

TR-192

株分け並列推論方式とその評価

増沢秀穂、久門耕一、板敷晃弘

佐藤 健、中村直人、相馬行雄

(富士通)

June, 1986

©1986, ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03) 456-3191 ~ 5
Telex ICOT J32964

Institute for New Generation Computer Technology

株分け並列推論方式とその評価

増沢 秀徳 久門 耕一 板敷 晃弘 佐藤 健 中村 直人 相馬 行雄
(富士通株式会社)

要約

日本の第5世代コンピュータシステムプロジェクト (FGCS) では、その目標として高速に推論を行う並列コンピュータを考えている。ここで述べる株分け並列推論方式は、その並列推論マシンに使われる並列推論方式の一つの候補である。

この論文では、実用指向の問題に対する株分け並列推論方式の評価を述べる。筆者等は、株分け方式と呼ぶ新しい並列推論方式を提案し、それに基づくOR並列推論実験システムを構築した。その実験システムで収集したデータにより、多くの並列性を持つ大きい問題に対して非常に有効であることを確認した。また、異なる問題での実験結果の考察及び並列処理効果の小さい問題の実行状況の解析等を通して並列処理効果を阻害している原因は主として株分け処理時間が大きいことであることを確認した。更に、並列処理効果の大きい問題に対する拡張した要素プロセッサ台数 (100 台) での予測性能向上比は約50倍であった。

この研究は、日本のFGCSプロジェクトの一環として通産省より委託されたものである。

1. はじめに

筆者等は、新しい並列推論方式として、株分け方式を提案した [久門85-1]。その方式に基づくOR並列推論実験システムを構築し [板敷85] その上で単純なゲームの問題、N-Queen を実行させて、株分け方式の有効性に対する感銘を掴んだ [相馬85]、[久門85-2]。その後、実用指向の問題として、文法的に制限した構文解析プログラムを対象に、株分け方式の評価を進めてきた。また、機能面としては、並列推論処理系へbagof 機能を組み込んだ [板敷85]。

本報告では、構文解析プログラム・トップダウン・パーザプログラム (DCG) * 及び・ボトムアップ・パーザプログラム (BUP) * [溝口85] を対象にして実験システムで収集したデータをもとに、DCG 及びBUP での並列処理効果を述べる。その後、異なる問題 (DCG, BUP, N-Queen) での実験結果の対比を通しての考察を述べ、更に、BUP の実験結果をもとに、並列処理効果が小さい問題に対してはその実行状況の解析を通して得た、効果

が小さい原因及び株分け方式の問題点等を述べ、また並列処理効果が大きい問題に対しては拡張した要素プロセッサ台数で実行した場合の予測性能向上比を述べる。第2章では試作した実験システムの概要を述べ、第3章ではDCG 及びBUP 等の実験結果を述べ、第4章ではDCG 及びBUP 等の実験結果に基づいた考察について述べる。

2. 実験システム

ここでは、株分け方式の原理、特徴、及び試作した実験システムについて簡単に述べる。

2.1 株分け方式

(1) 動作原理

図1-aは、Prologで記述された推論ベースの例である。矢印は、通常の逐次型推論での流れを示す。図1-bは、株分け方式の動作原理を示す。太い線で囲まれた部分は、PE 0での処理を示し、図1-aの矢印によって示された処理に対応する。もし、PE 1から仕事の要求があったなら

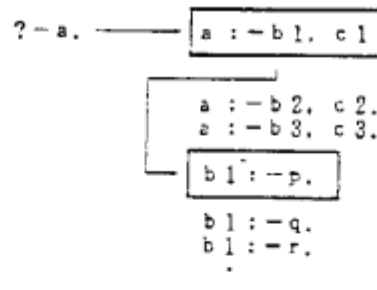


図1-a Prologで記述された推論ベース

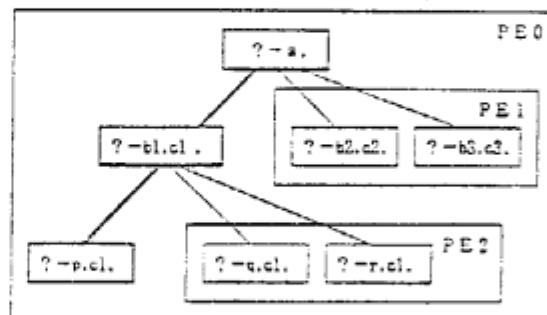


図1-b 動作原理

図1 株分け方式の原理

ば、仕事は探索木の根に最も近い枝根で分割され、残っている未処理の枝がPE 1に与えられる。更に、PE 2からの要求に対しても同様の処理が行われる。このように株分け処理（相馬85）を行い、複数のPEが協力して問題の解決にあたる。

(2) 特徴

株分け方式は、次のような特徴を持っている。

- ・ 各要素プロセッサ（PE）は、通常の逐次型処理における推論と同様に、完全なdepth-first 探索法に従って、探索木を処理する。
- ・ 他のPEから要求を受けたときだけ、仕事の分割及び分配処理を行う。

2. 2 実験システム

図2に実験システムのハードウェア構成を示す。実験システムは、マシン・インタフェース用の管理プロセッサ（MA）、15台のPEs、及び株分け方式用として試作した、用途が異なる二種類のネットワークからなる。

PE

PEは、推論処理及びその処理の間に他のPEからの要求に応じて仕事を分割し与える。各PE（現在は市販のパソコンを使用）上には株分け処理系を搭載しており、その性能は約0.5KLIPSである。株分け処理系とは、通常の逐次型推論処理系を基本とし、それに株分け方式のための制御機構（久門85-1）、（相馬85）を組み込んだものである。

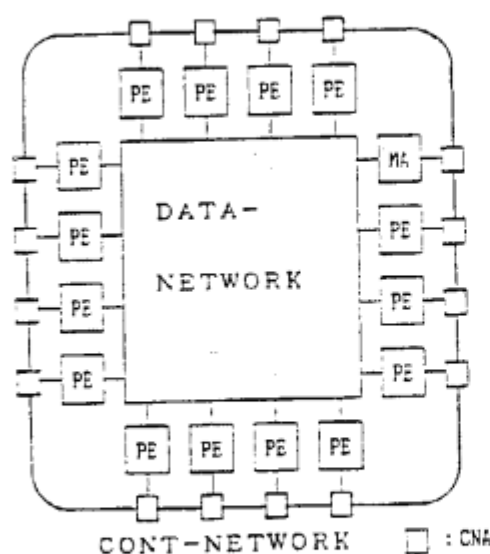


図2 実験システムの構成

CONTROL NETWORK

CONTROL NETWORK は、仕事を要求する為の通路である。このネットワークは、CONTROL NETWORK ADAPTER（CNA）を介して総てのPEをリング状に接続する。クロックの周波数は2MHzで、周回時間は最大 8μsである。そのネットワーク上を各PEの現在の状態（busy, idle）を示すパケット（12bitの並列転送）が常時回っている（久門85-2）。

DATA NETWORK

このDATA NETWORKは、仕事を転送する為の通路である。その構成は、4x4 スイッチを要素とした 2段の多段スイッチ・ネットワークである。転送方式は 8bit 並列DMA 転送であり、その性能は、ハードウェアとしては約 400 KB/s、ソフトウェア込みで約50KB/sである（久門85-2）。

3. 実験

ここでは、文法的に制限した構文解析プログラム、DCG 及びBUP、を対象にして、実験システムで収集した実験結果を述べる。はじめに評価プログラムについて述べ、それから並列処理効果及び通信回数等の実験結果について述べる。

3. 1 評価プログラム

評価プログラムは、3種類の異なる入力データを与えたDCG 及びBUP プログラムである。ここで、DCG プログラムとは、トップダウン・パーザ、具体的には構文解析木を根、つまり文を定義する規則のほうから解析の対象となる文の字づらへと分析していくプログラムであり、

表1 評価プログラムの問題の大きさ

入力の種類	大きさ（秒）	
	DCG 問題	BUP 問題
文章1	1 8	3 0
文章2	1 6 7	2 5 2
文章3	2 1 4 1	1 7 4 6

入力の種類	大きさ（秒）
6-Queen	5. 4
7-Queen	2 3
8-Queen	1 0 5
9-Queen	4 5 8
10-Queen	2 4 5 9

またBUP プログラムとは、ボトムアップ・パーザ、具体的には構文解析木を産、つまり解析の対象となる文の字づらのほうから組み上げていくプログラムである。

表1にDCG (ルール数:32, ファクト数:25), BUP (ルール数:43, ファクト数:103), 及びN-Queen についてそれらの問題の大きさを示す。ここでは、問題の大きさを単一のPEで実行したときの実行時間と定義する。

構文解析の結果は、解析木として得られるが、入力文章中に現れる「の」の係り方により、複数の解析の可能性があり、その結果複数の構文解析木が得られる。これらの複数の可能性が並列に処理される。

3. 2 実験結果

(1) 並列処理効果

図3に、評価プログラムにおいてPE台数を変化させた場合の並列処理効果を示す。なお、N-Queen 問題における実験結果も参考として示した。この図より、DCG 及びBUP 問題共に文章3のような大きな問題では、9-Queen, 及び10-Queen等と同様にほぼPE台数に比例した並列処理効果が得られていることがわかる。

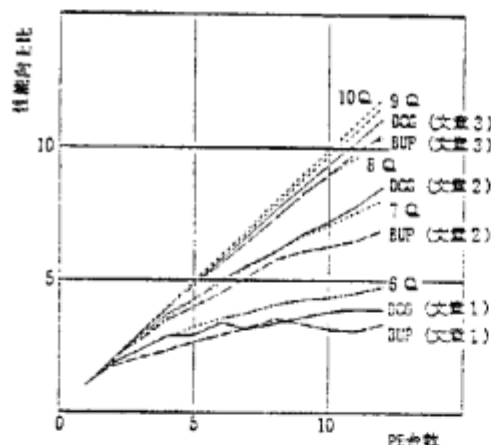


図3 並列処理効果

(2) 通信回数

図4に評価プログラムをPE台数=12で実行したときの問題の大きさと通信回数との関係を示す。ここでも、参考として、N-Queen 問題における実験結果を示した。この図の横軸は対数目盛で示しており、各問題においてシステム全体の通信回数は問題の大きさの対数に比例している。これは、問題の大きさが指数関数的に増加しても、通信回数は一次関数的にしか増加しないことを意味する。この傾向はN-Queen の実験結果でも見られるように、筆者等は、株分け方式はこのような並列処理に適した性質を有していると考えられる。

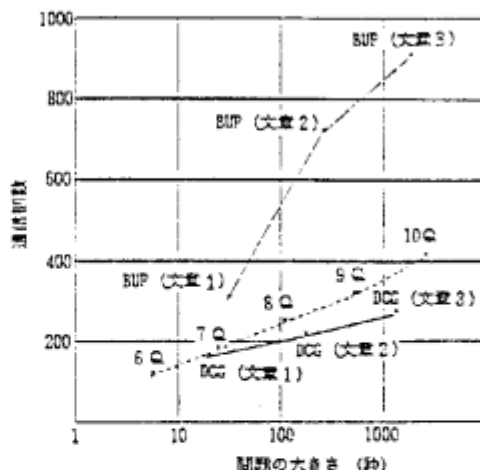


図4 問題の大きさと通信回数との関係 (PEs = 12)

4. 考察

ここでは、3. 2の実験結果をもとに、並列処理効果、及び異なる問題 (DCG, BUP, N-Queen) の実験結果の比較を通して考察を与え、更に BUP問題の実験結果を用いて並列処理効果が小さい問題に対してその実行状況の解析を通して得た、効果の小さい原因及び株分け方式の問題点等を述べる。その後、並列処理効果が大きい問題についてPE台数を100に拡張して実行した場合の予測性能向上比を述べる。

4. 1 並列処理効果

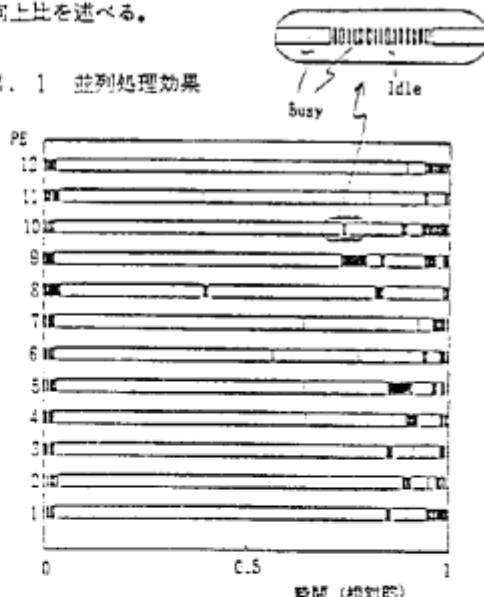


図5 BUP (文章3) 実行中の各PEの状態の変化 (PEs = 12)

図3より、評価プログラムの文章3、10-Queen、及び9-Queen では、ほぼ理想に近い並列処理効果が得られていた。これは、上記3. 2(1)で述べた並列処理に適した性質を持っている為、同種類の問題を実行した場合には大きい問題は相対的に通信回数、すなわち株分け処理に

よるオーバーヘッドが小さくなるからである。参考として、図5にPE台数=12で BUPの文章3の実行中の各PEの busy/idle状態の変化を示す。この図からわかるように、実行開始直後から実行終了直前の期間では殆ど全てのPE (12PEs) が通信も行わずに並列に動作している。

4. 2 異なる問題の対比

図6に DCG, BUP及び8-Queen 等をPE台数=12で実行したときの問題の大きさと並列処理効果との関係を示す。図6で、同程度の大きさの問題をみると、8-Queen, DCG, BUP の順で並列処理効果が大きい。以下では、8-Queen と DCG の文章2の対比を中心に述べる。

並列処理効果は、DCG の文章2では約8倍であるが、8-Queen では10倍以上である。しかし、通信回数については、図4から DCGの文章2の方が8-Queen よりも少ない。このように、通信回数が少ない DCGの文章2が8-Queen よりも並列処理効果が少なかった理由は以下の様である〔久門86-1〕, 〔久門86-2〕。

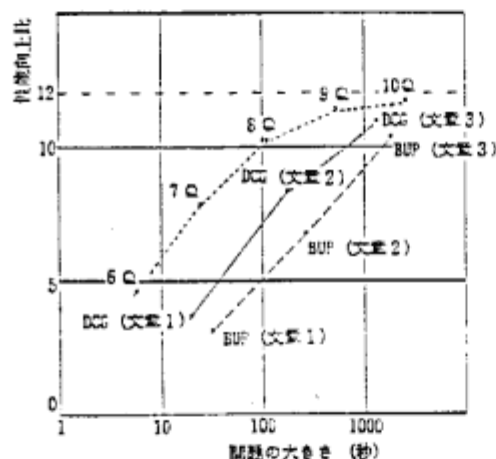


図6 問題の大きさと並列処理効果 (PEs = 12)

図7に DCGの文章2、及び8-Queen をPEs 12台で実行したときの実行開始時の各PEの状態の変化を示す。図7から、DCGの文章2は8-Queen よりも仕事と仕事との間隔が広いことがわかる。この仕事と仕事との間隔は、PEが、前の仕事を終ってidle状態になり、BUSY PEからの仕事の分配を待っている期間である。この期間は主として株分け処理に費やした時間と考えられる。従って、DCGの文章2の方が8-Queen よりも一回当りの株分け処理に要する時間を多く要すると考えられる。

図8に DCGの文章2、及び8-Queen 実行時の平均的な株分け処理に要した時間の詳細を示す。図8の DCGの文章2から、BUSY PEは推論中に IDLE PEからの要求を受けた後、分配出来る仕事を探し、そのような仕事があれば分配する仕事を作成し、その仕事を転送する。また、IDLE PEは転送された仕事を受け取り、推論が実行出来

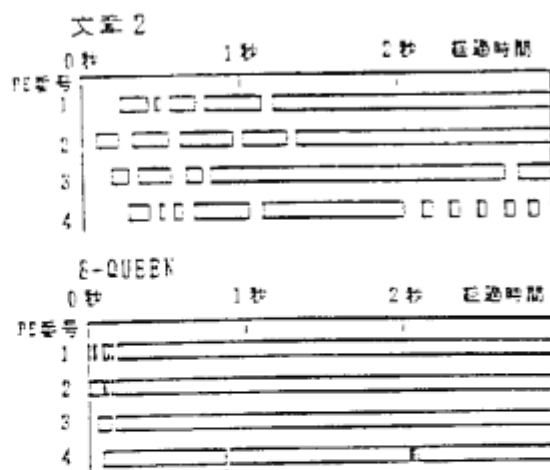
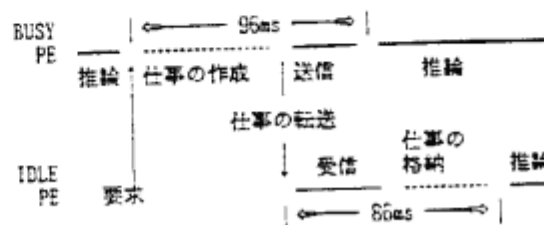


図7 実行開始時の各PEの状態の変化 (PEs = 12)

文章2:



8-Queen:

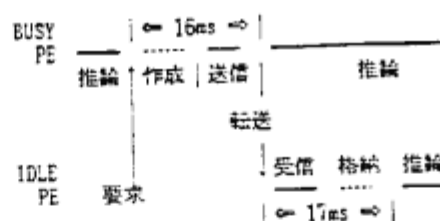


図8 株分け処理に要した時間の詳細

る内部形式に変換して格納し、その後推論処理に移る。ここで、BUSY PEは分配する仕事の作成、及び転送に約96ms、またIDLE PEでは転送された仕事の受け取り、及び格納に約86msの時間を要している。すなわち、BUSY PE及びIDLE PEで株分け処理に要した時間の合計は、約180msである。8-Queenでは、BUSY PEは分配する仕事の作成、及び転送に約16ms、またIDLE PEでは転送された仕事の受け取り、及び格納に約17msの時間を要している。すなわち、BUSY PE及びIDLE PEで株分け処理に要した時間は約30msであり、これは DCGの文章2のそれの1/6にすぎない。以上から、DCGの文章2は8-Queenに比べて通信回数は少ないけれども、一回当りの株分け処理時間が約6倍も要している為、仕事分配によるオーバーヘッドが大きく、並列処理効果において8-Queenよりも小さい。

8-Queenの文章2でも同様の傾向、すなわち株分け処理に要した時間が8-Queenの約4倍であり、また図4から通信回数がかなり多くなっている(8-Queen及びDCGに対してそれぞれ約2.9倍及び約3.4倍)為、並列処理効果が8-Queen及びDCGの文章2よりも低くなっている。

以上より、一つの文章に対する解析を並列にしかも独立に検査する場合、転送量は大きくなるが、それに伴い仕事の作成及び内部形式への変換を含む格納処理に要する時間が多くなる為、その時間を少なくする必要がある。

4. 3 BUPの文章1における並列処理効果

表2にBUPプログラムについての株分け処理系による実行を反映した論理的な性質を示す。表2から文章1では、平均並列度が45程度あるにもかかわらず、PE台数=12での性能向上比が4倍にも達していない(図3を参照の事)。ここでは、その原因を分析する。

表2 BUPプログラムの論理的性質

項目 プログラム	平均並列度 *1	推論のレベル数 *2 (段数)
文章1	45	1294
文章2	236	2081
文章3	1427	2388

(参考)

- *1 : 平均並列度 = 総推論数 / 推論のレベル数
- *2 : 探索木における根からの最長推論レベル段数

(1) 実験結果

実験結果より、PE台数=12で実行した時の全処理(busy/kabuwake/idle)の内訳は以下であった。

- ・ busy 時間 : 30.6% ・ kabuwake 時間 : 43.0%
- ・ idle 時間 : 26.4%

kabuwake時間とは全株分け処理時間の合計である。一面の株分け処理時間とは、図8で述べたように、具体的にはBUSY PEが転送する仕事の作成時間、仕事の送信時間、及びIDLE PEによる仕事の受信時間、仕事の格納時間からなる。

処理の内訳から、kabuwake及びidle時間等が全体の70%程度占めており、それらが性能向上を阻害している原因である。

実行状況を確認するとidle/kabuwake/busyの発生状況は次のようである(図9に文章1実行中の各PEの状態の変化を示す)。

- ・ idleは、実行時間の最初の1/3、及び最後の1/3期間で多く発生し、これらの期間で全idle期間の90%以上を占めている。

- ・ 通信は全体を通して一様に発生している(実行時間の最初の1/3の期間、真中の1/3の期間、最後の1/3の期間でそれぞれ112回、108回、125回の通信が発生している)。
- ・ 実行時間の真中の1/3期間では殆どのPEsが大きい塊の推論処理を行っている。

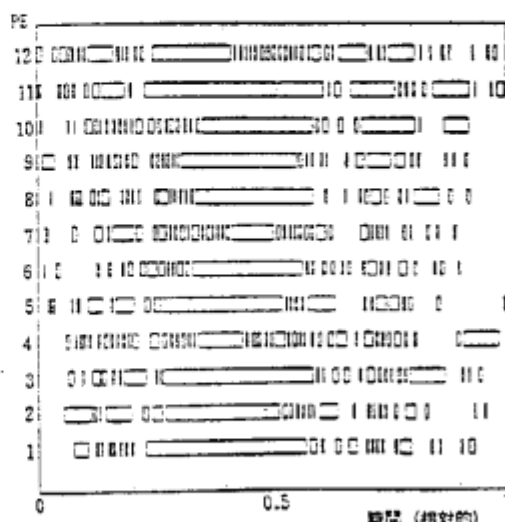


図9 BUP(文章1)実行中の各PEの状態の変化(PEs=12)

(2) 解析

idle/kabuwake時間が大きい原因を知る為、株分け処理系での実行を論理的に反映した文章1の探索木、及び図9に示した実行時のbusy/idle発生状況を用いて解析する。

① 探索木に出現する並列性の性質

図10に株分け処理系での実行を論理的に反映した文章1

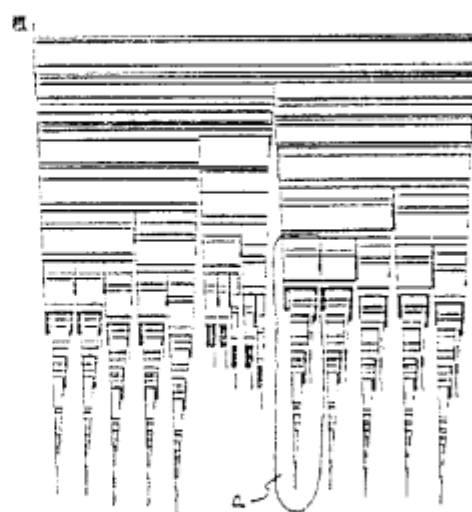
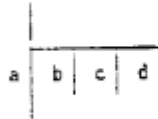


図10 BUP(文章1)の探索木

の探索木を示す。

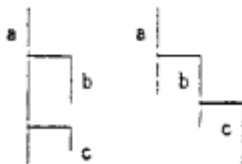
- ・ 探索木上の中間レベル以降で12個程度のある程度大きい塊の部分探索木 (図10のa) が存在する。
- ・ 存在する殆どの並列性は図11に示すものである。

図11-aに示す並列性 (箱型並列性という) は、ゴールが複数の同じ述語名のルールとのユニフィケーションにおいて複数ルールと成功するときが発生するものであり、図11-bに示す並列性は二つの枝に分かれた後、その内の一方は枝分かれの部分を持たず、また推論回数も少ないものである。



枝b,c,及びd は推論数が少ない。

図11-a 箱型並列性



(イ) (ロ)

- (イ) 枝b,及びc は推論数が少ない。
- (ロ) 枝b が枝分かれした後の枝a の推論数は少ない。枝c が枝分かれした後の枝b についても同様。

図11-b 箱型ではない並列性

図11 BUPで並列に処理出来る部分の性質

② 解析

探索木及び実行中の各PEの状態の変化等の解析の結果、kabuwake及びidle期間の大きい原因は以下のである。
kabuwake時間が大きい原因

- i. 一回当りの株分け処理時間が大きい。

図12に文章1の実行時における一回当りの株分け処理



図12 株分け処理時間の詳細

の内訳を示す。図12に示すように株分け処理において BUSY PE では45ms、IDLE PE では57ms要しており、推論処理時間 (約 2ms/推論処理) に比べて大きい。

- ii. 株分けされる仕事の大きさが小さい為、株分け処理回数が多い。

株分け処理時におけるBUSY PE 及びIDLE PE で株分け処理に費やす時間は、並列処理時のオーバーヘッドと考えられる。BUSY PE は、その費やす時間の間に約24個程度の推論処理が可能であり、24推論回数以下の仕事が全仕事数の約36% (推論回数が10以下の仕事が約35%) を占める。なお、株分け方式は一般的には通信回数は少ない (文章1, 文章2, 及び文章3等の実行時の通信回数 (株分け処理回数) はそれぞれ305 回, 714回, 及び900 回である)。

idle時間が大きい原因

- i. 株分け処理における仕事作成処理時間が大きい。

株分け処理時においてBUSY PE が転送する仕事を作成している間、IDLE PE はidle状態 (図12のIDLE PE における仕事待ち時間にあたる) である。この時間が推論処理時間に比べて大きい。全idle時間の40%近くはこれに起因する。

- ii. 問題に内在する並列性の性質に関する原因:

- ・ 方式的にその抽出に多くの時間を要する並列性

図11-aの並列性を抽出するとき、株分け方式では多くの時間を要する。これは株分け方式の原理に関するものであり、図13のように各PEが株分け処理を通して順次抽出する為である。もし、PE 1が枝b,c,d をそれぞれPE 2,

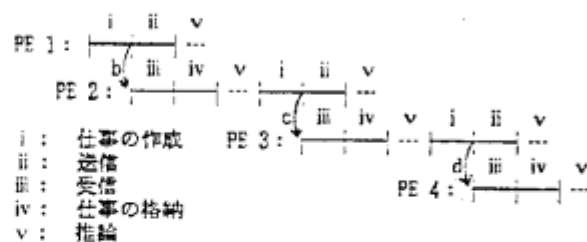


図13 箱型並列性を持つ部分に対する仕事の分配処理



図14 箱型並列性に対する効率的な仕事の分配処理

PE 3, PE 4 に同時期に分配出来れば、PE 2及びPE 3での仕事作成に要する時間は不要になる(図14)。これは、PE 2, PE 3, PE 4に転送される仕事においては、ルール番号(ルール番号とは枝の順序付けの為に使う指標である(相馬85)) ; 枝b, c, 及びd に対してそれぞれ 1, 2, 及び3) だけが異なる為であり、ルール番号を変える時間はほとんど無視出来るからである。更に、図13の場合よりも図14の場合の方が、PE 3, PE 4 は仕事を早く受け取ることが可能な為、PE 3, PE 4のidleな時間は小さくなる。

busy PE が早急に増えていかない並列性。

図11-bの並列性は、①の探索木に出現する並列性の性質で述べたことから抽出してもなかなか12台ものPEが並列に実行出来るような性質のものではない。

問題に内在するほとんどの並列性が上記の性質を有しており、その並列性を抽出する時期と一般的に複数PEによる並列実行効率が悪い実行序盤及び終盤等の時期とが重なり、更に株分け処理時間が大きい為に、idleなPEがすぐに発生し、なかなか全てのPEs がbusyにならずにidle期間が発生する。

以上のことからkabuwake及びidle時間等が大きくなり、この為に実行時間が長くなり並列処理効果が低くなっている。

(3) 問題点

以上のことから株分け方式の問題点を述べる。

- ① 一回当たりの株分け処理時間が大きい；
- ② 仕事の大きさの判別が不可能；
仕事の大小に関係なく株分けする。
- ③ 樹型並列性を抽出するのに多くの時間を要する；

4. 4 BUPの文章3に対する予測性能向上比

ここでは、並列処理効果が高い文章3をPE台数 = 100で実行したときの予測性能向上比について述べる。

(1) 株分け処理担当りの処理率

はじめに予測性能向上比を求める為に使用する株分け処理担当りの処理率という尺度を下式のように定義する。

株分け処理担当りの処理率 =

$$\frac{(\text{PE台数} = 1 \text{ での処理時間})}{(\text{一回当たりの平均株分け処理時間})} \times (\text{PE台数} = N \text{ での通信回数})$$

株分け処理担当りの処理率は、株分け処理時間を並列処理によるオーバーヘッドとみなして、その単位オーバーヘッド当りにどのくらい有効な処理が出来たかを表わ

す尺度であり、この値が大きい程性能向上比が高いことを意味する。

この株分け処理担当りの処理率を用いて、種々のプログラムの並列処理効果を比較した結果、実測データに基づいて比較した結果とほぼ一致した(図15)。これは、種々のプログラムの並列処理効果を同じ枠内で大雑把に比較出来る一つの尺度と考えられる。

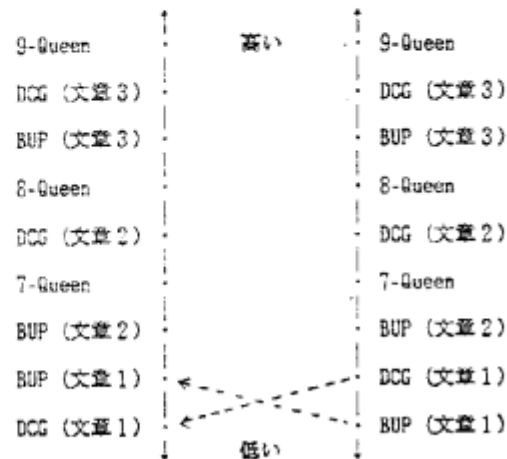


図 15-a 株分け処理担当りの処理率 図 15-b 実測データに従った順序付け
図15 並列処理効果のランク付け (PEs = 12)

(2) 予測性能向上比

上記の株分け処理担当りの処理率を使用して、予測性能向上比を得る。その方法の基本的な考え方は、同程度の株分け処理担当りの処理率を有するプログラムは同程度の理想性能向上比に対する割合を示すということである。尚、ここでPE台数 = Kで実行した時の性能向上比及びその理想性能向上比に対する割合は下記の式で与える。

$$\text{性能向上比} = \frac{(\text{PE台数} = 1 \text{ での処理時間})}{(\text{PE台数} = N \text{ での処理時間})}$$

$$\text{理想性能向上比に対する割合} = \frac{(\text{PE台数} = N \text{ での性能向上比})}{(N)}$$

予測性能向上比を得る方法の概略は、次の通りである。

PE台数 = Kで実行したときの予測性能向上比を得たいプログラムの株分け処理担当りの処理率と等しい値の株分け処理担当りの処理率を有するプログラムを見出し、そのプログラムの理想性能向上比に対する割合を実測データより求める。その割合を予測性能向上比を得たいプログラムの理想性能向上比に対する割合として使い、

予測性能向上比を求める。

以上の方法により求めた、文章3をPE台数=100で実行した時の予測性能向上比は約50倍である。

ここで、文章3をPE台数=100で実行した時の株分け処理相当りの処理率を得る為には、実行時の通信回数を求める必要があるが、通信回数は、文章3における既知データを使って得たPE台数と通信回数との関係(株分け方式においては比例関係となるという特徴がある[相馬85])から求めた。また、理想性能向上比に対する割合が既知のプログラムとしては文章1 (PE台数=5及び6で実行)を用い、次のことを仮定した。

仮定 文章1をPE台数=5(6)で実行した時の単位idle時間相当りの処理率と文章3をPE台数=100で実行した時の単位idle時間相当りの処理率とは等しい。(脚注)

5. まとめ

文法的に制限した構文解析プログラム“DCG”及び“BUP”を実験システムで実行した結果を用いて、株分け方式の評価を行なった。その結果、次の結論を得た。

- (1) DCG及びBUP問題でもN-Queen問題と同様に、大きい問題に対してはほぼPE台数に比例した並列処理効果があることを確認した。これは、DCG及びBUP問題でもN-Queen問題と同様の並列処理に対して好ましい性質を示した為である。すなわち、処理量が指数関数的に増えても、通信回数は一次関数的にしか増えない。これは、株分け方式の性質と考える。
- (2) 異なる問題(DCG, BUP及びN-queen)の実験結果の比較を通して、株分け処理に要する時間を小さく抑える必要があることがわかった。
- (3) 並列処理効果が小さい問題では、総株分け処理時間及びidle時間が実行時間の70%程度を占めていた。上記の原因を払る為に行なった実行状況の解析を通して、次のような株分け方式での問題を抽出した。
 - i. 一回当りの株分け処理時間が大きい。
 - ii. 仕事の大ききの判別が不可能である。
 - iii. 樹型の並列性の抽出に多くの時間を要する。
- (4) 並列処理効果が大きいプログラムのPE台数=100での予測性能向上比は約50倍である。

上記(2)及び(3)のiに対しては、現在作成中の株分け方式に対するコンパイラにおいて株分け処理時間を小さくする方式を試みており、そのコンパイラの実現、及び評価を待っているところである。また、その他の問題は株分け方式の基本原理解説するものであり、現在その対策を検討中である。

なお、筆者等は、株分け処理時間を小さくすることが

最優先課題と考えており、コンパイラ版によりその課題を克服することが可能であるという見通しがあり、株分け方式は有効と考える。

謝辞

最後に常日頃御指導を頂いておりますICOTの内田室長を始めとするPIGグループの方々、並びに当社の佐藤部門長、柳橋部長、林部長代理、及び服部主任研究員、またこのような興味深い研究に従事する機会を与えて下さいました山田所長に深謝致します。

参考文献

- [板敷85] 板敷, 他, “並列推論処理システム—改良型節単位処理方式の実験—”, 情報全大(30回), 7C-7, 1985年3月。
- [久門85-1] 久門, 他, “並列推論処理システム—改良型節単位処理方式—”, 情報全大(30回), 7C-8, 1985年3月。
- [久門85-2] 久門, 他, “株分け処理方式の評価”, 情報全大(31回), 2C-5, 1985年10月。
- [相馬85] 相馬, 他, “A NEW PARALLEL INFERENCE MECHANISM BASED ON SEQUENTIAL PROCESSING”, IFIP TC-10 Working Conference on Fifth Generation Computation Architecture, 1985年7月, 及び J.V.Woods, “A NEW PARALLEL INFERENCE MECHANISM BASED ON SEQUENTIAL PROCESSING”, fifth generation computer architecture, north-holland, 1986。
- [溝口85] 溝口, 他, “Prologとその応用”, 総研出版, 1985年。
- [久門86-1] 久門, 他, “「株分け」推論マシンの評価”, 情報全大(32回), 2R-5, 1986年3月。
- [久門86-2] 久門, 他, “KABU-WAKE: A NEW PARALLEL INFERENCE METHOD AND ITS EVALUATION”, COMPCON 86 Spring, 1986年3月。
- [板敷86] 板敷, 他, “「株分け」推論マシンにおけるbagof機能”, 情報全大(32回), 2R-6, 1986年3月。

(脚注) 仮定の数値的裏付けは次のようである。

・ 実験した種々のデータに対して、同程度の株分け処理相当りの処理率を有するプログラムは、同程度の単位idle時間相当りの処理率を示した。

・ 同一種類のプログラムは、その実行において同様のidle時間の発生傾向を示す。

尚、単位idle時間相当りの処理率は以下のように定義した。

$$\text{単位idle時間相当りの処理率} = \frac{(\text{PE台数} = 1 \text{ での処理時間})}{(\text{PE台数} = N \text{ での実行時の総idle時間})}$$