Available online at www.sciencedirect.com
ScienceDirect
Discrete Applied Mathematics 156 (2008) 3376–3399
DISCRETE APPLIED MATHEMATICS
www.elsevier.com/locate/dam

Integer version of the multipath flow network synthesis problem[☆]

S.N. Kabadi[a,*], R. Chandrasekaran[b], K.P.K. Nair[a], Y.P. Aneja[c]

[a] Faculty of Business Administration, University of New Brunswick, Fredericton, N.B., Canada E3B5A3
[b] Department of Computer Science, University of Texas at Dallas, Richardson, Texas, USA
[c] Odette School of Business, University of Windsor, Windsor, Ontario, Canada N9B3P4

Received 28 April 2004; received in revised form 15 November 2006; accepted 11 January 2008
Available online 21 February 2008

Abstract

We consider the following integer multipath flow network synthesis problem. We are given two positive integers q, n, $(1 < q < n)$, and a non-negative, integer, symmetric, $n \times n$ matrix R, each non-diagonal element r_{ij} of which represents the minimum requirement of q-path flow value between nodes i and j in an undirected network on the node set $N = \{1, 2, \ldots, n\}$. We want to construct a simple, undirected network $G = [N, E]$ with integer edge capacities $\{u_e : e \in E\}$ such that each of these flow requirements can be realized (one at a time) and the sum of all the edge capacities is minimum. We present an $O(n^3)$ combinatorial algorithm for the problem and we show that the problem has integer rounding property.

Keywords: Networks; Multipath flows; Network synthesis; Combinatorial algorithm; Strongly polynomial algorithm

1. Introduction

Gomory and Hu [10] and Mayeda [24] have considered the following continuous network synthesis problem, an elegant description of which can be found in [7]:

Given an integer $n > 1$ and a non-negative, symmetric, $n \times n$ matrix R, each non-diagonal element r_{ij} of which represents the minimum required flow value between nodes i and j in an undirected network on the node set $N = \{1, 2, \ldots, n\}$, construct an undirected network $G = [N, E]$ on node set N with non-negative, real-valued edge capacities $\{u_e : e \in E\}$, such that (i) all the flow requirements are met one at a time, (that is, for any $i, j \in N, i \neq j$, the maximum flow value in G from i to j is at least r_{ij}), and (ii) $\sum\{u_e : e \in E\}$ is minimum.

This problem, (and its generalization to the case of synthesizing a network with minimum weighted sum of edge capacities), has a polynomial size linear programming formulation [11], which can be solved in strongly polynomial time using Tardos' algorithm [29]. However, the Tardos' algorithm is known to be quite slow in practice. There is therefore a need for a special purpose, combinatorial, strongly polynomial algorithm which provides insight into the combinatorial structure of the problem and also works efficiently in practice.

☆ Research supported in part by research grants from the Natural Sciences and Engineering Research Council of Canada.
* Corresponding author. Tel.: +1 506 453 4869/3561.
E-mail address: kabadi@unb.ca (S.N. Kabadi).

0166-218X/$ - see front matter
doi:10.1016/j.dam.2008.01.015

In both [10,24], efficient combinatorial, strongly polynomial algorithms are presented for (the unweighted case of) the problem. The Gomory–Hu algorithm in [10] has a computational complexity of $O(n^2)$, and the synthesized network contains $O(n)$ edges. Also, when all the elements of the matrix R are integers, the edge capacities in the synthesized network are multiples of half. Alternate, combinatorial algorithms for the problem are presented in [12, 28].

In [4,26], an integer version of the problem of [10,24] is considered. Here, all the elements of the input matrix R and all the edge capacities of the synthesized network are required to be integers. In [4] and independently in [26], combinatorial algorithms of computational complexity $O(n^2)$ are presented for the problem, and it is shown that whenever $\max\{r_{ij} : j \in N - \{i\}\} > 1\ \forall i \in N$, the problem has *integer rounding property*, (that is, the difference between the sum of edge capacities in the optimal networks for the integer and continuous versions of the problem is less than 1). (As pointed out in [27], the algorithm in [4] is lacunary and does not apply to some cases.) Alternate algorithms for the problem are given in [19,27]. A strongly polynomial algorithm is given in [8] for a generalization of the problem to one in which we want to increase the edge capacities of a given network so as to meet the given flow requirements, such that the sum of additional capacities is minimum.

For the weighted case of the problem, strongly polynomial combinatorial algorithms for the continuous and integer versions are known only for the special case in which the network is restricted to be a cycle [15,16]. Results on some generalizations of the unweighted case of the problem to the case of 2-commodity flows are reported in [14,17], and to the case of hop-constrained flows are reported in [9,18].

The following concept of multipath flows was introduced and studied by Kishimoto [20], Kishimoto and Takeuchi [21,22] and Kishimoto, Takeuchi and Kishi [23] for improving the reliability of communication networks where edges are subject to failure:

Given an undirected network $G = [N, E]$, a source–sink pair (s, t) of nodes in N, a non-negative number u_e representing the capacity of edge e for each $e \in E$ and a positive integer q, a "q-path set" from s to t in G is a set of q edge-disjoint s–t paths in G. A q-path flow from s to t in G is an allocation of non-negative weights to the q-path sets from s to t such that for each edge $e \in E$, the corresponding flow on e, (i.e., the sum of weights assigned to the q-path sets containing the edge e), is no more than its capacity u_e. The flow value of a q-path flow is the sum of weights assigned to all the q-path sets.

Kishimoto [20] considered the problem of finding a q-path flow of maximum flow value for a given source–sink pair (s, t) of nodes in a network. For any cut separating s and t, he defined q-capacity of the cut in a particular way and showed that a max-flow min-cut theorem holds for q-path flows as well. He also provided a strongly polynomial algorithm for finding such a flow. Related results appear in [1,5,6]. In [2], the concept of q-path flow and many results on it are extended to the case of fractional values of $q > 1$.

In [3], the following continuous, multipath flow network synthesis problem is considered. We are given two positive integers q, n, $(q < n)$, and a non-negative, symmetric, $n \times n$ matrix R, each non-diagonal element r_{ij} of which represents the minimum requirement of q-path flow value between nodes i and j in an undirected network on the node set $N = \{1, 2, \ldots, n\}$. We want to construct a simple, undirected network $G = [N, E]$ with non-negative edge capacities $\{u_e : e \in E\}$ so that each of these requirements can be realized (one at a time) and the sum of all edge capacities is minimum.

When $q = 1$, this problem is precisely the classical network synthesis problem considered in [10,24]. In [3], an $O(n^3)$ combinatorial algorithm is presented for the case of arbitrary positive integer q, generalizing the results in [10, 24]. In [2], this result is extended to the case of fractional values of $q > 1$.

In this paper, we consider the integer version of the multipath flow network synthesis problem. That is, for integer-valued data, we require the values of edge capacities $\{u_e : e \in E\}$ of the output network to be integers. We present a strongly polynomial, combinatorial algorithm for arbitrary, positive integer $q > 1$ and we show that the problem has integer rounding property. That is, the difference between the optimal objective function values of the integer and the continuous versions of the problem is less than one.

After presenting in Section 2 notations and some basic definitions and results, we give in Section 3 an algorithm for the continuous version of the problem that produces an optimal solution to the problem such that when the elements of matrix R are integers, all the edge capacities of the output network are multiples of half. Finally, in Section 4 we further modify this algorithm to one that produces an optimal solution to the integer version of the problem.