Master thesis

Dynamic analysis of complex multibody systems using parallel computing

Piotr Poskart

Field of study: Automatic Control and Robotics, Speciality: Robotics

Academic year 2016/2017

Supervisor: dr inż. Paweł Malczyk

1. Introduction

The aim of this thesis was to analyze complex multibody systems using divide and conquer algorithm (DCA) implemented by the author. DCA algorithm is based on decomposition of the complex problem, related with dynamic analysis of large kinematic chains, into simplified subtasks. In parallel implementation this algorithm has $O(logn)$ time complexity on $O(n)$ parallel processors.

Using parallel computing algorithms of this type have a lot of applications e.g. in molecular dynamics to simulate interactions between particles or in robotics to analyze robot dynamics and to control them.

This thesis includes implementation of the DCA algorithm and tests of the implemented parallel algorithm efficiency on computer with multi CPUs and shared memory. The results were compared with the results of commercial applications for dynamic analysis. Accuracy of derived results and energy conservation of the systems were analyzed. The results of the dynamics analysis of simple systems composed of several members and systems composed of several thousand elements were also compared.

2. Implementation

The figures below shows idea of creating subassemblies in the DCA algorithm and distribution of parallel CPU threads to compute next bodies of the mechanism. Algorithm was implemented in *C++* and fine-grain parallelism was applied using barriers of *openMP* directives.

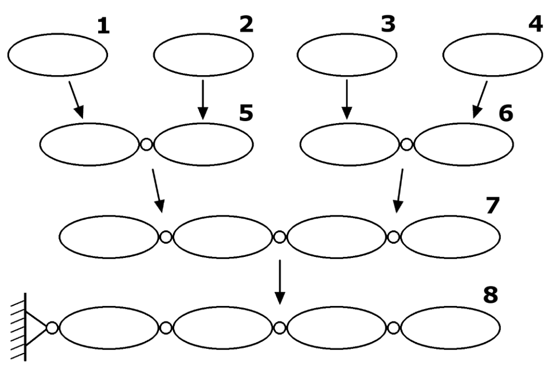

Figure 1: Recursive construction of a binary tree of subassemblies

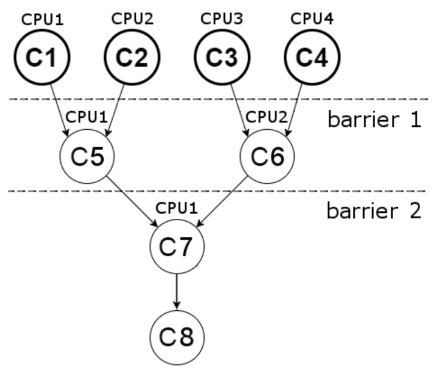

Figure 2: Distribution of threads for calculations in subassemblies

3. Numerical results

Numerical tests allowed to investigate the speedup coefficient of calculations S_p as a ratio of the algorithm's execution time in the sequential version to the time of algorithm calculations in the parallel version. The fastest speedup $S_p = 6.29$ was obtained for 16 threads of the 8-core *Intel Xeon* processor used in the tests.

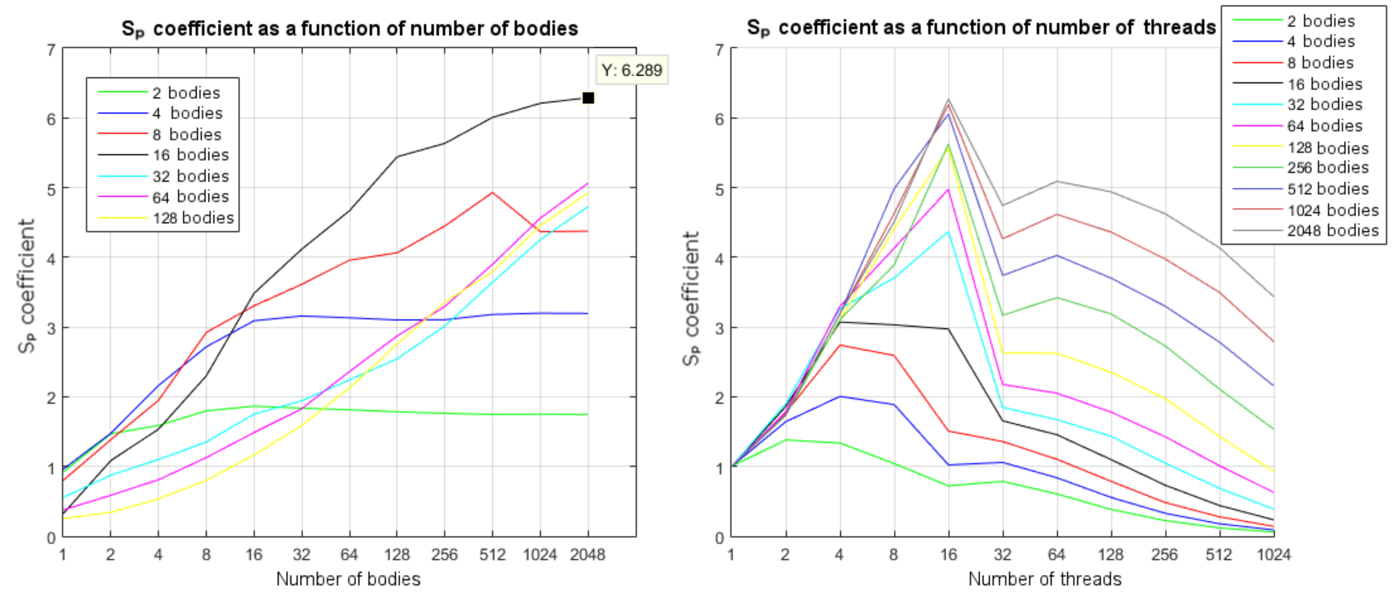

Figure 3: Figures of S_p speedup as a function of number of bodies and CPU threads

During the tests, we also observed an increase in the proportion of parallel calculations η_p in relation to the sequential part of the algorithm η_s $(\eta_p + \eta_s = 1)$. The figure 4 represents the share of the parallel calculations in the test of 16 CPU threads as a function of the number of bodies.

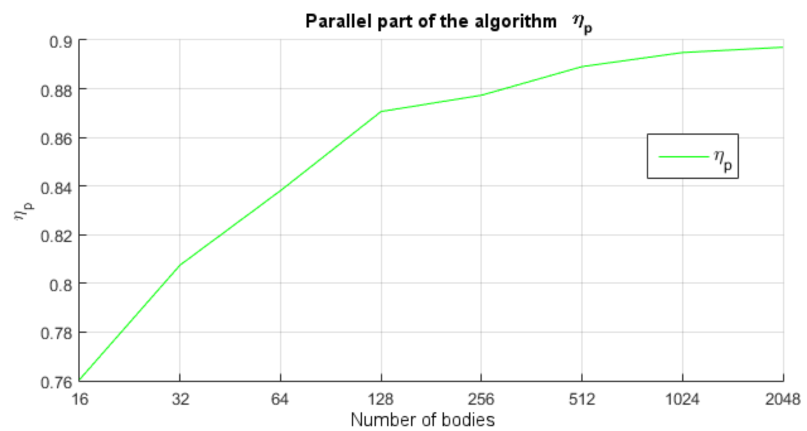

Figure 4: Parallel calculations η_p as a function of number of bodies

Additional observations

The maximum value η_p during the tests was obtained for the 16 bodies in dynamics computation using 2 CPU threads. It was $\eta_p = 0.93$. On this basis, the maximum possible speedup of the algorithm was estimated: $S_p^{max} \approx 14$.

4. Conclusions

- Efficient, parallel DCA algorithm for analysis of the dynamics of multibody systems was implemented, resulting in accuracy comparable with accuracy of the commercial applications
- The properties of the algorithm as a function of problem size and the number of parallel threads were studied
- Significant speedup S_p has been obtained, maximum speedup of the algorithm S_p^{max} was estimated

FACULTY OF POWER AND AERONAUTICAL ENGINEERING
WARSAW UNIVERSITY OF TECHNOLOGY
www.meil.pw.edu.pl

DIVISION OF THEORY OF MACHINES AND ROBOTS
INSTITUTE OF AERONAUTICS AND APPLIED MECHANICS
ztmir.meil.pw.edu.pl