

# 疎行列ベクトル積に対する OpenMP スケジューリング方式の分析

菱沼利彰<sup>†1</sup> 田中輝雄<sup>†1</sup> 長谷川秀彦<sup>†2</sup>

## 1. はじめに

われわれは、倍精度行列 A と倍々精度ベクトル  $x$  の積を高速化している[1]。4 並列環境において OpenMP のスケジューリング方式 `guided` を用いることで最大 3.7 倍マルチスレッド化の倍率が向上することがわかった。

しかし、いくつかの疎行列において性能がでない。たとえば行数  $N = 40798$ ,  $nnz = 2398220$  の疎行列 `TSOPF_FS_b162_c4` を `guided` で実行したときは約 2.2 倍、`static` では 1.5 倍にしかならない。これは各スレッドの担当する非零要素の数に偏りがあるためである。

本研究では、最適なスケジューリング方式を決定するために、the University of Florida Sparse Matrix Collection (<http://www.cise.ufl.edu/research/sparse/matrices/>) から取得した 1699 の倍精度疎行列と倍精度、倍々精度ベクトルの積に対し各スケジューリング方式の効果を分析した。

## 2. 実験

実験には Intel Core i7-2600K 3.4GHz 4core, DDR3-1333 16GB, Fedora16, Intel C/C++ Compiler 12.0.3 を用いた。コンパイルオプションは `"-O3 -xAVX -openmp -fp-model precise"`, OpenMP のバージョンは 4.6.3 である。行列の格納形式は CRS[2] である。

図 1 に、マルチスレッド化の倍率を示す。横軸は、各行の非零要素の数を標本とした分散である。

`static` は分散の大きい問題で性能が低く、`dynamic` と `guided` は分散の小さい問題で性能が低い。また、最も性能が良いものは、`static` が 1036, `dynamic` が 7, `guided` が 656 であった。分散の大きい問題ではどの手法も性能がでにくく、最も分散の大きい問題では、`guided` を用いても 2.3 倍であった。

性能が低い問題の多くは各スレッドの担当する非零要素のバランスが悪いため、担当する非零要素数を均等化するために、担当行数を決定する `balanced` 方式を導入した。最も性能が良いものは、`static` は 957, `dynamic` は 0, `guided` は 633, `balanced` は 109 となった。また、最も分散の大きい問題において、`balanced` を用いることで `guided` の 2.3 倍を 2.8 倍にできた。

これから、以下のような分散の値を用いたスケジューリング方式の切りかえ、`mix` を導入した。

```
if (variance  $\geq 5 \times 10^4$ ) then balanced
else if (variance  $\geq 10^3$ ) then guided
otherwise static
```

`mix` における分散、`balanced` における前処理の計算時間を含め、すべての疎行列ベクトル積を 100 回計算したときの時間は、最適なスケジューリング方式が 269 秒、`static` は 307 秒、`dynamic` は 549 秒、`guided` は 280 秒、`balanced` は 287 秒、`mix` は 278 秒だった。

`guided` の性能は最適なスケジューリング方式と比べ約 4% しか差がなく、非常に高いことがわかる。分散が 50000 以上の 76 問題において `mix` は `guided` より 59 個速く、全体で約 11% の高速化でき、分散が 100 以下の 1100 問題において `mix` は `guided` より 732 個 `guided` より速く、全体で約 3% の高速化ができた。

これらの結果から、分散によって疎行列ベクトル積のスケジューリング方式を使いわけることの有用性を示した。倍精度疎行列と倍精度ベクトルの積、さらに多くのマルチコア環境における結果はポスターで示す。

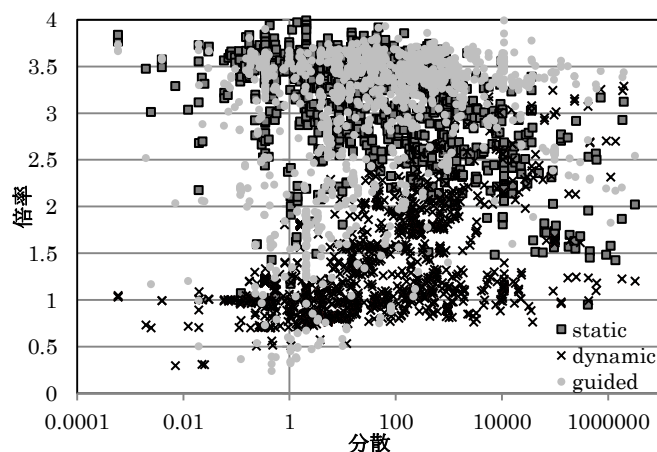


図 1 各スケジューリング方式の性能

## 参考文献

- [1] 菱沼, 他: AVX を用いた倍々精度疎行列ベクトル積の高速化, HPCS2013, Vol. 2013, pp.23-31 (2013)
- [2] Barrett, R., et al.: Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, SIAM (1994)

<sup>†1</sup> 工学院大学情報学部  
Faculty of Informatics, Kogakuin University

<sup>†2</sup> 筑波大学図書館情報メディア系  
Faculty of Library, Information and Media Science, University of Tsukuba