

## 2次元CFDによる並列計算機の検証模擬

原 田 公 一\*      吉 田 正 廣\*      中 村 絹 代\*

### Simulation of a Parallel Computer with a 2-Dimensional CFD Program

by

Koichi HARADA, Masahiro YOSIDA, Kinuyo NAKAMURA  
*National Aerospace Laboratory*

#### ABSTRACT

A simulation program for a parallel computer configured with 128 x 256 processing elements and the assembly language were implemented on the Numerical Simulator in the National Aerospace Laboratory.

The former simulates the memory access and register operation at the clock level and generates time diagrams along with operation results, the latter describes the procedures for the parallel computer.

In this paper, the parallel processing time by 128x256 processing elements is compared with the sequential processing time by the uni-processing element and a evaluating method for the parallel computer with a 2-dimensional computational fluid dynamics program is described.

#### 1. はじめに

将来型スーパーコンピュータの一形態である並列計算機の有用性を検証するため、並列計算機の動作をクロック単位で模擬するシミュレータおよびプログラム記述のためのアセンブラを開発した<sup>1,2)</sup>。

このたび、これらを用いて二次元非圧縮非定常ナビエーストークス方程式による解析プログラム処理時における並列計算機の性能について検証模擬を実施したので報告する。

#### 2. 並列計算機のイメージ

ここで想定した並列計算機は全体を制御・監視

する制御処理部、データ演算を行うデータ処理部および入出力処理部より成り、さらにデータ処理部はスカラデータ演算器とアレイデータ演算器等より成り、後者は128×256台の演算器要素と隣接結合による上下左右のシフトレジスタで終端がリング結合しているネットワーク（アレイデータメモリコントロールユニット）を経由して同数のアレイデータメモリ要素と結合している（図1）。

#### 3. 解析プログラムの概要

時刻  $t=0$  で一様流が円柱まわりに発生し、以後各時刻ごとに流れ場の渦度・流れ関数の計算、表面渦度の補正、表面圧力と抗力・揚力の計算を繰り返して2次元非定常流を追跡する解析プログラムを検証に用いた。図2に汎用機での計算例を

\* 航空宇宙技術研究所

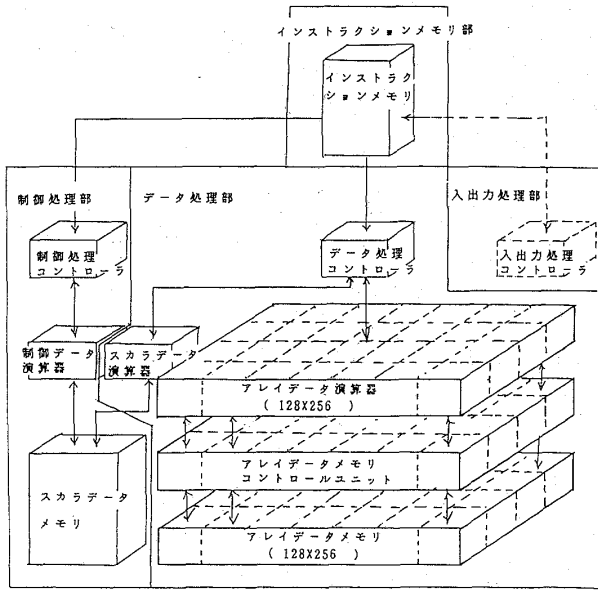


図1 並列計算機の構成

示す。

これを並列計算機に適合させるため、円柱まわりを256等分し、半径方向 $1 \sim e^{\pi}$  ( $\approx 23$ )を等比間隔になるように128分割し、各交点を計算のための格子点とした。また時間の刻み幅を0.0025とした。

なお、このプログラムは文献3を基に対称の条件をはずしたものである。

#### 4. 並列計算機への移植

解析プログラムを並列計算機に適合させる際に遭遇した事項について記す。また図3に移植した流れ関数の計算部を示す。

##### (1) 格子と演算器の対応

格子と演算器・記憶装置の対応付けは並列処理に適合させるための基本的条件であり、単に格子の分解能ばかりでなく、反復法や反復数に与える影響が大きい。2次元の格子を演算器・記憶装置に1対1で対応させると反復法では“旧値のみ”から新値を求めることになり、ヤコビ法を用いることになる。この時加速パラメータを用いることができない。また一方向に隣接する2格子、二方向に隣接する4格子を対応させると演算器を遊ばせることなくゼブラ法、偶奇法を用いることができる。図4に128×256格子点において逐次処理用のSOR法、並列処理用のヤコビ法・ゼブラ法を用

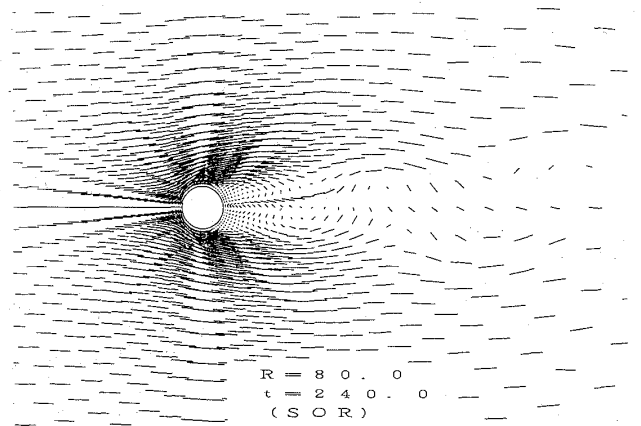


図2 計算例

い、収束条件を $10^{-5}$ 以下とした時の反復回数を示す。各々の加速パラメータは1.95, 1.00, 1.32である。

ヤコビ法・ゼブラ法ではSOR法に比べ初期の反復数が極端に多い。しかし以後、急激・単調に減少し、ついにはSOR法よりも少くなる。時刻をどこまで進めれば流れが落ち着くか定かでないが、計算法別に反復回数が全部で多くなるかどうかは速断できないようである。

##### (2) 並列処理とマスク操作

流れ関数の反復計算時における各格子点の修正量が収束誤差以内であるかを判定する場合、逐次処理では各点の修正量を条件と比較し、次の格子をチェックすることになるのに対し、並列処理では各格子点の収束判定が同時にできる。ただし全格子点が収束条件を満たし、反復計算を打ち切るかどうかを判定するためには各演算器の判定信号を一カ所で監視・判定する必要がある。

図5にその回路を示す。ここでは各演算器にマスク( $MK_{i,j}$ )を設定しておき、収束条件を満たさないものがあれば払う。各演算器のCRの内容がスカラデータ演算器に送られ、その内容が“空”であれば収束と判定できる。

さらに、表面圧力と抗力・揚力の計算には円柱に沿って積分する必要がある。逐次処理ではDOループで累積すれば済むのに対し、並列処理ではシフトとマスク操作による演算拒否の制御が必要である。図6にその様子を示す。図中“\*”はマスクによる演算抑制を示す。

```

679 :
680 :C*****
681 :C
682 :      DO 1700 L = 1 , 9999
683 :C
684 :      DO 1800 I=2,127
685 :      DELTA( I,1 ) = ( PSI( I+1,1 ) + PSI( I-1,1 )
686 :      + PSI( I,2 ) + PSI( I,256 ) ) / 4
687 :      + EE( I ) * ZETA( I,1 ) - PSI( I,1 )
688 :      DELTA( I,257 ) = DELTA( I,1 )
689 :      LPSI( I,1 ) = PSI( I,1 ) + CONST * DELTA( I,1 )
690 :      LPSI( I,257 ) = LPSI( I,1 )
691 :C
692 :      DO 1800 J = 2 , 256
693 :      DELTA( I,J ) = ( PSI( I+1,J ) + PSI( I-1,J )
694 :      + PSI( I,J+1 ) + PSI( I,J-1 ) ) / 4
695 :      + EE( I ) * ZETA( I,J ) - PSI( I,J )
696 :      LPSI( I,J ) = PSI( I,J ) + CONST * DELTA( I,J )
697 :1800 CONTINUE
698 :C
699 :      DO 1730 I = 2 , 127
700 :      DO 1730 J = 2 , 256
701 :      PSI( I,J ) = LPSI( I,J )
702 :1730 CONTINUE
703 :C*****C*****C
704 :C
705 :      L      0,0,K00000
706 :      T      0,0,L
707 :C
708 :A1800 L      0,0,L
709 :      IC     0,0,EMPT
710 :      T      0,0,L
711 :      L      1,0,K01001
712 :      SR     0,1,MINS
713 :      J      0,A1750
714 :C
715 :      MI
716 :      LA      1,0,ECND,MQNG,EMPT,0,C000,C000,I
717 :      SA      1,0,ECND,MQST,ZERO,0,C000,C000,K00001
718 :      SA      1,0,ECND,MQST,ZERO,0,C000,C000,K00127
719 :      FLA     0,0,ECND,MQNG,EMPT,0,C001,C000,PSI
720 :      FAA     0,0,ECND,MQNG,EMPT,0,M001,C000,PSI
721 :      FAA     0,0,ECND,MQNG,EMPT,0,C000,C001,PSI
722 :      FAA     0,0,ECND,MQNG,EMPT,0,C000,M001,PSI
723 :      FDA     0,0,ECND,MQNG,EMPT,0,C000,C000,KF0004
724 :      FSA     0,0,ECND,MQNG,EMPT,0,C000,C000,PSI
725 :      FLA     1,0,ECND,MQNG,EMPT,0,C000,C000,EE
726 :      FMA     1,0,ECND,MQNG,EMPT,0,C000,C000,AZETA
727 :      FARA    0,1,ECND,MQNG,EMPT
728 :      FTA     0,0,ECSR,MQNG,EMPT,0,C000,C000,DELTA
729 :      FMA     0,0,ECND,MQNG,EMPT,0,C000,C000,CONST
730 :      FAA     0,0,ECND,MQNG,EMPT,0,C000,C000,PSI
731 :      FTA     0,0,ECSR,MQNG,EMPT,0,C000,C000,PSI
732 :C
733 :C*****C*****C
734 :C
735 :      DO 1770 I = 1 , 127
736 :      DO 1770 J = 1 , 256
737 :      X1 = CONST * ABS( DELTA( I , J ) )
738 :      IF( X1 .GE. HOLD ) GO TO 1700
739 :1770 CONTINUE
740 :      GO TO 1750
741 :1700 CONTINUE
742 :1750 COUNT( K ) = L
743 :C*****C*****C
744 :C
745 :      MI
746 :A1770 FLA      0,0,ECND,MQST,MINS,0,C000,C000,DELTA
747 :      FLNA     0,0,ECST,MQNG,EMPT
748 :      FMA      0,0,ECND,MQNG,EMPT,0,C000,C000,CONST
749 :C
750 :      FSRA     1,1,ECND,MQST,ZERO
751 :      FLA      1,0,ECND,MQNG,EMPT,0,C000,C000,HOLD
752 :      FSRA     1,0,ECND,MQRS,MINS
753 :C
754 :      FSR      1,1,EMPT          :F2(1)=0
755 :      FSCR     1                :CMNCR2/5=F2(1),CMNCR3/6(K,
756 :      HCR
757 :      FLCR     1                :F2(1)=CMNCR2/5 AND MASK DF
758 :      FMV      1,1,ZERO         :SKIP IF ALL MASK IS SET
759 :      J        0,A1800
760 :A1750 L      0,0,L
761 :      T      0,0,COUNT
762 :C
763 :C
764 :C*****C*****C
    
```

流れ関数の計算

収束判定

図3 移植プログラム例

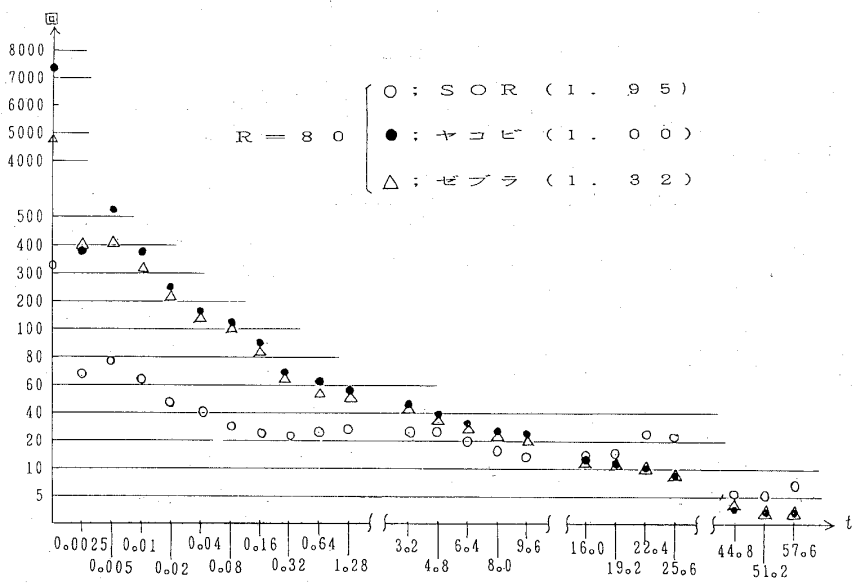


図4 反復回数の変化

(3) 境界条件とマスク操作

この解析プログラムでは渦度・流れ関数の計算時に上下左右の格子点の値を用いるため半径方向の先端(円柱表面)と終端(ここでは $e^*$ )ではその

の値を更新しない。また渦度・流れ関数の計算後、円柱表面の渦度の補正を行う。このため、内側の格子点と境界のそれとでは計算手続きが異なり、処理を拒否するマスク操作が必要である。

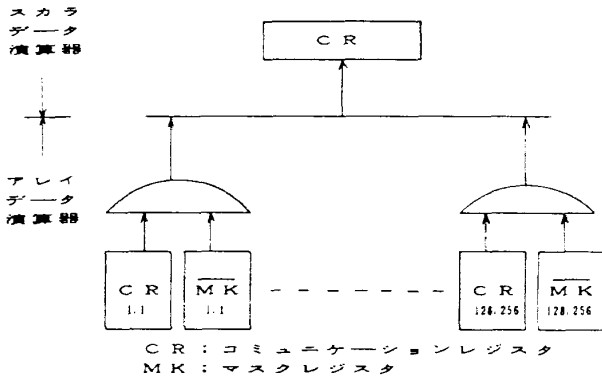


図5 収束の判定回路

5. 検証模擬と考察

一台の演算器による逐次処理と128×256台の演算器による並列処理との相対的処理効率を見極めることを目標とし、解析プログラムの流れを渦度・流れ関数の計算、表面渦度の補正、表面圧力・抗力・揚力の計算に分割し、各々について調べた(表)。

以下に考察を加え、性能向上策について述べる。

(1) 渦度の計算では演算器の台数(128×256台)

$$p(\theta)_{r=1} = \frac{4}{R} \int_0^\theta \left( \frac{\partial \zeta}{\partial \xi} \right)_{\xi=0} d\theta + C$$

$$\begin{cases} C_{Dp} = -\frac{1}{2} \int_0^{2\pi} p(\theta) \cos \theta d\theta \\ C_{Df} = -\frac{2}{R} \int_0^{2\pi} \zeta_{\xi=0} \sin \theta d\theta \end{cases}$$

$$\begin{cases} C_{Lp} = -\frac{1}{2} \int_0^{2\pi} p(\theta) \sin \theta d\theta \\ C_{Lf} = \frac{2}{R} \int_0^{2\pi} \zeta_{\xi=0} \cos \theta d\theta \end{cases}$$

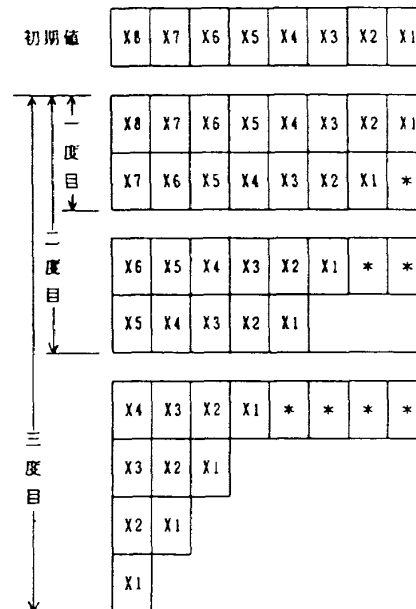


図6 表面圧力・抗力・揚力の計算

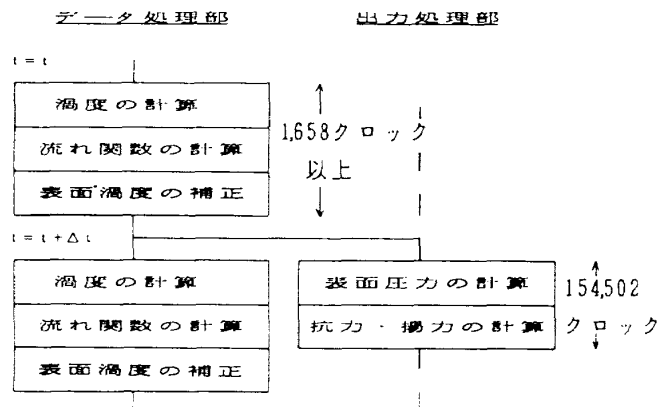


図7 負荷分散の考察

表 処理時間(クロック)

処理内容	逐次処理	並列処理	比
渦度の計算	24,463,783	788	31,045
流れ関数の計算	11,152,098/回	543/回	20,537
表面渦度の補正	77,165	327	235
表面圧力の計算	70,229	1,475	47
抗力・揚力の計算	84,273	3,719	22

に見合う性能が期待できる。

(2) 流れ関数の計算では期待の約6割しか性能が発揮されない。その主な原因は、この内容に収束判定の手続が含まれているためである。逐次処理では収束を満たさない格子点を発見すればただちに反復計算にもどることができ、全点を調べる必要がないのに対し、並列処理ではアレイデータ演算器からスカラデータ演算器に集められた信号を基に収束判定する必要があり、これが処理時間の約4割を占める。したがって、より一層の性能を達成するためには、全格子点での収束判定回路および命令系の最適化を図り<sup>4)</sup>、かつ、プログラム内での流れ関数本来の計算比重を増すことが考えられる。図4に示したように収束のための反復回数は始めは多く、以後減少して行くことから初期には100回単位で収束判定を行い、以後前回の反復回数を参考に収束判定手続の頻度を可変にすることにより実効上の性能向上を図ることができると思われる。

(3) 表面渦度の補正では円柱表面に割り当てられた格子点に対応する演算器のみ(256台)が動作しており、ほぼ予想通りである。しかし、大部分の演算器が停止しているため、計算法の改善が望まれる。

(4) 表面圧力と抗力・揚力の計算は他の部分に比べ極端に効率が低い。その原因は、円柱表面上の格子だけで演算が行われ、かつ円柱表面に沿って積分(累積)計算があるため、データ転送が行われ、各々処理時間の約3割5分、5割5分を占めるためである。したがって、結合網を改良しても約70倍、50倍にしかならないことは注意を要する。

性能向上策としてはこの結果が渦度・流れ関数の計算に使用されないことから他の処理部(例えば入出力処理部)に移動でき、またすべきと思われる。ただし、渦度・流れ関数の計算と表面渦度の補正に要する時間以内でこれらを処理しようとするれば、その処理部はスカラデータ演算器の約100倍の性能を有するものでなければならない(図7)。

(5) 渦度・流れ関数の計算および表面渦度の補正だけを観ると隣接結合の並列計算機による2次元非圧縮非定常ナビエーストークス方程式による円柱まわりの流れ場解析に対しては、約21,000倍の性能が達成できるものと思われる。

なお、並列計算機の構成、演算器、結合網、命令系、計算法等を見直し、より一層の性能向上を図ることが今後の課題である。

謝辞：流れ場解析プログラムの作成にあたり、当所の井上数値シミュレータ研究室長より助言を頂いたことを記し、感謝の意を表する。

## 参考文献

- 1) 原田：“並列計算機のアーキテクチャシミュレータ”，航空宇宙技術研究所資料，TM-583，1988年3月。
- 2) 原田，“並列計算機のアセンブラ”，航空宇宙技術研究所資料，TM-586，1988年6月
- 3) M. Kawaguti & P. Jain: “Numerical Study of a Viscous Fluid Flow Past a Circular Cylinder.” J. Phys. Soc. Japan, 21 (1966) 2055-62.
- 4) 原田：“並列計算機のシミュレーション”，第6回航空機計算空気力学シンポジウム論文集，pp. 49~54，航空宇宙技術研究所，SP-9，1988年12月。

