

PS 並列化の速度向上評価と 負荷均等化粗粒度方式の提案

湯川 高志 松澤 和光 石川 勉

NTT 情報通信処理研究所

プロダクションシステムの照合フェーズ並列化による速度向上度を、解析により評価する方法を述べ、基本的な並列化方式の解析評価による比較を行う。速度向上評価法は、照合に用いられる Rete ネットをモデル化し、そのモデルに基づいた速度向上推定の一般式を構築することにより、様々な並列化方式の、広い問題領域にわたる評価を可能とする。また、解析結果より、通信オーバーヘッドの少ない粗粒度方式では、小ルール時に負荷ばらつきによる速度低下が問題となる点を指摘し、これを改善する負荷均等化方式を提案する。本方式は、従来の構造並列にデータ並列を融合することにより負荷を均等化を図っている。さらに、本方式を解析、実験により評価し、小ルール数での速度向上改善を確認した。

Performance Analysis of a Parallel Production System and its Load Balancing Scheme

Takashi Yukawa Kazumitsu Matsuzawa Tsutomu Ishikawa
NTT Communications and Information Processing Laboratories
3-9-11 Midori-cho Musashino-shi Tokyo, 180 Japan

An analytical evaluation scheme for speed-up of a parallelized production system is discussed, and the performance of basic parallel algorithms are evaluated. A model of the Rete-network, which is used for match phase of production systems, is introduced to calculate the speed-up for a wide spectrum of expert-system properties, which are applied on production systems. In addition, a load balancing scheme is proposed for course-grained parallelization. This scheme provides high efficiency even though the size of the applications are small. The scheme is evaluated experimentally to prove its performance.

1. はじめに

エキスパートシステム (ES) の開発が本格化するにつれて、その推論機構であるプロダクションシステム (PS) の高速化が要求されるようになってきた。そのため、PS の推論を並列処理により高速化する研究が盛んになってきている [1] [2] [3]。

この並列化研究においては、PS 推論動作である認知サイクルのうち、その時間の 90% 以上を消費する照合フェーズが対象とされている。

従来、並列化方式の評価には実在の ES を使ったシミュレーションが用いられていた。しかし、速度向上は、一般に対象とする ES の問題特性に大きく依存するため、これでは用意した ES の問題特性での評価しか得られなかった。

本稿では、Rete アルゴリズムベースの PS を対象とした、速度向上度の解析的評価法について述べる。解析的評価法は、想定した ES 特性から速度向上を計算によって求めることにより、様々な方式の広い範囲の問題特性にわたった概略的な評価を可能とする。

本評価法では、速度向上を解析的に計算するために、次の様なアプローチをとっている。

- 1) 速度向上に影響の大きい ES の代表的パラメータの抽出。
- 2) 照合に用いられる Rete ネットの構造、動作の概略を、少数のパラメータで表現できるモデルの構築。
- 3) 上記 Rete ネットモデルに基づいた、問題特性と照合特性との関係の定式化。
- 4) 照合特性による、分割損、負荷ばらつきの表現と、粒度、通信コストをパラメータとした速度向上推定の一般式の導出。
- 5) 粒度の異なるいくつかの基本方式に対する上記パラメータの算出と、ES 特性を想定した速度向上の評価。

更に、評価の結果を考察し、粗粒度方式では小ルール時の負荷ばらつきが問題となることを指摘する。そして、粗粒度方式にデータ並列性を融合することにより、負荷ばらつきの改善を

図った方式を提案し、解析および実験により、その効果を確認する。

2. Rete ネットのモデル化

本章では、PS 照合フェーズに最も一般的に用いられている Rete アルゴリズムについて説明し、アルゴリズムの核である Rete ネットをモデル化して、その構造、動作と問題特性との関係を定式化する。

2.1 Rete アルゴリズム

PS は、if-then 型のルール知識に基づき、事実知識を格納したワーキングメモリ (WM) を書き換えることにより推論を行うシステムである。この推論は、照合・競合解消・ルール実行の各フェーズからなる認知サイクルの繰り返しにより進められるが、このなかで、照合フェーズの処理量が格段に大きい。そのため、照合を効率的に処理するアルゴリズムがいくつか考えられており、代表的なものが Rete アルゴリズムである。

Rete アルゴリズムでは、ルールの条件部を Rete ネットとよばれるネットワークに展開する (図 1)。この Rete ネットに、照合の途中結果を蓄積し、認知サイクル毎には WM の変化分のみをトークンとして Rete ネットに流すことにより照合を効率化している。

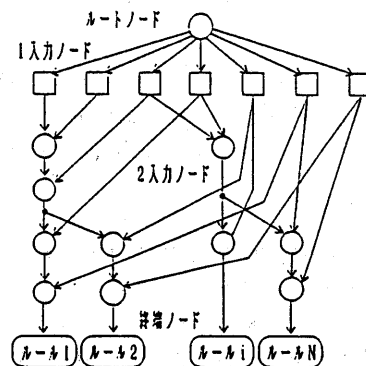


図 1 Rete ネット

Rete ネットは、トークンが最初に入力されるルートノード、ルール条件部中定数との比較を行なう部分を処理する 1 入力ノード、条件

要素間の変数同士の突合せを行なう2入力ノード、照合に成功したトークンが到達する終端ノードから構成されている。また、照合の中間結果は、各ノードに付随するメモリノードに蓄えられる。

2.2 エキスパートシステムの問題特性

Rete ネットの構造、動作の推定に必要な問題特性として表1～2に示すものがある。これらは、ESソースプログラムの分析、あるいは、ES設計段階での見積りによって得られる代表的なパラメータである。

ここで、 S_0 は、ES中の全条件要素に対する同一定数テストをもつ条件要素の割合である。Rete ネットでは同一の定数テストを行う1入力ノードを共有するため、1入力ノード数の推定のために、これが必要となる。

アクション部の更新動作比率 β_m は、ESの実行要素(WM生成・更新・削除の動作記述)に占める更新動作の割合である。更新は、削除と生成のふたつの動作を内部的に行うため、ルール実行で生成されるトークン数に影響する。

また、WMの要素(WME)数は、生成、削除動作により、推論中に変化するが、ここではその平均値を用いることにする。

表1 ルールのパラメータ

パラメータ	記号	パラメータ	記号
ルール数	R	ルール当りのアクション数	A
ルール当りの条件要素数	E	アクション部更新動作比率	β_m
同一定数テストの比率	S_0		

表2 WMのパラメータ

パラメータ	記号	パラメータ	記号
WME数	W	クラス数	L

2.3 Rete ネットのモデル

Rete ネットのモデル化に際しては、2入力ノードのふるまいを主に主眼を置く。これは、2入力ノードの処理量が他のノードに比べて非

常に大きく、また、全ての種類のノードは必ず2入力ノードに接続されているためである。したがって、2入力ノードのモデル化により、他のノードの特性も付随的に求めることができる。

2.3.1 2入力ノードのモデル化

代表的ないくつかのESについて、Rete ネットのふるまいが調べられているが^[4]、この分析から以下の事実が導き出される。

《事実》

- ・ルール間での同一条件照合の共通処理のために2入力ノードは出力枝に分岐を持つが、その分岐数は比較的少ない。
 - ・1ルールあたりの条件要素数の度数分布は、条件要素の増大に従って指数関数的に減少する。
- これをベースに、2入力ノードを以下のようにモデル化する(図2)。

[2入力ノードモデル]

- ・出力枝の一定の分岐率Bを持つ
- ・一つのノードは確率Pで終端する

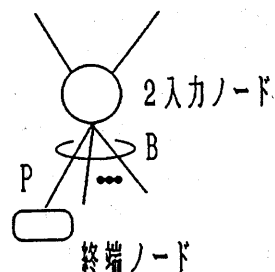


図2 2入力ノードのモデル

これにより、2入力ノードのふるまいを分岐数Bと終端確率Pのふたつのパラメータで規定できる。このモデルでは、どの出力枝も一定確率Pで終端ノードに接続されているので、第1段目の2入力ノードから終端ノードまでのノード段数は、指数関数的に減少する度数分布を示す。2入力ノードの段数=ルールの条件要素数+1であるため、上記の事実によくあう。

このモデルを規定するパラメータと問題特性の関係は以下の様になる。

《EとPの関係》

ノードの接続段数は指数関数的な分布となり、その平均値 e は、

$$e = \frac{P}{1 - (B - P)}$$

E = e + 1 であるから、

$$P = (B - 1) + \frac{1}{E - 1}$$

このことから、分岐数 B が決まれば、このモデルは平均条件要素数 E から一義的に決定できる。B は異なるルール間で共通な条件要素の比率に依存するため、正確に求めるにはソースプログラムを分析する必要がある。しかし一般には、共通部分が少ないもので 1.1~1.2、共通部分が多いものでも 1.7~1.9 程度である。

2.3.2 Rete ネットの諸特性

以上のモデルと問題特性より、Rete ネット構造的な特性および処理量特性は、表 3~4 の様に表される（導出は付録参照）。

処理量に関しては、認知サイクル毎、あるいはノード毎に異なるが、速度向上を計算するには、その平均値がわかれば十分なため、導出に際し以下の仮定を設けた。

[仮定]

- ・各クラスに対する条件要素の数は均一
- また、トークンもこれに対し均等に発生
- ・各ノードの照合成功率は一定
- (定数テスト: α_c , 変数照合: α_j とする)

ここで、 α_c および α_j は照合するルールの絞り込み方に依存するが、一般には、定数テストを粗くして、変数照合で最終的に絞り込むので、 α_c は 10~数 10%、 α_j は 1% 以下である。なお α_j は認知サイクル毎の照合成功ルール数（設計時に見当がつく）から逆算する事もできる。

表 3 Rete ネットの構造的な特性

1入力ノードの総数	D_1	$R(E-1)(1-S_c)$
2入力ノードの総数	D_2	R/P
1入力ノードから2入力ノードへの接続枝数	S	$D_2 \{2-(B-P)\}$

表 4 Rete ネットの処理量特性

活性化される1入力ノード数	O_1	
1入力ノードを通過するトークン数	T_t	$S\alpha_c/L$
活性化される2入力ノード数	O_2	$T_t \cdot (E-1)$
2入力ノードでの平均照合回数	j	$(W/L)\alpha_c$
1認知サイクルで発生するトークン数	M	$A(1+\beta_m)$

3. 基本並列化方式と速度向上度

本章では、上記の照合特性に基づき、ES 特性と並列化粒度から速度向上を計算する推定式を導出する。また、粒度が異なるいくつかの基本的な方式に推定式を適用し、評価を行なう。

3.1 速度向上推定の一般式

1 認知サイクルを逐次処理した場合の処理時間を T_0 とすると、並列化時の処理時間 T は、

$$T = \frac{QT_0}{N} \gamma + T_c$$

ここで、

- N : プロセッシングエレメント (PE) 数、
- Q : 処理の分割による総処理量の増加係数
- γ : 負荷ばらつきによる処理時間の増加係数
- T_c : PE あたりの通信時間である。

ところで、通信時間 T_c は、PE あたりの処理量に比例するとして、

$$T_c = \frac{QT_0}{N} C_0$$

と書くことができる。 C_0 は方式に固有の通信オーバーヘッド係数である。

以上より、速度向上 U は以下の様になる。

$$U = \frac{N}{Q} \cdot \frac{1}{\gamma + C_0}$$

処理量増加係数 Q は、方式が決まれば Rete ネットモデルから容易に求めることができる。

また、負荷ばらつき係数 γ および通信量係数 C_0 については、以下でより具体的に検討する。

《負荷ばらつき係数 γ 》

通信オーバーヘッドを除いた処理時間を T_p とすると、これは最も時間のかかった PE により支

配されるので、

$$T_P = \max(T_1, T_2, \dots, T_N)$$

(T_i : PE_iの処理時間)

となる。実際には、PSは多くの認知サイクルの繰り返しによって推論を行うので、速度向上計算にはこの平均値を用いる。

負荷ばらつき係数は、負荷均等と想定した時の処理時間と T_P との比と考えられるから、

$$\gamma = \frac{\max(T_1, T_2, \dots, T_N)}{\sum T_i / N}$$

となる。

負荷ばらつきは、並列処理の粒の処理量分布および粒同士の逐次性が原因となっておこるため、照会特性からこれらを抽出することにより γ を定式化できる。そのために、

粒の数: n

並列処理の各粒: g_1, g_2, \dots, g_n

粒 g_i の処理時間: t_i

と定義する。また、逐次的に処理されるべき粒がある場合は番号を適切につけかえて、それらの集合を作り、それぞれを

$$G_1 = \{g_1, g_2, \dots, g_{x1}\}$$

$$G_2 = \{g_{x1+1}, g_{x1+2}, \dots, g_{x2}\}$$

⋮

$$G_m = \{g_{x(m-1)+1}, \dots, g_{xm}\}$$

と書ことにする。即ち、全く独立に処理できる粒は、 m 個である。また、

G_i の処理時間: T_{Gi}

とする。すなわち、 T_{Gi} は集合 G_i に含まれる粒の処理時間を総和したものになる。

粒の逐次性により、 G_i に属する粒がどの様に割り付けられたとしても、最後の要素 g_{xi} の処理が終了するまでには、少なくとも T_{Gi} の時間が必要になる。

集合の最後の要素を各PEに同一個数割り付けると考えれば、1PEあたり m/N 個となる。

粒の番号を適当につけかえれば、 T_i は、

$$T_1 = T_{G1} + T_{G2} + \dots + T_{G(m/N)}$$

$$T_2 = T_{G(m/N+1)} + T_{G(m/N+2)} + \dots$$

$$+ T_{G(2m/N)}$$

⋮

これより、 γ は以下の様にして計算できる。

1) Rete ネットの特性より、処理量の分布および逐次処理されるべき粒の数の分布はわかるから、これらを合成して、上式の T_{Gi} の分布を計算する。

2) 上で求めた分布を持つ母集団から取り出した m/N 個の標本値の総和の分布(すなわちPE処理時間 T_i の分布)を計算し、その平均値 $E(T_i)$ を求める。

3) T_i の分布を持つ母集団から、 N 個の標本値を取り出した場合の最大値の分布を計算し、この平均値 $E(\max T_i)$ を求める。

$$4) \gamma = \frac{E(\max T_i)}{E(T_i)}$$

を計算する。

このモデルでは γ の性質は以下の様になる。

- ・各PEに割り付けられる粒が多ければ、処理時間は平均化され、処理量分布、逐時性の影響はともに少ない。逆に、並列度が小さければ、これらの影響が顕著になる。

- ・逐次性大の場合は、 G_i が含む粒の数が多く、粒の処理量ばらつきは平均化されるため、 γ はこの逐時性に支配される。

- ・逐時性小の時は、処理量分布の影響が顕著になる。

《通信オーバーヘッド係数 C_o 》

1認知サイクルでの総照会回数を N_m とし、これに対する通信回数の比率を C_R とする。また1回の照会時間を T_m 、これに対する1データ通信の時間の比率を C_c とする。これよりPEあたりの平均通信時間 T_c は、

$$T_c = \frac{N_m \cdot C_R \cdot T_m \cdot C_c}{N}$$

となる。ところで、 $N_m \cdot T_m$ は QT_o に等しいから、

$$T_c = \frac{QT_o}{N} C_R \cdot C_c$$

したがって、

$$C_0 = C_R \cdot C_C$$

ただし、バス結合型の様に通信競合が起るアーキテクチャの場合には、 C_C はこれによる遅延を含めて考える必要がある。

3.2 並列化粒度と基本方式

Retenetの構造の並列性に注目すると、以下の並列粒度が抽出できる。

- (1) トップノード：一段目の2入力ノード(トップノード)とそれに連なる全てのノード
- (2) ルール
- (3) ノード

更に、データの並列性に注目して、

- (4) ジョインスライス：ノード内の1回の照合の粒度を抽出することができる(図3)。

表5 各方式のパラメータ

パラメータ	トップノード	ルール	ノード	ジョインスライス
粒の数 n	$\frac{T_t}{L}$	$\frac{Q \cdot T_t}{L}$	$\frac{T_t \cdot M}{L}$	$\frac{T_t \cdot M \cdot j}{L}$
Q	≈ 1	RE/S	1	1
処理量 分布	粒内の ノード数	粒内の ノード数	無視可	無視可
逐次性	なし	なし	ノードの 従属数	無視可
C_R	≈ 0	≈ 0	1/j	2

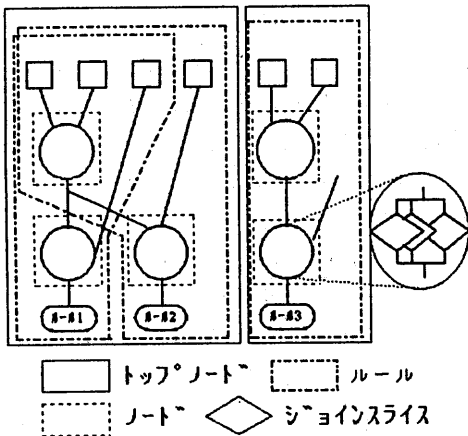


図3 Retenet並列化の粒度

これらの粒度に対応する基本的なアーキテクチャとしては以下が考えられる。

- ・トップノード・ルール：放送機能を持つ(トークンを全PEに配布するため)バス結合型
- ・ノード：ツリー、メッシュ等ネットワーク型
- ・ジョインスライス：トークン配布、収集機構を持つ図4に示す様なアーキテクチャ

Retenetモデルを用いて、これら方式の速度向上の計算に必要なパラメータを表5の様を求めることができる。

トップノード、ルール並列で、粒に含まれるノード数は、連結したノード数に等しく、これは、 B が小さいことから、条件要素数の分布とほぼ相似である。また、ノード並列では、ノードの処理量分布も負荷ばらつきの原因となるが、前節での考察から逐次性が支配的であり、無視できることがわかる。

3.3 速度向上度の評価結果

中規模のESを想定して速度向上を推定した結果を図5に示す。ESの問題パラメータは以下の通りである。

R	300	W	250	B	1.2
E	3.5	A	2.0	S_c	0.3
L	7	β_m	1.0		

また、速度向上に影響が大きいと考えられるR, E, Bを変化させた場合の変動範囲を図6に示す。ただし、各粒度の本質的な傾向を調べるために、 $C_C = 0$ とした。

更に、トップノード方式を実際のESを用いて実測した結果を図7に示す。サンプルとしては、ルール数23、MWE数1000の診断型

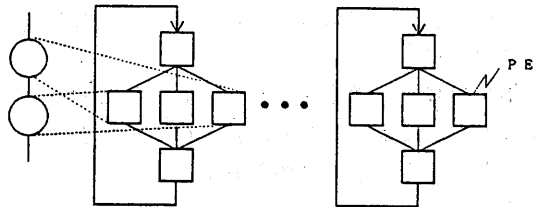


図4 ジョインスライスの基本アーキテクチャ

ESを用いた。実線は、推定式によるもので、良くあっていることがわかる。

3.4 各方式の比較

トップノード、ルールの粗粒度方式は、本質的に大きな速度向上が得られないことが図6からわかる。これは、並列処理の粒が少なくかつ粒の処理量が異なることに起因する、負荷ばらつきのためである。しかし、照合動作中に通信が不要なため、通信コストの高いアーキテクチャに適用できる。

ノード、ジョインスライスの細粒度方式では、潜在的には大きな速度向上が得られるが、通信コストの影響が大きい。このため、高速な通信機構を用いなければ、その潜在力を十分に引き出すことはできない。

PSの総合的な高速化を考えた場合、照合が約90%であることから、照合のみの速度向上は十数倍が妥当である。この観点で各方式をみると、中～大規模ESに対しては簡易なアーキテクチャで実現できる粗粒度方式で十分である。また、小ルール時には、粗粒度方式の負荷ばらつきを改善するか、通信コストの小さいアーキテクチャに細粒度方式を適用する必要がある。

将来、他のフェーズが高速化され、照合にさらなる速度向上が要求される場合には、通信専用ハードを用いるなどして、細粒度方式の潜在力を十分に引き出す必要がある。

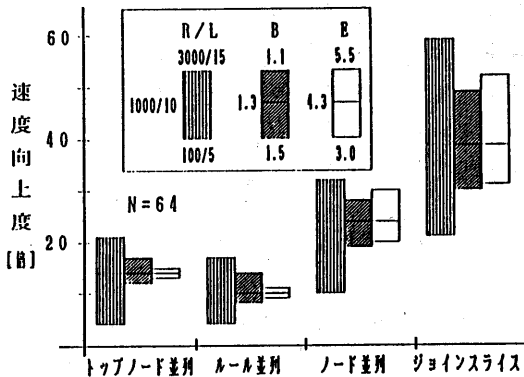
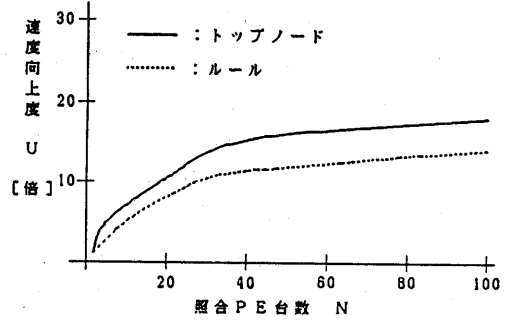
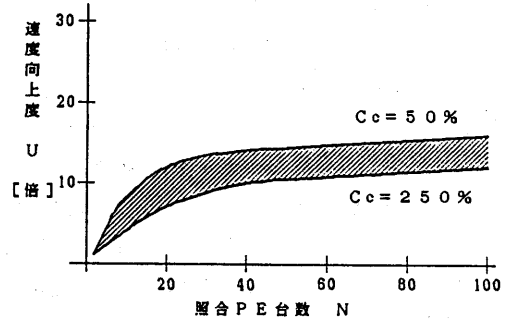


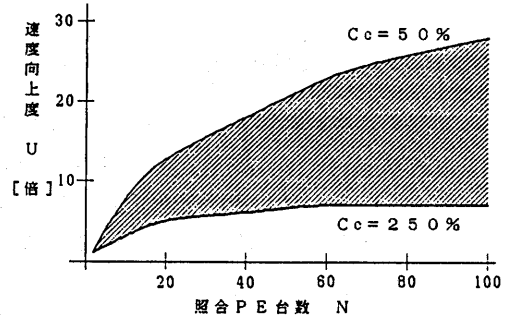
図6 速度向上の問題依存性



(a) トップノード・ルール



(b) ノード



(c) ジョインスライス

図5 解析評価による速度向上特性

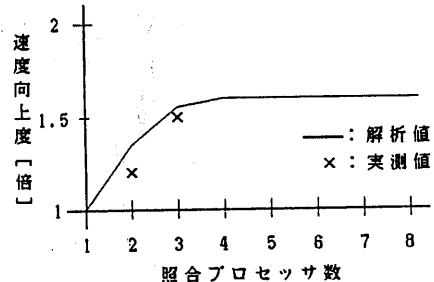


図7 速度向上度の実測値

4. 負荷均等化を図った粗粒度並列化方式

トップノード方式は、通信頻度が少ないため、比較的簡単なアーキテクチャにも適用が可能という特徴を持つ。しかし、負荷ばらつきの影響により、ルール数が少ない時には、十分な速度向上が得られないという欠点がある。

本章では、粗粒度方式にデータ並列性を融合させる事により、負荷ばらつきを抑え、小ルール数でも十分な速度向上が得られる改良方式を提案する。

4.1 データ方向並列化の原理

2入力ノードは、入力枝上のメモリノードに蓄積された全てのトークンとの突合せを行っており、ここにデータ並列性が存在する。したがって、複数のPEに、ノードとそれに付随するメモリノードを重複して割り付け、以下の処理を行うことによりその並列性を引き出すことができる。

[2入力ノードの並列処理]

トークン蓄積はどれか一つのPEが行なう
トークンの突合せは全てのPEで行なう

蓄積をどれかのPEが選択的に行うことにより、メモリノード上のトークンは各PEに分散される。一方、突合せは全PEが行うため、対向枝上で同様の蓄積動作により分散されたト

ークンと完全にしかも並列に照合が行われる。

すなわち、ひとつのトークンに対する上記の動作は、対向枝からの照合に備えたトークンの分散蓄積と、実際の並列照合処理の両方を同時に行っていることになる(図8)。また、この並列化では、トークンが入力された後は、PE間でのデータ交換は一切不要である。

4.2 データ並列を融合したトップノード方式

本方式は、トップノード方式の構造並列性をそのまま使い、最上段の2入力ノードのみに上記のノード並列化を施すことにより負荷の均等化を図った方式である。

その動作概要を図9に示す。最上段ノードにトークンが入力され並列照合が行われると、その照合結果が入力される下段ノードも自動的に

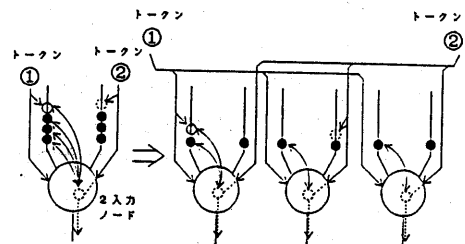


図8 2入力ノードの並列処理

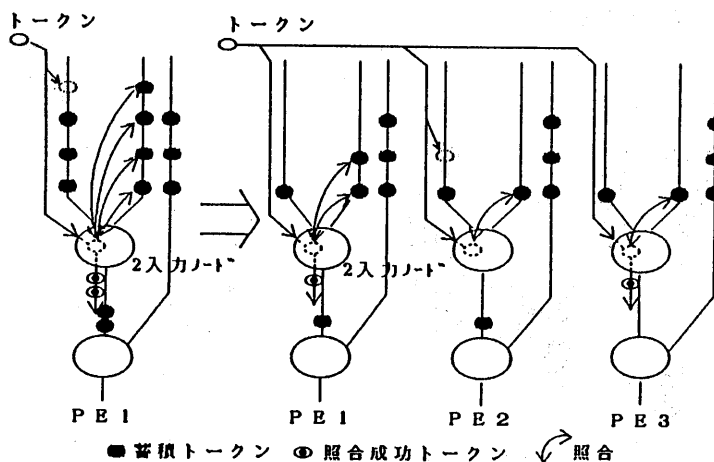


図9 負荷均等化トップノード方式

並列化される。また、2入力ノード間に蓄積されるトークンも分散される。一方、2段目以降のノードの右側からトークンが入力された場合には、全てのPEがトークン蓄積を行うが、この場合でも、対向枝のトークンは分散されているため照合は並列化される。

このように、最上段のノードのみにデータ並列化機能を持たせることにより、従来の特徴である通信不要性を保ったまま、全体を並列に処理できる。

ここで、データ方向の並列度は、原理的にはメモリノードに蓄積されたトークン数まで可能であるが、実際には、トークン数に応じて並列度を動的に変化させるのは困難であるし、2段目以降のメモリノードに重複して蓄積されるトークンにより、メモリ消費が大きくなるため、数〜十数にとどめるのが得策である。

本方式の解析評価結果を図10に示す。ESパラメータは、3章での解析と同一のものを用い、データ並列度は3とした。WME数が多ければ、小ルール数でも十分な速度向上が得られる。

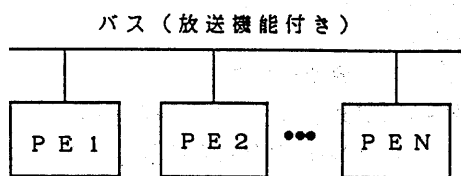


図11 実験機のアーキテクチャ

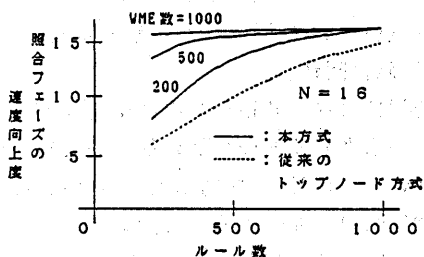


図10 負荷均等化方式の解析評価

4.4 実験による評価

本方式の効果を確認するために、実験を行なった。実験システムは、バス結合型で、ルール実行で発生したトークンを同時にPEに転送できるように放送機能を備えている(図11)。

3章と同じESを用いた実測結果を図12に示す。従来方式では50%であった並列化効率(U/N)が65%にまで向上している。

今後は、より大規模なESを用い、より多くのPEで実験する予定である。

5. まとめと今後の課題

ESの条件要素数の度数分布が指数関数的に減少している点に注目して定式化したReteneットのモデルを用い、逐次性・処理量分布に起因する負荷ばらつき、通信オーバーヘッド等を考慮して、速度向上推定の一般式を導出した。

また、この推定式を用いて、粒度の異なる4種類の基本方式を評価し、粗粒度方式では負荷ばらつきが、細粒度では通信オーバーヘッドが問題となる事を指摘した。

更に、粗粒度方式で小ルール数・大WMEのES領域をカバーする事をねらいとして、データ並列性を融合させた、トップノードの改良方式を提案した。また、本方式を解析、実験により評価し、小ルール数での速度向上の改善を確認した。

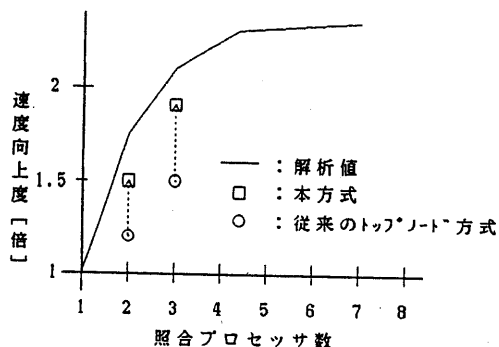


図12 負荷均等化方式の実験評価

今後、ルール実行・競合解消も並列化された場合^{[5][6]}、照合も更に高速性が要求される。そのために、潜在能力の高い細粒度方式について、通信コストの影響の少ない改良方式を検討してゆく予定である。

謝辞 日頃ご指導いただき、NTT情報通信処理研究所知識処理研究部 村上研究部長、ならびに、ご討論いただいた知識処理研究部の諸氏に感謝致します。

<< 参考文献 >>

- [1]A. Gupta, C. L. Forgy, A. Newell and R. Wedig. "Parallel Algorithms and Architectures for Rule-Based System", Proc. of ISCA, 1986.
- [2]M. A. Kelly and R. E. Seviora, "A Multiprocessor Architecture for Production System", Proc. of AAAI, 1987
- [3]鈴岡, 藤田, J. R. Canfield, 小柳. "フログクソシステムの並列処理方式", 人工知能全国大会論文集, 1988
- [4]A. Gupta and C. L. Forgy, "Measurements on Production System", Tech. Report CMU-CS-83-167, Carnegie-Mellon Univ., 1984
- [5]石田, "フログクソシステムの並列実行可能性の解析", 信学論 Vol. J71-D No. 3, 1988
- [6]長野, 松澤, "フログクソシステムの非同期実行方式", 信学全大論文集, 1989

付録 Rete ネット特性の導出

1) 1入力ノードの総数: D_1

M - M 間で共通の定数リストがなければ、

$R(E-1)$ であるが、 S_c だけ共通化されているので、

$$D_1 = R(E-1)(1-S_c)$$

2) 2入力ノードの総数: D_2

ノードは確率 P で終端するため、 m 段目の2入力ノードの終端数 b_m は、ノード数を D_{2m} として、

$$b_m = D_{2m} \cdot P$$

この総数は M - M 数 R に等しいから、

$$R = \sum b_m = P \sum D_{2m}$$

従って、2入力ノードの総数 D_2 は、

$$D_2 = R/P$$

3) 1入力ノードから2入力ノードへの接続枝数 S

Reteネットでは、2入力ノードの右入力枝は、必ず1入力ノードに接続されている。また、1段目では、左入力枝も1入力ノードに接続される。従って、

$$S = D_{21} + \sum D_{2m}$$

D_2 を D で表わして代入すれば、

$$S = D_2 (2 - (B-P))$$

4) 活性化1入力ノード数 O_1

活性化される1入力ノードは、入力されたトークンのクラスと等しいクラスをもつ。クラス毎の条件要素数が均等との仮定より、

$$O_1 = D_1/L$$

5) 1入力ノード通過トークン数 T_t

照合に成功する1入力ノード数は、 $O_1 \cdot \alpha_c$ 。1入力ノードは平均 S/D_1 の分岐を持ち、これら全てにトークンが配布されるから、

$$T_t = O_1 \cdot \alpha_c \cdot S/D_1 = S \cdot \alpha_c/L$$

6) 活性化2入力ノード数 O_2

2入力ノードの平均従属段数は $E-1$ なので、これと T_t との積が、活性化2入力ノード数となる。

7) 平均照合回数 j

照合対象となる蓄積トークンの数は、全 WM のうち定数リストを通過するものの数に等しいため、クラスあたりの WME 数と定数リスト成功率との積になる。

8) 1認知サイクルで発生するトークン数 M

トークンは生成・削除動作で1個、更新動作で2個発生するため、実行要素数と $(1+\beta n)$ との積になる。