Improved Constraints for Multiprocessor System Scheduling

Martin Grajcar, Werner Grass
{grajcar|grass}@fmi.uni-passau.de
University of Passau, Germany

Abstract

MILP-based models are useful for finding optimal schedules and for proving their optimality. Because of the problem complexity, model improvements have to be investigated.

We analyze the constraints necessary for precluding resource conflicts, present novel formulations, and evaluate them. The efficiency of the solution process can be improved significantly by selecting the proper formulation.

The scheduling problem

Scheduling applications in computer science range from HL-Synthesis through Real-Time Embedded Systems to HW/SW-Co-design. We deal with a STATIC scheduling problem for a heterogeneous multiprocessor system, where TASKS have to be assigned to PROCESSING MODULES and to be ordered. COMMUNICATIONS and BUS STRUCTURE are also considered, the goal is to minimize MAKESPAN or LATENCY and hardware COSTS, while also respecting other constraints.

It is a deterministic NP-complete optimization problem [2]). There are many heuristics dealing with it, e.g. List Scheduling [5] or Genetic Algorithms [3, 4].

Unfortunately, there is no heuristic giving useful guarantees with respect to the quality of its result. While facing hard deadlines, heuristics may fail to find any solution at all.

For this reason we focus on an EXACT solution procedure, such as BRANCH-AND-BOUND working with an MILP-MODEL. Most constraints can be incorporated in the model in a quite straighforward way [6].

Focus of the work

The RESOURCE CONFLICT constraints (preventing two tasks (or communications) from using one resource at the same time) require special attention mainly for the following reasons:

- They can be incorporated into an MILP-model only by using at least one artificial boolean variable and multiple equations per constraint.
- They grossly outnumber all other constraints, so they enlarge the model in both directions (variables and equations) significantly, thus having a great impact on the performance.
- There is no obvious "best" way for expressing them.

Results

We investigated 15 different formulations (all but one novel) of resource conflict constraints on 30 most demanding examples (some taken from [1, 7]) having up to 100 tasks. We achieved an improvement of factor 2.3 compared to [1]. Note, that this improvement can be combined with improvements based on tuning the branch-and-bound etc.

Interestingly, the best CPU times were achieved neither for the smallest formulation nor for the smallest number of nodes. We believe that our improved formulations may also be used for other problems dealing with resource conflicts. The full version of this paper is available from the authors.

References

[1] A. Bender: *MILP Based Task Mapping for Heterogeneous Multiprocessor System;* EURO-DAC'96 with EURO-VHDL'96, Geneva, 1996, p. 190

[2] M. R. Garey, D. S. Johnson: *Computers and intractability - a guide to the theory of NP-completeness;* Freeman, 1979

[3] M. Grajcar: *Genetic List Scheduling Algorithm for Scheduling and Allocation on a Loosely Coupled Het. Multiproc. System;* Proc. 36th DAC, New Orleans, 1999

[4] M. Grajcar: *Strengths and Weaknesses of Genetic List Scheduling for Het. Systems;* Proc. 2nd Int. Conf. on Appl. of Conc. to Sys. Design, Newcastle upon Tyne, U.K., 2001

[5] Y.-K. Kwok, I. Ahmad: *Dynamic Critical-Path Scheduling;* IEEE Trans. on Par. and Distr. Systems, 1996, Vol. 7

[6] G. L. Nemhauser, L. A. Wolsey: *Integer and Combinatorial Optimization;* J. Wiley and Sons Ltd, 1988

[7] S. Prakash, A. C. Parker: *SOS: Synthesis of Application-Specific Heterogeneous Multiprocessor Systems;* Journal of Parallel and Distributed Computing 16, 1992, p. 338–351

Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02)
1530-1591/02 $17.00 © 2002 **IEEE**

IEEE COMPUTER SOCIETY