

トランスピュータによる高速数値演算

松坂知行*

High Speed Numerical Calculation by Transputer

Tomoyuki MATSUZAKA

Abstract

Transputers developed by INMOS are characterized by high speed and parallel operation. The author compared the computational speed of Transputers with that of a 32 bit personal computer, through some examples of matrix calculations and ordinary differential equations. The results showed that the Transputers have excellent performance both on computational speed and parallel operation.

1. まえがき

コンピュータのハードウェア、ソフトウェアの飛躍的な発展により計算機援用解析技術が登場し、従来実験のみに頼っていた研究分野にもコンピュータが適用されるようになってきた。すなわち計算力学、計算物理学、計算化学、計算電磁気学などと呼ばれる研究分野が登場しコンピュータを駆使して、実験では達成不可能であった現象の解明が行われるようになってきた。このような研究は自然科学の分野にとどまらず社会科学の分野にも及んでいる。このような複雑な現象の計算量は常に膨大であり、現実的な利用時間内に意味のある解を得るためには⁽¹⁾

- a) 超高速演算機能
- b) CPUの専有
- c) 大容量の主記憶

が要求される。

超高速演算機能という要求に対しては、スーパーコンピュータ（ベクトルコンピュータ）が開発されておりより一層の高速化が進められている。しかしかに高速のCPUであっても、ユーザ数の増加が、演算の高速化を上回

平成元年12月15日受理

* 情報システム工学研究所・電気工学科教授

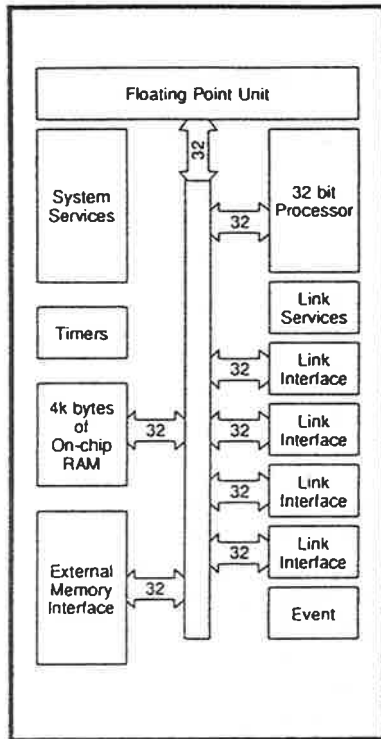


図1 トランスピュータの内部構造

るならば、ターンアラウンド時間は伸びる一方であり、物理的に意味のある結果を出すための必要時間が増加する。このため3つの条件を満足する計算機として普及型のスーパーコンピュータや、ミニスーパーコンピュータ、高速なワークステーションが開発され、ユーザに提供されるようになってきた。しかしユーザの要求は限りがなく、より高速かつ低価格なパーソナルな環境で数値計算を行いたいという要望が増えてきた。このような要求に応えるCPUとしてトランスピュータが注目されている。トランスピュータは

- a) 単一の計算エンジンとしても高速で、ワークステーションの演算速度に匹敵する。
- b) 並列化により演算を高速化できる。

という特徴をもっている。

トランスピュータのソフトウェアは従来OCCAMで記述されていたが1988年英国の3L社より並列FORTRAN, 並列Cが発表され、FORTRANの数値計算に馴染んでいたユーザにも容易に利用が可能になってきた。

本稿ではトランスピュータの概要、数値演算能力を述べ、実際の数値演算に最もよく現れる行列計算、微分方程式の求解に適用した結果について述べる。

2. トランスピュータの概要^{(2),(3)}

トランスピュータは英国のインモス社(INMOS)が開発したマイクロプロセッサであり、16ビットと32ビットがある。32ビット版には浮動小数点演算プロセッサ(FPU)を内蔵しないT414と浮動小数点演算プロセッサ(FPU)を内蔵したT800がある。20MHz版のT800は公称10MIPS, 1.5MFLOPSという高速演算処理能力が備わっている。

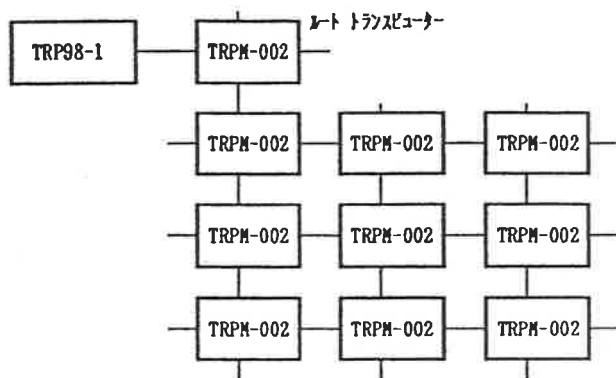
またトランスピュータは並列計算用のCPUとして設計されている。並列計算に必要なCPU間のデータ交換は10または20Mbpsという高速のシリアルリンクにより行われる。図1はトランスピュータの内部構造を示す。図示の通りこの通信リンクは4チャンネルある。

一方このような並列計算だけでなくトランスピュータは計算エンジンとし

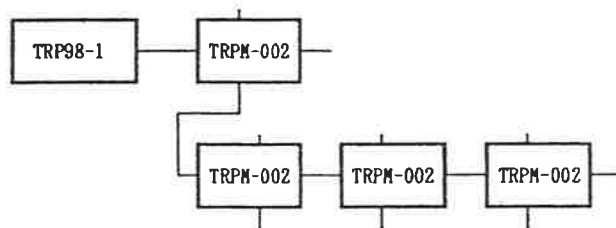
でも優れた性能をもっている。いくつかのベンチマークテスト、実際の応用プログラムの実行テストにより T 8 0 0 は V A X - 8 6 0 0 級の演算能力を発揮することが報告されている⁽¹⁾。このトランスピュータを搭載したボードは P C - 9 8 0 1 用、U N I X ワークステーション L U N A 用として多くの会社より市販されており、従来慣れ親しんだ O S (M S - D O S , U N I X) で使用することができる。

またトランスピュータは問題の構造に応じて、図 2 に示すようにアレイ接続、パイプライン接続、ツリー接続が可能である。

格子接続



パイプライン接続



ツリー接続

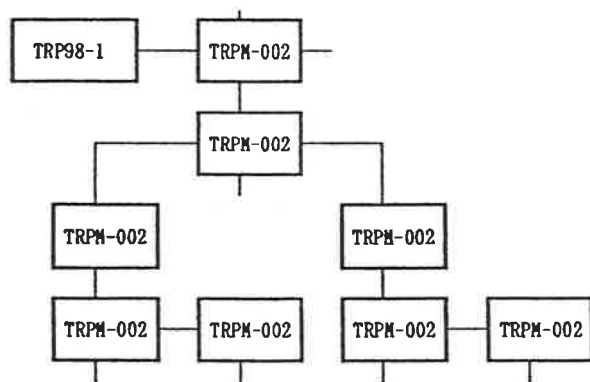


図 2 トランスピュータの接続方法

3. ソフトウェア

トランスピュータは従来、並列処理用専用言語であるOCCAMが用いられていた。OCCAMは並列処理専用の言語として専門家の間では主流の言語であるが、一般のユーザにとってはFORTRANあるいはCの方がはるかに使い易く、これらの言語のサポートにより今後トランスピュータがいろいろな分野に普及してゆくものと思われる。

並列FORTRAN（英国、3L社）の機能は

- a) 標準FORTRAN互換機能
- b) タスク割り付けによる並列化機能
- c) 非割り付け並列化機能（FLOOD FILL機能）

である。基本的にはFORTRAN 77にCPU間通信機能を追加したものであり、計算エンジンとして使用するかぎりは標準的なFORTRAN 77と同じである。

タスク割り付けによる並列化機能は、分割されたタスク間通信を書き加えた複数の並列走行可能なプログラムを複数のトランスピュータで実行させる機能である。この時各タスクを各CPUに割り付ける必要がある。

非割り付け並列化機能は分割されたタスクが完全独立でタスク間通信が不要の場合のみ有効な並列化機能であり、ネットワークの中にあるトランスピュータはすべて平等であり、手の空いたCPUが順次分割されたタスクを受けもち、処理終了後ルートコンピュータに結果を返す。

並列FORTRANではコンパイル、リンクはトランスピュータ上で実行されるようになっており、ホストCPUのOSは入出力の管理とコンパイラおよびソースファイルの転送を行うのみである。

4. トランスピュータの数値演算速度

本節ではトランスピュータ単体の演算速度を調べるためFORTRANによる実数の四則演算、三角関数、指数関数、対数関数の実行時間を求めてみた。この結果は表1、表2の通りである。表1は単精度（32ビット）、表2は倍精度（64ビット）である。なお表中SUN-3/50のデータは文献（1）のデータである。またパソコンはPC-9801XL²である。これらの表に示す結果で明らかのように、トランスピュータ（T800）は単精度、倍精度とも四則演算に関しては極めて高速であり、ワークステーションの約1.4～2.0倍である。単精度の実行結果から分かるように加算の実行時間は0.65μsであり、公称値（1.5MFLOPS）に近い値が出ている。また関数演算ではほぼワークステーションと同程度である。また倍精度の関数演算ではワークステーションよりは若干遅いが、四則演算に関してはやはり1.0～2.0倍速い。実際の応用プログラムでは通常四則演算と関数

表1 トランスピュータの実行時間
(単精度) (単位: μ SEC)

機種	PC-9801XL ²		Sun-3/50
CPU	T800-20	80386 +80387	68020-15 6889 -16.67
ソフトウェア	PARALLEL _FORTRAN	PROFESSIONAL _FORTRAN	FORTRAN77 (f77-f68881)
+	0.65	9	14.2
-	0.75	9	14.2
*	0.95	9	15.6
/	1.25	12	17.4
SQRT	7.61	32	14.2
SIN	24.67	95	31.4
COS	26.16	100	31.2
TAN	24.64	70	35.8
EXP	28.52	128	37.8
LOG	25.91	62	45.0
LOOP	0.85	2	

を掃き出し法で解いた結果を表3に示す。MS-DOS上の標準FORTRANではMS-DOSのメモリ管理が640KB以下であるので行列のサイズは200x200が最大であった。一方トランスピュータではこの制約にとらわれないので行列サイズを大きくでき、500x500でも計算可能である。なお演算は1個のトランスピュータを用い、単精度で行っている。表3の結果からわかるようにトランスピュータによる演算速度は17倍という驚異的な速度を示している。

演算が混在しているのでT800はワークステーションの数倍の処理速度と推定される。

5. 数値計算への応用

4節でトランスピュータ単体の四則演算速度を求めたが、本節では数値計算に関する応用問題の代表として最も基本的で、かつ頻繁に用いられる連立一次方程式、行列計算、微分方程式を取り上げ、種々のケースについて演算速度を比較する。微分方程式に関する応用問題では筆者の開発した連続系シミュレーション言語DECS^{(4),(5)}へ適用した。また計算速度の比較はT800による並列FORTRANと80386+80387によるPROFESSIONAL FORTRANで行った。

5.1 連立一次方程式

連立一次方程式

$$Ax = B \quad (1)$$

$$A (n \times n),$$

$$B (n \times 1),$$

$$x (n \times 1)$$

表2 トランスペュータの実行時間
(倍精度) (単位: μ SEC)

機種	PC-9801XL ²		Sun-3/50
CPU	T800-20	80386 +80387	68020-15 6889 -16.67
ソフトウェア	PARALLEL _FORTRAN	PROFESSIONAL _FORTRAN	FORTRAN77 (f77-f68881)
+	0.80	9	16.0
-	0.90	9	15.8
*	1.55	10	17.2
/	2.15	13	19.2
SQRT	14.45	32	15.0
SIN	46.31	94	32.6
COS	38.40	99	32.6
TAN	46.51	70	37.2
EXP	51.69	126	39.0
LOG	44.77	63	46.2
LOOP	0.85	3	

では密であるためと思われる。

(2) 並列処理で行った場合

つぎに行列のサイズを 100×100 としてトランスペュータを複数個並べて並列処理を行った場合の演算速度の向上の程度について述べる。

まず並列処理を行うためにつぎのようにタスクを分割する。

a) トランスペュータ2個の場合

行列A (i, j) を偶数行と奇数行に分割し、それぞれ行列B (i, j) と並列処理により乗算を行い、最後に全体をまとめる。

図3はタスクの割りつけを示す。ここで

M 全体の処理を行うタスク

5.2 行列の乗算

つぎに2つの行列A、Bの積を求め結果をCに入れる計算

$$A \cdot B = C \quad (2)$$

$$A (n \times n),$$

$$B (n \times n),$$

$$C (n \times n)$$

を行った場合の演算速度の比較を行う。この例題では1個のトランスペュータで演算を行った場合と複数個のトランスペュータで並列処理を行った場合の比較を行った。

(1) 単一処理で行った場合

1個のトランスペュータと、80386+80387を用いた場合の処理速度の比較を行った。その結果は表4の通りである。演算速度を比較してみるとトランスペュータは80386+80387の4~7倍で、行列サイズが大きくなるにつれて差は開く傾向にある。しかし5.1ほどの差は生じていない。計算速度にあまり差の生じない理由は、行列の要素が例題5.1では粗であるのに対し例題5.2

表3 連立一次方程式の実行時間
(単位: SEC)

機種	PC-9801XL ²	
CPU	T800-20	80386 +80387
ソフトウェア	PALALLEL _FORTRAN	PROFESSIONAL _FORTRAN
行列サイズ		
100*100	4.401	79
200*200	35.312	615
300*300	121.174	—
400*400	286.311	—
500*500	558.127	—

表4 行列の積の実行時間
(単位: SEC)

機種	PC-9801XL ²	
CPU	T800-20	80386 +80387
ソフトウェア	PALALLEL _FORTRAN	PROFESSIONAL _FORTRAN
行列サイズ		
50*50	0.584	2
100*100	4.824	17
150*150	16.457	109
200*200	39.353	—
300*300	139.465	—
400*400	330.394	—

W1 Aの奇数行とBの積を求めるタスク

W2 Aの偶数行とBの積を求めるタスク

を意味する。トランスピュータへの割り付けはM、W1をトランスピュータT1、W2をトランスピュータT2とした。

b) トランスピュータ4個の場合

行列A(i, j)を

A(i₁, j) (i₁=1, 5, 9, ...)

A(i₂, j) (i₂=2, 6, 10, ...)

A(i₃, j) (i₃=3, 7, 11, ...)

A(i₄, j) (i₄=4, 8, 12, ...)

のように分割し、それぞれ行列B(i, j)と並列処理により乗算を行い最後に全体をまとめる。図4はタスクの割り付けを示す。ここで

M 全体の処理を行うタスク

W1 A(i₁, j)とB(i, j)との乗算

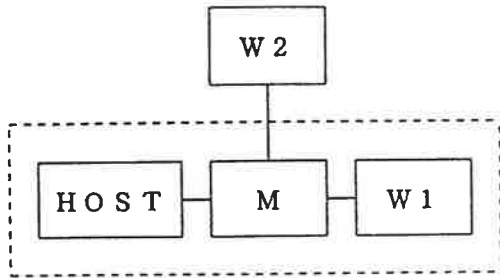
W2 A(i₂, j)とB(i, j)との乗算

W3 A(i₃, j)とB(i, j)との乗算

W4 A(i₄, j)とB(i, j)との乗算

を意味する。トランスピュータへの割り付けはM、W1をトランスピュータT1、W2をトランスピュータT2、W3をトランスピュータT3、W4をトランスピュータT4とした。表5はトランスピュータの個数と演算速度の

関係を示す。表より分かるように演算速度はおおよそトランスペュータの個数に比例していることが分かる。理想的には正確に比例するはずであるが、トランスペュータ相互間、タスク相互間の通信時間があるために個数と正確には比例しない。しかしシリアルリンク方式で通信を行っているためバス競合などの問題がないため、共通バス方式に比較して演算速度の低下の度は少ない。



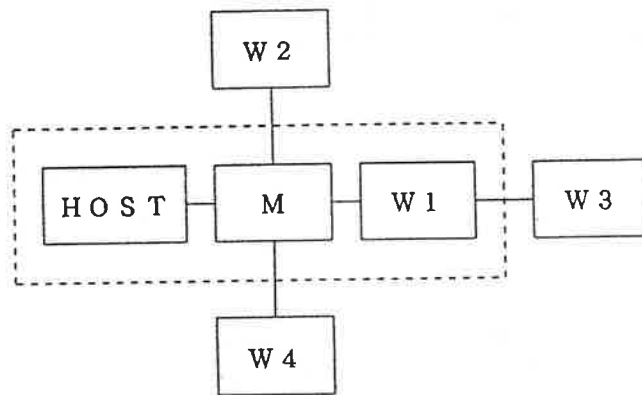
- M 全体のタスク
- W1 Aの奇数行とBの積を求めるタスク
- W2 Aの偶数行とBの積を求めるタスク

図3 トランスペュータのタスク割りつけ

5.3 連立常微分方程式への適用

つぎに連立常微分方程式の求解に適用してみた。本節の目的は筆者の開発した連続系シミュレーション言語DEC Sの高速化を計るための基礎研究のためである。例題としては図5のようなR-L-C直列回路を用いた。

本回路より



- M 全体の処理を行なうタスク
- W1 Aの1, 5, 9, ...行とBとの積を求めるタスク
- W2 Aの2, 6, 10, ...行とBとの積を求めるタスク
- W3 Aの3, 7, 11, ...行とBとの積を求めるタスク
- W4 Aの4, 8, 12, ...行とBとの積を求めるタスク

図4 トランスペュータのタスク割りつけ

$$R \frac{dq}{dt} + L \frac{d^2q}{dt^2} + \frac{q}{c} = E \quad (3)$$

が成立する。ここで

$$x_1 = q \quad (4)$$

$$x_2 = \frac{dq}{dt} \quad (5)$$

とおくことにより (3) 式は

$$\frac{dx_2}{dt} = (E - \frac{x_1}{c} - R x_2) / L \quad (6)$$

となり (5)、(6) 式より R-L-C 直列回路の状態方程式が得られる。この状態方程式を 4 次 Runge-Kutta 法を用いて解いた。つぎにトランスペュータ 1 個の場合と 3 個の場合の演算速度を比較した。3 個の場合は抵抗の値が異なる 3 つのケースを同時に並列処理するものである。この演算時間の比較を表 6 に示す。この結果はトランスペュータ 1 個の場合でも通常の FORTRAN の約 10 倍の演算速度が達成できることを示している。また 3 個の場合は 1 個の場合の 2.3 倍の演算速度である。これらの結果からトランスペュータを採用した場合には 1 個の場合でも大幅な演算速度の向上が期待できるものと思われる。

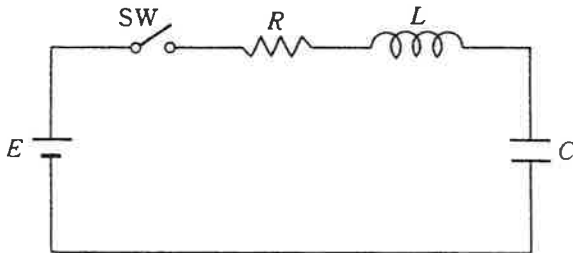


図5 RLC 直列回路

6. むすび

以上トランピュータ単体の演算速度、並列化による演算速度の向上、応用プログラムへの適用結果について述べた。得られた結論をまとめると

- (1) トランピュータ単体の四則演算の演算速度は 80386+80387 の FORTRAN の約 10 倍、関数演算に関しては約 2~3 倍である。
- (2) 行列計算に適用した結果、80386+80387 の FORTRAN と比較した場合、積の計算では約 5 倍、連立一次方程式の掃き出し法による解法では約 17 倍の演算速度が達成できた。ただし演算速度は行列要素のスパース性にも依存するのでこれらの数字は一応の目安である。
- (3) 行列計算に複数のトランスペュータを用いた場合、演算速度はおおよそトランスペュータの個数に比例する。トランスペュータではシリアルリンクによりデータ交換が行われるので、バスの競合などの問題

がないため、個数にはほぼ比例して演算速度が向上するものと思われる。

また行列計算の場合タスクの分割は極めて容易であり、したがってソフトウェアの作成も容易である。

- (4) 連立常微分方程式へ適用したところ、80386+80387のFORTRANと比較した場合、トランスピュータ1個で約10倍である。またパラメータを3回変化させて計算させ、各処理毎にトランスピュータを割り当てた場合、演算速度は約2.3倍であった。連立常微分方程式の並列処理に関するタスク分解は、行列演算と比較して可成り複雑であるので今後さらに研究する必要がある。

表5 行列の積を複数個のトランスピュータで計算した例
(単位:SEC)

機種	PC-9801XL ²
CPU	80386+80387
トランスピュータの使用個数	実行時間 (行列 100*100)
1	4.401
2	2.861
4	1.519

トランスピュータの高速演算性能、並列処理機能を生かす分野としては画像処理、FEM, BDM, シミュレーション、ロボット制御、ニューロコンピュータなどが考えられ、今後いろいろな分野へ応用されてゆくと思われるが、ソフトウェアを作成した感じでは、並列FORTRANのエラーメッセージがもっと親切であって欲しいことである。

7. 参考文献

- (1) 井門 並列C及び並列FORTRANを用いたトランスピュータによる高速数値計算、2nd Transputer/Occam International Conference, p101, April, Tokyo, 1989
- (2) トランスピュータボード TRPM-002取扱い説明書、コンカレントシステムズ
- (3) INMOS Engineering Data, IMS T800 Transputer
- (4) 松坂 パソコンから大型機まで実行可能な連続系シミュレーション言語、SENAC Vol.21, No.3, p76-88
- (5) T.Matuzaka and S.Ookawa, Differential equation based continuous system simulation language implemented on personal computers, BICSC Proceedings, p319-324, Bejin, China, October, 1989
- (6) 松坂 トランスピュータによる数値シミュレーション、計測自動制御学会東北支部第116回研究集会資料、資料番号116-5、1989年10月

表6 連立常微分方程式の演算速度の比較

(単位: SEC)

機種	PC-9801XL ²
CPU	80386+80387
ソフトウェア	実行時間
BASIC	33
BASICC	11
PROFESSIONAL -FORTRAN	1.35
PARALLEL -FORTRAN *1	0.148
PARALLEL -FORTRAN *2	0.063

*1 トランスピュータを1つ使用

*2 トランスピュータを3つ使用