Resilient Countertrading

Friedrich Eisenbrand
EPFL

Niko Schöll
GreenCircle

September 13, 2021

Problem description

This project is a collaboration with GreenCircle, a young start-up based in Berlin which provides technical and logistical infrastructure for circular economy (see https://www.green-circle.co/). As an experiment, they are organizing a large-scale barter exchange (trade without money). To implement it, they face the following problem: there is a set of participants. Each participant has one object that he or she *offers* and one object that he or she *desires*. The participants are willing to countertrade the object in possession for the one that they desire.
The goal is now to find trading *cycles* that satisfy a maximum amount of participants. These cycles should also be constrained in length so that a *failure* of trading by one participant does not affect too many other participants.

In mathematical terminology, this problem is a *maximum cost circulation problem*. Here, the objects are *nodes* V of a directed graph $D = (V, A)$ and the participants are directed arcs. The arc/participant (u, v) represents the the desired object u and the offered object v of the respective participant. Without length constraint to deal with resilience, one then has a maximum cost circulation problem, where each arc has capacity and cost one. This can be solved in polynomial time, see, e.g. [1].

Project objective

The main goal is to implement a software tool that efficiently finds resilient cycles in the described graph. The software must quickly find an optimal solution on the dataset provided by the company. Ideally, the software should not rely on commercial packages. First computational results on an artificial dataset are due in October. A solution for the actual dataset is due Mid-November.
Furthermore, the complexity status of the constrained problem has to be researched in the literature. Integer programming models and efficient cutting plane/separation techniques need to be developed, implemented and tested.

Prerequisites

1. *Discrete Optimization*
2. Programming skills
3. Entry test: Coding of min-cost flow problem/solution on an artificial graph on 10 notes with NetworkX and Gurobi.

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows.* Prentice Hall Inc., Englewood Cliffs, NJ, 1993. Theory, algorithms, and applications.