1P07

並列分散処理向きハウスホルダ変換の実験

○ 村上 弘
首都大学東京・数理情報科学専攻（〒192-0397 八王子市南大沢1-1）

要約：　ハウスホルダ直交変換は行列数値計算に於いて直交分解（ベクトルの組の正規直交化，行列の QR 分解），固有値解析（対称行列の三重対角化，非対称行列のヘッセンベルグ化），特異値解析（行列の上二重化），などに利用され重要である．そこで，大規模行列のハウスホルダ直交変換を並列分散システム上で効率良く処理する方法を検討する．

記憶の参照局所性を高め，分散並列化への適合度を向上させることを狙い，通常のハウスホルダ型直交変換を行列の領域分割に沿って二段化し，プロセッサ間相互の通信量や処理の同期頻度を低減する．方法を再帰的に適用すると，多段化も可能である．

算法

$m{\times}n$ 行列 A は縦長（$m{\gg}n$）とする．今回の説明では省略するが，必要なら各段の QR 分解に列ピボット操作を加えることもできる

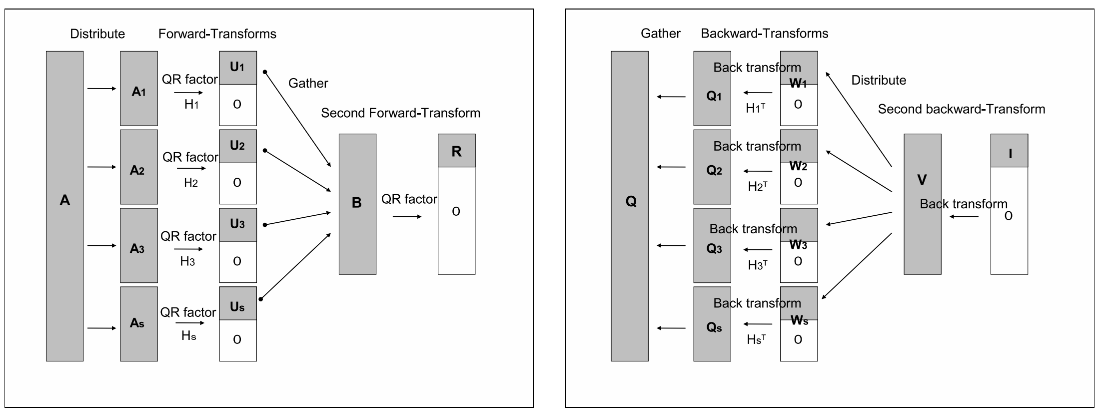

図 1: 正変換の計算，逆変換の計算

上三角化（正変換，図 1 左）

- **第一段目：** A の行方向を s 個に分けて，小行列を $A^{(1)}, A^{(2)}, \ldots, A^{(s)}$ とする．$A^{(i)}$ は $m_i{\times}n$ 行列で，$m=\sum_{i=1}^{s} m_i$，但し $m_i{\ge}n$ とする．$A^{(i)}$，$i=1,2,\ldots,s$ のそれぞれに対して独立に，左側から n 個の鏡映変換 $h_1^{(i)}, h_2^{(i)}, \ldots, h_n^{(i)}$ を順に適用して上三角化を行ない，得られる $m_i{\times}n$ 行列を $U^{(i)}$ とする（$H^{(i)} \equiv h_n^{(i)} \cdots h_2^{(i)} h_1^{(i)}$ と書けば，$H^{(i)}A^{(i)} = U^{(i)}$ である）．$U^{(i)}$ の行は先頭の n 個を除いて全て $\mathbf{0}$ である．

- **第二段目：** s 個の行列 $U^{(i)}$，$i{=}1,2,\ldots,s$ の各先頭の n 個の行を集めた $(sn){\times}n$ 行列を B とする．B を再び直交変換を用いて QR 分解し，$B=VR$ とする，但し V は列直交な $(sn){\times}n$ 行列で，R は $n{\times}n$ の上三角行列．この R が行列 A の QR 分解で得られる上三角行列である．

A を上三角化する左側からの直交変換は，二種類の操作「直交変換 $H^{(i)}$，$i{=}1,2,\ldots,s$」,「B の上三角化で用いた直交変換」を（行の添字の対応を考慮して）順に適用したものになる．それらは，陰的な表現で保持される．

陽的な Q の構成（逆変換，図 1 右）

A の QR 分解 $A=QR$ の Q を陽的な構成は次のようにする．V の sn 個の行を順番に「連続する n 個ずつの行」の組 s 個に分ける．$i{=}1,2,\ldots,s$ に対して第 i 番目の組を $m_i{\times}n$ 行列 $W^{(i)}$ の先頭の n 個までの行とし，残りの行には $\mathbf{0}$ を入れる．次に $H^{(i)}$ の逆変換を $W^{(i)}$ に作用させ，$m_i{\times}n$ 行列 $Q^{(i)} = H^{(i)T}W^{(i)}$ を作る（保存しておいた $A^{(i)}$ の三角化に用いた直交変換 $H^{(i)}$ の情報により，$W^{(i)}$ の左側から逆変換 $H^{(i)T}$ を作用させる）．$Q^{(i)}$，$i{=}1,2,\ldots,s$ を縦に並べた $m{\times}n$ 行列が求める Q である．

実験

本方法を用いて HA8000 の 1 ノード（AMD Opteron 8356，2.3GHz，4CPU（=16 コア））のシステム上で，コア数を変えて測定した例を示す（表 1，図 2）．コアあたりの理論性能上限は 9.2GFlops（倍精度演算）である．コンパイラには intel Fortran version 10.1 を用いた．

表 1: 実効演算速度，$m = 144000$，縦ブロック長 1200（120 分割）二段階目の QR 分解にも，縦分割ブロック化を適用．倍精度，Gflops 値

N	1core	2core	3core	4core	5core	6core	8core	10core	12core	15core
10	2.01	3.93	5.72	7.07	7.89	9.35	11.7	14.5	17.2	20.6
20	2.37	4.71	6.89	8.89	10.9	12.5	15.2	18.6	21.3	25.2
30	2.54	5.03	7.49	9.76	11.9	13.4	17.5	21.2	24.5	28.8
40	2.63	5.25	7.65	10.2	12.4	14.5	18.7	22.5	25.9	30.3
50	2.65	5.23	7.78	10.1	12.9	14.9	19.0	22.2	26.6	31.7
60	2.57	5.12	7.61	9.91	12.0	14.4	18.7	22.5	26.4	31.4
70	2.49	4.90	7.24	9.44	11.7	13.5	18.3	22.0	25.8	30.7
80	2.39	4.76	7.02	9.40	11.4	13.2	17.6	21.3	25.0	29.5
90	2.32	4.57	6.79	8.81	11.1	12.6	16.1	20.7	24.1	28.9
100	2.28	4.50	6.60	8.65	10.9	12.6	15.8	20.2	23.1	28.3

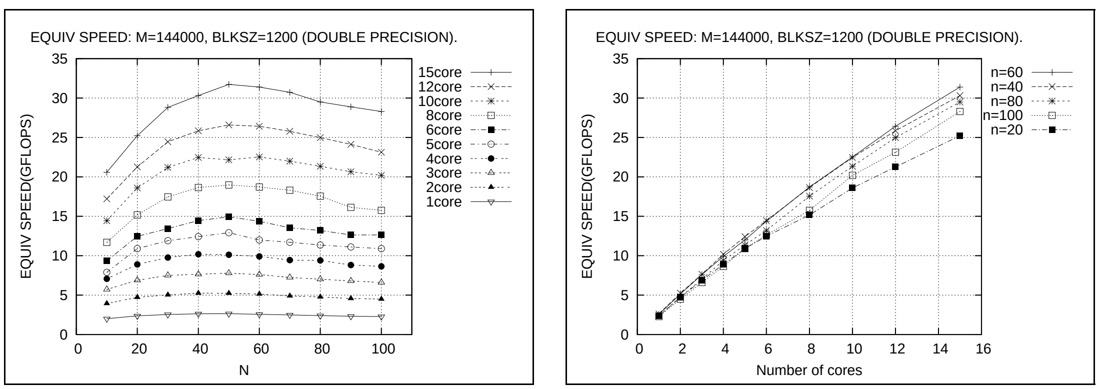

図 2: 実効演算速度，$m = 144000$，縦ブロック長 1200（120 分割）

既に良く知られた，複数の基本鏡映変換を蓄積して各列に施す技法，さらに蓄積した基本鏡映変換を変形しブロック変換化する技法を今回の方法と併用すると，さらに高速化ができる．

詳しいことは当日の会場のポスターで解説する．

参 考 文 献

[1] Bischof C. and Van Loan C.: The WY representation for products of Householder matrices, SIAM J. Sci. Stat. Comput., **8**, No.1, (1987), 2–13.

[2] Dongarra J. J., Hammarling S. J. and Sorensen D. C., LAPACK Working Note #2; Block Reduction of Matrices to Condensed Forms for Eigenvalue Computations, ANL/MCS-TM-99, (1987).

[3] Golub G. H. and Van Loan C. F., Matrix Computations, 3rd ed., The John Hopkins University Press, Baltimore London, 1996.

[4] Schreiber R. and Van Loan C.: A storage efficient YW representation for products of Householder transformations, SIAM J. Sci. Stat. Comput., **10**, No.1, (1989) 53–57.

[5] Stathopoulos A., and Wu K., A block orthogonalization procedure with constant synchronization requirements, SIAM J. Sci. Comput., **23**, No.6 (2002), 2165–2182.

[6] Wilkinson J. H. and Reinsch C., Handbook for Automatic Computation, Vol.II, Linear Algebra, Springer-Verlag, Berlin Heidelberg New York, 1971.