

PARALLELIZATION TOOLS AND INDUSTRIAL APPLICATIONS

M.BARAVYKAITE¹, R.ČIEGIS², A.JAKUŠEV³ and G.ŠILKO⁴

Vilnius Gediminas Technical University

Saulėtekio av. 11, LT-10223 Vilnius – 40, Lithuania

E-mail: {¹milda.baravykaite, ²rc}@fm.vtu.lt, ³a.jakusev@kemek.lt, ⁴gs@sc.vtu.lt

Parallelization tools are used to ease the development of parallel programs from sequential ones. That is a common situation in industrial applications of parallel programming. The aim of parallelization tools is to save the user from creation of the parallel algorithms from scratch [1].

Several possibilities of parallelization tools are analysed. This paper will present object oriented tools for data parallelism and parallelization of master-slaves, and branch and bound algorithms.

Parallelization tool for data parallelism introduces data structures that can be comfortably used in sequential programming and have useful features that allow automatic parallelization.

Branch and bound is an algorithm paradigm, which has to be filled out for each specific problem type [3], i.e. we should define the branch and bound strategies (algorithms), that are appropriate for the given problem. Branch and bound algorithm parallelization tool offers several parallel algorithms that can be obtained from the sequential branch and bound algorithm. Parallelization requires the user to implement several classes, that naturally arise from the branch and bound algorithm logic, and to use a tool to achieve parallel programs. Adding additional features to these classes will allow to implement more efficient parallel algorithms.

Some industrial applications of our *master-slaves* tool are presented in [2].

REFERENCES

- [1] M. Baravykaitė and R. Šablinskas. The template programming of parallel algorithms. *Math. Model. Anal.*, **7**, 2002, 11 – 20.
- [2] M. Baravykaitė, R. Belevičius and R. Čiegis. One application of the parallelization tool of Master – Slave algorithms. *Informatica*, **13** (4), 2002, 393–404.
- [3] J. Clausen. *Parallel branch-and-bound - principles and personal experiences*. Parallel Computing in Optimization, Kluwer Academic Publishers, 1997, 239 – 267.