

# H行列ベクトル積のスレッド並列化手法に関する性能評価

川村 卓人<sup>1</sup>, 深谷 猛<sup>2</sup>, 岩下 武史<sup>2</sup>, 伊田 明弘<sup>3</sup>

<sup>1</sup> 北海道大学 大学院情報科学研究科, <sup>2</sup> 北海道大学 情報基盤センター,

<sup>3</sup> 東京大学 情報基盤センター

e-mail : iwashita@iic.hokudai.ac.jp

## 1 はじめに

近年, H行列法(階層型行列法) [1] が活発に研究されている. 例えば, 境界要素解析の高速化のために H行列を用いる事例が報告されているが, この場合, H行列を係数とする連立一次方程式を反復法で解くことになる. 本研究では, その過程で必要となる, H行列とベクトルの積(H行列ベクトル積)の計算に着目し, 最近のマルチコアCPU環境におけるスレッド並列化手法について検討する.

## 2 H行列の概要

本研究では, 分散H行列ライブラリ HACApK [2] において生成された H行列を扱う. 本ライブラリで生成された H行列はブロック(以降, 葉行列)に分割された構造を有しており, 各葉行列は, フルランクの密行列, または低ランクの行列(2個の長方形行列の積で表現)となっている. なお, H行列が有用となる応用では, 大半の葉行列が低ランク行列となっており, 行列の次元に対して, データサイズや関連する演算量が相対的に小さくなる.

## 3 H行列ベクトル積のスレッド並列化手法

H行列  $A$  とベクトル  $x$  の積  $y = Ax$  を考える. H行列の構造から, この計算は,  $A$  の各葉行列に関して  $x$  の対応する部分との積を計算し, その結果を  $y$  の該当部分に加える, という形で処理できる. ここで, 各葉行列と  $x$  (の一部)との積は独立して計算が可能である. そのため, 葉行列をスレッドに分配し, それぞれのスレッドで独立して  $y$  の断片を計算し, 最後に  $y$  に関して縮約処理を行うことで, H行列ベクトル積をスレッド並列化することができる. 実際, HACApK や文献 [3] でも, このアプローチで H行列ベクトル積をスレッド並列化している.

上述の手法でスレッド並列化をした場合, 各葉行列ごとの演算量が異なるため, スレッド間の負荷の不均一を抑えることが重要になる. この点に対して, HACApK では, 最初に演算量

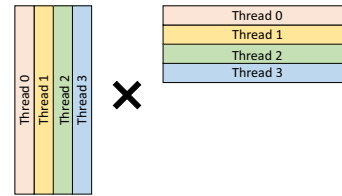


図1. 低ランク葉行列のスレッド分割手法(4スレッド時)

を見積もり, それがスレッド間で出来るだけ均一になるように葉行列の分配を行っている. 一方, 文献 [3] では, 最初に, 演算量がある閾値以上となる葉行列を全スレッドで処理し, その後, 残りの葉行列に関して動的負荷分散(OpenMPのdynamic)を用いてスレッドに分配して処理をする方法を提案している.

どちらの手法についても, スレッド数を変化させた場合の性能等が十分に調査されているとは言えず, 特に最新のマルチコア・メニーコアプロセッサ上の実装については検討の余地が残っている. 例えば, 図1のような低ランク葉行列のスレッド分割手法は, 棟形らの方法と比べてスレッドの同期回数を削減することができるため, 一考の余地がある. そこで, 本研究では, スレッド並列化手法についてより詳細な性能評価・検討を行う. なお, 性能評価結果については, ポスター発表当日に報告する.

## 参考文献

- [1] W. Hackbusch: A Sparse Matrix Arithmetic Based on H-matrices. Part I: Introduction to H-matrices, Computing, Vol. 62, No. 2, pp. 89–108 (2009).
- [2] A. Ida et al.: Parallel Hierarchical Matrices with Adaptive Cross Approximation on Symmetric Multiprocessing Clusters, Journal of Information Processing, Vol. 22, No. 4, pp. 642–650 (2014).
- [3] 棟形克己 他: 動的負荷分散による階層型行列計算の並列化, 情報処理学会研究報告, Vol. 2015-HPC-148, NO. 5, pp. 1–15 (2015).