ARTICLE IN PRESS

Discrete Optimization ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Discrete Optimization

www.elsevier.com/locate/disopt

Complete mixed integer linear programming formulations for modularity density based clustering

Alberto Costa[a,*], Tsan Sheng Ng[b], Lin Xuan Foo[b]

[a] *National University of Singapore and ETH Zurich, Future Resilient Systems, Singapore*
[b] *Department of Industrial and Systems Engineering, National University of Singapore, Singapore*

ARTICLE INFO

Article history:
Received 18 April 2016
Received in revised form 29 March 2017
Accepted 30 March 2017
Available online xxxx

Keywords:
Clustering
Modularity density
Mixed integer linear programming
Reformulations

ABSTRACT

Modularity density maximization is a clustering method that improves some issues of the commonly used modularity maximization approach. Recently, some Mixed-Integer Linear Programming (MILP) reformulations have been proposed in the literature for the modularity density maximization problem, but they require as input the solution of a set of auxiliary binary Non-Linear Programs (NLPs). These can become computationally challenging when the size of the instances grows. In this paper we propose and compare some explicit MILP reformulations of these auxiliary binary NLPs, so that the modularity density maximization problem can be completely expressed as MILP. The resolution time is reduced by a factor up to two order of magnitude with respect to the one obtained with the binary NLPs.

1. Introduction

Given an undirected unweighted graph $G = (V, E)$, where V is the vertex set and E is the edge set, cluster analysis refers to finding a partition of V into disjoint groups called clusters (or communities) such that vertices in the same cluster are densely connected to each other and less connected to those in other clusters. This is a very important problem with applications in many fields, e.g., social networks [1], recommender systems [2], biology and bioinformatics [3,4].

To date, there have been numerous methods proposed to identify clusters in a graph. Some methods do not require a function to be optimized, for example the Girvan and Newman's heuristic [1] where the edge with highest 'betweenness' (i.e., number of shortest paths running along that edge) is iteratively removed. Other approaches can be based on some rules (usually related to the number of neighbors of vertices inside and outside their corresponding cluster) that each cluster must respect. In this category we can find the *strong* and *weak* definition of Radicchi et al. [5], the *semi-strong* and *extra-weak* definitions of Hu et al. [6], and the *almost-strong* definition of Cafieri et al. [7]. Alternatively, clustering can be expressed in terms of

* Corresponding author.
E-mail addresses: costa@lix.polytechnique.fr (A. Costa), isentsa@nus.edu.sg (T.S. Ng), foolinxuan@u.nus.edu (L.X. Foo).

http://dx.doi.org/10.1016/j.disopt.2017.03.002
1572-5286/

an objective function to optimize, and this is interesting from a mathematical programming point of view. One of the commonly used objective functions is *modularity*, which is defined as the fraction of edges within clusters minus the expected fraction of such edges in a random graph with the same degree distribution. It is to be noted that the problem of clustering based on maximization of the modularity function is *NP*-hard [1,8,9].

Notwithstanding its *NP*-hardness, clustering solutions arising from modularity maximization also present some practical issues, in particular, *resolution limit* and *degeneracy*. The former refers to the possibility of small clusters being merged with other clusters and thus not detected [10]. The latter occurs when there are several high quality local optima and a global optimum cannot be easily found [11]. In order to overcome the resolution limit issue associated with modularity maximization, an alternative clustering measure, *modularity density*, was proposed in [12]. The main difference with respect to modularity is that modularity density takes into account the size of the clusters. More precisely, according to [13] the modularity density D can be defined as:

$$D = \sum_{c \in C} \left(\frac{2m_c - \bar{m}_c}{n_c} \right), \tag{1}$$

where C is the set of clusters (whose cardinality $|C|$ is not know in advance), m_c is the number of edges having both end vertices inside cluster c (inner edges), $\bar{m}_c$ is the number of edges having one end vertex in c and the other one outside c (cut edges), and n_c is the number of vertices inside cluster c (i.e., the size of the cluster).

Unfortunately, the optimization problem based on maximizing modularity density, termed as the *Modularity Density Maximization* (MDM) problem, is not easy to solve. More precisely, MDM was formulated as a binary Non-Linear Programming (0–1 NLP) problem in [12], and in general non-linear, non-convex problems having integer variables are solved by means of MINLP (Mixed-Integer Non-Linear Programming) solvers, which may converge slowly to the optimal solution even for small size instances. Nevertheless, as pointed out in [14,15], it is still unclear if MDM is *NP*-hard, since the proof provided in [12] is wrong.

In [13] some Mixed-Integer Linear Programming (MILP) reformulations of MDM were proposed. However, to implement these MILP reformulations, the authors require the solution of a set of auxiliary 0–1 NLP problems. Albeit these auxiliary 0–1 NLPs are easier to solve than the MDM problem presented in [12], they can still be quite challenging when the size of the instances is large. Hence, it is important to find efficient ways to solve them. In this paper, we derive several exact MILP reformulations of the above-mentioned auxiliary NLP problems, thus obtaining complete MILP formulations of MDM. To do so, we employ some reformulation techniques, e.g., linearization of bilinear terms, expansion of integers in power of two, and reformulation of fractional programs. Finally, we perform numerical studies to show that the proposed MILP reformulations are more computationally efficient, especially when problem instances scale up.

The rest of the paper is organized as follows: In Section 2 we introduce the techniques employed later to linearize the non-linear problems. In Section 3 we give a formal problem statement of the MDM and the related auxiliary 0–1 NLPs. The MILP reformulations of these problems are then presented in Sections 4 and 5. In Section 6 we also prove that the optimal solutions of the MILPs and those obtained in [13] by solving the continuous relaxation of the 0–1 NLPs are the same, i.e., we can discard the integrality constraint from the 0–1 NLPs and the solution is still integral. Computational results showing the efficiency of our new models are reported in Section 7. Finally, conclusions are drawn in Section 8.

2. Preliminaries: reformulation techniques

In this section we introduce the techniques that will be employed later to derive the MILP reformulations of the 0–1 NLPs.

Please cite this article in press as: A. Costa, et al., Complete mixed integer linear programming formulations for modularity density based clustering, Discrete Optimization (2017), http://dx.doi.org/10.1016/j.disopt.2017.03.002.