Presentation Abstract

Parallelization of Matrix Partitioning in Construction of Hierarchical Matrices Using Task Parallel Languages

ZHENGYANG BAI[1,a]   TASUKU HIRAISHI[2]   HIROSHI NAKASHIMA[2]
AKIHIRO IDA[3]   MASAHIRO YASUGI[4]

Hierarchical matrix ($\mathcal{H}$-matrix) is an approximated form to represent $N \times N$ correlations of N objects. $\mathcal{H}$-matrix construction is done by partitioning a matrix into submatrices, followed by calculating element values of these submatrices. This paper proposes implementations of the matrix partitioning using task parallel languages, Cilk Plus and Tascell. The matrix partitioning is divided into two steps: cluster tree construction by dividing objects into clusters hierarchically, and block cluster tree construction by finding out all cluster pairs that satisfy an admissiblity condition from the cluster tree. Because the cluster tree constructed and traversed in these steps is unpredictably unbalanced, it is expected that we can parallelize these both steps efficiently using task parallel languages. To obtain sufficient parallelism in the cluster tree construction, we not only execute recursive calls in parallel, but also parallelized inside of each recursive step. In the block cluster tree construction, we assigned each workers its own space to store the cluster pairs without using locks. As a result, comparing to a sequential implementation in C, we achieved a 11.8 times speedup using Cilk Plus for the cluster tree construction and a 38.8 times speedup using Tascell for the block cluster tree construction, both on a 36-core Xeon processor.

---

[1]   Graduate School of Informatics, Kyoto University, Kyoto 606–8501, Japan
[2]   Academic Center for Computing and Media Studies, Kyoto University, Kyoto 606–8501, Japan
[3]   Information Technology Center, the University of Tokyo, Bunkyo, Tokyo 113–8658, Japan
[4]   Department of Artificial Intelligence, Kyushu Institute of Technology, Iizuka, Fukuoka 820–8502, Japan
[a]   haku.shouyou.63a@st.kyoto-u.ac.jp