

領域分割法のマルチコアPCクラスタ向け性能最適化

河合浩志(東大)、荻野正雄(九大)、
塩谷隆二(東洋大)、吉村忍(東大)

Performance Optimization of Domain Decomposition Method for Multi-core PC Cluster

Hiroshi Kawai (Univ. of Tokyo), Masao Ogino (Kyushu Univ.), Ryuji Shioya (Toyo Univ.) and Shinobu Yoshimura (Univ. of Tokyo)

キーワード: 領域分割法, 構造解析, マルチコア, PCクラスタ, 有限要素法
(Domain Decomposition Method, Structural Analysis, Multi-core, PC Cluster, Finite Element Method)

1. はじめに

領域分割法 (Domain Decomposition Method: DDM) は有限要素法 (FEM) による数値シミュレーションの並列化のための有効な方法の一つである。一般にDDMでは、解析領域全体を複数の部分領域に分割し、これら部分領域ごとに仮の境界条件を与えてそれぞれ有限要素解析を行う。そして、仮に与えられた境界条件による領域間の不釣合いを反復計算により順次補正していく。すでに本手法を用いた数億自由度規模の大規模構造解析がオープンソースCAEシステムADVENTURE [1]により実現されている。

一方で、近年HPC環境の主流となった大規模PCクラスタにおいて、ノード内並列性能がますます重要になってきている。特にCPUごとのコア数が増加していく中で、有限要素法を始め従来の数値計算アルゴリズムを来るべきメニーコア時代に適応させていく必要が生じている。

本研究では、DDMにおける主要な計算負荷である部分領域ごと有限要素計算について、そのマルチコアアーキテクチャ向け各種ソルバーを開発し、またこれらのマルチコアPCクラスタでの性能評価を行う。

2. 部分領域FEMソルバー

ここでは、本研究において開発されたDDMにおける部分領域FEMソルバーについて紹介する。これらはみなキャッシュやSIMD拡張命令の利用を考慮したマルチコアCPUアーキテクチャ向け実装となっている。

[DS: Direct Storage] 直接法ベースで、係数行列やLDL分解結果などをメインメモリ上に保存しておく。

[DSF: Direct Storage-Free] 同じく直接法ベースで、DDM反復ごとに毎回キャッシュ上で係数行列のアセンブルや行列分解などを行う。

[IS: Iterative Storage] CG法などの反復法ベースで、係数行列や前処理行列などをメインメモリ上に保存しておく。

[ISF: Iterative Storage-Free] 同じく反復法ベースで、DDM反復ごとに毎回キャッシュ上で係数行列や前処理行列を作成する。

このうち、DSF、ISおよびISFについては、CPU内のキャッシュ上で計算の大部分が行われるため、メインメモリへのアクセス頻度が低く、マルチコアCPUアーキテクチャ向けの実装となっている。ただし、それぞれキャッシュ容量制限に注意する必要がある。

大規模PCクラスタでの計算ノードあたりの実効効率を評価するための第一歩として、前記の各種部分領域FEMソルバーについて、インテル社の4コアCPU、Core i7 940 (2.93GHz)を搭載したPCでの性能評価を行った。なお、SSE命令を用いるためのSIMDベクトル化を行っている。ベンチマーク結果として、DSFではピーク性能の6割を達成した。また、ISで2割、ISFで16%程度となった。また、本来メモリアクセスインテンシブなDSについても、その境界条件処理および反力評価部分についてこれをElement-by-Element計算に置き換えることで計算時間の短縮を実現するとともにピークの2割から3割の性能を達成した。

発表ではこれらの各種ソルバーについてのより詳細な性能分析を報告するとともに、東大T2Kなどの大規模PCクラスタにおけるDDMコードの並列性能およびピーク性能比について議論する。

参考文献

- [1] ADVENTURE プロジェクトHP:
<http://adventure.sys.t.u-tokyo.ac.jp/jp/>