5G－10

掃き出し補数法の簡略化による コストに基づく仮説推論

二田 文之 石塚 満
東京大学工学部電子情報工学科

1 はじめに

仮説推論とは真偽の不明な事柄（仮説）について取り敢えず真であると考えて推論を進め，ゴールが証明できればその仮説が正しかったものと考え，ゴールを導くことができない場合はその仮説は誤りであり別の仮説を考える，という形式の推論方式である．このような推論により，知識ベースに不完全な知識を含めることができるため知識ベースの能力や汎用性を高めることができ，診断，設計といった実用的な問題にも応用する事が可能である．

しかし仮説推論では，知識ベースの仮説間の矛盾チェックについて問題規模に対し最悪で指数オーダの時間がかかるため，推論速度の低下が問題となる．推論時間の短縮法としてはこれまでに冗長計算の回避による効率化，近似解法による計算時間の短縮等の方法が提案されてきた[1]．これらのうち掃き出し補数法を利用した仮説推論システム，知識構造に基づいて掃き出し操作を効率的に行うNBP法[3]は，準最適解を高速に求めることのできる手法として非常に有効である．本稿でも掃き出し補数法をベースにした準最適解探索を考察する．

なお準最適解というのは，ゴールを無矛盾で導く要素仮説の重みの和が必ずしも（最小）最適ではないということである．また知識ベースは，真であることがわかっている背景知識と真偽が不明で互いに矛盾の可能性のある知識からなり，背景知識はホーン節集合で与えられる。これは以降の議論で共通のものとする．

2 0-1整数計画問題への置き換え

0-1整数計画法の利用では真，偽をそれぞれ1,0に対応させandｰ,orを以下の例のように置き換えることで問題を解く．文献[5]では重みつきのアブダクション問題で以下の変換を行うことで，ランダムに生成した問題の97%を単体法のみで解けることを示している．

1. $p \leftarrow q_1 \vee \cdots \vee q_n$
2. $p \leftarrow q_1 \wedge \cdots \wedge q_n$

$$\Downarrow$$

1. $p \leq q_1 + \cdots + q_n$
   $p \geq q_i \qquad (i = 1, \cdots, n)$
2. $p \geq q_1 + \cdots + q_n - (n-1)$
   $p \leq q_i \qquad (i = 1, \cdots, n)$

また文献[6]では上の変換のうち1の2番目の式,2の最初の式がそれぞれ不要であることを示している．これはゴールの証明を目的とする時には，これらの式が必ずしも成り立っていなくても良いことを意味する．

我々の扱う仮説推論では上の変換の他に，仮説間の矛盾を表す式の変換を与えなければならない．この変換は以下のように行えば良い．

- $inc \leftarrow q_1 \wedge \cdots \wedge q_n$

$$\Downarrow$$

- $q_1 + \cdots + q_n \leq n - 1$

3 掃き出し補数法

0-1整数計画問題の計算複雑度はNP困難であり，最適解の探索時間は変数の数に対し指数的に増大してしまう．これを解決するため，掃き出し補数法[4]ではまず単体法で実数最適解を得て，その周辺で探索を行う．これにより，単体法で高速に解が得られる可能性をもちつつ，実数最適解周辺の探索で準最適解を得ることができる．

掃き出し補数法のアルゴリズムは，かなり長くなるためここでは割愛する．文献[2]に詳しい記述があるので，そちらを参照して頂きたい．

4 本手法のアルゴリズム

掃き出し補数法は単体法を元にしており，構造変数を非基底状態に，スラック変数を基底状態にしようとする．単体法を理解していなければ動作がわかりづらく，また基底と非基底状態を交換する相手を探さなければならない．

Cost-based Hypothetical Reasoning on
a Simplification of Pivot-and-Complement Method
Tomoyuki Futada, Mitsuru Ishizuka
Dept. of Info. & Commun. Eng., Univ. of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, 113, JAPAN

本稿では掃き出し補数法でスラック変数を使用しない手法を考察する. これによりアルゴリズムが理解しやすくなり, 更に構造変数とスラック変数の交換の相手を探す必要がなくなり各基底変数の値の更新もなくなるため高速化がはかれることが期待される.

我々の使用したアルゴリズムを以下に示す.

1. 初期フェーズ.
   整数条件を [0,1] 間の実数に緩和して, 単体法を用いて目的関数値を最小にする最適実数解を求める. もし, この最適解が整数解になっていれば, 終了.

2. 探索フェーズ.
   制約条件を満たしたまま実数値をとっている変数で値を 0 あるいは 1 にできるものを探す. もしあれば目的関数に関して最良 (最小) になるものについて実行する. なければ 4 へ.

3. 現在得ている実行可能解を丸めによって実行可能 0-1 整数解が得られる場合, これを得て改善フェーズへ. 得られなければ 2 へ.

4. 実数値をとっている変数を 0-1 にしたときに, 制約条件に違反する度合を示す実行不可能度指数が最小になる変数の値を交換する. ここで実行不可能度指数とは, 各変数が実行可能領域をはみ出した量の和である.

5. 各変数について以下の操作を行う.

   - 値が 0-1 であれば, 値を反転させ (0 ならば 1 に,1 ならば 0 にする), 実行不能度指数を計算する.
   - その他の値であれば値を 0,1 にしそれぞれについて実行不能度指数を計算する.

   以上の計算結果で実行不能度指数が最も改善されるものについて値を交換する. 値が 0,1 以外の場合には実行不能度指数が変化しなくても, 他に改善される変数がなければ交換を行う. 改善されるものがない場合 7 へ.

6. 現在の解が実行可能でないならば 5 へ. 実行可能なら 3 へ.

7. 2 つの 0-1 変数の補数をとることによって実行不可能度指数が改善されるならば最初に見出した補数化操作を行い 6 へ. 改善されるものがなければ, 探索に失敗したとして終了.

8. 改善フェーズ.
   実数最適コストを Z_{LP}, 現在の実行可能 0-1 整数解による目的関数値を $Z_{current}$ とし, もし $Z_{current} - Z_{LP} < 1$ の場合は現在の解が最適解であるので終了.

9. 1 つの 0-1 構造変数の補数をとることで, 実行可能性を保ったまま目的関数値を改善できる場合, 最も改善できる変数について実行し再び改善フェーズへ. このような変数が見出せなかった場合, 解の改善が行えないとして終了.

なお知識構造から制約不等式への変換は単体法では 2 節の変換を使っているが, それ以降では 2 節の 1 の最初の式,2 の 2 番目の式のみを用い, 更に矛盾知識については右辺を n-1 以下から n 未満としている. これによって変数の値の動ける領域が広がり, 探索が失敗することが大幅に減少している.

5 結果と評価

ランダムに生成した 26 問題を解いた結果, 掃き出し補数法を改善した NBP と同等の速度, コストの解を得ることができた.

6 まとめ

スラック変数を使用しない掃き出し補数法と, その有効性について論じた. 現在, 大規模な問題での評価, 計算時間の理論的考察, 知識構造を利用した改善フェーズの考察等を行っている.

参 考 文 献

1) 石塚: 仮説推論の計算量と高速化メカニズム, 人工知能学会誌, Vol.9, No.3, pp.342-349 (1994).
2) 石塚: 知識の表現と高速推論, 丸善 (1996).
3) 大澤, 石塚: 仮説推論における準最適解を多項式時間で計算するネットワーク化バブル伝搬法, 電子情報通信学会論文誌 D-2 Vol.j77-D-2, No.9, pp.1817-1829 (1994).
4) E.Balas and C.Martin: Pivot and Complement – A Heuristic for 0-1 Programming, Management Science, Vol.26, pp.86-96 (1980).
5) Santos Jr, E.: A linear constraint satisfaction to cost-based abduction, Artificial Intelligence Vol.65, pp.1-27 (1994).
6) Santos Jr, E., Santos S. E.: Polynomial solvability of cost-based abduction, Artificial Intelligence 86, pp.157-170 (1996).