

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評価

2M-6

西山 智

小野 智弘

小花 貞夫

国際電信電話株式会社 研究所

1. はじめに

網管理等の分野でのエキスパートシステム (ES) は、例えば網のトラフィック情報等大量のデータを用いて推論 (大量データ処理) し、かつ網から発生するアラーム等の外界の変化に実時間で応答 (実時間処理) する必要がある。筆者らは、このような ES の構築を容易とするために、大量データ処理と実時間処理が可能な実時間推論データベースシステムを実装した^{[1][2]}。本稿では、その評価について報告する。

2. 実時間推論データベースシステムの概要

本システムは、以下の特徴を持つ。

(1) 大量データ処理

推論機能をデータベースに組み込む (図 1(a)) ことにより、従来の汎用 ES では扱う事が困難な大量データをデータベースから直接用いて推論できる。

(2) 実時間処理

マッチングアルゴリズムとして、最新のデータ (シード) から入れ子ループによる最良優先探索を行い、また探索中にも外界からのデータの変化を割り込みにより受け付ける (図 1(b)) アルゴリズム^[2]を使用している。これにより、ルールの実行を待つことなく一定時間で外界の変化を推論に反映するという、従来の汎用 ES シェルでは困難な機能を実現している。

ソフトウェア構成を図 2 に示す。開発者はアプリケーションプログラムを OPS83^[3] 準拠のルール記述言語で記述し、ルールコンパイラがそれを C++ に変換する。そのソースとライブラリを C++ コンパイラによりコンパイルし、その応用に対する実時間推論データベースシステムを作成する。開発者はアプリケーション上で宣言するだけで、主記憶上の非永続データとデータベース上の永続データを区別なく推論に使用できる。作成されるソフトウェアは、推論プロセス、DB I/F プロセス、外部 I/F プロセスの 3 プロセスからなる。これら 3 つのプロセスは、共有メモリおよびメッセージキューを介して情報を共有する。

3. 評価

3.1 評価環境

評価では、図 3 に示すルールシステムを使用した。図 3 において、クラス a, b の宣言部の [persistent] はデータ

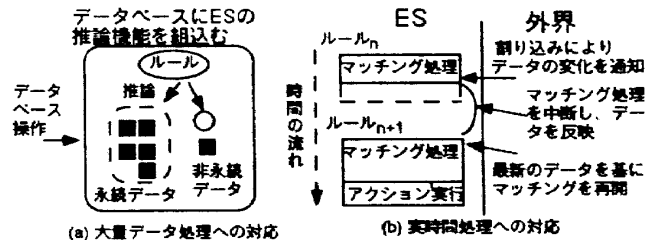


図 1: 大量データ処理と実時間処理の実現

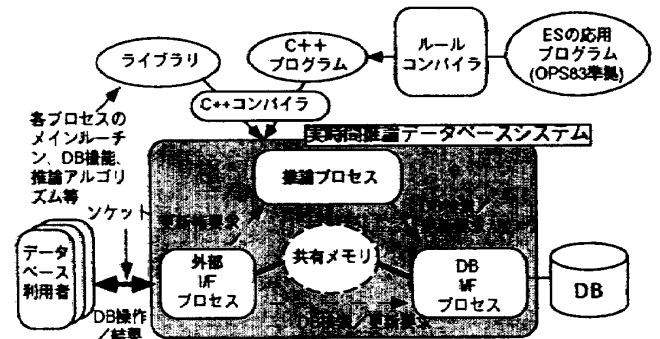


図 2: ソフトウェア構成

```

type a = [persistent] element (
    a1: integer ;
    a2: symbol ; );
type b = [persistent] element (
    b1: integer ;
    b2: symbol ; );
type c = persistent element ( c1:integer; );
rule match_AB { -- 条件節 (CE) を 2 つ含む
    &CE1 ( a ) ;
    &CE2 ( b b1 = &CE1.a1 ; ); -- a と b を join --
    -->
    -- 実際のアクション無し -- }
rule match_C {
    ( c ) ;
    -->
    -- 実際のアクション無し -- }
    
```

図 3: 評価に使用したルールシステム

の永続/非永続に応じてデータの永続性を示すキーワード persistent を付与した/しないことを示す。このルールシステムに対して、クラス a および b の初期に生成するデータ件数を変化させた。評価は、Sun SPARC Server 1000 (Solaris2.5.1) 上で行い、非永続データ格納用に 2M バイト、永続データのバッファ用に 3M バイトの共有メモリを割り当てた。

"Evaluation of Database System with Real-time Inference Mechanism for Expert Systems" by Satoshi NISHIYAMA, Chihiro ONO and Sadao OBANA, KDD R&D Labs.

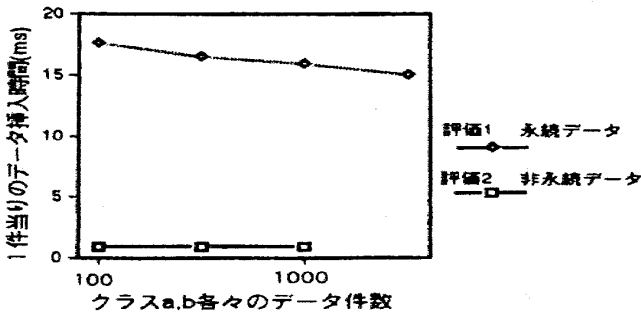


図 4: データ作成時間

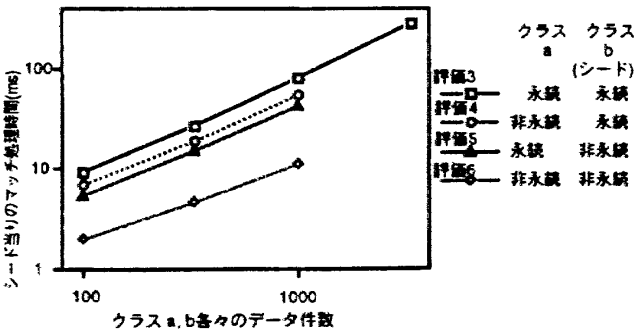


図 5: マッチング処理時間

3.2 永続データと非永続データによる性能変化

永続データと非永続データを用いる場合の推論性能を比較するために、クラス a および b をそれぞれ永続データ、非永続データとした場合の、データ作成時間を測定した (評価 1,2)。また、クラス a, b と永続、非永続の全ての組合せに対してマッチング時間を測定した (評価 3~6)。なお、クラス a, b の順で作成する事によりクラス b のデータが常にシードとなるようにした。図 4 に 1 データ当りの作成時間を、また図 5 に 1 シード当りのマッチング時間を示す。

(1) データ作成性能: 評価 2 より非永続データの作成性能は、データ件数にかかわらずほぼ一定であった。評価 1, 2 より永続データの性能は、非永続データに対して 1/20~1/17.7 程度であり、データ件数の増加に伴い改善する傾向にある。DB I/F プロセスでデータをバッファしている事から、そのヒット率の向上によると考えられる。データ件数をクラス a, b それぞれ 3.3×10^3 件とした場合、評価 2 は共有メモリ不足により作成できなかったが、評価 1 は DB に格納しているためデータ作成が行えた。このことから、使用できるメモリ量にかかわらず、大量のデータを用いて推論が行える事が実証できた。

(2) データ検索性能: 実装で使用したマッチングアルゴリズムでは、シードを起点にルール内の残りの条件節 (CE: Conditional Element) を入れ子ループにより検索するため、シード当りのマッチング処理の計算量はル-

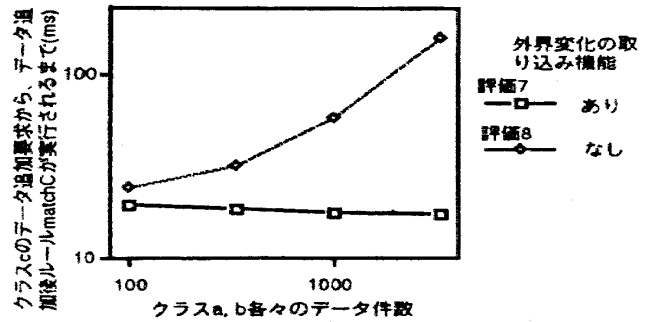


図 6: 外界の変化の取り込み時間

ル内 CE 数を c, データ件数を n とすると、 $O(n^{c-1})$ である。評価に用いたルール match_AB の CE 数は 2 であるので、シード当りのマッチング時間はデータ件数に比例する。評価の結果、評価 3~6 のいずれの場合にもほぼデータ件数に比例する処理時間となった。また評価 3 は評価 6 の 1/4.5~1/7.3 の性能であった。評価 4,5 は評価 3,6 のほぼ中間の性能であったが、評価 5 の方が約 20% 高速であった。評価 5 は入れ子ループの内側、評価 4 は外側が永続データである事からバッファのヒット率の違いによるものと考えられる。

3.3 外界の変化の取り込み性能

クラス a, b が両方とも永続データの場合において、マッチング処理中に外部 I/F プロセスがクラス c のデータの追加要求を受信してから、推論プロセスがルール match C を実行するまでの時間を測定した。外界の変化取り込みを行う場合 (評価 7) と、その機能を停止した場合 (評価 8) の実行時間を図 6 に示す。評価 7 ではデータ件数 (シード当りのマッチング時間) にかかわらずほぼ一定の時間で、match C が実行されている。一方評価 8 では、他のルール起動後に初めて外界の変化が取り込まれるため、ほぼデータ件数に比例して実行時間が延びた。このことから、データ件数にかかわらず外界の変化に対して実時間性が実現できていると言える。

4. おわりに

本稿では、大量データ処理と実時間処理が可能な実時間推論データベースシステムの設計の評価について報告した。評価の結果、使用できるメモリ量に係わらず大量のデータを用いて推論が行える事と、データ件数にかかわらず外界の変化に対して実時間応答性が実現できている事を確認した。最後に日頃御指導頂く KDD 研究所 村上 仁己取締役所長、鈴木 健二副所長に感謝します。

参考文献

- 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 回情処全大 3D-09, (1995).
- 小野 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-推論アルゴリズム-,” 第 51 回情処全大 3D-10, (1995).
- C. Forgy: “The OPS83 User's Manual Ver. 2.2,” (1986).