ICOT Technical Memorandom: TM-0950

TM-0950

エネルギー最小化法による記号的推論の制御

橋田 浩一

August, 1990

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191～5
Telex ICOT J32964

Institute for New Generation Computer Technology

エネルギー最小化法による記号的推論の制御*

橋田 浩一
(財) 新世代コンピュータ技術開発機構 (ICOT) 第3研究室
hasida@icot.or.jp

1 導言

認知過程における情報処理は、非常に多様な情報の伝播を含む。これをモデル化するには、情報伝播の方向を限定しない制約[1]の形で知識や信念を表現する必要がある。記号的制約を表現するには記号論理を用いればよいが、それだけでは情報処理の制御を欠き、また、文の解釈の優先度のような認知のアナログ的側面を記述できない。そこで、記号的制約の上にアナログ量の**ポテンシャル・エネルギー**を定義し、エネルギーの最小化原理に従って情報処理を分散的に制御すると同時に、解釈の優先度のような側面をも捉えることを考える。アナログ的情報による実行制御 [4] や曖昧性の解消 [3, 6] などは、これまで別個に論じられて来たが、それらはエネルギーの概念を用いた一般的な枠組の下に統一できる。

2 エネルギーと活性拡散

記号的な制約を 1 階述語論理の節形式のプログラムによって表現する。要素式[2]をノード、変数や述語の共有などによるそれらの間の参照関係をリンク[3]と考えることにより、この表現をネットワークと見なせる。たとえば、(1)[4]というプログラムは (2) のようなネットワークである。

```
(1) +p(X,Z) -q(X,Y) -r(Y,Z).
    +q(A,B) -A=a.
```

*本稿は、日本認知科学会第 7 回大会 (1990 年 7 月、於九州工大) において同題で発表した内容を拡張したものである。

[1]本稿で述べる制約のシステムは、Prolog のような宣言的意味論を持たない。重要なのは、宣言的意味論を持つことではなく、多様な情報伝播を許容することである。

[2]p(X) のような通常の要素式と X = f(Y) のような変数の束縛を総称してここでは要素式と言う。また、小文字で始まる名前は述語などの定項、大文字で始まる名前は変数とする。

[3]ただし、これらのリンクは 3 つ以上のノードを一度に結び付けることもあるので、一般には通常のリンクではなく、超リンク (hyper link) である。

[4]各リテラルは、その正負に応じて要素式の前に + または - を付けて表記する。たとえば、

$$\forall X, Y\{s(X,Y) \vee \neg p(X) \vee q(Y) \vee X \neq f(Y)\}$$

という節を、ここでは

```
+s(X,Y) -p(X) +q(Y) -X=f(Y).
```

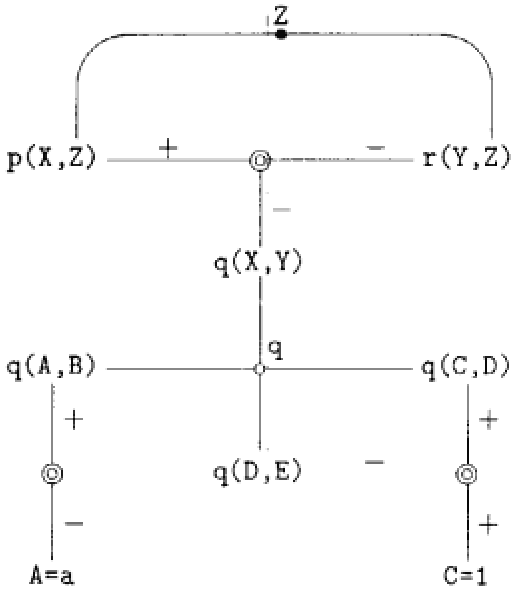
と書く。節の中のリテラルの順序には意味がなく、これは

```
-p(X) +s(X,Y) -X=f(Y) +q(Y).
```

などと書いても同一の節を表わす。

+q(C,D) +C=1 -q(D,E).

(2)



ここで、◎は節に相当する(超)リンクの束であり、∘は述語に相当するリンクの束であり、•は変数に相当するリンクの束である。節と要素式を結ぶリンクのラベルである正負の符号は、その節中におけるその要素式のリテラルとしての符号を表わす。簡単のため、(2)では、Z以外の変数に対応する構造は省略してある。

各要素式は活性度(出力)を持っている。活性度は(0,1)区間の実数値であり、その要素式の表現する要素命題が真である主観的確率とする。この活性度の関数として、制約のネットワーク上にエネルギーを定義する。すると、エネルギー最小化原理によって情報処理の制御を行なうことができる。情報処理には連続的な側面と離散的な側面とがある。前者は活性拡散(spreading activation)、後者は論理プログラムに関する記号計算である。

制約全体のエネルギー関数を、要素式、節、述語、および変数などの、制約の各部分のエネルギー関数の和と考える。以下本節では、制約の各部分の局所的な論理的意味を反映するようにこれらの部分的なエネルギー関数を定義し、それらによって起動される活性拡散について述べる。一般には、制約全体のエネルギーを E、要素式 α の活性度を x_α としたとき、α に対する活性化入力を $-\frac{\partial E}{\partial x_\alpha}$ とする。これは、コネクショニスト・モデルなどでも用いられている標準的な考え方である。

要素式 α のエネルギー関数を以下の式で定義する。

(3) $$-b_\alpha x_\alpha + T\left\{x_\alpha \log x_\alpha + \overline{x_\alpha}\log\overline{x_\alpha}\right\}$$

x_α は α の活性度、T は(簡単のため)全ての要素式に共通の正の定数とする。T を温度(temperature)[5]、b_α を α の偏向(bias)と言う[6]。b_α が大きいほど α が仮説し難い[7]。b_α は定数ではなく、後で(13)

[5] ここでは物理的次元を無視している。

[6] 偏向を含むこの第1項のエネルギーは、Hobbsの枠組における仮説コスト(assumability cost) [3] に対応する。ただし、このエネルギーは文脈(x_α の値など)によって変化する点が仮説コストと異なる。

[7] (3)の第1項により、α は外部入力 b_α を受けることになる。

に示すように、他のエネルギーに依存して決まる[8]。任意の値 a について、$\overline{a} = 1 - a$ とする。第2項によって x_α の値は $(0,1)$ 区間に制限される[9]。

述語 π のエネルギー関数を以下のように定める。

$$(4) \qquad -\sum_{\alpha \neq \beta} r_{\alpha\beta}(x_\alpha - \frac{1}{2})(x_\beta - \frac{1}{2}) + B_\pi \sum_\gamma x_\gamma \prod_\delta \overline{x_\delta}$$

第1項の x_α は述語 π を持つ要素式 α の活性度、$r_{\alpha\beta}$ は非負の定数である。第1項により、$r_\alpha\beta$ が大きい程 α と β は同一の真理値を持ち易い。第2項の γ は π を持つ要素式のうち負リテラルとして参照されているものであり、δ は、π が限定述語[10]ならば π の定義節の頭部の正リテラル、自由述語[11]ならば π を持つ正リテラルである。

(4) の第2項は、γ が真になるには δ のうちのいずれかが真でなければならないことを表わす。B_π は非負の実数であり、その値は π が限定述語ならば大きく、自由述語ならば小さい。この項は完備化または閉世界仮説 (CWA; closed world assumption) に相当する。即ち、定義節を持たない限定述語に関しては δ が存在しないから、$\prod_\delta \overline{x_\delta}$ が1となり、要素式 γ は、この項の影

[8]従って実は、ポテンシャル・エネルギーによって規定できない力の場ができてしまう。ポテンシャル・エネルギーというのは、さしあたって数学的扱いを簡単にするために考えただけであり、一般には適切な力の場があればよい。いずれにせよ、それでうまく行くかどうかは実験してみないとよくわからない。

[9]この項は、sigmoid 関数 $1/(1+\exp(-x/T))$ の逆関数を積分したものである。活性拡散が収束した状態では、$\frac{\partial E}{\partial x_\alpha} = 0$ であるから

$$x_\alpha = \frac{1}{1+\exp(-I_\alpha/T)}$$

が成立する。ここで I_α は、E の他の項に基づく α への入力の総和である。これからわかるように、T が大きい程 x_α は I_α の正負に従ってそれぞれ1と0に近い値を取りやすい。sigmoid 関数は上記以外のものでもよいだろう。

活性拡散のひとつの方法は、各時点で上記の式に従って各要素式の活性値を更新してゆくことである。以下に示すように E は x_α の2次以上の項を含まないから、I_α は x_α を含まない。従って、x_α の更新は、α 以外の要素式の活性値を固定した場合の E の最小値を与える。ゆえに、活性値を変化させるような更新は必ず E を減少させるから、様々な要素式に対する更新が同期的か非同期的かによらず、この方法による活性拡散は収束する (Zorn の補題)。しかも、E の値が極小でない限り、$\frac{\partial E}{\partial x_\alpha} \neq 0$ であるような α が存在するから、活性拡散が続く。従って、この活性拡散は E を極小値 (最小値とは限らない) に収束させる。最小値に至る可能性を高めるには、ボルツマン・マシン (Boltzmann machine) やガウシアン・マシン (Gaussian machine) [1] の場合のように、I_α には確率的変動を含めるなどの方法がある。

[10]**限定述語** (bound predicate) とは、定義節を用いて完備化 (completion) によって定義される述語である。(これはもちろん、アナログ的側面を捨象した近似的な定義に過ぎない。) 述語 π の**定義節** (definition clause) とは、$\beta\ B$. の形に書かれる節である。ここで、β は π を持つ正リテラルであり、この節の頭部 (head) と言う。ただし、β においては + を省略する。また、B はリテラルの集合である。定義節以外の節には**先頭節** (top clause) と**自由節** (free clause) がある。先頭節は:$-B$. の形、自由節は B. の形をしている。(1) に示した節は全て自由節である。

たとえば、述語 p の定義節が

```
p(X) -q(X,a).
p(f(X)) -r(X).
```

の2個であれば、p は

$$\forall A\{p(A) \Leftrightarrow \{\exists Y(q(A,Y) \wedge Y = a) \vee \exists X(A = f(X) \wedge r(X))\}\}$$

によって定義される。(p を持つ正リテラルを含む節は自由節かも知れないので、たとえ p が限定述語であっても、そのような節が全て p の定義節とは限らない。) 定義節を持つ述語は全て限定述語であり、定義節を持たない限定述語は充足不能である。束縛 X = f(Y) における f は2引数の自由述語と見なせる。

[11]限定述語以外の述語を自由述語 (free predicate) と言う。

響によって強さ B_π の抑制性の入力を受ける。一般に、π を持つ正リテラルの個数 (π が限定述語ならば、定義節の個数) が少ないほど、γ に対するこの項の効果による抑制性の入力が強く、また、各々の δ に対するこの項の効果による興奮性の入力が強いので、(4) の第1項による δ から γ への興奮性の入力も強い。従って、γ に対する発想推論が起動され易い。逆に、正リテラルの個数が多いほど、γ に対する発想推論は起動され難い[12]。

なお、(4) では、要素式の否定を証明するための発想推論とか、第2項の δ 同士が抑制し合うことなどについては考慮していない。これらの点をエネルギー関数に反映させるには、推論の中で各述語がどのように使われるかについてさらに詳しく考察する必要があるだろう。

変数 ξ のエネルギー関数を下記のように定義する。

$$(5) \qquad \sum_{\sigma \neq \tau} d_\sigma \overline{y_\sigma} d_\tau \overline{y_\tau}$$

y_σ は ξ の生起 σ を含むリテラルの活性度である。d_σ は σ が ξ の束縛に課する制限の強さを表わす非負の実数値であり、σ の**依存係数** (dependency coefficient) と言う。たとえば -X=f(Y) における X の位置の依存係数は大きく、Y の位置の依存係数は 0 とする。また、以下のように定義される述語 member の第1引数の依存係数は 0 である。

```
(6) member(E,[E|_]).
    member(E,[_|S]) -member(E,S).
```

簡単のため、各述語の各引数位置ごとに正負のリテラルの依存係数は決まっているとする。たとえば、ある節の中のリテラル -p(X) と別の節の中のリテラル -p(Y) の引数位置の依存係数は等しい。ただし後述のように、ある要素式が同時に正および負のリテラルとして参照されている場合には、その各引数位置の依存係数は 0 である。

活性の低い2つの (同一かも知れない) リテラルの間に変数に関する依存関係[13]がある場合、その変数のエネルギーは高い。たとえば、

```
(7) p(X) -q(X) -r(X).
```

という節の3つのリテラルの各引数位置をそれぞれ σ、τ および η とおくと、変数 X のエネルギー関数は

$$(8) \quad d_\sigma \overline{y_\sigma} d_\tau \overline{y_\tau} + d_\tau \overline{y_\tau} d_\eta \overline{y_\eta} + d_\eta \overline{y_\eta} d_\sigma \overline{y_\sigma}$$

となる。$\overline{y_\tau}$ と $\overline{y_\eta}$ はそれぞれ要素式 q(X) と r(X) の活性度であるから、この第2項 $d_\tau \overline{y_\tau} d_\eta \overline{y_\eta}$ はこれらの要素式の活性度が共に高いときに大きな値を持つ。従って、これらの要素式は互いに抑制し合う。また、第1項と第3項は要素式 p(X) の活性度が高いときは値が小さい。他の2つの要素式の活性度が高いとき、p(X) はそれらから興奮性の活性化を受ける。

節 Φ のエネルギー関数を下のように定める。

$$(9) \qquad A_\Phi \sum_{\gamma \neq \delta} c_\gamma y_\gamma c_\delta y_\delta + D_\Phi \prod_\gamma \overline{c_\gamma y_\gamma}$$

y_γ を Φ の中のリテラル γ の活性度 (要素式 α の活性度 x に対し、正リテラル +α の活性度を x とし、負リテラル -α の活性度を $\overline{x}$ とする) とし、A_Φ、D_Φ は非負の定数とする。c_γ は (0, 1] 区

[12] 次節を参照。

[13] ある変数に関する**依存関係** (dependency) とは、その変数が2つの (同一かも知れない) リテラルの2つの異なる引数位置に出現することである。

間の定数であり、γ の**関与係数** (commitment coefficient) と言う。また、上の式には現われないが、節の中の各リテラルには**偏向係数** (bias coefficient) というもうひとつの定数が割り当てられている[14]。

(9) の第1項は同時に2つ以上のリテラルの活性が高い場合に大きな値を持つ。これにより、高々1個のリテラルだけが真であることが望ましいことになる。従って、たとえば要素式pの活性が高い場合、

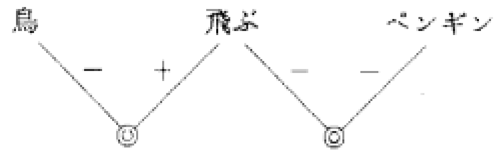
(10) `+p -q -r.`

という節によって、qとrとはpから興奮性の活性化を受ける。これは、pが与えられたときに(10)から仮説としてqとrとを導くという発想推論 (abduction) に相当する。

一方、(9) の第2項の値は全てのリテラルの活性が低い場合に大きくなる。これは少なくとも1個のリテラルが真であるべきことを意味する。従ってたとえば、要素式pの活性が低く、かつ要素式qの活性が高ければ、(10) によって要素式rは抑制性の入力を受ける。これは演繹推論 (deduction) に相当する。

(9) の A_Φ が大きいほど発想推論が起動され易く、D_Φ が大きいほど演繹推論が起動され易い。たとえば、下のような2つの節を考える。

(11) a. -鳥 +飛ぶ.
　　 b. -ペンギン -飛ぶ.

(12)


ここで、$D_a < D_b$ としよう。すると、要素式「鳥」の活性が高く「ペンギン」の活性が低ければ「飛ぶ」の活性は高くなる。しかし、「鳥」の活性が高くても「ペンギン」の活性が同程度に高ければ「飛ぶ」の活性は低くなる。こうして非単調な演繹的推論が制御できる。また、A_a が大きく A_b が小さいと考えると、aを用いて「飛ぶならば鳥だろう」という発想推論が行なえるが、bを用いた「飛ばないならばペンギンだろう」という発想推論は起動され難い。このように、エネルギー関数の適切な設定によって発想推論の制御もできる。

要素式 α のの偏向は、その正リテラルとしての偏向から負リテラルとしての偏向を差し引いたものである。リテラル γ の偏向は、下の式の値と0のうちの最大値とする。

(13) $$-\sum_{\Phi}(\Phi\text{ のエネルギー関数の第1項による }\gamma\text{ への入力})\times(\Phi\text{ における }\gamma\text{ の偏向係数}) + R\times J_\gamma$$

ここで、Φ は γ を含む節である。γ が正リテラル $+\alpha$ ならば $J_\gamma = J_\alpha$ であり、負リテラル $-\alpha$ ならば $J_\gamma = -J_\alpha$ である。J_α は、述語、変数、および節のエネルギー (要素式 α 自身のエネ

[14]節は、詳しくは次のように書く。

```
+.7/.5 p(X) -.9 q(X) 2.3(4.1).
```

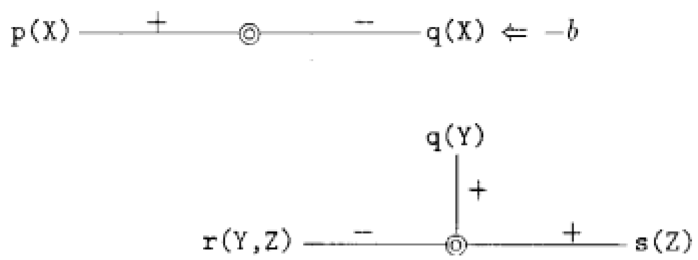
この節を Φ とすると、$A_\Phi = 2.3$、$D_\Phi = 4.1$ である。また、第1の正リテラルの関与係数は0.7、偏向係数は0.5で、第2の負リテラルの関与係数は0.9、偏向係数は0である。これらの係数の指定を省略した場合は、デフォルト値がとられる。関与係数のデフォルトは1、偏向係数のデフォルトは0である。

ルギーは除く)の各項による要素式αへの入力の総和である。Rは回復率(recovery ratio)と言い、全てのリテラルに共通の定数である[15]。因子化とは、次節で述べる記号演算の1種である。

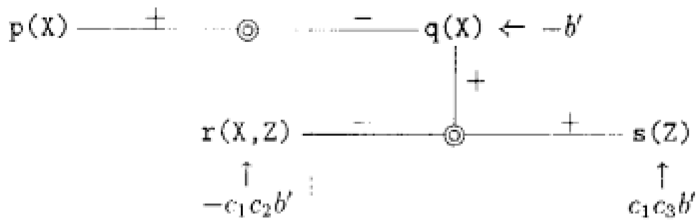
3 記号計算の制御

以下では簡単のため、全ての基本的な記号演算を、同一の述語を持つ2つの異なる要素式を単一化する[16]操作として捉える[17]。このとき、2つの別の要素式が単一化し、さらにそれらの対応する引数(変数)同士が単一化する[18]。後は、単一化の結果である要素式の偏向およびその各引数位置の依存係数を決めれば、この演算の後のエネルギー関数が定まる。2つの要素式の単一化には以下の2通りの場合がある。

第1の場合は、2つの(同一かも知れない)節において逆符号のリテラルとして参照されている2つの要素式を単一化する[19]場合で、この演算を接続(connection)と言う。たとえば、(14)の

(14)


2つの節を要素式q(X)とq(Y)とを単一化することによって接続すると、(15)のようになる。

(15)


ここで、⇐は要素式への外部入力(偏向)を表わす。bは(13)によって決まる正の値とする。一般に、正(負)リテラルに対する正(負)の偏向は、そのリテラルを単一化するような接続によって、相手の節の他のリテラルに分配される[20]。分配の重みは関与係数の積によって定義されると考えられる。たとえば(14)においては、r(X, Z)は$-c_1c_2b'$とc_1c_2b'という外部入力を受けると見なす[21]。これによって、r(X, Z)に対する記号推論が生じ易くなるが、エネルギーはこれらの外部入力がない場合と同じである。c_1、c_2、およびc_3はそれぞれ、下の節におけるq(Y)、r(Y, Z)、

[15] Rの値は多くの0でない偏向係数より少し小さい程度が望ましいと思われれる。

[16] 要素式を単一化する前に、それらを参照している節を複写しておいてもよい。

[17] 実際には、そのような計算の並列実行に相当するプログラム変換[2, 5](依存伝播; Dependency Propagation)を用いる。

[18] こうして、要素式と変数が複数の節において共有され、大域変数(transclausal variable)となることがある。

[19] これはいわゆる融合法(resolution)に対応する。このとき、単一化の結果である要素式の各引数の依存係数は0とする。

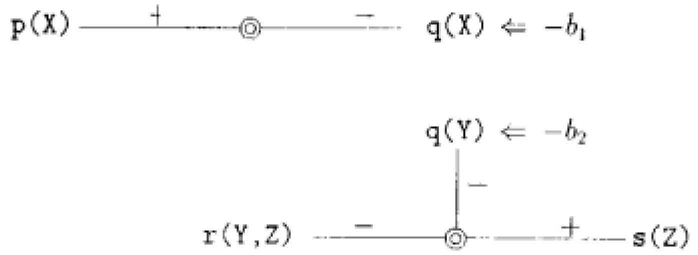
[20] これは、Hobbs [3]の枠組の一般化になっている。

[21] これだと活性拡散の計算が複雑になり過ぎるかも知れない。また、たとえばr(X, Z)に対する記号推論によってq(X)の偏向が小さくなった場合、r(Y, Z)だけでなくs(Z)に対する記号推論の優先度も下がることになるが、実はそうでない方がよいと思われる。

s(Z) の関与係数である。また、単一化の結果である要素式の各引数位置の依存係数は全て 0 とする。

第 2 の場合は、2 つの (同一かも知れない) 節において同符号のリテラルとして参照されている 2 つの要素式を単一化する場合で、この単一化を**因子化** (factoring) と言う。たとえば、(16) において q(X) と q(Y) とが因子化すると、(17) を得る。ここで、(17) における -q(X) は 2 つ

(16)



(17)

の節に属するので、2 つの偏向係数を持つことになる。因子化する 2 つのリテラルが同一の節中にある場合は、単一化によってできるリテラルの関与係数はもとの 2 つの関与係数の最大値、偏向係数はもとの 2 つの偏向係数の最小値とする。また、述語を共有する同符号のリテラルの同じ引数位置の依存係数は等しいから、単一化によってできるリテラルの各引数位置の依存係数は、もとの 2 つのリテラルの対応する依存係数に等しい。

述語を共有し、かつ対応する引数が全て等しいような 2 つの要素式は、義務的に単一化 (接続または因子化) する。また、1 つの変数が 2 通りに束縛されている場合、2 つの束縛は義務的に因子化する。たとえば、-X=f(Y) と -X=f(Z) が共存するとき、これらの束縛は単一化し、Y と Z は同一の変数となる。

記号計算における局所的な演算も、それによって見込まれるエネルギーの減少に応じた優先度で発火する。しかし、記号的演算はネットワークの形を変え、従ってエネルギー関数の定義を変えるから、記号計算の優先度を定めるには、各演算に伴うと予測されるエネルギーの減少を、そのエネルギー関数の形の変化から予測しなければならない、という点が活性拡散よりも複雑である。

簡単のため、記号的推論は活性拡散が収束した段階で行なわれると考えよう。そのとき、ある要素式が興奮性と抑制性の入力を同時に受けているならば、いずれかの入力を弱めることによって局所的なエネルギーが減少することが期待できる。従って、ある要素式に対する記号演算の優先度は、その要素式が受ける興奮性の入力の強さと抑制性の入力の強さ、およびその演算によってどの入力をどのくらい変化させることができるかに応じて決まる。たとえば、要素式のエネルギー関数 (3) の第 2 項は記号演算によって変化しないので、計算の制御にはあまり関係がないだろう。接続および因子化という演算によってエネルギーがどのように変化するかは、さらに具体的な文脈によって異なるので、処理の制御や解釈の優先度を規定するには、それに応じた場合分けが必要と思われる。以下では、典型的な場合の一部について述べる。

全体として、計算の目的は、先頭節の否定を説明すること[22]、つまり、エネルギーの小さな複合節を上記のような演算によって先頭節から作ることである。先頭節の :- の左辺に活性の高

[22]先頭節が (A) の場合、説明すべき命題は (B) である。

い隠れた正リテラルがあって、右辺の各リテラルはそこから抑制性の入力を受けるとする。これは先頭節の否定を仮説することに相当する。

たとえば、下の先頭節の２つの要素式はこの節の中で興奮性の入力を受ける。

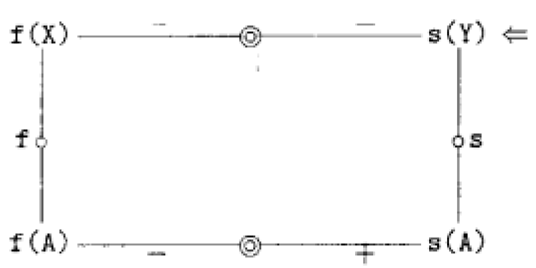
(18) :- -f(X) -s(Y).

ここで、s(Y) が負の偏向を持っているとすると、s(Y) は興奮性の入力と抑制性の入力を同時に受けることになる。さらに、

(19) -f(A) +s(A).

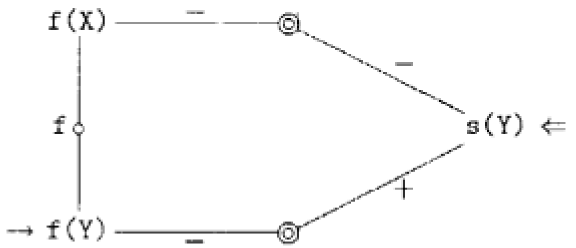
という節を考えると、これらの節は (20) に示すようなネットワークをなす。このネットワーク

(20)



中で、f(X) から述語 f のリンクを伝わって (4) の第１項によって f(A) に興奮性の活性化が及び、それが節 (19) のエネルギー関数 (9) の第２項によって s(A) の活性を高め、さらに述語 s を介して s(Y) に興奮性の入力を与える。こうして、s(Y) は２つの興奮性の入力と１つの抑制性の入力を受けることになる。そこで s(Y) と s(A) とを単一化して２つの節を接続すると、s(Y) の偏向は f(Y) に送られ、(21) のようになる。これは発想推論の一種であり、要素式 s(Y) に対する

(21)



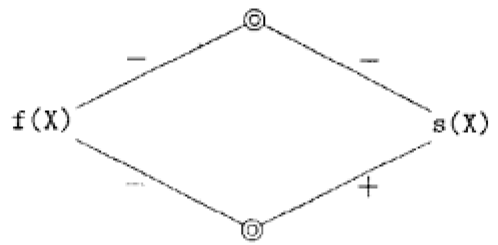
興奮性の入力を合わせたものの強さと抑制性の入力の強さとに応じた優先度で起動される。(21) では f(Y) が f(X) からの興奮性の入力と外部からの抑制性の入力を受けているので、f(Y) と f(X) とを因子化すると、(22) が得られる。ここで、(21) の f(Y) の偏向と f(X) の偏向の最大値 (-f(Y) の偏向と -f(X) の偏向の最小値の符号を逆転したもの)、つまり後者が (22) の f(X) の偏向になっている。こうして、先頭節のエネルギーを高めるような外部入力は消失した。

(20) で s(A) の活性が高いと考えるとこのような下降的な処理が起動されるが、上昇的な処理もまた活性拡散によって自然に実現される。もしも s(Y) に抑制されて s(A) の活性が低いとすると、それはさらに (9) の第２項によって f(A) を抑制する。こうして f(A) は興奮性の入力と抑制性の入力の両方を受けることになり、f(X) との因子化が起動されて、(23) となる。さらに s(Y) と s(X) とが単一化すると、(22) となる。その際、(23) の s(Y) の偏向は一担 (22) の f(X)

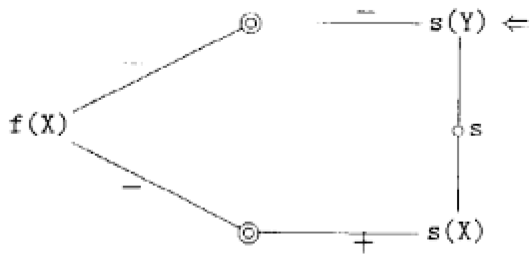
(A) :- -p(X) -q(X,Y).

(B) $\exists X,Y\{p(X) \wedge q(X,Y)\}$

(22)



(23)



に受け継がれるが、その値は (23) における f(X) の偏向より小さいので、後者が (22) の f(X) の偏向となり、結局 s(Y) の偏向は消失し、上と同様の結果を得る。

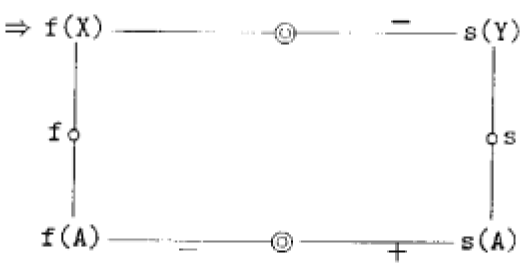
(20) から (22) に至るのに (21) を経るか (23) を経るかは、接続と因子化の優先度によって決まる。前記の通り、ある述語 (上の場合は s) を持つ正リテラルを含む節が多ければ、その述語を持つリテラルに関する接続の優先度は低い。しかし、(20) において (19) を経て s(A) が s を持つ他の正リテラルに比べて際だって高い活性を持っているとすれば、(19) を用いた接続の優先度は高くなる。活性拡散によるこのような上昇的な情報の流れにより、たとえば語の意味からの辞書引きのような処理が効率よくできることが期待される。

この例において節 (19) が選択されたのは、述語 f と s を経た活性拡散による所が大きい。これに対し、たとえば下の節は同様の活性拡散に関与しないので、用いられ難い。

(24) -q(B) +s(B).

この例題によって、自然言語におけるある種の照応現象が説明できる。たとえば、「フェリーが来た」という発話の後で「その船」によってそのフェリーを指すことができるが、これは上記の (20) から (22) に至る推論によって制約のエネルギーが下がる、ということに相当する。ここで、「フェリー」という不定記述 (indefinite description) は偏向が 0 のリテラル -f(X) に対応し、「その船」という定記述 (definite description) は偏向の大きな (抑制性の外部入力を受ける) リテラル -s(Y) に対応すると考える。X と Y はそれぞれ「フェリー」と「船」の指示対象を表わす。これに対し、「船が来た」という発話の後で「そのフェリー」によってその船を指すという言葉遣いはやや不自然である。これは、(25) において f(X) の偏向を消去する方法がさしあたっては

(25)



存在しないことに相当すると考えられる。

上で述べたのは偏向や節のエネルギー関数による抑制性の活性化が処理を引き起こす例であるが、リテラルへの抑制性の活性化の主なものとしては、他に変数のエネルギー関数 (5) によるものがある。つまり、変数に関する依存関係も偏向と同様に処理を起動する。この場合も、本質的には上記と同様の処理が行なわれる[23]。いずれの場合にも、どの節を用いるかなどに関する優先度は、エネルギー関数によって同様に定まる。つまり、興奮性の活性化と抑制性の活性化を共に受けているリテラルを処理する際、接続による発想推論の場合にはなるべく活性の高いリテラルと単一化し、因子化の場合にはなるべく活性値の異なるリテラルと単一化する、というのが一般的方針である。

4 結言

エネルギーの最小化から種々のヒューリスティクスが自然に導かれることを論じた。より複雑な例題を用いて上述の方法を吟味すること、およびコネクショニズムや論理プログラミングにおける学習法の利用を検討することなどが、今後の課題であろう。また、制約全体を熱力学的に開いた系にするため、エネルギーの供給をいかにして行なうかという問題もあり、この問題は時間の扱いにも関わるものと思われる。

参考文献

[1] Akiyama, Y., Yamashita, A., Kajiura, M. and Aiso, H. (1989) 'Combinatorial Optimization with Gaussian Machines,' *Proceedings of IJCNN'89*, pp. 533-540.

[2] Hasida, K. (1990) 'Sentence Processing as Constraint Transformation,' *Procedings of ECAI'90.*

[3] Hobbs, J., Stickel, M., Martin, P., and Edwards, D. (1988) 'Interpretation as Abduction,' *Proceedings of the 26th Annual Meeting of ACL*, pp. 95-103.

[4] Suttner, C.B. and Ertel, W. (1989) *Automatic Acquisition of Search Guiding Heuristics*, ms.

[5] Tuda, H., Hasida, K., and Sirai, H. (1989) 'JPSG Parser on Constraint Logic Programming,' *Proceedings of the European Chapter of ACL'89.*

[6] Waltz, D., and Pollack, J. (1985) 'Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation,' *Cognitive Science*, Vol. 9, pp. 51-74.

[23] ただし、制約変換 [5, 2] の場合には、変数に関する依存関係は透過 (penetration) や融合 (fusion) などの演算を引き起こす。偏向によって引き起こされる処理は、本原稿で述べたような演算の並列実行に相当する。