A Study on Petri Net-based Formulation and Iterative Optimization for Integer Linear programming

A Study on Petri Net-based Formulation and Iterative Optimization for Integer Linear Programming

Abstract

This thesis investigates two significant obstacles encountered in utilizing discrete optimization techniques in various fields the difficulty in mathematical programming formulation, and the high possibility of unreasonable computation time.

For the difficulty in mathematical programming formulation, we propose a Petri net based mathematical programming approach to combinatorial optimization, in which we generate integer linear programming problems from Petri net models instead of the direct mathematical formulation. We classify combinatorial optimization problems into ordinary and time-dependent problems.

For ordinary problems, we present autonomous Petri net modeling, where we obtain fundamental constraints derived from Petri net properties and additional problem-specific ones. For time-dependent problems, we propose a colored timed Petri net modeling, where we generate variables and constraints for time management and for resolving conflicts.

Our Petri net approach can drastically reduce the difficulty of the mathematical formulation. That is, (1) the Petri net modeling does not require deep knowledge of mathematical programming and linear model formulation technique, (2) our automatic formulation allows us to generate a large size of integer linear programming problems, and (3) the Petri net modeling approach is flexible for input parameter changes of the original problem.

For solving large-scale integer linear programming problems efficiently, we present a metropolis-type optimization algorithm based on mathematical programming. The algorithm reduces the size of mathematical programming problems by setting values to the variables specified in the simulated annealing process so that a solver can solve the reduced problem.

Our algorithm has a good characteristic in that the reduced problem at each iteration has better or equal quality feasible solutions. Therefore, the iterative approach improves solution quality efficiently.

The experimental evaluation shows the effectiveness of our algorithm for some class of problems, where our approach obtains a good quality of solutions in a reasonable computation time for hard classes of integer linear programming problems compared to the exact solver, which requires unreasonable computation time.