AP1000を用いた下三角疎行列の直接解法

5F−4

川北 淳平, 野寺 隆
慶應義塾大学 理工学部

1 はじめに

下三角疎行列Lを係数とする方程式$Lx = b$を, 複数個の右辺bについて解く必要がある場合に, Lの逆行列を利用する並列解法が知られている (Alvarado et al.[1], 川北 [5] 参照).

この解法では, 下三角疎行列Lを次のように分解する.

$$L = L_1 L_2 \cdots L_n \tag{1}$$

ただし

$$\begin{aligned}(L_k)_{ik} &= L_{ik} && for \quad k \le i \le n \\ (L_k)_{jj} &\equiv 1 && for \quad j \ne k \\ (L_k)_{ij} &\equiv 0 && otherwise\end{aligned}$$

である.

ここでL_kをいくかのグループに分けて, 行列Lを次のように表すことができる.

$$L = \prod_{k=1}^{m} P_k \tag{2}$$

このとき, 各P_kの逆行列P_k^{-1}を計算しておけば, 複数の右辺bについて,

$$x = L^{-1} b = \prod_{k=m}^{1} P_k^{-1} b \tag{3}$$

と解くことができる.

Lの逆行列L^{-1}は, 一般に下三角疎行列Lよりも非ゼロ要素が多くなってしまうが, 各P_kに対し, その逆行列P_k^{-1}が新たな非ゼロ要素を生成しないようなグループ分けを考えることにより, 行列Lの疎という性質を生かすことができる.

また, L^{-1}をこのように分解することにより, 並列計算機においては, $P_k^{-1}x$の計算に必要なすべての乗算を同時に行なうことができるので, 並列計算機において高速に処理することができる.

この解法では, グラフの概念をとり入れた前処理により, 行列Lの分解 (2) を決定するが, 複数の右辺bを想定しているので, 分解アルゴリズムよりも, 式 (3) の計算を高速に行なうことが重要となる.

実際の式 (3) の計算では, 行列P_k^{-1}の各非ゼロ要素にたいして1回ずつ乗算をおこなうので, 並列計算機のセルの個数が, 各P_k^{-1}の非ゼロ要素数を上回っていれば, 全非ゼロ要素を別々のセルに配置して, $P_k^{-1}b$の乗算をワンステップですますことができる. この場合, 式 (3) の全対の計算量は, グループ数mに依存する.

したがって, セル数などの制約がなければ, グループ数を減らすことが有効である.

2 分解アルゴリズム

下三角行列は, DAG(*Directed Acyclic Graph*) によってあらわすことができる. 下三角行列Lの有向グラフを$G(L)$とすると, $G(L)$の頂点V, 有向辺Eは,

$$\begin{aligned}V &= \{1, 2, \cdots, n\} \\ E &= E(L) \equiv \{(j, i) \mid L_{ij} \ne 0\}\end{aligned}$$

である. ただし, 有向辺(j, i)は, 頂点jから頂点iへ伸びる矢印である. すべての有向辺(j, i)について$j < i$という関係が成り立つ.

このDAGを用いて, 下三角行列Lを分解するアルゴリズムが, RP1アルゴリズムである ([1]). 図1は, RP1アルゴリズムによって, DAGの頂点をグループ分けしたものである. この頂点のグループ分けに対応して, 下三角行列Lの分解を行なうものである. また, 頂点の番号は, 行列のリオーダリングに相当する.

ところで, 図1の左側の数字は, 頂点のレベルを表しているが, 図2に示すように、下三角行列は, それと対応するDAGにおける最大レベルと同じ数の行列に分

Parallel Solution of Triangular Systems for Parallel Computer
Junpei Kawakita, Takashi Nodera
Faculty of Science and Technology, Keio University
3-14-1 Hiyoshi kōhoku, Yokohama, Kanagawa, 223 Japan

解することができる．つまり，同じレベルに存在する頂点間には辺が存在しないので，同レベルに属する頂点の集合をひとつのグループとすることができるからである．また最大レベルにおける頂点と，そのひとつ前のレベルに属する頂点をひとつのグループとすることができる．

このように分解すると，分解された下三角行列の数 m は，せいぜい RP1 アルゴリズムと同じで，多くの場合はるかに大きくなってしまう．

ところで，本解法は下三角行列 P_k の逆行列を求める必要があるが，このようなレベルによる分解では，P_m をのぞく P_k $(k = 1, \cdots, m-1)$ の逆行列は簡単に求めることができるという長所がある．

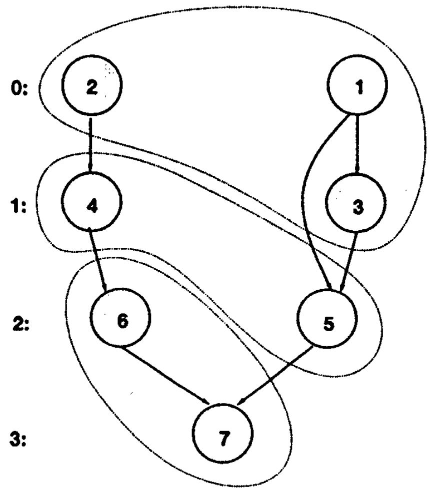

図 1: RP1 アルゴリズム

3 まとめ

AP1000 では，2 次元トーラス構造をつかって，$P_k^{-1}b$ を計算する時の加算をグローバル演算することが考えられるが，扱う問題が疎行列であるため，データ配置の方法もいろいろ考えられる．セルの数が，分解された各行列 P_k の非ゼロ要素数よりも少ない場合，どのようにセルの負荷を分散させるかということも，重要な問題であると考えられる．

当日,AP1000 を使った数値実験について詳細に報告する．

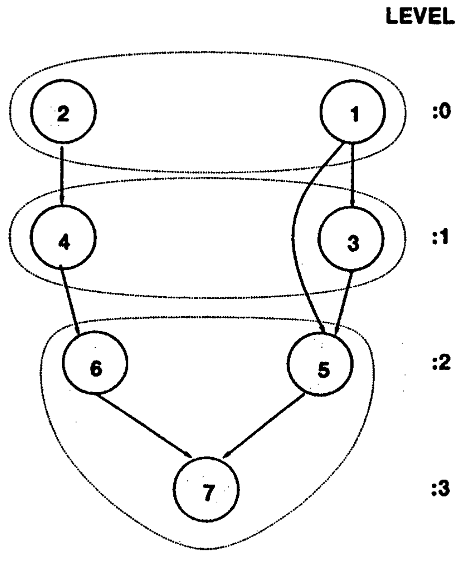

図 2: レベルによる分解

参 考 文 献

[1] Fernando L. Alvarado, Robert Schreiber, Optimal Parallel Solution of Sparse Triangular Systems, SIAM J. SCI. COMPUT. Vol. 14, No.2, pp. 446-460, March 1993

[2] R. Betancourt, An Efficient Heuristic Ordering Algorithm for Partial Matrix Refactorization, IEEE Trans. Power Systems, 3 (1988), pp. 1181-1187

[3] Fernando L. Alvarado, David C. Yu, R. Betancourt, Partitioned Sparse A^{-1} Methods, IEEE Trans. Power Systems, 5 (1990), pp. 452-459

[4] J.J. Du Croz, N.J.Higham, Stability of methods for matrix inversion, IMA J. Numer. Anal., 12 (1992), pp. 1-19

[5] 川北 淳平, 直接法における下三角疎行列の並列解法について, 情報処理学会研究報告 HPC 51(1994)