Huawei_TCS_DFVS_Solver Description

Yuming Du
SMART, School of Computer Science and Technology, Huazhong University of Science & Technology, Wuhan, Hubei, China

Qingyun Zhang
SMART, School of Computer Science and Technology, Huazhong University of Science & Technology, Wuhan, Hubei, China

Junzhou Xu
Huawei TCS Lab, Shanghai, China

Shungen Zhang
SMART, School of Computer Science and Technology, Huazhong University of Science & Technology, Wuhan, Hubei, China

Chao Liao
Huawei TCS Lab, Shanghai, China

Zhihuai Chen
Huawei TCS Lab, Shanghai, China

Zhibo Sun
SMART, School of Computer Science and Technology, Huazhong University of Science & Technology, Wuhan, Hubei, China

Zhouxing Su
SMART, School of Computer Science and Technology, Huazhong University of Science & Technology, Wuhan, Hubei, China

Junwen Ding
SMART, School of Computer Science and Technology, Huazhong University of Science & Technology, Wuhan, Hubei, China

Chen Wu
Huawei TCS Lab, Shanghai, China

Pinyan Lu
Huawei TCS Lab, Shanghai, China

Zhipeng Lv
SMART, School of Computer Science and Technology, Huazhong University of Science & Technology, Wuhan, Hubei, China

Abstract

In the challenge of Parameterized Algorithms and Computational Experiments 2022, a solver needed to find the best solution of each Directed Feedback Vertex Set instance. We submitted our heuristic solver Huawei_TCS_DFVS_Solver, which treated the challenge as a problem of finding the longest sequence that was a topological ordering of a subgraph of the graph. The solver consists of three parts: (1) preprocessing, (2) initial sequence generation, and (3) simulated annealing that introduces adaptive heating and intercomponent parallelism.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms

Keywords and phrases Directed Feedback Vertex Set, Local search, Heuristic search

Digital Object Identifier 10.4230/LIPIcs.PACE.2022.1

Supplementary Material *Software (Source Code)*: `https://doi.org/10.5281/zenodo.6638370`

1 The challenge and the sequence

For a directed graph $G = (V, A)$, the Directed Feedback Vertex Set (DFVS) is a subset $X \subseteq V$ such that, when all vertices of X and their adjacent arcs are removed from G, the remainder is acyclic. In the challenge of Parameterized Algorithms and Computational Experiments (PACE) 2022, a solver needed to find the minimum DFVS of the input graph. Additionally, for heuristic algorithms, the process was limited in 10 miniutes.

Each DFVS and the Directed Acyclic Graph (DAG) obtained by removing vertice in the DFVS from the graph are one-to-one correspondence, so to find the minimum DFVS we can look for the maximum DAG.

A topological ordering is possible if and only if the directed graph is acyclic. A DAG has at least a topological ordering, at the same time a topological ordering contains exactly all vertices of the DAG. So, similar to [1], we can solve the challenge by finding the longest sequence. The sequences represent topological orderings of subgraphs of the graph.

2 The solver

The solver consists of three parts: (1) preprocessing, (2) initial sequence generation, and (3) simulated annealing. The preprocessing uses some simple rules to reduce the size of the graph. The initial sequence generated speeds up the search process. The simulated annealing is the main part of the searching process.

2.1 Preprocessing

The solver extracts preprocessing rules from [2] and applies them repeatedly to the input graph until the graph keeps unchanged in a round. The rules are listed below.

- When the in-degree or out-degree of a vertex is zero, mark the vertex as "in-DAG" and remove the vertex from the graph.
- When the vertex is a direct successor of itself, mark the vertex as "in-DFVS" and remove the vertex from the graph.
- When the in-degree of a vertex is one, mark the vertex as "in-DAG" and merge the vertex into its direct predecessor. A similar rule is applied to vectice whose out-degree are ones.
- Obtain the Strongly Connected Components (SCCs) of the graph ignoring the bidirected arcs. Arcs whose heads and tails are in different components can be removed.
- When any two vertice in a vertice set S are connected by a bidirected arc, S is called a bidirected clique. If there's a vertex v not connected to any vertex out of S, mark v as "in-DAG" and others in S as "in-DFVS", remove S from the graph afterwards.
- If a point m has a non-bidirected arc to n, m is called a non-bidirected predecessor of n and n is called a non-bidirected successor of m. Considering an arc from m to n, if all the non-directed predecessors of m are predecessors of n, the arc can be removed from the graph without affecting the solution. The same is true for the case where all the non-directed successors of n are successors of m.

2.2 Initial sequence generation

Compared with starting from an empty sequence, generating an initial sequence and starting from it can speed up the search process. The solver uses a greedy strategy to generate the initial sequence. After backing up the graph, the solver loops until the graph is empty, after which the graph is restored. In each round the solver finds the vertex with maximum score

and attaches it to the end of the sequence, then the solver removes it and other vertice that violate topological ordering rules from the graph. Here, the score of a vertex v is calculated as the in-degree of v plus the out-degree of v and all its predecessors.

2.3 Simulated annealing

The basic framework for the simulated annealing is taken from [1]. In each round the solver probabilistically receives moves until a certain number of moves or a timeout. In the former case the solver scales down the temperature and continues the cycle, and in the latter it outputs the longest sequence in history. After getting the final sequence, the solver filters out the vertice in the preprocessed graph that are not in the final sequence and integrates them with the vertice marked as "in-DFVS" in the preprocessing to get the final result.

2.3.1 Move mechanism

The action of inserting a new vertex into the sequence and removing other vertice that violate the topological ordering rules is called a move. Making a move needs two arguments: the new vertex and the insert position. For the former, the solver randomly pick a vertex from all vertice in the preprocessed graph but not in the sequence as the new vertex. For the latter, same as [1], the solver randomly chooses between two positions for the insert position: (1) in front of the most preceding vertex in the sequence among the successors of the new point, or (2) after the most posterior vertex in the sequence among the predecessors of the new point.

2.3.2 Adaptive heating

In the late stage of simulated annealing, the sequences are difficult to make a move, which affects the efficiency of local search. Therefore, the solver introduce adaptive heating to improve the diversification.

A round of simulated annealing fails when a longer sequence is not found in the round. Here we refer to one single cycle containing $maxMove$ moves as a round. After a certain number of consecutive failures, the temperature is raised to that before the failures. The upper limit of the number of consecutive failures is also raised when heating.

2.3.3 Intercomponent parallelism

We can note that more than 1/5 of the sample graphs given in PACE 2022 have multiple SCCs after preprocessing. Thus, the solver implements the mechanism of intercomponent parallelism by performing simulated annealing in each component sequentially in each round, keeping parameters, such as the temperature and the upper limit of the number of consecutive failures, independent among the components.

References

1 Philippe Galinier, Eunice Lemamou, and Mohamed Wassim Bouzidi. Applying local search to the feedback vertex set problem. *Journal of Heuristics*, 19(5):797–818, oct 2013. doi:10.1007/s10732-013-9224-z.

2 Hen-Ming Lin and Jing-Yang Jou. On computing the minimum feedback vertex set of a directed graph by contraction operations. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 19(3):295–307, mar 2000. doi:10.1109/43.833199.