Abstract of Equation based parallelization of Modelica models

Marcus Walther Volker Waurich Christian Schubert Dr.-Ing. Ines Gubsch
Dresden University of Technology
{marcus.walther, volker.waurich, christian.schubert, ines.gubsch}@tu-dresden.de

Modelica has become a widely used standard to describe physical simulation models. Compiling such a model into binary code can be performed by applications like Dymola, SimulationX or OpenModelica. However, all these tools only create a single thread simulation code out of standardized Modelica models, which does not allow for a speed-up with modern multi-core CPUs. This is due to the dependencies among the model equations which have to be considered in order to distribute the tasks amongst several threads. To gain a speed-up from this architecture, software programs have to be partitioned into several independent parts. A common representation of these parts is called a task graph or data dependency graph[1]. The authors of this article have developed a module for the OpenModelica Compiler (OMC), which creates, simplifies and schedules such task graphs. The tasks are created based on the BLT (block lower triangular)-structure[2], which is derived from the right hand side of the model equations. A noticeable speed-up for fluid models on modern six-core CPUs can be achieved, as shown in figure 1.

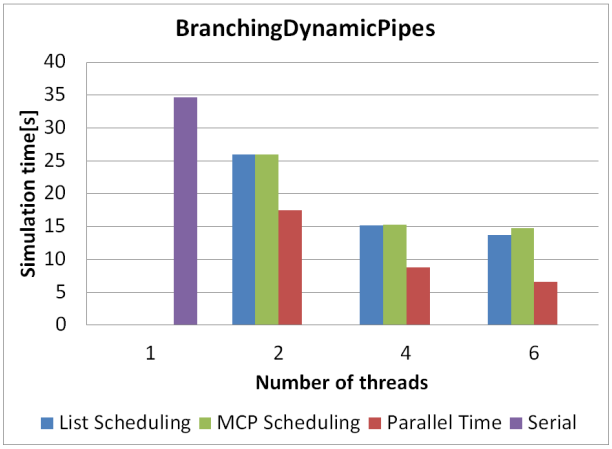

Figure 1: *Benchmark of the dynamic pipes example*

References

[1] P. Aronsson, *Automatic Parallelization of Equation-Based Simulation Programs*. Institutionen för datavetenskap, 2006.

[2] F. Casella, "A strategy for parallel simulation of declarative object-oriented models of generalized physical networks," *Linköping University Electronic Press, Linköpings universitet*, 2013.