

6F-8 競合解消戦略に基づく照合アルゴリズム(CREST)

坂口 聖治 小池 博 富田 浩
 ((株) 日立製作所 システム開発研究所)

1. はじめに

高速推論アルゴリズムとしてデータ駆動型照合方式があり、著名なものに `rete`¹⁾ や `TREAT`²⁾ がある。また、これらの改良型アルゴリズム^{3, 4, 5)} も数多く提案されている。しかし、これらの提案アルゴリズムでは、再照合処理の高速化に重点をおいているため、ルールが成立する状態数が多く、しかも、競合解消戦略によって選択されたルールの実行によって前回の状態に著しい変化が生じる割当て問題や、スケジューリング問題などに対する効果は薄かった。これらの問題の解決案としてルールを複数の群に分割し、推論対象ルール群がアクティブ状態へ遷移したとき中間成立状態の整合をとるアルゴリズム⁶⁾ が提案されている。しかし、このアルゴリズムも同一ルール群内の競合集合が多ければ、意味をなさない。ここで提案するアルゴリズム(CREST)は、競合解消戦略の性質を用いて、専用の照合ネットワークの生成と競合集合の部分絞込みをおこなうことで、実際に実行される照合処理と算出される競合集合の量を軽減し、この問題に対処するものである。

2. 競合解消戦略とその性質

競合解消戦略は、競合集合そのものを絞り込むものとルールを特定することによって競合集合を絞り込むものがある。競合解消戦略は、これらを組合せて指定される。本稿では、次の戦略を前提として説明する。

- ・順序付最新性 (OR: Ordered Recency strategy)
 ルール条件をみたまデータのタイムタグを条件出現順に比較し、新しいものを優先する戦略
- ・最新性 (R: Recency strategy)
 ルール条件をみたまデータのタイムタグを比較し、新しいものを優先する戦略
- ・詳細性 (SC: Special Case strategy)
 ルール条件の詳細さ(条件数)を比較し、チェック項目の多いルールを優先する戦略
- ・確実性 (C: Certainty strategy)
 ルールに付加された確実度を比較し、より確実度

の高いルールを優先する戦略

- ・ルール記述順 (RO: Rule Ordered strategy)
 ルールが記述された順番を比較し、先に記述されたルールを優先する戦略

これらの戦略のOR, R戦略は競合集合そのものを絞り込むタイプの戦略であり、SC, C, RO戦略はルールを特定することによって競合集合を絞り込むタイプの戦略である。以後、本稿では、前者をOR系戦略、後者をRO系戦略と呼ぶ。

3. 競合解消戦略の最適化

競合解消戦略は、その組合せ方によって競合状態が解消されない可能性が残る。そこで、必ずOR+RO戦略を追加することによって、この問題を回避し、これによって発生する冗長戦略を次のように抹消する。

- ①同一戦略が複数指定された場合、2番目以降の該戦略を抹消する。
- ②OR戦略とRO戦略がともに指定された場合、以降の競合解消戦略を全て抹消する。
- ③RO系戦略が複数指定されている場合、RO戦略以降の該戦略の全てを抹消する。

以後、本稿では特に断らないかぎり、上述の冗長戦略抹消処理後の競合解消戦略を単に競合解消戦略と呼ぶ。

4. 競合解消戦略の性質に着目した処理の軽減

競合解消戦略のRO系戦略は、ルール内の競合状態に影響を与えず、OR系戦略は、ルール間の競合状態に直接影響を与えない。そこで、①RO系戦略によってルール間の優劣を求めておき、②OR系戦略によってルール内の競合戦略をもとめておけば、競合集合の各要素の優劣を決定する処理を軽減することができ、競合解消処理の効率を向上させることができる。

ルール間の競合解消戦略的な優劣は、RO系戦略によって、あらかじめ該当ルールをソートしてチェーン

構造しておくことができる。ソーティング戦略は、指定された競合解消戦略からSC, C, RO戦略のものを抽出することで得られる。ただし、RO系戦略の間にOR系戦略の列が存在する場合、これらが戦略列を分断するため、複数キーのソーティングとなる。つまり、競合解消処理は、このルールチェーンを順番にたどり、各ルールのターミナルノードに蓄積されているルールの代表インスタンスエーションを比較することでルール群内競合解消処理をおこなう、その後、アクティブルール群間の競合解消をおこなえば、求める競合解消戦略の結果が得られる。

5. 競合解消戦略に基づくルール条件照合処理

前述した競合解消戦略のうち、R戦略について説明する。OR+RO戦略がデフォルトで追加されるため、ルール内競合解消戦略は、R+OR戦略である。CRESTの条件照合処理は、チェックデータ派生処理と条件チェック処理によって競合解消戦略後にセレクトされるインスタンスエーションを含む部分競合集合を求めることで処理量の軽減を実現している。

いま、派生元データ組が(1, m, n)となっているとすると、これより派生される1次派生データ組リストは、+1のハミング距離を持つものであり、

(1₁, m, n) (1, m₁, n) (1, m, n₁) の3つであり、それぞれのリストは、派生元リストの(1, m, n)よりCR順位が低いのは自明である。2次以降の派生リストは、以前のリストにハミング変化をおこしたノード

(白抜き文字部分)を含めて右側のみを対象としてリストを変化させる。同時に存在する候補データ組リスト中にRの競合

	スパン1	スパン0
0	a	d i j
1		f h
2	b c	e g k

図1 αノード状況の例

解消順位が等しいデータ組が存在する場合は、OR順位の高いデータ組が、チェック候補データ組リストである。図1のようにαノードがスパン0→スパン1のように更新された場合、派生リストは、図2のように

派生リスト導出 (スパン0)

- 0: [(a, f, e)]
- 1: [(a, b, e)(d, f, g)(i, f, e)]
- 2: [(d, f, g)(d, h, g)(i, f, e)]
- 3: [(d, f, g)(d, h, g)(i, f, e)]
- 4: [(d, h, g)(i, f, e)] (d, f, k)で条件が成立

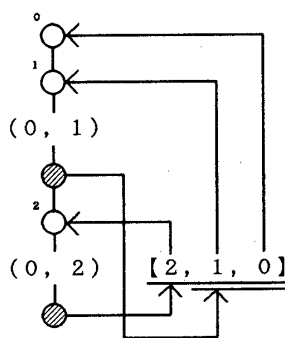
派生リスト導出 (スパン1)

- 0: [(a, f, b)(d, h, g)(i, f, e)]
- 1: [(a, b, e)(a, f, g)(d, f, b)(d, h, g)(i, f, e)]
- 2: [(a, f, e)(a, h, g)(d, f, b)(d, h, g)(i, f, e)]
- 3: [(a, h, e)(a, f, g)(d, f, b)(d, h, g)(i, f, e)]

図2 派生リストの状況

なる。ここでは、スパン0において(d, f, k)で条件が成立し、その組合せによってa, b, cのデータが流入してきたものと仮定している。

条件チェック処理は、β照合条件を図3のような有向グラフ(RICG: Rule Instansiation Check Graph)にしたがって処理をおこなう。



したがって処理をおこなう。前述のチェックデータ派生処理によって派生されたデータ組をRICGの0流入点から流す。図中の()内は、データ間の条件をあらわし、条件が不成立の場合【】内に記述された位置のデータを選移させる。もし該位置のデータが選移できなければ、順に右へ移動し選移データを選択動作を繰り返し、このRICG

図3 RICGの例

をスルーしたデータ組が求めるルールインスタンスエーションである。

6. まとめ

割当て問題や、スケジューリング問題などは、ルールが成立する状態数が多く、しかも、計算しておいた過去の中間状態が無駄になることが多い。CRESTアルゴリズムは、求められた競合集合から効率よく競合解消処理をおこなう方式に加えて、この無駄となる計算を最小に抑える照合処理を実現するものであり、従来不向きとされていた問題に対して有効なアルゴリズムである。

[参考文献]

- [1] Forgy, C.L: A Fast Algorithm for the many Pattern/Many Object Pattern Match Problem: Artif. intell, Vol. 19, pp.17-37(1982).
- [2] Miranker, D.P: TREAT: A better match algorithm for AI production system: National Conference of Artificial Intelligence(1987).
- [3] 田野 他: 知識ベース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(1987).
- [4] 荒屋 他: プロダクションシステムにおいえる効率的パター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(1988)
- [5] 稲葉 他: TREATアルゴリズムにおけるコードサイズ最適化: 第39回 情報処理全国大会予稿集, 7D-4(1989).
- [6] 斎 他: スケジューリング高速化のための知識分割処理方式: 第39回 情報処理全国大会, 2B-1(1989).