European Journal of Operational Research 238 (2014) 711–723

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

Single-commodity robust network design problem: Complexity, instances and heuristic solutions

Eduardo Álvarez-Miranda [a,b], Valentina Cacchiani [b], Andrea Lodi [b,*], Tiziano Parriani [b], Daniel R. Schmidt [c]

[a] DMGI, Universidad de Talca, Merced 437, Curicó, Chile
[b] DEI, University of Bologna, Viale Risorgimento 2, I-40136 Bologna, Italy
[c] Institut für Informatik, Universität zu Köln, Pohligstrasse 1, 50969 Köln, Germany

ARTICLE INFO

Article history:
Received 6 August 2013
Accepted 21 April 2014
Available online 30 April 2014

Keywords:
Robustness
Network design
Complexity
Heuristic
Proximity search

ABSTRACT

We study a single-commodity Robust Network Design problem (RND) in which an undirected graph with edge costs is given together with a discrete set of balance matrices, representing different supply/demand scenarios. In each scenario, a subset of the nodes is exchanging flow. The goal is to determine the minimum cost installation of capacities on the edges such that the flow exchange is feasible for every scenario. Previously conducted computational investigations on the problem motivated the study of the complexity of some special cases and we present complexity results on them, including hypercubes. In turn, these results lead to the definition of new instances (random graphs with $\{-1,0,1\}$ balances) that are computationally hard for the natural flow formulation. These instances are then solved by means of a new heuristic algorithm for RND, which consists of three phases. In the first phase the graph representing the network is reduced by heuristically deleting a subset of the arcs, and a feasible solution is built. The second phase consists of a neighborhood search on the reduced graph based on a Mixed-Integer (Linear) Programming (MIP) flow model. Finally, the third phase applies a *proximity search* approach to further improve the solution, taking into account the original graph. The heuristic is tested on the new instances, and the comparison with the solutions obtained by Cplex on a natural flow formulation shows the effectiveness of the proposed method.

1. Introduction

Network design problems arise in many different areas, such as transportation and telecommunication. Recently, the class of *robust network design* problems has received increasing attention. The term robust can represent the capability of the network to cope with disruptions or to deal with different traffic scenarios in different times of the day, as is the case of our work.

In this paper, we study the *single-commodity* Robust Network Design problem (RND) defined as follows. We are given an undirected graph $G=(V,E)$, a cost vector (c_e) $(e \in E)$ and an integer *balance* matrix $B=(b_i^q)$ $(i \in V, q=1,\ldots,K)$. The qth row b^q of B is called the qth *scenario*.

For a given scenario, we call a node with nonzero balance a *terminal*. More specifically, a node i with positive balance is called a *source* and we call the balance of i its *supply*. A node with negative balance is called a *sink* and its balance is called *demand*.

Let us denote by (i,j) and (j,i) the arcs (directed from i to j and from j to i, respectively) corresponding to edge $e=\{i,j\} \in E$. In addition, let us call $f_{i,j}^q \in \mathbb{Z}_+$ the integral amount of flow that is sent along arc (i,j) from i to j in scenario q and by f^q the corresponding flow vector.

RND calls for determining integer capacities $(u_e) \in \mathbb{Z}_+^{|E|}$ $(e \in E)$ with minimal costs $c^T u$ such that, for each q $(q=1,\ldots,K)$, there is a directed network flow f^q in G that is feasible with respect to the capacities and the balances of the qth scenario. In particular, the flow f^q $(q=1,\ldots,K)$ must fulfill the following constraints:

1. $f_{i,j}^q + f_{j,i}^q \leqslant u_e$ for all edges $e=\{i,j\} \in E$, which imposes that the sum of the flows going along every edge (in both directions) must respect the installed edge capacity, for every scenario.
2. $\sum_{\{i,j\} \in E}(f_{i,j}^q - f_{j,i}^q) = b_i^q$ for all nodes $i \in V$, which implies that the flow must satisfy the required integer balances.

An overall natural model for RND reads as follows

---

* Corresponding author. Tel.: +39 0512093029; fax: +39 0512093073.
E-mail addresses: e.alvarez@unibo.it (E. Álvarez-Miranda), valentina.cacchiani@unibo.it (V. Cacchiani), andrea.lodi@unibo.it (A. Lodi), tiziano.parriani@unibo.it (T. Parriani), schmidt@informatik.uni-koeln.de (D.R. Schmidt).

http://dx.doi.org/10.1016/j.ejor.2014.04.023
0377-2217/

$$\min \sum_{\{i,j\}\in E} c_{ij} u_{ij} \tag{1}$$

$$\sum_{j:\{j,i\}\in E} f_{ji}^q - \sum_{j:\{i,j\}\in E} f_{ij}^q = b_i^q \quad \forall i \in V,\ q = 1,\ldots,K \tag{2}$$

$$f_{ij}^q + f_{ji}^q \leqslant u_{ij} \quad \forall \{i,j\} \in E,\ q = 1,\ldots,K \tag{3}$$

$$f_{ij}^q \geqslant 0 \quad \forall \{i,j\} \in E,\ q = 1,\ldots,K \tag{4}$$

$$u_{ij} \in \mathbb{Z}_+, \quad \forall \{i,j\} \in E \tag{5}$$

where the objective function (1) is to minimize the total cost of the installed capacities. Constraints (2) ensure flow-conservation in each scenario and impose to satisfy the required balances. Constraints (3) model that the capacity of an edge is at least as large as the flow it carries. Integral flows are enforced through integrality of the capacity variables, as all balances are integral (Ford & Fulkerson, 1957).

As described in Buchheim, Liers, and Sanità (2011), an example of a practical application of the considered problem is the following: some clients wish to download some programs stored on several servers. For a client, it is not important which server he or she is downloading from, as long as the demand is satisfied. In other words, we consider servers that store identical data: examples are video on demand or large data centers in which one mirrors his data over several locations. This is opposed to multi-commodity network design, in which point-to-point connections are considered, i.e., each client requests a specific server. In addition, we consider the robust version of the problem: at different times of the day, the demands may change (e.g., different clients show up), and the goal is to design a network that is able to route all flow in all different scenarios. In particular, we consider a finite list of demands, i.e., we sample different times of the day.

Contribution of the paper. Preliminary computational investigations have been performed on classical graphs from the literature with random balances (Buchheim et al., 2011) and on special hypercubes with $\{-1, 0, 1\}$ balances (Álvarez-Miranda et al., 2012). The results in both papers have shown that the former instances are surprisingly easy for a general-purpose Mixed-Integer Programming (MIP) solver on the natural flow-formulation (1)–(5), while the latter instances are structurally difficult. The first contribution of the paper is in studying the complexity of some RND special cases[1] associated with the above instances and enlightening the reasons of the observed computational behavior. Second, based on the complexity results, we propose a new family of randomly generated RND instances that are computationally challenging for the natural flow formulation already for $|V| = 50$ and $K = 10$. Third, motivated by those instances (available upon request from the authors), we propose new Matheuristic approaches that provide high-quality approximated solutions for large graphs (tests are reported for $|V|$ up to 500) in short computing times.[2] The motivation for developing Matheuristic methods, i.e., optimization algorithms made by the interoperation of Metaheuristics and Mathematical Programming (MP) techniques, instead of classical Metaheuritic relies on the possibility of working with a MP model. This allows, in general, the exploitation of effective and reliable software for solving sub-problems as, for example, the local search MIP neighborhoods, and guarantees a certain flexibility in accommodating modifications and extensions to the original version of the problem.

Organization of the paper. Section 2 reviews the (vast) related literature by pointing out differences and similarities. In Section 3 we present the complexity results we achieved on special classes of instances, while Section 4 describes the proposed heuristic algorithm and its performance is reported in Section 5. Finally, in Section 6 we draw conclusions and describe ideas for future research.

2. Related literature

The work on classical (i.e., non-robust) network design goes back as far as the early 1960s where it was studied by Chien (1960) and Gomory and Hu (1961, 1962). Since then, network design has evolved to a vast field of research which we cannot fully discuss in the scope of this article. We rather refer to Chekuri (2007) for a complete overview and restrict ourselves to a few exemplary related works that are of direct importance for us here.

The common theme of network design problems is installing optimum-cost capacities in a given network topology such that a set of traffic requests can be routed through the network. In practice, however, the traffic requests are not exactly known in advance. This can be due to measuring errors or simply because they cannot be predicted (Ben-Tal & Nemirovski, 2000). Here, the robustness comes in: Following an idea by Soyster (1973), Ben-Tal and Nemirovski (1999) coined the term of an *uncertainty set* that is added to the model and contains all possible (or likely) scenarios against which the robustness should protect.In particular, they consider ellipsoidal uncertainty sets. Bertsimas and Sim (2004) introduced *Γ-robustness* as a general model for robustification in MILPs. They consider a polyhedral uncertainty set where the parameter Γ is used to control the number of coefficients that can simultaneously assume different values from the nominal ones in each constraint. Since then, robust network design has been very actively studied. The notions of *network topology*, *cost*, *capacity*, *traffic request* and *routing* can vary – as well as the exact way in which the problem is *robustified*.

In this paper, we study a *worst-case* robust model in the sense of Ben-Tal and Nemirovski (1999). This means that our solutions must be feasible for all the scenarios from the uncertainty set. The uncertainty set is *finite* and *explicitly given* as part of the input (an idea that goes back to Minoux (1981)). We use an *undirected* graph as the network topology and allow *dynamic routing* (each scenario may be routed on different paths). Furthermore, we assume *linear costs* for the capacities and integer multiples of a unit capacity may be installed on each edge. Each node specifies its traffic request by a scalar number that gives its supply or demand and each such traffic request may be routed on an arbitrary number of paths (the routing is *splittable*) as long as each edge carries an integer amount of flow in total. Therefore, the underlying flow model is a standard *single-commodity*, splittable network flow in our case.

To the best of our knowledge, only two prior publications on this specific problem exist. The problem was first studied by Buchheim et al. (2011). They gave an exact branch-and-cut algorithm that solves a flow-model MIP through sophisticated general-purpose cutting planes. Lately, Álvarez-Miranda et al. (2012) introduced a capacity-based MIP-model, and discussed a preliminary set of results of the biennial “Vigoni 2011–2012” between the universities of Köln and Bologna.

Atamtürk (2000) considers a variant of the non-robust single-commodity network design problem where integer multiples of a facility with fixed capacity can be installed on each arc. Ortega and Wolsey (2003) report on the performance of general MIP solvers on various network design problems and develop an exact algorithm for the single-commodity fixed-charge network design problem (all arcs may be bought at a fixed-charge and then be used at full capacity).

A close variant of single-commodity RND is the *multi-commodity* robust network design problem. Here, the traffic requests specify the amount d_{ij} of flow that should be exchanged among all pairs of nodes

---

[1] The RND problem is strongly NP-hard (Sanità, 2009).

[2] A preliminary version of the heuristic approaches described here was introduced in Álvarez-Miranda et al. (2012) where the first phase of the investigation on RND, which was the topic of the “Vigoni 2011–2012” project between the University of Köln and the University of Bologna, was summarized.