

4D-4

ボルツマンマシンによる論理型言語の推論実行*

川原 英哉 村越 英樹 舟久保 登†

東京都立科学技術大学‡

1 はじめに

論理型言語は多くの分野で注目されている。論理型言語におけるプログラミングでは、その問題に内在する事実と関係に注目し、それを記述するだけでよく、問題を解くアルゴリズムを与える必要のない点が大きな特徴である。しかし、そのプログラムの推論実行は、通常、「反駁手続き」と呼ばれる繰り返し手続きによって実現されているため、膨大な処理時間がかかる。よって多くの場合、人間がある程度アルゴリズムを与え、効率化を図らなければならない。これゆえ、並列化により推論処理を高速化するための研究が多数ある。しかしながら、これらのアプローチは、依然、「反駁手続き」を基盤としたもので、結果、そのシステムは複雑で柔軟性に欠けるものとなってしまう。

そこで本報告では、論理型言語の推論実行を組合わせ最適化問題に帰着し、ボルツマンマシンの最適化機械としての能力を利用して処理するという、新しいアプローチの推論実行方法を提案する。またこれを実際に簡単な問題に対して適用し、その正当性を確認した。

2 ボルツマンマシンと最適化

ボルツマンマシンとは、対称相互結合型ニューラル・ネットワークの一種であり、「ネットワークのエネルギー」を最小化するようにその状態を変化する(ようにできる)という特徴がある。そしてこの特性を利用して、ある種の組合わせ最適化問題を解くことができる。

ネットワークのエネルギーとは、Hopfieldらによって導入された概念で[1][2]、以下のように定義される。 $(x_k(t))$: 時刻 t における k 番目のユニットの状態, v_k : そのしきい値

$$E(t) \equiv - \sum_{i,j} w_{ij} x_i(t) x_j(t) - \sum_i v_i x_i(t) \quad (1)$$

ボルツマンマシン[3]の各ユニットの状態遷移は、非同期、確率的に行なわれる。つまり、ある(k 番目の)ユニットについて、それが0のときと1のときのエネルギーの差を ΔE_k とするとき、次式で表される確率 $P_{x_k=1}$ で x_k は1となる。 T はネットワークの温度と呼ばれる。

$$P_{x_k=1} = \frac{1}{1 + e^{-\frac{\Delta E_k}{T}}} \quad (2)$$

そして、エネルギーをうまく最小化するために、「焼きなまし(Simulated Annealing)」と呼ばれる手法(はじめは T を大きくして、次第に小さくする)が利用される[4]。

*Inference Mechanism for Logic Programming Language with Boltzmann Machines

†Hideya KAWAHARA(E-mail: kawahara@fml.ec.tmit.ac.jp), Hideki MURAKOSHI and Noburu FUNAKUBO

‡Tokyo Metropolitan Institute of Technology

最適化問題をボルツマンマシンで解くには、その評価関数を2次形式で表現し、エネルギー関数と係数比較して、結合係数、しきい値を定め、ネットワークを構成すればよいわけである。

3 推論実行とそのエネルギー関数

論理型言語の推論実行をボルツマンマシンに行なわせるためには、そのエネルギー関数を、論理型言語によって記述されたプログラムから決定しなくてはならない。しかしその際、「変項」の扱いが問題となる。そこで以下のような手順(変項をグローバルな扱いとし、その「個体」との対応を表すユニットを導入する)でプログラムを展開し、エネルギー関数をもとめる(図1)。

- (1) 「事実」は、変項を介在させたかたち書き換えておく。
- (2) 各「ルール」の変項を、全てのルールにグローバルな変項と対応付ける(可能な組合せ全て)ことによって、プログラムを展開する。
- (3) ルールの頭部(“: -”より左)が同じものは選言でまとめる。
- (4) “: -”を同値(“ \leftrightarrow ”)と読み換える。これは、「与えられた知識で証明出来ないものは偽とする」ことを意味する。
- (5) 以上の結果得られたルール群の、

「述語」(「グローバルな変項」, ...)
「グローバルな変項」=「個体」

それぞれにユニットを対応付け、ネットワーク全体のエネルギー関数を求める。

- (6) さらに「1つの変項にはただか1つの個体しか割り当てられない」ことを保証するために、(5)で求めたエネルギー関数に以下の項を付加する。

$$\sum_i \left\{ \sum_j \sum_{k>j} x_{G_i=c_j} x_{G_i=c_k} \right\} \quad (3)$$

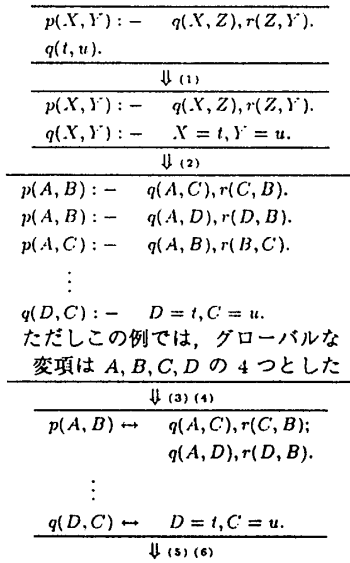
ここで、 $x_{G_i=c_j}$ はグローバルな変項 G_i に個体 c_j が割り当てられていることを表すユニットである。

この際必要となる、論理演算を行なわせるためのエネルギー関数である E_{and} 等は、簡単に求められ、それらは以下のようになる。

$$E_{and}(x_a, x_b, x_p) = x_a x_b - 2x_a x_p - 2x_b x_p + 3x_p$$

$$E_{or}(x_a, x_b, x_q) = x_a x_b - 2x_a x_q - 2x_b x_q + x_a + x_b + x_q$$

$$E_{inv}(x_a, x_r) = 2x_a x_r - x_a - x_r + 1 \quad (4)$$



$$\begin{aligned}
 E = & E_{and}(x_{q(A,C)}, x_{r(C,B)}, x_{s0}) \\
 & + E_{and}(x_{q(A,D)}, x_{r(D,B)}, x_{s1}) \\
 & + E_{or}(x_{s0}, x_{s1}, x_{p(A,B)}) \\
 & + \dots \\
 & + x_{D=s} x_{D=t} + x_{D=t} x_{D=u} + x_{D=u} x_{D=s}
 \end{aligned}$$

ここで、 $x_{q(A,C)}$ は命題 $q(A,C)$ に
 割り当てられたユニットを、 x_{s0} 等は展
 開の途中で必要となるユニットを表わす

図 1: プログラムの展開手順

これらのエネルギー関数は、それぞれ、 x_a, x_b, x_p が論理積、 x_a, x_b, x_q が論理和、 x_a, x_p が否定の関係にあるときにおいてのみ、最小値をとるものである。

このようにして求められたエネルギー関数は、プログラムで示された論理関係が満たされたときのみ最小値をとる。よってボルツマンマシンの最適化機械としての機能を利用し、推論実行(無矛盾な関係を導く)をすることができる。

4 簡単な問題への適用

以上の方法を基礎に、ボルツマンマシンにより推論実行をする論理型言語処理系のプロトタイプを構築し、実際に簡単な問題に対して適用し、その正当性を確認した。

図 2 は、その処理系の概略である。ネットワーク・コンパ

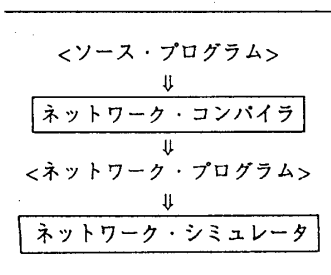


図 2: 処理系の概略

イラは、§3 の手順を機械的に行ない、論理型言語のソース・プログラムをネットワークのユニット間の結合等の情報(ネットワーク・プログラム)に変換するものである。またネットワーク・シミュレータは、ネットワーク・プログラムを読み込み、ボルツマンマシンをシミュレートするものである。

「問い合わせ(“?- ”)」は、その質問に対応するユニットのどれかが必ず 1 となるように、エネルギー関数に項を追加することにより行ない(「質問」が真であるという仮定)、その真偽判定は、ネットワーク・シミュレータの結果(エネルギーの値とユニットの最終状態)を読む(満たす場合、エネルギーは 0。その条件はユニットの状態によりわかる)ことにより行なう。

ここでは、この処理系で以下のようなプログラムを実行した結果を紹介する。

```

father( namihei, sazae ).
mother( fune, sazae ).
father( masuo, tarao ).
mother( sazae, tarao ).
grandmother( X, Y ) :-
    mother( X, Z ), mother( Z, Y );
    mother( X, Z ), father( Z, Y ).

?- grandmother( X, Y ).
    
```

グローバルな変項は 3 つとし(必要なユニットの数は 69 個)、また、「焼きなまし」の温度スケジュールは以下の式に従い、 $T_0 = 300.0$, $\tau = 5.0$ とした。

$$T(t) = \frac{T_0}{1 + \frac{t}{\tau}} \tag{5}$$

そして、打ち切り回数(その回数で収束していなかった場合は、はじめからやり直す)を 1000 回とした場合、以下のような出力が結果として得られ、収束回数の期待値は、約 600 回であった(100 回の試行の結果から求めた)。

```

grandmother(A,C) mother(A,B) mother(B,C)
A=fune B=sazae C=tarao
    
```

5 おわりに

本報告では、以上のような方法で、述語論理式で記述された論理型言語のプログラムをボルツマンマシン上に展開し、その挙動を用いて処理するという、新しいアプローチの論理型言語処理系を提案した。また、そのプロトタイプを構築し、簡単なサンプルで動作を確認した。

今後の課題としてはより大規模な問題についてシミュレーションを行なって、その有効性、問題点などを明らかにすることがあげられる。

参考文献

- [1] J.J.Hopfield, *Neural network and physical systems with emergent collective abilities*, Proc. Natl. Acad. Sci. USA 79,2554 (1982).
- [2] J.J.Hopfield and D.W.Tank, *Neural Computation of Decision in Optimization Problems*, Biol. Cybern., Vol 52, pp. 141-152 (1985).
- [3] G.E.Hinton, T.J.Sejnowski, D.H.Ackley, *Boltzmann Machines: Constraint Satisfaction Networks That Learn*, Tech. Rep. CMU-CS-84-119, Carnegie-Mellon University (1984).
- [4] S.Kirkpatrick et al., *Optimization by simulated annealing*, Science, vol. 220, no. 4598, pp. 671-680 (1983).