仮説推論問題における問題例の考察

松尾 豊　石塚 満
東京大学工学部
{matsuo, ishizuka}@miv.t.u-tokyo.ac.jp

Abstract

仮説推論の準最適解を高速に得るさまざまな手法が提案されているが、各手法のパフォーマンスは用いる問題例に依存する。手法を正確に評価するには、問題例に対する分析が必要である。本論文では、SAT における研究を参考に、与えられた仮説推論問題がだいたいいくつくらい解をもつか、という見積もりを行うことにより、その問題の難しさを示すひとつの指標を提案する。この指標を用いることで、仮説推論問題がどのくらい難しいかを解を得る前にあらかじめ知ることができる。ここでは、ランダムに生成した問題に対し実際に解が得られたかどうかを調べ、この指標の有効性を検証する。

1　はじめに

仮説推論（もしくはアブダクション）は、真か偽か不明な事柄をとりあえず真と考えて（仮説を立てて）推論を進め、矛盾なくゴール（観測された事実）が説明できれば、立てた仮説は正しかったと考える形式の推論である。基盤性と実用性の両面で有用な枠組みであるが、NP 完全または困難な問題であるため、推論速度が大きな問題となり、高速に仮説推論問題を解く手法がいくつか開発されている。推論パスネットワークによる推論法は、仮説推論の厳密解を得る手法としては究極的な速度を達成している。また、ネットワーク化バブル伝搬法 [7] や SL 法 [5] は、仮説推論の準最適解を多項式時間で得る手法であり、大規模な問題に対しても、高速に最適解に近い解を得ることができる。また、最近では、変数や制約をそれぞれひとつのプロセッサと考え、それらのインタラクションにより探索が進むという視点に立つ、並行プロセスによる高速仮説推論法 [6] が提案されている。この視点により、いくつかの仮説推論問題および SAT 問題のアルゴリズムを統一的に見渡すことができ、また、より高速に解を得る、より最適解に近い解を得るなどと柔軟にアルゴリズムを変更できるというメリットがある。

本稿では、この一連の高速推論法から一歩わき道にそれ、これらのアルゴリズムを実験的に評価する際に用いる問題例について考察する。

2　仮説推論問題の解析における問題点

仮説推論法の計算オーダを評価する際には、理論的な探索時間の考察や実験的なパフォーマンスにより評価を行うのが通例である。理論的な考察は、計算オーダの最悪ケースを求めるものであり、実際にアルゴリズムを使用する際に問題となる平均ケースのパフォーマンスとは関係がないため、ある問題例に対して実際にアルゴリズムを走らせる実験的な評価を用いる場合が多い [9]。

その際に用いる問題例としては、応用例（設計、診断問題）から得られた問題を利用する場合や、ランダムに生成した問題により評価する場合がある。しかし、仮説推論において「問題をランダムに生成する」というのは非常に曖昧な表現である。解法の提案者に都合のよいように「ランダムに」問題を生成することは簡単である。したがって、用いる問題を適切に分析することが、より正確に仮説推論の解法を評価するために不可欠である。また、応用例から得られた問題を利用する場合においても、その問題がどのくらい難しい問題なのかによって、アルゴリズムの評価も変わってくる。

仮説推論と関係が深い SAT(satisfiability) 問題に対しては、ベンチマーク [1] が整備されており、ランダムに生成する問題という定義も明瞭である。仮説推論問題は SAT 問題として表すこともできるが、各節がホーン節である、仮説からのボトムアップによりゴールを真としなければならない、などの特殊な性質がある。そのため、解法もこの性質をうまく利用することが必要であるし、仮説推論問題を分析する際にも SAT 問題とは異なる方法を用いる必要がある。

3　SAT と仮説推論における問題例の生成法

3.1　3-SAT におけるランダムな問題例の生成

ランダムに 3-SAT 問題を作るには以下のような手順を踏む。

1. 変数の数 N、および節の数 L を与える。
2. N 個の変数の中から重複しないようにランダムに 3 つ変数を選ぶ。それぞれの変数を確率 0.5 で反転（真偽を逆転）させる。
3. 得られた節が今までに得られたものと同じなら 2 へ戻る。
4. 得られた節の総数が L に達していれば終了。そうでなければ、2 へ。

このように、3-SAT 問題をランダムに作るのは非常にシンプルであり、ランダム性というのがごく自然な形で導入されている。3-SAT の場合には、パラメータとして N と L だけ与えれば問題を生成することができる。

ここで L/N の値が問題の性質を知る重要な指標となる。実験的には L/N の値が 4.3 より大きくなると、急激に問題が難しくなる。これが、phase transition と呼ばれる現象であり、以前からこの閾値が存在することが報告されていた。最近では、フラクタルの理論とも結び付けられ、統計的な側面からの説明が進んでいる [3]。

3.2　仮説推論におけるランダムな問題例の生成

一方、ランダムに仮説推論の問題例を作るには、次のような問題点がある。

- 与えるべき条件が多い。（変数の数（そのうち仮説の数）、節の数（そのうち inc ルールの数）、各ルールにおけるボディ部の変数の数の分布、ゴールとする変数など）
- 節を作る際、ランダムに変数を選ぶと、多くの場合、ゴールと関連する仮説の数は非常に小さくなる。

そこで、例えば、ヘッド部の変数と番号の近い変数を選ばれやすくするなどの工夫をすると、意図的に大きな問題を作ることができる。

しかし、このような工夫をした上で問題を作成し解法を評価することが、どれくらい妥当であるのか、はっきりとしない。問題の難しさを表す何らかの指標が必要である。

4　新たな指標：仮説推論の難しさの指標

4.1　3-SAT の場合

3-SAT の phase transition 現象に対する初期の理論的研究には、次のようなものがある。

N 個の変数があるとき、その解空間は 2^N（個）である。一方、例えば、“$a \vee \bar{b} \vee c$.” という節を加えたとき、解空間全体のうち、{ a=false, b=true, c=false } という可能性が取り除かれる。a,b,c だけに着目すると、$2^3 = 8$ つの候補のうち 1 つだけ実行不可能となるので、解空間全体は 7/8 になると見なすことができる。したがって、節が L 個あれば、解空間全体は $(7/8)^L$ 倍となる。もちろん、ある節が実行不可能にする解空間がすでに実行不可能である場合も当然あるが、ここでは非常に大雑把にいって、だいたい 7/8 になるということである。

したがって、実行可能解の個数は、$2^N\ (7/8)^L$ であると概算することができる。この値が 1 を越えれば、実行可能解が存在する可能性が高い。

$$2^N \left(\frac{7}{8}\right)^L > 1 \quad \text{よって} \quad L/N > 5.19$$

この 5.19 という数字が、非常に大雑把ではあるが、phase transition 現象の起こる L/N の理論的な予測値である [2]。もちろん、理論的な予測値に関する研究もその後進んでおり、例えば、Kamath は $L/N > 4.758$ なら高い確率で解を持たないことを、Frieze-Suen はこの値の下限が少なくとも 3.003 であることを示している [4]。

4.2　仮説推論問題の場合

以上の議論をそのまま仮説推論問題に当てはめることはできない。なぜなら、ひとつの節に含まれる変数の数が 3 と決まっていないこと、生成可能な要素仮説が限られていることなどのためである。しかし、前節で述べた

解の数の概算の考えは参考になる。ここでは、この考え方を仮説推論に適用してみる。

仮説推論問題の解空間は、仮説の数を H とすると 2^H である。次に、inc ルールによりこの解空間がどのくらい削られるか考える。

$$inc \leftarrow h1, h2.$$

というルールを加えた場合、$h1 = true, h2 = true$ の場合にだけ実行不可能となるので、解空間は $3/4$ になる。一般的にボディ部の変数の数を b とすると、inc ルールによって解空間は、$(2^b - 1)/2^b$ 倍になる。

さらに、一般のルール（正のリテラルを 1 つもつ節）の場合には、少し複雑である。

$$a \leftarrow b, c.$$

というルールについて考える。このルールがなかった場合、a, b, c が全く独立に真偽の値を取ると仮定すると、a が真となるのは表 1 に示すように 8 通りの組み合わせ中、4 通りである。

一方、このルールが付け加わった時を考えると、表 2 のようになる。したがって、上位ノード a が真を取る組み合わせの数は 5/4 になる。上位ノードが真となる組み合わせが p 倍になると、単純に考えてゴールが真となる組み合わせも p 倍になると考えられる。

表 1: a, b, c が全く独立に真偽をとる場合

a	F	F	F	F	T	T	T	T
b	F	F	T	T	F	F	T	T
c	F	T	F	T	F	T	F	T

表 2: $a \leftarrow b, c.$ を考慮する場合

a	F	F	F	**T**	T	T	T	T
b	F	F	T	T	F	F	T	T
c	F	T	F	T	F	T	F	T

これを一般化して、各ルールのボディ部の数を b とすると、解の個数は $(2^b + 1)/2^b$ 倍になると考えられる。

以上をまとめると、解の個数は

$$K = 2^H \times \prod_{i \in NormalRule} \frac{2^{b_i} + 1}{2^{b_i}} \times \prod_{j \in IncRule} \frac{2^{b_j} - 1}{2^{b_j}}$$

であると概算することができる。ここで、乗除を加減に置き換えるるために $K' = \log_2 K$ とする。**K′ は大きいほど実行可能解を持つ可能性が高い、すなわち易しい問題であり、小さいほど実行可能解を持つ可能性が低い、すなわち難しい問題である**と考えられる。

K' を求めるには次のような手順で求める。

- K' := 要素仮説の数とする。
- and ルールに対して、ボディの数を b とすると、$K' := K' + log(2^b + 1) - log(2^b)$ とする。すべての and ルールについて繰り返す。
- inc ルールに対して、ボディの数を b とすると、$K' := K' + log(2^b - 1) - log(2^b)$ とする。すべての inc ルールについて繰り返す。

この計算は、ルール数に対して線形なオーダで計算できるので、計算時間は問題にならない。

ひとつのルールによる K' の増減は表 3 のようになる。つまり、ボディ部の変数の数が大きいルールは問題の難しさにほとんど寄与しない。

表 3: ひとつのルールによる K' への影響

ボディ部の数	and	inc
1	0.5850	-1
2	0.3219	-0.4150
3	0.1699	-0.1926
4	0.0875	-0.0931
5	0.0444	-0.0458
6	0.0224	-0.0227
7	0.0112	-0.0113
8	0.0056	-0.056
9	0.0028	-0.028
10	0.0014	-0.0014

5 実験

前節までの議論はあくまでも理論的なものであるから、その指標 K' が実際に問題の難しさを表していることを確かめる必要がある。そこで、文献 [6] と同様の方法で生成した問題 805 問について、仮説推論法（並行プロセスによる高速仮説推論法）により解が得られたかどうかを K' の値によって分析したものが表 4 である[1]

[1] もちろん、全解探索を行っているわけではないので、実行可能解があっても発見できない場合もあるが、今のところ、解が得られない＝実行可能解がないとしている。

K' の値が 56 以下では、約 2/3 の問題で解が得られなかったが、それを越えると急に解が得られる確率が高くなっており、230 以上ですべての問題で解が得られている。したがって、この問題群に関しては、K' が 60 以上か以下かで、問題が解ける確率が高いかどうか判断することができる。

6　今後の課題

仮説推論の解法を正確に評価するには、問題例の検討が不可欠である。特定の問題に対する理論的な考察は [8] などでされているものの、より大雑把に確率的に問題の性質を知ろうという研究は、海外で特に Gomes, Selman, Hoos らのグループを中心に活発に議論が行われており、本研究も仮説推論という立場から何らかの貢献ができるものと考えている。

ここに述べた仮説推論問題の分析法は、研究の初期段階のものであり、今後はより詳細な検討を進める予定である。例えば、問題の規模が大きくなったときに、ここでの計算どおりの結果が出ないことも分かっており、統計的なモデルを考慮に入れるなど、改良を加えていきたいと考えている。

表 4: ランダムに生成した問題例の分析 (節の数が 100 以上)

K'	Unsolved の数	割合
⇑ *hard*		
41.0	8/12	0.667
56.0	16/25	0.640
72.9	8/37	0.216
88.9	14/66	0.212
104.8	12/77	0.156
120.8	10/74	0.135
136.8	9/59	0.153
152.8	9/81	0.111
168.7	9/87	0.103
184.7	6/61	0.098
200.7	2/57	0.035
216.6	1/33	0.030
232.6	0/42	0.
248.6	0/30	0
264.6	0/31	0
280.5	0/8	0
296.5	0/8	0
312.5	0/10	0
328.4	0/5	0
344.4	0/1	0
360.4	0/1	0
⇓ *easy*		
	計 104/805	0.129

参考文献

[1] http://dimacs.rutgers.edu/.

[2] John Franco and Marvin Paull. Probabilistic analysis of the davis putnam procedure for solving the satisfiability problem. *Discrete Aplied Mathematics*, Vol. 5, pp. 77–87, 1983.

[3] C. Gomes, B. Selman, and N. Crato. Heavy-tailed phenomena in satisfiability and constraint satisfaction problems. *J. of Automated Reasoning*, Vol. 24, pp. 67–100, 2000.

[4] Anil Kamath, Rajeev Motwani, Krishna Palem, and Paul Spirakis. Tail bounds for occupancy and the satisfiability threshold conjecture. *FOCS'94*, pp. 502–511, 1994.

[5] 松尾, 二田, 石塚. SL 法：線形・非線形計画法の併用によるコストに基づく仮説推論の準最適解計算. 人工知能学会誌, Vol. 13, No. 6, pp. 953–961, 1998.

[6] 松尾, 石塚. 並行プロセスによる高速仮説推論法. 第 120 回 知能と複雑系研究会, 2000.

[7] 大澤, 石塚. 改良型ネットワーク化バブル伝播法による低次多項式時間仮説推論法. 人工知能学会誌, Vol. 10, No. 1, pp. 123–130, 1995.

[8] 大澤, 石塚. コストに基づく仮説推論を多項式時間で達成する新しい十分条件. 人工知能学会誌, Vol. 13, No. 3, pp. 415–423, 1998.

[9] B. Selman, H. Kautz, and D. McAllester. Ten challenges in propositional reasoning and search. *Proc. IJCAI-97*, pp. 50–54, 1997.