A conflict-learning based algorithm to solve the p-center problem*

Thomas Bazaille, Laure Brisoux Devendeville, Chu-Min Li, Corinne Lucet

Laboratoire MIS (UR 4290), Université de Picardie Jules Verne, France

{thomas.bazaille, laure.devendeville, chu-min.li, corinne.lucet}@u-picardie.fr

Mots-clés : *p-center, tree search, conflict*

1 Introduction

The p-center problem belongs to the family of facility location problems whose goal is to place p centers in a network in order to minimize the maximum distance between a client vertex and a center. There are many applications for this problem, such as the location of fire stations, ambulance depots, charging stations for electric vehicles etc. The p-center problem is NP-hard even in some simplified formulations where strong assumptions are made about the structure of the graph. Here, we propose an exact method to solve the p-center problem. In the literature several types of exact methods have been proposed, based on : integer programming model [3, 2] or SAT formulation [4]. In this paper, the initial optimization p-center problem is decomposed in several decision problems in which the radius is set. To solve the decision p-center problem an exact method based on a search tree algorithm is used. The branching strategy is done according to the learning process stemmed from the SAT resolution techniques, i.e. learning from previous failures in order not to repeat them. Unit propagation is ensured by employing the set covering formulation that allows to apply dominating reduction rules [1]. In order to make better use of the learnt results during the development of the tree, restart processes are applied regularly. Preliminary experimentation have been done on some well known instances.

2 Problem definition

Given an undirected graph $G = (N , E)$, where N is the set of n nodes $\{v_1, v_2, ..., v_n\}$ and E is the set of edges. The distance of the shortest path between any two nodes u and v is denoted by $d(u, v)$. The p-center problem is to find a set of facilities $S \subseteq N$, such that $|S| = p$ and the objective function : $f = max_{v \in N} min_{u \in S} d(u, v)$ is minimized. We notice that this problem is equivalent to solve a sequence of set covering problems (SCP) as proposed by Minieka [6], each corresponding to a given allowed maximum distance D. If a solution exists for SCP with a distance D, it ensures that the optimal solution for the p-center problem is less than or equal to D. Otherwise, the optimum solution is greater than or equal to $D + 1$.

3 Resolution method

First, given a distance D we create a decision graph G' of G in which each edge between two vertices v_i and v_j is defined if vertex v_j can cover vertex v_i within distance D. Then, the method used to solve this set covering problem is based on a search tree that enumerates all possible centers. The main features in such a tree are : (1) how to choose the next center to open and (2) what are the propagation implied by these choices. To exhibit propagation we

* This work has been funded by the French Agence Nationale de la Recherche, reference ANR-19-CHIA-0013-01

use the specific reduction rules [1]. These rules determines which vertices have to be centers or not. In consequence, they remove some vertices from the decision graph. To choose the next potential center, we used a heuristic based on SAT problem one [5]. In SAT problem, thanks to the implication graph, when a conflict occurs, the weight of clauses involved in this conflict increases. Then the SAT heuristic chooses its variables in the more conflictual clauses. The main idea is to reduce the search tree by quickly pruning branches. We propose to extend it to our problem. A conflict occurs if some nodes could not be covered, the weight of vertices involved in the conflict increases. So the heuristic we proposed selects the next vertex according to its weight. A sketch of our algorithm is given in algorithm 1. To improve this method, regularly a restart is done in order to exploit the learnt vertex weights allow to develop a tree smaller than the previous ones.

Algorithm 1 Pcenter (G', C, V', p)

Require: $G' = (N, E)$ a graph, $C \subseteq N$ a set of vertices, $V' \subseteq N \backslash C$ a set of vertices, p an integer

$C, V' \leftarrow reductionGraph(G', C)$
if $cover(G', C)$ **then**
    return true
else
    if $component(G'(V')) > p - |C|$ **then**
        $analyse\_conflict(G', C)$
        return false
    else
        $V'' \leftarrow V'$
        while $V'' \neq \emptyset$ **do**
            $v \leftarrow choice(G', C, V'')$; $V'' \leftarrow V'' \backslash \{v\}$
            $out \leftarrow Pcenter(G', C \cup \{v\}, V' \backslash \{v\}, p)$
            if out **then**
                return true
            end if
        end while
        return false
    end if
end if

4 Perspectives

Next step we will introduce more reduction rules to reduce the search space. Further objective will be to extend our works to solve the capacitated p-center problem with costs and time dependencies.

Références

[1] J. Alber, M. R. Fellows, and R. Niedermeier. Polynomial-time data reduction for dominating set. *Journal of the ACM*, 51(3) :363–384, May 2004.

[2] H. Calik and B. C. Tansel. Double bound method for solving the p-center location problem. *Computers & Operations Research*, 40(12) :2991–2999, December 2013.

[3] M. S. Daskin. *Network and discrete location : models, algorithms, and applications.* Wiley-Interscience series in discrete mathematics and optimization. Wiley, New York Chichester, 1995.

[4] X. Liu, Y. Fang, J. Chen, Z. Su, C. Li, and Z. Lu. Effective Approaches to Solve P -Center Problem via Set Covering and SAT. *IEEE Access*, 8 :161232–161244, 2020.

[5] J.P. Marques-Silva and K.A. Sakallah. GRASP : A Search Algorithm for Propositional Satisfiability. *IEEE Transactions on Computers*, 48(5) :506–521, May 1999.

[6] E. Minieka. The m-center problem. *SIAM Review*, pages 138–139, 1970.