Comparison of CP and IP Techniques for Data Transfer Planning *

Petr Holub
Institute of Computer Science, Masaryk University
Botanická 68a, Brno, Czech Republic

Miloš Liška, Hana Rudová and **Pavel Troubil**
Faculty of Informatics, Masaryk University
Botanická 68a, Brno, Czech Republic

Introduction

Planning of data transfers and allocation of network resources becomes an interesting problem when bandwidth of a requested traffic is comparable to a capacity of available networks. An interactive high-quality media distribution for collaborative environments is one of the important applications of the problem. The CoUniverse framework (Liška and Holub 2008) was proposed to handle these applications automatically for end-users. A planning module is one of the crucial components of the framework. Each demand is represented by a transfer of media stream from a producer to a set of consumers. To plan each demand, a proper distribution of the stream through a set of distributors is computed. There is no temporal reasoning since all allocated resources must be available all the time during the interaction. To allow for real-time experience with the CoUniverse, it is necessary to compute a plan within a few seconds. Fast planner responses also allow a graceful reaction to resource failures or changes such as link outage. More precisely, the CoUniverse monitors the network and can process replanning from scratch if needed.

Variety of network optimization problems were described in the survey published by Simonis (2006). The problem we are interested in is related to a path placement problem where the decision about allocating path for a demand should be done. Since each demand corresponds to a data transfer from a producer to a set of consumers, a communication graph is a tree with the producer in its root, consumers at leafs and media distributors at internal nodes. Therefore, we call the problem tree placement. The link-based model for this problem was described in (Holub, Rudová, and Liška 2010) using techniques of constraint programming (CP). Java constraint solver Choco 2.1.1[1] was applied to implement the model. The constraint programming solver was improved over the years since its first implementation (Liška and Holub 2008) and allows solving of medium-sized data instances currently. A new integer programming (IP) solver was proposed in (Troubil and Rudová 2010) and utilizes the Gurobi Optimizer 3.0.2[2]. This solver was implemented in the CoUniverse as a comparative approach to improve solution capabilities of the planning module and enlarge the size of solved problems. Since all the constraints are linear, it seemed as a natural extension of the solution approach. Our current intent is a parallel development of both solvers and their improvement based on experiences with both methodologies.

*This research is supported by the Ministry of Education, Youth and Sports of the Czech Republic under the project 0021622419.

[1] `http://choco.emn.fr`

[2] `http://www.gurobi.com`

Model with Linear Constraints

We shortly describe basics of models. For detailed formulation we refer to (Holub, Rudová, and Liška 2010; Troubil and Rudová 2010).

The underlying network consists of nodes where consumers, producers or distributors (particular applications) reside. Directed links in the network correspond to logical connection between the nodes (links do not represent physical topology of a network, since it is not generally known by end-users). Possible congestion on a physical link shared by several logical links is detected by the CoUniverse and avoided by replanning.

The decision variables are now represented by $x_{s,l}$ binary variables which refer to a transfer of a demand (called stream) s over a link l. The goal is to minimize the overall sum of latencies of individual streams scheduled onto the links. There is a set of constraints related to stream bandwidth and link capacity. They express that the total bandwidth of streams planned on each link cannot exceed the capacity of the link. Further, exactly one $x_{s,l_{\text{out}}}$ variable equals one among variables for stream s and for all links l_{out} going from producer of the stream s. Similarly, exactly one $x_{s,l_{\text{in}}}$ variable equals one among variables for stream s and for all links l_{in} to a consumer of the stream s. If there is neither consumer of a stream s nor distributor on a node, no link l_{in} ending at the node is used by s (i.e., $x_{s,l_{\text{in}}} = 0$). Similarly, if there is neither producer of s nor distributor on a node, no link l_{out} beginning at the node is used by s. Currently, constraints for nodes with distributors are linear only in the IP solver. They state for each node v with a distributor that (1) at most one $x_{s,l_{\text{in}}}$ variable for all links l_{in} going into the node v and for all streams s can be equal to one; (2) for each stream s, the number of non-zero $x_{s,l_{\text{in}}}$ variables for links l_{in} going into the node v (denoted $I_s x$) must be smaller or equal than the number of non-zero $x_{s,l_{\text{out}}}$ variables for l_{out} going out of the node v (denoted $O_s x$), i.e., $I_s x \leq O_s x$;

(3) it cannot distribute a stream s it does not receive, i. e., $O_s x \leq outdeg(v) * I_s x$ where $outdeg(v)$ is the number of the links beginning at v. Finally, it is necessary to post constraints prohibiting cycles among distributors (and connecting consumers to such cycles). The CP planner currently implements two variants of the constraint. The first one is based on the idea that at most $(k-1)$ variables $x_{s,l}$ among k nodes with distributors can be set to one for each stream s (this variant is also implemented in the IP solver). The second variant considers the distance from a root (node with a producer). Certainly, it must be smaller for a source node of each edge than for its target node.

Extensions

Current development of both models is directed towards improvement of the problem formulation. Further changes will involve problem extensions such as transcoding of the media streams by distributors, inclusion of partial information about network topology or consideration of more complex optimization criteria. Interesting area for the development is related to replanning and self-stabilizing mechanisms to support dynamic characteristics of the problem.

Having the actual problem in mind, we are now exploring various directions. We are comparing both solvers and implementing all described constraints in both solvers simultaneously. We also explore usefulness of a number of redundant constraints. Namely distributor constraints from the described linear model are to be added to the CP solver. Alternative cycle elimination constraint may be investigated in the IP solver. However, this is probably not a critical change. In the context of the traveling salesman problem (Applegate et al. 2007), Pataki (2003) discusses that the linear programming relaxation of a similar MTZ formulation is weaker than the other cycle elimination constraint based on the subtour elimination. A more interesting option consists in reformulation of the problem in the context of integer multicommodity network flows (Ahuja, Magnati, and Orlin 1993) since formulations based on network flows might improve effectiveness of the IP solver namely. Last but not least, we would like to apply the idea of combination the current tree placement model with the placement of independent paths similar to (Zerola et al. 2009). Such model would integrate our decision variables $x_{s,l}$ with the decision variables $x_{s,l,c}$ denoting whether a stream s is transferred by a link l to a consumer c. Even though this model was considered for a different problem involving transfers of large scale data to different data servers, it is also applicable in our case.

Current Results

The CoUniverse framework with the CP solver has been deployed on various use cases. It is used for distributed classroom taught at the Louisiana State University and distributed to a number of universities in the U.S., Czech Republic and Argentina. It was demonstrated in meetings such as SuperComputing'07 convention for communications covering even destinations from the United States, Czech Republic and Asia. For an example see Figure 1.

Experimental data sets with different topologies were generated based on those real-life problems. Comparison of both solvers shows that the first new IP solver is capable of solving larger data sets (Troubil and Rudová 2010) than the original CP solver which promises very interesting results in the above mentioned further developments.

Figure 1: Data transmissions computed by CoUniverse during GLIF 2007 meeting.

References

Ahuja, R. K.; Magnati, T. L.; and Orlin, J. B. 1993. *Network Flows*. Prentice-Hall.

Applegate, D. L.; Bixby, R. E.; Chvátal, V.; and Cook, W. J. 2007. *The Traveling Salesman Problem: A Computational Study*. Princeton University Press.

Holub, P.; Rudová, H.; and Liška, M. 2010. Data transfer planning with tree placement for collaborative environments. Under minor revision in *Constraints*.

Liška, M., and Holub, P. 2008. CoUniverse: Framework for building self-organizing collaborative environments using extreme-bandwidth media applications. In *Euro-Par 2008 Workshops – Parallel Processing*, volume 5415 of *Lecture Notes in Computer Science*, 339–351. Springer.

Pataki, G. 2003. Teaching integer programming formulations using the traveling salesman problem. *SIAM Review* 45(1):116–123.

Simonis, H. 2006. Constraint application in networks. In Rossi, F.; van Beek, P.; and Walsh, T., eds., *Handbook of Constraint Programming*. Elsevier. 875–903.

Troubil, P., and Rudová, H. 2010. Integer programming for media streams planning problem. In *MEMICS 2010 Proceedings*, 175–183.

Zerola, M.; Barták, R.; Lauret, J.; and Šumbera, M. 2009. Efficient multi-site data movement in distributed environment. In *Proceedings of the 10th IEEE/ACM International Conference on Grid Computing (GRID)*, 171–172. IEEE.