulfils the constraints.

4.2.2. Heuristic approach

In the second variant, presented in Algorithm 2, only one RDP is considered, but in order to achieve a high quality of the result the convergence processes utilize a heuristic aggregation of all the metrics. The aggregation is based on the non-linear technique that has been earlier presented in [11].

For the constraints set $C = \{c_1, c_2, \ldots\}$, the aggregated tree metric is defined as follows: $m_{aggr}(t) = \max\{(m_1(t)/c_1), (m_2(t)/c_2), \ldots\}$.