# PACE22_FVSP_HUST_SCP: A heuristic algorithm of directed feedback vertex set problem

**YuMing Du**
School of Computer Science and Technology, Huazhong University of Science & Technology, China

**QingYun Zhang**
School of Computer Science and Technology, Huazhong University of Science & Technology, China

**JunZhou Xu**
Huawei TCS Lab Shanghai, China

**ShunGen Zhang**
School of Computer Science and Technology, Huazhong University of Science & Technology, China

**Chao Liao**
Huawei TCS Lab Shanghai, China

**ZhiHuai Chen**
Huawei TCS Lab Shanghai, China

**ZhiBo Sun**
School of Computer Science and Technology, Huazhong University of Science & Technology, China

**ZhouXing Su**
School of Computer Science and Technology, Huazhong University of Science & Technology, China

**JunWen Ding**
School of Computer Science and Technology, Huazhong University of Science & Technology, China

**Chen Wu**
Huawei TCS Lab Shanghai, China

**PinYan Lu**
Huawei TCS Lab Shanghai, China

**ZhiPeng Lv**
School of Computer Science and Technology, Huazhong University of Science & Technology, China

## Abstract

A directed graph is formed by vertices and arcs from one vertex to another. The feedback vertex set problem (FVSP) consists in making a given directed graph acyclic by removing as few vertices as possible. In this write-up we outline the core techniques used in the heuristic feedback vertex set algorithm, submitted to the heuristic track of the 2022 PACE challenge.

**2012 ACM Subject Classification** Mathematics of computing → Graph algorithms

**Keywords and phrases** directed feedback vertex set, local search, simulated annealing, set covering

**Digital Object Identifier** 10.4230/LIPIcs.PACE.2022.1

**Supplementary Material** The source code is available on GitHub (https://github.com/1774150545/PACE-2022.git)
*Software (Source Code)*: `https://doi.org/10.5281/zenodo.6644066`

## 1 Problem Description

Let $G = (V, E)$ be a directed graph, where $V$ is the vertex set and $E \subseteq V \times V$ is the edge set. The feedback vertex set problem is to find a minimum subset $X \subseteq V$ such that, when all vertices of $X$ and their adjacent edges are deleted from $G$, the remainder subset $G' \subseteq G$

is acyclic. The subset $X$ is a feedback vertex set of the graph $G$. In the challenge of PACE 2022, for heuristic algorithms, the process was limited in 10 miniutes.

## 2 Reduction Rules

Reduction Rules play an important role in solving FVSP because it improves the efficiency of algorithm. A contraction (reduction) operation reduces the original graph while it preserves the information necessary for finding the minimum feedback set. We adopt eight reduction rules to eliminate some vertices and edges, and deduce that some vertices must be included in the optimal solution. Five contraction operations have been proposed in Levy and Low (1988) [4]. More recently, three new contraction operations have been presented in Lin and Jou (1999) [5]. Our implementation adopted the directed graph reduction method in [5]. The reductions simplifies the graph efficiently and does not take too long.

## 3 Simulated Annealing

In order to tackle FVSP, we propose a simulated annealing algorithm which is based on the local search framework. The earliest idea of simulated annealing algorithm (SA) was proposed by N. Metropolis et al. [1] in 1953. It generates an initial solution, than iteratively improves the incumbent solution by a local search procedure. At each iteration of the algorithm, it first evaluates the neighborhood of the current solution and determines whether to perform the current neighborhood move by $\Delta T$. If the current solution improves the best solution found so far, then the best solution is updated with the current. Finally, when the specified termination condition is met, the algorithm terminates and returns the best solution.

### 3.1 Initialization

This procedure is divided into two stages to generate an initial solution. In the first stage, the original problem is transformed into a vertex cover problem [2, 6] and then solves the problem by executing a heuristic algorithm under limited time. The execution time of the heuristic algorithm is determined according to the ratio $r_b$ of bidirectional edges. The greater the value of $r_b$, the greater the execution time.

$$r_b = \frac{2 * \sum_{i,j \in V} (e_{ij} * e_{ji})}{|E|} \tag{1}$$

In the second stage, the descent heuristic algorithm is used to further optimize the solution. Finally, the initial solution $X_0$ of the problem is obtained.

### 3.2 Neighborhood Structure and Evaluation

A directed graph with no directed cycles is named a directed acyclic graph (DAG). Every DAG has a topological ordering, i.e. an ordering of its vertices such that the starting-point of every arc occurs earlier in the ordering than the endpoint of the arc. Conversely, the existence of a topological ordering in a graph proves that this graph is acyclic.

In order to solve FVSP, we adopts a neighborhood move based on add and remove operations [3]. Specifically, an add operation consists of inserting a new vertex $j \notin X$ at some particular position into the sequence and, at the same time, a remove operation used to remove the vertices that would now violate the precedence constraint. Based on the incumbent solution $X$, performing the operations produces a new neighborhood solution $X'$.

To obtain the best neighboring solution and improve the incumbent solution $X$, our solver uses the insert policies that in-coming and out-coming neighbors. Specifically, a vertex $v$ can be inserted in the sequence at only two different positions, which are after its numbered in-coming neighbors, or just before its numbered out-going neighbors.

However, the simulated annealing algorithm used in [3] tends to end prematurely, so after reaching the local optimum, we increase the temperature of simulated annealing appropriately, and randomly delete some vertices in the topology sequence in order to jump out of the local optimum, so that the algorithm can rerun. The algorithm terminates until the limit time is reached. In addition, in the later period of the simulated annealing run, a cache acceleration method is also applied.

In particular, we lazily transform the FVSP to the set covering problem when the number of cycles is not too large, and use the set covering algorithm to further optimize current solution $X$.

## 3.3 Data structure

To efficiently update topological sequences, we used a data structure that can perform insert, delete and compare the order of two vertices in constant amortized time.

In our implementation, a method of labeling vertices is used. Each vertex in the sequence has a label that determines their order in the topological sequence. The labels are sparse enough, which is achieved by continuously dividing sufficiently large intervals, so that there will be spare labels between every two adjacent vertices in the topological sequence. When a vertex is inserted, it is assigned a label that is half the sum of the preceding and succeeding labels in topological order. If the spare label does not exist, the label of related vertices in the topology sequence needs to be adjusted.

## References


1 Dimitris Bertsimas and John Tsitsiklis. Simulated annealing. *Statistical science*, 8(1):10–15, 1993.

2 Yuning Chen and Jin-Kao Hao. Dynamic thresholding search for minimum vertex cover in massive sparse graphs. *Eng. Appl. Artif. Intell.*, 82:76–84, 2019.

3 Philippe Galinier, Eunice Adjarath Lemamou, and Mohamed Wassim Bouzidi. Applying local search to the feedback vertex set problem. *J. Heuristics*, 19(5):797–818, 2013.

4 Hanoch Levy and David W. Low. A contraction algorithm for finding small cycle cutsets. *J. Algorithms*, 9(4):470–493, 1988.

5 Hen-Ming Lin and Jing-Yang Jou. Computing minimum feedback vertex sets by contraction operations and its applications on CAD. In *Proceedings of the IEEE International Conference On Computer Design, VLSI in Computers and Processors, ICCD '99, Austin, Texas, USA, October 10-13, 1999*, page 364. IEEE Computer Society, 1999.

6 Changsheng Quan and Ping Guo. A local search method based on edge age strategy for minimum vertex cover problem in massive graphs. *Expert Syst. Appl.*, 182:115185, 2021.