Approximation Algorithms for Finding a Maximum-Weight Spanning Connected Subgraph with given Vertex Degrees *

Alexey E. Baburin and Edward Kh. Gimadi

Sobolev Institute of Mathematics, prospekt Akademika Koptyuga 4, 630090 Novosibirsk, Russia

Abstract. In the paper a problem of finding a maximum-weight spanning connected subgraph with given vertex degrees is considered. The problem is MAX SNP-hard, because it is a generalization of a well-known Traveling Salesman Problem. Approximation algorithms are constructed for deterministic and random instances. Performance bounds of these algorithms are presented.

1 Introduction

Let $G(V, E)$ be a complete n-vertex undirected graph without loops with a non-negative weight function w of edges. There are known integers d_i $(i = 1, \ldots, n), 1 \le d_i \le n$.

In [6] the problem of finding a realization of a set of integers as degrees of the vertices in a subgraph G' of G is formulated. (This set is called a graphical partition of a number p, where $p = \sum_{i=1}^{n} d_i$). It is clear that for every such realization the number p is even and $d_i \le n-1$ for each $i = 1, \ldots, n$. These conditions are not sufficient. For example, a set $D = (3, 3, 3, 1)$ is not a graphical partition. A constructive criterion of realizability of a set of integers is implied from the following statement presented in [7]:

Theorem 1. *Let $n > d_1 \ge d_2 \ge \ldots \ge d_n$ and $\sum_{i=1}^{n} d_i$ is even. Then a partition $D = (d_1, \ldots, d_n)$ is graphical if and only if a modified partition $D' = (d_2 - 1, d_3 - 1, \ldots, d_{d_1+1} - 1, d_{d_1+2}, \ldots, d_n)$ is graphical.*

An optimization appearance of the problem was described in [2] by means of reduction to the problem of finding a maximum-weight matching.

The problem of finding a maximum-weight spanning connected subgraph with given vertex degrees appeared in [4] and was denoted by $CSSDP$. Denote by $W^*(G)$ the weight of an optimal solution for an instance G of the problem, by $W_A(G)$ – the weight of a solution presented by approximation algorithm $\bar{A}$ for the same instance.

* This research was supported by the Russian Foundation for Basic Research (grant 02-01-01153), program of supporting of leading science schools of Russia (project "Nauchnaya Shkola - 313.2003.1"), and INTAS (grant 00-217)

$\Delta_A = \min_G \frac{W_A(G)}{W^*(G)}$ is called *a performance ratio* of algorithm A.

An approximation algorithm was presented in [4] for the metric case of $CSSDP$. The algorithm only worked for the case when all integers d_i are even. The performance ratio of the algorithm exceeded $1 - \frac{1}{d(d+1)}$, where $d = min\{d_i | i = 1, \ldots, n\}$. In this paper we present an approximation algorithm and show its performance ratios for different classes of the problem $CSSDP$.

An algorithm for random instances of the problem was also presented in [4]. We improve the performance estimates by presenting a modification of the algorithm and show its asymptotical optimality for a wider class of input instances.

Different applications of the problem and its variations appeared in [1].

2 Approximation algorithm A' for deterministic instances of $CSSDP$

Below we consider a complete n-vertex undirected graph $G = (V, E)$ without loops with a non-negative weight functions w of edges. Integers d_i $(i = 1, \ldots, n), 2 \leq d_i \leq n$ are given.

2.1 Algorithm A' description

We propose an algorithm A' for solving $CSSDP$ on the graph. First, it constructs an optimal solution for a relaxation of the problem (without the requirement of connectivity). Then the components of connectivity are patched into one component that is a feasible solution for the problem.

Stage 1. Using Gabow's algorithm [3] a spanning subgraph G' of G with given vertex degrees is found.

Stage 2. Components of connectivity $C_1, \ldots, C_\mu$ are found. The components are reordered so that the minimum edge of G' belongs to C_1. If $\mu = 1$ then the subgraph G' is the output of algorithm A', else the stage 3 is performed.

Stage 3. For every component C_i $(i = 1, \ldots, \mu)$ a set S_i of edges is formed. The set S_1 consists of all edges in C_1. For $i > 1$, S_i consists of all edges in C_i that do not form a component of 2-connectivity (that are not *bridges*). In every set S_i an edge $e_i = (u_i, v_i)$ of minimum weight is found.

Stage 4. Let $i = 1$. Put $p_{\mu+1} = q_1 = u_1$, $p_1 = v_1$.

Stage 5. If $w(q_i, u_{i+1}) \geq w(q_i, v_{i+1})$ put $q_{i+1} = v_{i+1}$, $p_{i+1} = u_{i+1}$. Otherwise put $q_{i+1} = u_{i+1}$, $p_{i+1} = v_{i+1}$.

Stage 6. Put $i = i + 1$. If $i < \mu - 1$ the stage 5 is performed, else the stage 7 is performed.

Stage 7. If $w(q_{\mu-1}, u_\mu) + w(u_\mu, p_{\mu+1}) \geq w(q_{\mu-1}, v_\mu) + w(v_\mu, p_{\mu+1})$ put $q_\mu = v_\mu$, $p_\mu = u_\mu$. Otherwise put $q_\mu = u_\mu$, $p_\mu = v_\mu$.

Stage 8. Edges $e_1, \ldots, e_\mu$ are removed from G'.

Stage 9. Edges $(q_1, p_2), (q_2, p_3), \ldots, (q_\mu, p_{\mu+1})$ are added to G'.