

ショートノート**条件付連想 Rete ネットワーク†**

荒屋真二‡

プロダクションシステムの高速実行方式として Rete ネット方式がよく知られている。各ルールの動作部と条件部との相互関係に関する知識を用いて、Rete ネットをさらに構造化したものが、連想 Rete ネット方式である。この改良方式では、トークンを Rete ネットの途中の 1 入力ノード（連想ノード）から直接流せるので照合範囲が局所化される。しかし、発火ルールの動作部に変数を含む場合には、あらかじめテスト結果がわかっている 1 入力ノードにもトークンが流れるという無駄があった。本論文では、条件付連想ノードという概念を導入することにより、このような無駄な照合を完全に除去できることを示す。

**1. まえがき**

プロダクションシステム (PS) の高速化手法として、推論実行前にルール条件部をコンパイルする Rete アルゴリズムが良く知られている<sup>1)</sup>。しかし、このアルゴリズムにはまだかなり無駄なパターン照合が残されており、いくつかの改良方式が提案されてきた<sup>2)~4)</sup>。

これら改良方式の中で最も効率的なものは連想 Rete ネット方式である<sup>4)</sup>。なぜなら、この方式は、他方式がトークンを常にルートノードから流すのに対し、Rete ネットの途中のノード（連想ノード）からトークンを直接流すため、パターン照合の範囲を局所化できるからである。しかし、この方式も無駄なパターン照合を完全に除去しているわけではない。すなわち、連想ノードの下流においては、ノードテストの結果があらかじめわかるにもかかわらず、推論実行の度にパターン照合が繰り返されるからである。

本論文では、従来の連想ノードを拡張した条件付連想ノードという概念を導入することにより、上述の無駄なパターン照合を回避する方法を提案する。この方法では、各動作項に含まれる変数に対する条件（連想条件）と、その条件を満足したときにトークンが必ず到達する最も深いノード（条件付連想ノード）を各動作項に附加しておく。推論実行時には、トークンが連想条件を満足すれば、直接、条件付連想ノードから流される。この方式では、発火ルールの動作項に含まれる変数に関するパターン照合と、条件パターン間にまたがるパターン照合だけが推論実行時に行われる。

† A Conditional Associable Rete Network by SINJI ARAYA  
(Department of Communication and Computer Engineering,  
Faculty of Engineering, Fukuoka Institute of Technology).

‡ 福岡工業大学工学部通信工学科

次章では連想 Rete ネットの概要とその問題点を述べる。3 章では条件付連想 Rete ネットを提案すると共に、その有効性を示す。4 章では条件付連想 Rete ネットの生成法を述べる。5 章では本論文をまとめたあと、今後の課題について述べる。

**2. 連想 Rete ネットとその問題点**

連想 Rete ネットとは、作業記憶要素 (WME) の追加・削除を引き起こす動作項から、連想ノードへ至るリンクを、Rete ネットに追加したものである。ここで、連想ノードとは、WME が Rete ネットをトークンとして流れるとときに必ず到達する最も深いノードで、かつ、その下流にある端末ノードのうち少なくとも一つに到達する可能性があるノードである。

図 1 は Rete ネット、連想 Rete ネット、および条件付連想 Rete ネットを比較するために三つを合体して書いたものである。図 1 において、Rete ネットはルール  $R_i$  の動作項 ( $MAKE C_i \wedge A_{i1} 1 \wedge A_{i2} < X > \wedge A_{i3} 3 \wedge A_{i4} 4 \wedge A_{i5} 5$ ) からルートノードへのリンクだけが付いたものである（このリンクは実際には存在しないが、Rete ネットではトークンが常にルートノードから流されることを強調するために図示したものである）。連想 Rete ネットは、 $R_i$  の動作項から 1 入力ノード “ $A_{i2} > 7$ ” へのリンクだけが付いたものであり、ノード “ $A_{i2} > 7$ ” がその動作項の連想ノードである。条件付連想 Rete ネットについては次章で述べる。

今、ルール  $R_i$  が発火し、そのとき動作項の変数  $< X >$  は 8 に束縛されているものとする。また、動作パターン ( $C_i \wedge A_{i1} 1 \wedge A_{i2} < X > \wedge A_{i3} 3 \wedge A_{i4} 4 \wedge A_{i5} 5$ ) は、ルートノードから端末ノード  $R_j$  に至るパスを

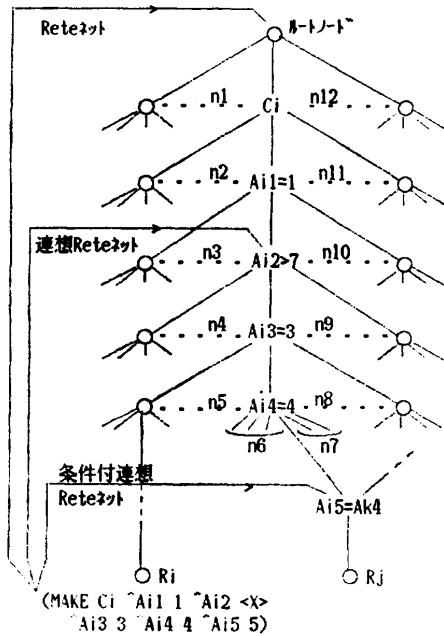


図 1 Rete ネット、連想 Rete ネット、および  
条件付連想 Rete ネット  
Fig. 1 Rete net, associative Rete net, and  
conditional associative Rete net.

通る時にのみ、すべてのパタン照合に成功するものとする。Rete ネットでは、 $R_i$  の動作項によって WM に追加される ( $C_i \wedge A_{i1} 1 \wedge A_{i2} 8 \wedge A_{i3} 3 \wedge A_{i4} 4 \wedge A_{i5} 5$ ) をトークンとしてルートノードから流す。一方、連想 Rete ネットでは、そのトークンを連想ノード “ $A_{i2}>7$ ” から直接流す。ゆえに、Rete ネットに比較して、明らかに失敗に終る  $n_1 + n_2 + n_3 + n_{10} + n_{11} + n_{12}$  回以上のパタン照合と、明らかに成功する 2 回のパタン照合 ( $C_i, A_{i1}=1$ ) をせずに済む。しかし、連想ノード “ $A_{i2}>7$ ” の下流の 1 入力ノードにおいては、 $n_4 + n_5 + \dots + n_9$  回以上の照合に失敗し、2 回の照合に成功する。これらのパタン照合を推論実行時に行なうことは明らかに無駄である。なぜならば、ノード “ $A_{i3}=3$ ” とノード “ $A_{i4}=4$ ” を経て 2 入力ノード “ $A_{i5}=A_{k4}$ ” に至るパスだけが成功するということは、動作項の対応する属性値が定数であるので、推論実行前にわかるからである。

この例からもわかるように、連想 Rete ネットでは、動作項に含まれている変数に関するテストを行うノードが上流にあるほど、その下流において無駄なパタン照合が発生する。

### 3. 条件付連想 Rete ネット

前章で述べた連想 Rete ネットの無駄なパタン照合を回避する方法として、動作項の変数に関するテストができるだけ下流で行なうように Rete ネットを最適化することが考えられる。しかし、これには次のような問題点が存在する。すなわち、あるパスを通過する可能性のある動作項は一般に複数存在し、それらの動作項に含まれる変数は同一属性に対するものとは限らない。ゆえに、ある一つの動作項に対して最適化しても、他の動作項に対しては最適となるとは限らない。また、この種の最適化は、同一条件をできるだけ一つのノードにまとめるという、Rete ネット本来の利点を損うことになる。

そこで本論文では、Rete ネット部分には修正を加えず、前章で述べた連想ノードを拡張した条件付連想ノードというものを導入した条件付連想 Rete ネットを提案する。図 1において、ルール  $R_i$  の動作項から出ている 3 本のリンクのうち、2 入力ノード “ $A_{i5}=A_{k4}$ ” へのリンクだけをもつものが条件付連想 Rete ネットである。この 2 入力ノードをその動作項の条件付連想ノードと呼ぶ。また、トークンが条件付連想ノードに送出される条件を連想条件と呼び、この場合、それは “ $A_{i2}>7$ ” である。

推論実行時にルール  $R_i$  が発火すると、まず、( $C_i \wedge A_{i1} 1 \wedge A_{i2} 8 \wedge A_{i3} 3 \wedge A_{i4} 4 \wedge A_{i5} 5$ ) が WM に追加される。次に、この WME は連想条件 “ $A_{i2}>7$ ” を満たしているので、2 入力ノード “ $A_{i5}=A_{k4}$ ” の左メモリにトークンが入れられ、トークン処理が開始される。ゆえに、パタン照合回数は、連想 Rete ネットに比べ、 $n_4 + n_5 + \dots + n_9 + 2$  回以上減少する。もし、 $<X><=7$  のときには、連想条件を満たさないのでトークンは送出されない。この場合のパタン照合回数は、連想 Rete ネットと同じとなる。

このように、条件付連想 Rete ネットは、連想条件が満たされたときには、失敗および成功のパタン照合回数が減少する。一方、連想条件が満たされないときには、連想 Rete ネットの照合回数とほとんど同じとなる。図 1 は、連想条件が一つの場合の例であるが、一般には、連想条件は複数個存在する（4 章参照）。条件付連想 Rete ネットのメモリ効率は、連想条件を格納しておくためのメモリが余分に必要なため、連想 Rete ネットよりも悪くなる。また、動作項の変数の数が多くなるほど、連想 Rete ネットの場合と同様

に、推論効率化の効果は減少する。なぜならば、動作項に変数が増えると、連想 Rete ネットでは連想ノードが上流の方（ルートノードに近い方）に位置するようになり、条件付連想 Rete ネットでは連想条件の数が増加するからである。しかし、条件付連想 Rete ネットの推論効率が、Rete ネットや連想 Rete ネットよりも劣ることは絶対にない。

#### 4. 条件付連想 Rete ネットの生成法

条件付連想ノード及びその連想条件は、連想 Rete ネットを生成するときに付随的に入手できる。連想ノードは、WM に対する追加・削除を引き起こす動作項の動作パターンをトークンとして、Rete ネットのルートノードから流すことによって効率的に求められる<sup>4)</sup>。ここで動作パターンとは、動作項から動作関数 make や remove を除いたものである。一般に、動作パターンには変数が含まれるので、各ノードでのテスト結果は確定しない。ゆえに、テスト結果として成功、失敗のほかに成功可能というものが導入され、成功可能なときにもトークンが後続ノードに送られるようになっている。こうして、トークンがある端末ノードに到達したときに、ルートノードからその端末ノードへ至るパスの中で最も上流に位置する成功可能ノードが連想ノードとなる。

さて、条件付連想ノードは、上述のパスに含まれる最も上流にある 2 入力ノードである。もし、そのパスの中に 2 入力ノードが一つもないときには、端末ノードが条件付連想ノードとなる。2 入力ノードでは二つ以上のトークンが関連するので、あらかじめテスト結果が成功と確定することではなく、推論実行時にトークンを流す必要がある。

一方、連想条件はそのパスに含まれる、成功可能と判定された 1 入力ノードの条件の集合である。特殊な場合として、そのパスの中に、成功可能と判定された 1 入力ノードが一つも存在しないこともあります。このときには、連想条件がないことを意味し、条件付連想ノードと連想ノードは完全に一致する。

以上の生成法からもわかるように、1 入力ノードが条件付連想ノードとなることはない。ゆえに、推論実行時にトークンが 1 入力ノードに流れることはない。ゆえに、WM の初期値をあるルールで設定するようになるならば、Rete ネットの 1 入力ノードの部分を除去することも可能である。しかし、WM の初期値を

ルールによって設定しないときや、条件付連想 Rete ネットを効率的に生成するためには 1 入力ノードの部分を残しておく必要がある。

#### 5. あとがき

本論文では、PS の高速化のための連想 Rete ネット方式における無駄なパタン照合の存在を明らかにし、条件付連想ノードという概念を導入することにより、それを回避できることを示した。提案方式の効果は、本文でも述べたように、ルール動作部の変数の数の影響を受ける。ゆえに、その定量評価に当っては、多くの実際的プログラムを用いて評価データを蓄積して行かなければならない。

**謝辞** 本研究の一部は福岡工業大学附属エレクトロニクス研究所より助成を受けた。ここに、心より謝意を表する。

#### 参考文献

- 1) Forgy, C. L.: Rete: A Fast Algorithm for the Many Pattern/Many Object Pattern Match Problem, *Artif. Intell.*, Vol. 19, pp. 17-37 (1982).
- 2) 荒屋ほか：プロダクションシステムのための高速パターン照合アルゴリズム、情報処理学会論文誌, Vol. 28, No. 7, pp. 768-775 (1987).
- 3) 田野ほか：知識ベースシステム構築用ツール EUREKA における高速処理方式、情報処理学会論文誌, Vol. 28, No. 12, pp. 1255-1268 (1987).
- 4) 荒屋ほか：プロダクションシステムにおける効率的パタン照合のための連想 Rete ネットワーク表現、情報処理学会論文誌, Vol. 29, No. 8, pp. 741-748 (1988).

(昭和 63 年 5 月 2 日受付)  
(昭和 63 年 9 月 5 日採録)



荒屋 真二（正会員）

昭和 24 年生。昭和 47 年東北大学工学部通信工学科卒業。同年三菱電機(株)入社。昭和 60 年福岡工業大学通信工学科助教授。この間、昭和 58 年 4 月～9 月神戸大学工学部非常勤講師。工学博士（東京大学）。人工知能、知識工学、コンピュータミュージックの研究に従事。昭和 57 年電気学会論文賞受賞。電気学会、電子情報通信学会、人工知能学会、IEEE 各会員。