Computers and Operations Research 87 (2017) 83–97

Contents lists available at ScienceDirect

Computers and Operations Research

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

A variable neighborhood search and simulated annealing hybrid for the profile minimization problem

Gintaras Palubeckis*

Faculty of Informatics, Kaunas University of Technology, Studentu 50–408, 51368 Kaunas, Lithuania

ARTICLE INFO

Article history:
Received 10 January 2017
Revised 25 May 2017
Accepted 2 June 2017
Available online 3 June 2017

Keywords:
Combinatorial optimization
Matrix profile
Graph profile
Simulated annealing
Variable neighborhood search

ABSTRACT

Given an undirected simple graph G, the profile minimization problem (PMP) is to find an ordering of the vertices of the graph G such that the sum of the profiles of all its vertices is minimized. The profile of the vertex v in position i is defined as $\max\{0, i - h_v\}$, where h_v is the position of the leftmost vertex among all vertices adjacent to v in G. We propose an approach for the PMP, which combines a variable neighborhood search (VNS) scheme with the multi-start simulated annealing (MSA) technique. The solution delivered by MSA is submitted as input to the VNS component of the method. The VNS algorithm heavily relies on a fast insertion neighborhood exploration procedure. We show that the time complexity of this procedure is $O(n^2)$, where n is the order of G. We have found empirically that it is advantageous to give between 50 and 75% of the computation time to MSA and the rest to VNS. The results of the computational experiments demonstrate the superiority of our MSA-VNS algorithm over the current state-of-the-art metaheuristic approaches for the PMP. Using MSA-VNS, we improved the best known solutions for 50 well-recognized benchmark PMP instances in the literature. The source code implementing MSA-VNS is made publicly available as a benchmark for future comparisons.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

The *profile minimization problem* (PMP) is an important member of a wide set of combinatorial optimization problems on permutations. It can be stated as follows. Suppose that we are given an undirected simple graph $G = (V, E)$ with vertex set V and edge set E and a permutation of its vertices $p = (p(1), \ldots, p(n))$, where $n = |V|$ is the order of G and $p(i)$, $i \in \{1, \ldots, n\}$, is the vertex in the ith position of the permutation. We denote by $h_{p(i)}$ the leftmost position $j < i$ such that $(p(i), p(j)) \in E$. If no such j exists, then $h_{p(i)}$ is set to i. The *profile* of vertex $p(i)$ is defined as the difference between i and $h_{p(i)}$. With these notations, the PMP can be expressed as

$$\min_{p \in \Pi} F(p) = \sum_{i=1}^{n} (i - h_{p(i)}) \tag{1}$$

where Π is the set of all permutations of $V = \{1, \ldots, n\}$. The largest of the vertex profiles

$$\varphi(p) = \max_{1 \leqslant i \leqslant n} (i - h_{p(i)}) \tag{2}$$

is called the *bandwidth* of the graph G. An example of an instance of the PMP is shown in Figure 1. Historically, the PMP has received the greatest attention in the development of methods for solving sparse systems of linear equations of the form

$$Ax = b, \tag{3}$$

where $A = (a_{ij})$ is a symmetric $n \times n$ matrix, b is an n-vector, and x is the n-vector of unknowns. The system (3) is represented by the graph G in which a pair of vertices i and j are joined by an edge if and only if $a_{ij} = a_{ji} \neq 0$. The problem of solving (3) arises in various contexts in science and engineering, especially in situations where finite-element analysis is used. In many applications, the matrix A is sparse. It is well known that the direct methods for solving (3) perform better in terms of execution time when nonzero entries of the sparse matrix A are grouped around the main diagonal. A possible way to obtain such a matrix is to symmetrically permute the rows and columns of A using a good solution p to the PMP instance defined by A. For a thorough discussion on the applicability of profile minimization techniques to the solution of a system of linear equations, the reader is referred, for example, to the book of Tewarson (1973) and to the recent survey article by Davis et al. (2016). A number of other applications of the PMP have been identified in the literature. These include an approach by Xu et al. (2013) for sparse matrix-vector multiplication optimizations using graphics processing units. The au-

* Corresponding author.
E-mail address: gintaras.palubeckis@ktu.lt

http://dx.doi.org/10.1016/j.cor.2017.06.002
0305-0548/© 2017 Elsevier Ltd. All rights reserved.

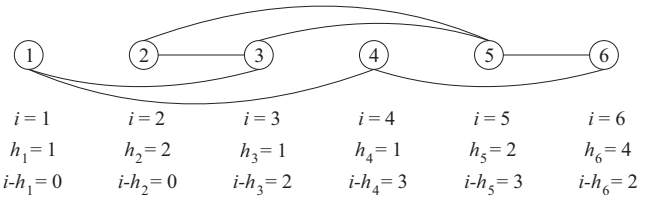

Fig. 1. Solution $p = (1, \ldots, 6)$ to an instance of the PMP in which $h_{p(1)} = h_1 = 1$, $h_2 = 2$, $h_3 = 1$, $h_4 = 1$, $h_5 = 2$, $h_6 = 4$ and $F(p) = 0 + 0 + 2 + 3 + 3 + 2 = 10$.

thors have shown that, with matrix bandwidth/profile minimization techniques, both cache usage enhancement and index compression can be enabled. Higham (2003) considered the PMP in the context of small world networks. He investigated the use of profile minimization algorithms for so-called small world reordering problem. Berry M. et al. (1996) dealt with the PMP in the information retrieval setting. They used the reordering algorithms to produce narrow-banded hypertext matrices for cluster identification. Mueller et al. (2007) reported on an application of profile reduction techniques in the graph visualization domain. The authors used graph ordering heuristics for visual analysis of data sets represented by visual similarity matrices. Meijer and de Pol (2015) suggested application of profile minimization algorithms in symbolic model checking. They addressed the problem of static ordering of variables in decision diagrams representing formulas. Bolanos M. et al. (2012) considered the PMP in relation to a new approach for computing entropy rate for undirected graphs. The matrix bandwidth and profile reduction techniques can be used to obtain lower bounds for graph entropy.

Lin and Yuan (1994) have shown the PMP to be NP-hard. Therefore, polynomial-time exact algorithms for this problem have been proposed only for restricted classes of graphs, e.g., wheels, complete bipartite graphs (Lin and Yuan, 1994), trees (Kuo and Chang, 1994), and triangulated triangles (Guan and Williams, 2003).

Because of the hardness of the problem, the majority of the research on the PMP has focused on designing heuristic and metaheuristic-based algorithms. One of the most widely studied ways of obtaining a satisfactory solution to the problem has been the use of constructive methods. They generate permutations by starting from an empty solution and gradually assigning the vertices of the graph to free positions of the permutation. One of the first constructive algorithms for the PMP was proposed by Cuthill and McKee (1969). Their algorithm generates the level structure rooted at a vertex of minimum degree. The vertices are assigned labels (in other words, the vertices are placed in the permutation) in increasing level order. George (1971) observed that reversing the obtained permutation yielded better solutions. In the literature, this approach is referred to as the Reverse Cuthill-McKee algorithm (RCM). Gibbs et al. (1976) presented another constructive technique for reducing the profile of a sparse matrix. This heuristic first finds the endpoints of a pseudo-diameter. Then it constructs a level structure and finally applies a fast numbering procedure similar to that used in RCM. At about the same time, Gibbs (1976) proposed a variation of this heuristic, which is currently known as the Gibbs-King algorithm. Sloan (1986) developed an algorithm consisting of two distinct stages. In the first stage, a pair of pseudo-peripheral vertices are located. They serve as start and end vertices for the second (numbering) stage of the method. Other vertices are chosen according to a priority function composed of two terms, one of which tries to reduce the increase in the profile and another takes into account the distance between the considered vertex and the end vertex. The algorithm was shown to be superior to previous methods (Sloan, 1986). Several enhancements to the Sloan algorithm have also been proposed (Duff et al., 1989; Kumfert and Pothen, 1997; Reid and Scott, 1999). Barnard et al. (1995) and Paulino et al. (1994a); 1994b) developed algorithms for the PMP which are based on spectral properties of the adjacency matrix of the graph. In these algorithms, the vertices are ordered according to the eigenvector corresponding to the smallest positive eigenvalue of the Laplacian matrix associated with the given graph. Hu and Scott (2001) presented a multilevel algorithm for profile reduction. Their approach combines a graph coarsening technique with the Sloan algorithm on the coarsest graph.

In the literature, there have been several local search or metaheuristic-based algorithms proposed for solving the PMP. The earliest such algorithm appeared in 1985 by Armstrong (1985). This algorithm is based on the simulated annealing (SA) paradigm. The move type that is applied in this implementation of SA is an interchange of two randomly selected vertices. The initial temperature is chosen such that almost any interchange is likely to be accepted. The algorithm incorporates a reheating mechanism which is triggered when the system has been cooled too quickly during the previous iterations of SA. Another SA algorithm for the PMP has been proposed by Lewis (1994). Like SA approach of Armstrong, it proceeds by performing random pairwise interchanges of vertices. The initial temperature depends on the profile of the starting solution. Throughout cooling, the temperature is lowered by a factor of 0.9 or 0.95. At each iteration, the algorithm computes the change in profile as a result of performing a move. Computational experience with this algorithm has been reported for several SA schedules. Hager (2002) developed two exchange methods for improving a given solution to the PMP. One of them, called down exchange, involves iteratively shifting a vertex to the right by one position at a time. Another exchange method, called up exchange, explores the solutions which can be obtained by shifting a vertex to the left in a similar manner. When combined together, these methods essentially can be categorized as a local search (LS) algorithm. The neighborhood of a solution in this approach consists of all those solutions that can be obtained by relocating a vertex from its current position in the permutation to a different one. Reid and Scott (2002) presented a significantly improved implementation of the Hager's exchange methods. An especially good performance, in terms of running time, was provided by the developed version of the up exchange algorithm. Kaveh and Sharafi (2012) proposed an algorithm for the PMP, which is based on the metaheuristic optimization method known as charged system search. Numerical experiments have demonstrated the effectiveness of this algorithm. Koohestani and Poli (2014) presented a hyper-heuristic approach based on genetic programming for evolving an enhanced version of the Sloan algorithm. The authors combined this version with the local search technique. Each step in their implementation of local search consists of swapping positions of two vertices. The approach was shown to outperform six existing algorithms for the PMP. More recently, Koohestani and Poli (2015) developed a genetic programming system for profile reduction of sparse matrices. They tested this method against several state-of-the-art heuristic techniques. Sánchez-Oro et al. (2015) proposed a scatter search algorithm for solving the PMP. They considered two solution improvement methods. One of them relies on performing pairwise interchanges of vertices. Another method involves the use of vertex insertion moves. The authors have found that the latter method was much faster than the former one. Sánchez-Oro et al. reported the results of extensive computational experiments. Their algorithm improved best known solutions for a number of benchmark PMP instances.

In some applications, e.g., in direct sparse matrix methods, it is important that a satisfactory solution to the PMP be provided very quickly. In this respect, metaheuristic-based methods are inferior to the fast heuristic techniques. However, considering direct matrix methods, Lewis (1994) as well as Reid and Scott (2002) have

Download English Version:

https://daneshyari.com/en/article/4958984

Download Persian Version:

https://daneshyari.com/article/4958984

Daneshyari.com