1X-1

ベクトル計算機向き逐次代入型計算の高並列化

○吉原　郁夫　　村松　晃　　中尾　和夫
（㈱ 日立製作所　システム開発研究所）

1．はじめに

大規模シミュレーションでは、偏微分方程式の数値解を求める過程で現われる、連立一次方程式の求解に多大の時間を必要とする。差分法や有限要素法で離散化した場合、係数行列はスパースとなるが、その非零要素が多い特徴を生かすため、反復解法がよく用いられる。

反復解法のうち高収束なものは、たいてい逐次代入型計算を含んでいる。例えば、SOR法、ICCG法、ILUCR法などがそうである。この型の計算は、そのままではベクトル計算機の性能を出しにくいため、リストベクトルを用いてデータ参照関係を適正化して、ベクトル化する方法が採られる[1]。

しかし、それでもまだ格子点計算の、始めの部分と終わりの部分のループが短いため、やや効率が悪いという問題が残っている。この点を解決すべく、複数の端点から計算を始めることにより演算の並列度を高め、この部分のループを長大化するとともに、ループ数を削減することを考えた。

2．ＳＯＲ法の高並列化

2.1　従来のＳＯＲ法のベクトル処理法

SOR法のベクトル処理法を、二次元ポアソン方程式 $\Delta U(x,y)=\phi(x,y)$ の求解を例に説明する。SOR法では、U_{ij}を DOループのインデックス順に次のように更新して行く。

$$U^*_{ij} \leftarrow \omega\{\alpha(U^*_{i-1\,j}+U_{i+1\,j})+\beta(U^*_{i\,j-1}+U_{i\,j+1})-\phi_{ij}\}+(1-\omega)U_{ij}$$

（ただし、α，βは定数，ωは加速係数，*印は更新済の記号）

インデックス$_{ij}$順の計算では、更新された値を直ちに使うため、ベクトル化されない。従来は、図1に示す①,②,③,……の斜線上の格子点データごとにループを構成するようにして、ベクトル化していた[1][2]。

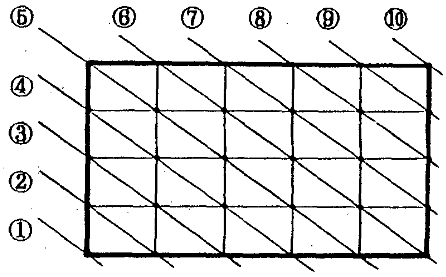

図１　従来のＳＯＲ法計算順序

2.2　高並列な計算リストの構成法

従来法は、左下隅を起点とし、値の更新がインデックス$_{ij}$順の計算と全く同じになるように計算して行く。しかし、どこを起点に計算を始めても同じ値に収束する筈である。例えば、右上隅から左下隅に向い⑩,⑨,⑧,……としても、これは DOループを逆回ししたにすぎないから、同じ値に収束する。

両方の計算順序を併用することを考える(図２)。左下隅と右上隅の両端から、計算を始めると①と⑩，②と⑨，③と⑦，……が対になるが、互いに離れているので独立な計算である。従って、同じループ

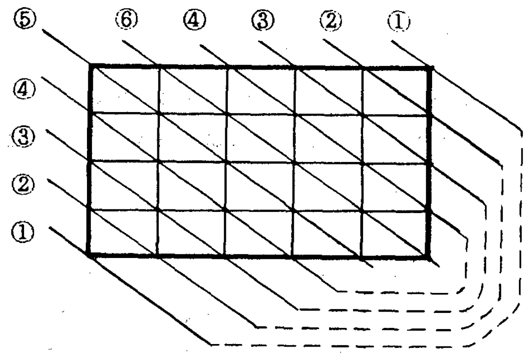

図２　高並列なＳＯＲ法計算順序（起点数２）

Highly Parallel Vectorization of Recurrence Calculatoion
Ikuo YOSHIHARA , Akira MURAMATSU , Kazuo NAKAO
Systems Development Laboratory , Hitachi,Ltd.

にいれてもベクトル化できる。このようにすると、従来法ではループの長さが1,2,3,……と増えて行ったのに対し、2,4,6,……と増えて行き、ちょうど2倍になる(図3)。また、中央の点が求まったとき、全格子点の計算を終了したことになるから、ループ数は約 1/2になる。

さらに上の考えを拡張し、他の隅からも同時に計算を始めることを考える。四隅から計算を始めると、ループ長さは4,8,12,……と増えて行くが、最後の方のループでは段々ループが短くなって行く(図3)。ループ数は両端から始めた場合と同じ[NS/2+1]である。ただし、[ ]はガウスの記号,二次元問題ではNS=NX+NY-1である。

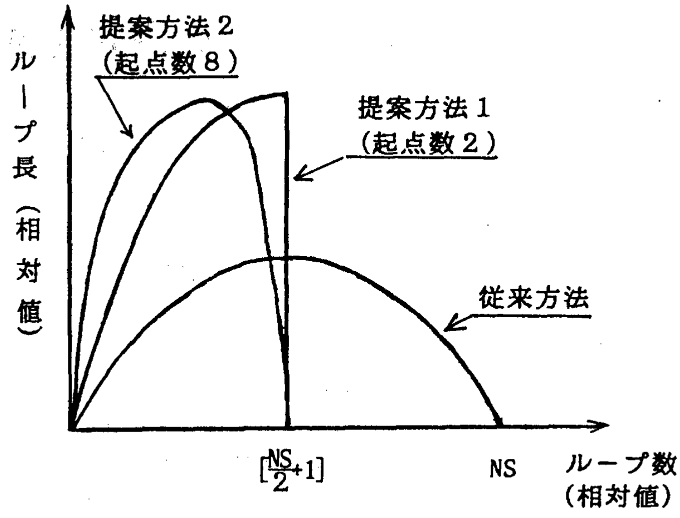

図３ ループの長大化

３．数値実験

現実に解きたい問題の多くは、三次元であるから、三次元ラプラス方程式の求解を例に、提案方法の効果を検証する。

計算起点のとり方は２～８まで可能であが、ここでは、両極端の２と８を採り上げる。空間分割数が約３万(32*32*32)、x,y,z軸方向の差分の刻み幅を等しくし、相対残差ノルムが10^{-10}に達するまで反復計算を行う。表１に示す如く、ベクトル長が長くなり、かつループ数が半減することが確認された。さらに、提案方法１では、収束回数も少なくなる。偏微分方程式の境界値問題は、「周囲の値が内部を決定する」と言える。提案方法１は従来法より、周囲の影響が伝わり易いため収束が速くなる、のではないかと考える。数ケースの実験すべてにわたり、提案方法１は従来法より収束が速い。

４．おわりに

多数の隅から計算を開始することにより、SOR法の演算並列度を高める方法を見出した。二次元問題の場合、計算の起点付近のループ長は従来法の 2~4 倍となり、三次元問題の場合 2~8 倍となる。ループ数はいずれの場合も従来法の約半分である。

ベクトル計算機では、ループが短いと加速率が低いので、短ループの解消は効果的である。特に、日常的に解く、中規模問題に対し有効である。

＜参考文献＞

1) 村田 他:スーパーコンピュータ,共立出版(昭60)

2) 後 他:ベクトル計算機による数値計算、数値解析研究会 2-1, pp1~10(昭 57)

表1 高並列化の効果

方法	ループ数	数値実験例			備考
		ループ数	平均ループ長	収束回数	
従来方法	NS	94	348	108	起点数1
提案方法1	$[\frac{NS}{2}+1]$	48	683	95	起点数2
提案方法2	$[\frac{NS}{2}+1]$	48	683	112	起点数8
備考	NS=NX+NY+NZ-2	空間分割数 32*32*32			