

エキスパートシステムのための実時間推論データベースシステムの設計

3D-10

- 推論アルゴリズム -

小野 智弘

西山 智

小花 貞夫

鈴木 健二

国際電信電話株式会社 研究所

1. はじめに

筆者らは、網管理分野等におけるエキスパートシステムを構築するために、大量のデータ (WME:Working Memory Element, 推論に用いる事実) を扱え、実時間応答を実現した推論機能を持つデータベースシステムの開発を行なっている^[1]。本システムでは、LEAPS アルゴリズム^[2]を拡張して外界からの WME の変更に対して実時間で応答し、かつ永続的 WME を扱うことができる推論アルゴリズムを採用している。本稿ではこの推論アルゴリズムの概要を報告する。

2. LEAPS アルゴリズム

一般にエキスパートシステムでは図1の例に示すようなルールと WME を用いて、以下の処理を繰り返す認識-実行サイクルによって目的の動作を実行する。

[マッチング] CE(Conditional Element:WME の満たすべき条件) と WME を照合して発火可能なインスタンシエーション(実行可能なルールと、対応する WME の組)を見つける。

[競合解消] 実際に発火させるインスタンシエーションを選択する。

[ルール実行] RHS(Right Hand Side:アクションの並び) を実行する。



図1: ルールと WME の記述例

LEAPS は、発火する優先度の高いものから順にマッチングを行ない、発火可能なインスタンシエーションを見つけると同時に発火させるアルゴリズムである。図2(a)に示すように、CE 每に α メモリ (AMEM) を設け、各 CE にマッチした WME のタイムスタンプ (WME をユニークに識別するタグ) と属性 (WME の内容) を格納する。また、シード (推論の起点) とすべき WME を指

"Design of Database System with Real-Time Inference Mechanism for Expert Systems -Inferencing Algorithm-" by Chihiro ONO, Satoshi NISHIYAMA, Sadao OBANA and Kenji SUZUKI, KDD R & D Laboratories

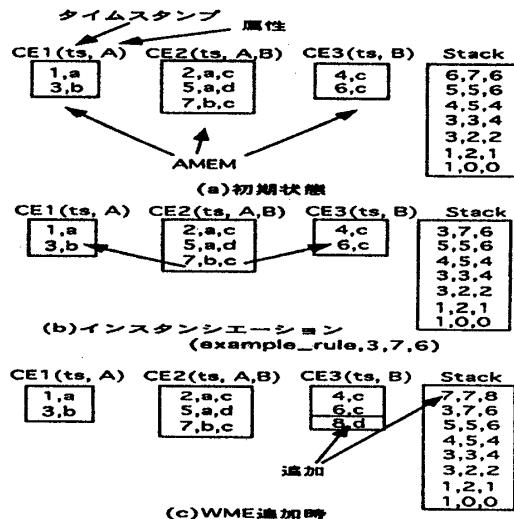


図2: LEAPS アルゴリズム

定るためにスタックを設け、マッチングの途中結果や新たに追加された WME のタイムスタンプの組をプッシュする。

例えば、スタックの最上位 (図2(a) では (6,7,6)) をポップしてマッチングを始め、発火可能なインスタンシエーション (example_rule,3,7,6) を見つけた時点で途中結果 (3,7,6) をスタックに保存し (図2(b))、RHS を実行する。その後スタックから次のシードをポップし、マッチングを再開する。RHS の実行によって WME が追加された場合は図2(c) に示すように、AMEM とスタックへ WME のタイムスタンプを追加することでマッチングに反映させる。

3. LEAPS アルゴリズムの適用性

3.1 LEAPS の利点

(1) RETE^[3] や TREAT^[4] に比べて高速である^[2]。

(2) RETE や TREAT が、競合解消の完了までは外部からの割り込みをかけることができないのでに対して、LEAPS は発火可能なインスタンシエーションを見つける度に発火させるアルゴリズムであるため、マッチング中に割り込みをかけることができる。

3.2 LEAPS への拡張の必要性

LEAPS を実時間推論データベースへ適用する際には、以下の 2 つの拡張を行なう必要がある。

外部からの WME の変更に対する実時間対応

LEAPS では WME の変化の要因は RHS の実行による WME の変更のみであったが、網管理システム等の応用を考えると、アラーム等の外界からの WME の変更が生じるため、外部からの WME の変更も変化の要因となる。従って、外部からの WME の変更に対して、認識一実行サイクルの中で最も時間を要するマッチング部において外界の変化を受け付け、実時間の応答を可能とする必要がある。

永続的 WME への対応 LEAPS は全ての WME をシードとしてマッチング操作を行なうアルゴリズムであり、かつ全ての WME がプログラムの起動後に生成される。ところが、実時間推論データベースでは非永続 WME のみならず、プログラムが起動された時点で既に存在する永続的 WME も扱わなければならぬ。この場合、タイムスタンプ、AMEM、スタック等の扱い方を考える必要がある。

4. LEAPS アルゴリズムの拡張

4.1 外部からの WME の変更に対する実時間対応

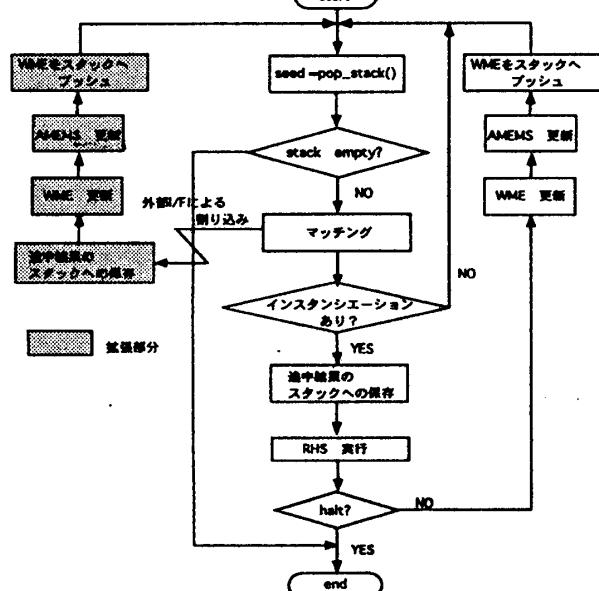


図 3: 拡張した LEAPS の認識一実行サイクル

図 3 に拡張した LEAPS の認識一実行サイクルを示す。マッチングループ中に外界からの WME の変更を割り込みとして受け付けると、直ちにマッチングを中断して途中経過(マッチングのループで最後に評価した WME の組)をスタックにプッシュする。続いて WME の変更を反映させ、対応する AMEM に WME を追加し、スタックにもプッシュする。一連の操作が終了した後に、スタックからシードをポップしてマッチングを再

開する。

4.2 永続的 WME への対応

永続的 WME に対応するために行なった拡張点をタイムスタンプ、AMEM、スタックについて述べる。

タイムスタンプ LEAPS ではマッチング順序を制御するために各 WME にタイムスタンプが必要である。そこで、タイムスタンプの値を永続的とし、永続/非永続の区別なく現在のタイムスタンプの最大値+1 を新しい WME に付与する。これにより、WME の新旧関係とタイムスタンプ値の大小関係が常に正しく保たれる。

AMEM 本アルゴリズムではルールは動的に変化しないことを仮定しているため、過去のプログラムで行なった(WME の CE に対する)マッチング結果を保存する AMEM の内容は、新しいプログラムでも利用可能である。従って永続的な WME に対しては永続的な AMEM を設ける。1 つの WME が多数の AMEM に格納される場合もあるが、等価な CE 間で AMEM を共有することで、大規模な永続的 WME に対しても AMEM に必要な領域を少なくすることができる。

スタック スタックについては、過去のプログラムでの内容は新しいプログラムには必ずしも有効でないため、永続化せずにメモリ上におく。推論プログラム起動時にシードとする必要のあるものについてはアプリケーション側で明示的にスタックに積み、その後は起動後のデータの変化差分のみをシードとして推論を行なうこととした。

5. おわりに

本稿ではエキスパートシステムを構築するための実時間推論データベースシステムにおける推論アルゴリズムを報告した。本アルゴリズムは LEAPS を拡張し、推論中に割り込みをかけることによって外界からの WME の変更に対して実時間で応答可能とした。また、永続的 WME を扱うために、タイムスタンプ、AMEM やスタック等に変更を加えた。今後は、本アルゴリズムを実装し、評価を行っていく予定である。最後に日頃御指導頂く KDD 研究所 浦野所長に感謝します。

参考文献

- [1] 西山 他：“エキスパートシステムのための実時間推論データベースシステムの設計—全体概要—,” 第 51 回情報全大, 3D-9, (1995)
- [2] Daniel P. Miranker et. al.: “On the Performance of Lazy Matching in Production Systems,” Proc. of Eighth National Conference on Artificial Intelligence, (1990).
- [3] Forgy C. : “RETE: A Fast Match Algorithm for the Many Pattern/Many Object Pattern Match Problem,” Artificial Intelligence, no.19, pp.17-37, (1982)
- [4] Daniel P. Miranker : “TREAT: A Better Match Algorithm for AI Production Systems,” Proc. of the 1987 National Conference on Artificial Intelligence, (1987)