Theoretical Computer Science 630 (2016) 117–125

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Approximation algorithms for submodular vertex cover problems with linear/submodular penalties using primal-dual technique ☆

Dachuan Xu [a,*,1], Fengmin Wang [a], Donglei Du [b,2], Chenchen Wu [c,3]

[a] College of Applied Sciences, Beijing University of Technology, 100 Pingleyuan, Chaoyang District, Beijing 100124, PR China
[b] Faculty of Business Administration, University of New Brunswick, Fredericton, NB, E3B 5A3, Canada
[c] College of Science, Tianjin University of Technology, Tianjin 300384, PR China

ARTICLE INFO

Article history:
Received 7 September 2014
Received in revised form 1 April 2016
Accepted 3 April 2016
Available online 19 April 2016
Communicated by Z. Cai

Keywords:
Penalty
Submodular
Vertex cover
Primal-dual
Approximation algorithm

ABSTRACT

The notion of penalty has been introduced into many combinatorial optimization models. In this paper, we consider the submodular vertex cover problems with linear and submodular penalties, which are two variants of the submodular vertex cover problem where not all the edges are required to be covered by a vertex cover, and the uncovered edges are penalized. The problem is to determine a vertex subset to cover some edges and penalize the uncovered edges such that the total cost including covering and penalty is minimized. To overcome the difficulty of implementing the primal-dual framework directly, we relax the two dual programs to slightly weaker versions. We then present two primal-dual approximation algorithms with approximation ratios of 2 and 4, respectively.

1. Introduction

1.1. Review of vertex cover and submodular function

A general vertex cover problem involves an undirected graph $G = (V, E)$ with vertex set V and edge set E. For each vertex $i \in V$, there is an associated nonnegative cost $c(i)$. The central issue in the vertex cover problem is which a vertex subset should be selected as the vertex cover to cover all the edges. A vertex cover in G is a vertex subset $S \subseteq V$, if every edge in E is incident to at least a vertex in S. The problem is to determine a vertex cover such that the total covering cost is minimized. The vertex cover problem is a fundamental and widely investigated combinatorial optimization problem (cf. [17]). It is well-known that the vertex cover problem is NP-hard (cf. [23]), which cannot be solved efficiently unless $P = NP$.

A set function $f(\cdot) : 2^V \to R_+$ is submodular iff it satisfies $f(X \cup Y) + f(X \cap Y) \le f(X) + f(Y)\ \forall X, Y \subseteq V$. It has the property of decreasing marginal return. Submodular functions occur in many mathematical models in operations research,

☆ A preliminary version of this paper appeared in the Proceedings of the 20th International Computing and Combinatorics Conference, 2014.

* Corresponding author.
E-mail address: xudc@bjut.edu.cn (D. Xu).

[1] The first author is supported by NSFC (No. 11531014) and Collaborative Innovation Center on Beijing Society-Building and Social Governance.
[2] The third author is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) grant 283106.
[3] The fourth author is supported by NSFC (No. 11501412).

http://dx.doi.org/10.1016/j.tcs.2016.04.005
0304-3975/

economics, engineering, computer science, and management science (cf. [9,11]). In combinatorial optimization, submodular functions play a key role somewhat similar to that played by convex/concave functions in continuous optimization. They exhibit sufficient structures so that a mathematically beautiful and practically useful theory can be developed. Lovász [25] establishes a direct connection between submodularity and convexity: the submodularity of a set function can be characterized by the convexity of a continuous function obtained by extending the set function in an appropriate manner. Furthermore, there has been extensive work on submodular function optimization (cf. [10,11,13,19,26]).

Iwata and Nagano [20] extend the classical vertex cover problem to the submodular case, where, given a nonnegative submodular function $C(\cdot): 2^V \rightarrow R_+$, the objective of the submodular vertex cover problem (SVC) is to find a vertex subset $S \subseteq V$ to cover all the edges that minimizes the submodular cost $C(S)$. Based on this model, we introduce the notion of penalty, which has been considered earlier in the context of the Steiner tree, TSP, and facility location (see [6–8,14,24] and references therein). In this paper, we extend the SVC to the penalty counterparts, i.e. the submodular vertex cover problem with penalties (SVCWP). Not all the edges are required to be covered by a vertex cover, and the uncovered edges should be penalized. The problem is to determine a vertex subset to cover some edges and penalize the uncovered edges such that the total cost including covering and penalty is minimized. Since the penalty cost function may be linear or submodular, we consider two models, namely the submodular vertex cover problem with linear penalties (SVCLP) and the submodular vertex cover problem with submodular penalties (SVCSP).

1.2. Previous work

Since the vertex cover problem is NP-hard, numerous work has focused on developing approximation algorithm for this problem. Among the extant literatures, several techniques have been developed, such as LP-rounding and primal-dual methods. For instance, Hochbaum [16] presents an LP-rounding 2-approximation algorithm for this problem, and Bar-Yehuda and Even [2] propose a primal-dual 2-approximation algorithm. Moreover, depending on the number of the vertices or the maximum degree of the graph, there are some approximation algorithms that achieve $2 - o(1)$ ratio (cf. [3,15,21]). On the other hand, Knote and Segev [22] prove that the lower bound for approximation is $2 - \epsilon$ for any $\epsilon > 0$ under the unique game conjecture.

Generalizations and variants of the vertex cover problem have been studied (cf. [1,12]). Below we only review some results related to the SVCWP. Iwata and Nagano [20] introduce the submodular vertex cover problem and present a 2-approximation algorithm using the convex programming rounding technique. Hochbaum [18] introduces the generalized vertex cover problem and presents a 2-approximation algorithm using the LP-rounding technique. This problem (a.k.a. the prize-collecting vertex cover problem [4]) is essentially the vertex cover problem with linear penalties in our terminology. Using the primal-dual technique, Bar-Yehuda and Rawitz [5] propose a 2-approximation algorithm. Taking the maximum degree d of the given graph into consideration, Bar-Yehuda et al. [4] give a local-ratio $(2 - 2/d)$-approximation algorithm.

1.3. Our work

- We firstly formulate the SVCWP (in particular the SVCLP and SVCSP) as linear integer programs. This class of problems generalizes the classic vertex cover problem, submodular vertex cover problem [20], and generalized vertex cover problem [18].
- In order to use the primal-dual technique to design approximation algorithms and control the dual ascending process in polynomial time, we relax the dual programs of the linear program relaxations for the SVCLP and SVCSP to slightly weaker versions.
- Implementing the primal-dual framework directly on the relaxed dual programs, we present two primal-dual approximation algorithms with 2 and 4 ratios for the SVCLP and SVCSP respectively.

1.4. Organization

The remainder of the paper is organized as follows. Sections 2 and 3 offer two primal-dual approximation algorithms with ratios of 2 and 4 for the SVCLP and SVCSP respectively. We finish the paper by drawing some conclusions and possible directions for future research in Section 4.

2. Primal-dual approximation algorithm for the SVCLP

2.1. Model formulation

Inputs and cost components

- $G = (V, E)$: undirected graph with vertex set V and edge set E.
- $C(\cdot): 2^V \rightarrow R_+$: nonnegative submodular function with $C(\emptyset) = 0$.
- $C(S)$: covering cost for each subset $S \subseteq V$.
- p_e: penalty cost for each edge $e \in E$.