

実時間知識処理をめざした木探索制約推論の高速化手法

3N-4

遠城 秀和

NTT データ通信（株）開発本部

1 はじめに

近年の知識処理技術の進歩により、プロセス制御やオンライントランザクション処理のような実時間処理分野においても知識を使ったより複雑な処理の実現が求められている。これらの実時間処理においては、レスポンスタイムなどの時間条件を満たす保証がシステム実現上必要である [1]。しかし、従来の知識処理では、答えが有限時間内に求められる保証がなく、与えられた時間内に答えを求めるというレスポンスタイムの保証をすることができなかった。このため、実時間処理分野への知識処理技術の応用が困難であった。さらに上限保証した推論処理の高速化も与えられた時間内に推論処理を完了するために重要である。

本報告では、知識処理の一推論方法であり上限保証のある制約推論法において、制約充足手法の併合法が木探索法に変換可能なことを用い、併合法で提案されている高速化手法を木探索法に応用する方法を述べる。

2 実時間環境における知識処理のレスポンスタイム

従来のエキスパートシステムなどの知識処理では、「利用者などから質問を受け付け、質問に対する推論結果を答として返す」という処理が基本的流れになる。一方、実時間処理ではオンライン処理のトランザクションやプロセス制御のセンサデータなどが外部イベントとして入力され、そのイベントの処理結果が出力として返される。したがって知識処理を実時間処理に導入するには、外部イベントを知識処理への質問に変換し（入力変換処理）、推論結果をそのイベントに対する出力に変換する（出力変換処理）ことで可能となる。（図1）

実時間処理では、外部イベントの入力時からイベントに対する返事を返すまでを通常のレスポンスタイムとし

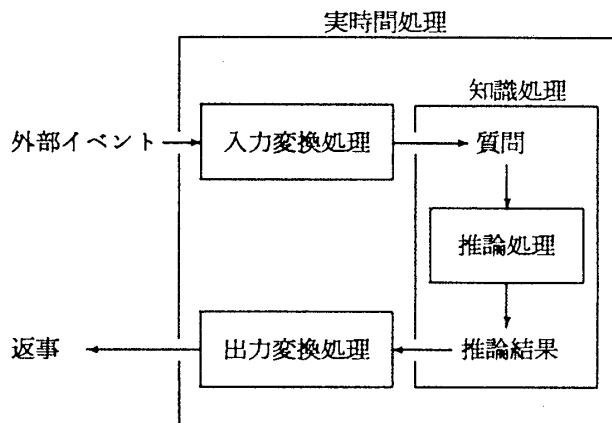


図1: 実時間環境における知識処理

ている。それと同様に、実時間環境における知識処理では外部イベントの入力時から、入力変換処理、推論処理、出力変換処理を順番に実行し、返事を返すまでをレスポンスタイムとすることになる。

したがって知識処理のレスポンスタイムを保証するには、処理全体に与えられた時間を分割し、決められた各時間内に入力変換処理、推論処理、出力変換処理の終了する保証が必要である。入力変換処理と出力変換処理は従来の実時間処理のモジュールとして実現でき、従来方法で与えられた時間内の終了を示すことが可能である。そこで、実時間環境における知識処理のレスポンスタイムは、入力変換処理、推論処理と出力変換処理の各処理時間の和として推定できる。

また、入力変換処理の処理時間と出力変換処理の処理時間は推論処理の処理時間に較べ十分に短く、入力変換処理や出力変換処理に割当てた時間内の終了保証は推論処理に較べ容易である。このため、実用的なレスポンスタイムの保証には、与えられた時間内での推論処理の終了を示しかつ推論処理を高速に行なうことが重要である。

3 推論処理における計算量

与えられた時間内での推論処理の終了を証明するには、推論処理時間に上限が存在しなければならない。さらに、レスポンスタイムの推定に用いるには、係数の値も含め

た完全な計算式の形での計算量推定法が必要である。

これまでに知識処理の推論方法としては、ルール、論理、セマンティックネット、フレーム、制約などが提案されている。その中で制約推論は、変数が有限個数の値を取る場合を対象として計算量理論による処理時間上限の解析が進んでいる [2] [3] [4]。しかし、これらの推定方法は既に提案されている高速化手法に十分対応していない。また、制約推論の解法である木探索法と併合法の比較も試みられている [5] が、高速化手法を含めて比較することが高速な手法の選択としては重要である。

4 木探索法の高速化

併合法で提案されている高速化手法が木探索法に適用可能であれば、併合法と木探索法は同等の処理速度を出すことが出来、手法選択の自由度が増す。そこで、制約充足手法の併合法が木探索法に変換可能なことを用い、併合法で提案されている高速化手法を木探索法に応用する方法を以下に述べる。

4.1 併合法の木探索法への対応付け

以下に示す方法で横型探索である併合法から縦型探索である木探索法への対応付けが可能である [5]。(図 2)

- (1) 頂点拘束ネットワークの各頂点毎に頂点の制約条件を満たす値の組を保持する変数(導入変数)を導入する。
- (2) 導入変数の順序は併合していく順序と同じにする。
- (3) 導入変数間の制約条件は、一つ前までの全ての導入変数と自導入変数間で、原問題の変数に関する整合性とする。

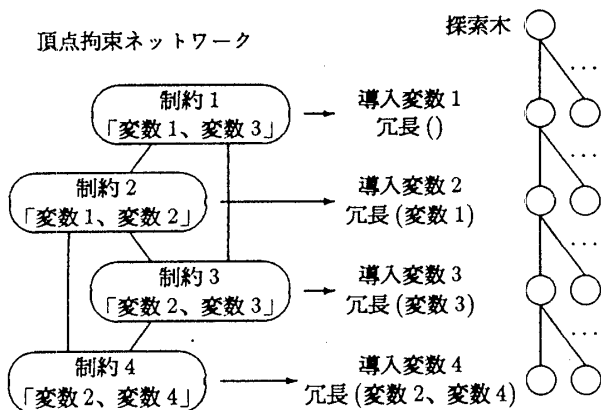


図 2: 併合法の木探索法への対応例

この対応付けによって、併合法では蓄積される途中解を利用した高速化をしなければ、この木探索処理は、併合処理は同じ処理時間で実行出来ている。

逆に、途中解の蓄積をすれば併合法における高速化手法を木探索処理に利用可能になる。

4.2 併合法における高速化手法の木探索法への適用

併合法における高速化手法としては、制約条件の構造を考慮しながら冗長な変数を縮退させる方法が提案されている [6]。冗長変数の縮退操作により、それ以降は冗長変数を併合処理で用いないので探索空間が削減される。縮退した変数との対応を復元する情報は途中解に付加して蓄積している。

対応付けられた木探索にこの高速化手法を導入するには、通常処理に以下の動作の追加が可能である。

- (1) 既に求められた解を蓄積する。
- (2) 冗長変数を持つ導入変数の値が新しく求められる毎に、冗長変数を縮退して出来る途中解が既に求められているか検査する。
- (3) 求められているいれば、それ以降の答は既に求められていることが解り、木探索を中断する。解は、既に求められていた途中解の先の答と現在の途中解とを合成し生成する。

5 まとめ

本報告では、知識処理の一推論方法であり上限保証のある制約推論法において、制約充足手法の併合法が木探索法に変換可能なことを用い、併合法で提案されている高速化手法を木探索法に応用する方法を述べた。

参考文献

- [1] JIS X 0010-1987 情報処理用語(操作技法及び機能)
- [2] 遠城: “実時間知識処理をめざした制約推論のレスポンスタイム推定法”, 第44回情処全大, 2Q-4 (1992)
- [3] 遠城: “実時間知識処理をめざした制約推論における木探索制約充足手法のレスポンスタイム推定法”, 第47回情処全大, 1P-2 (1993)
- [4] 窪田、内野、李、山下、西原: “併合法による制約充足の並列化効果について”, 第46回情処全大, 7A-1 (1993)
- [5] 遠城: “実時間知識処理をめざした制約推論のための制約充足手法比較”, 第46回情処全大, 7A-2 (1993)
- [6] 塩澤、西原、池田: “拘束条件の構造を考慮した整合ラベリング問題の解法”, 情報処理学会論文誌, Vol.27, No.10, pp927-935 (1986)