

IV-36

ニューラルネットワークモデルの交通制御問題への適用 (その2)

北海道大学工学部 学生員 関 俊一
正 員 中辻 隆
正 員 加来 照俊

1. はじめに

今日、世界の多くの都市が慢性的な交通渋滞問題に直面している。交通マネジメントシステムは、現有施設を空間的、時間的に有効利用することによってこの問題に対処しようとするものである。将来の交通マネジメントシステム、特に交通制御システムにおいては、制御方法を交通状況に応じて自律的に変化させる「自己組織化制御」がその基本理念となっている。これまでに、ファジィ論理やエキスパートシステムを応用することによって自己組織化能力を有する交通制御システムの開発が試みられている。

ニューラルネットワークモデルは、エキスパートシステムにおけるif-then 演算のような論理的演算を必要とせず、過去の入出力の結果のみから自律的に学習を行うことができるので、非線形、非定常、あるいは非論理的構造の表現能力に優れているといわれている。ニューラルネットワークモデルのこのような特性を、交通制御に関わる諸問題、例えば、交通変量の短期予測、非計測変量の推定、最適制御、あるいはインシデントの検知等に応用することによって、効率的な施設の運用を計ることができる。

本研究は、最適交通制御問題、特に最適スプリットの推定問題に対して階層型のニューラルネットワークモデルの適用を試みたものである。

2. ニューラルネットワークモデル

神経回路網(ニューラルネット)は神経細胞(ニューロン)が多数結合して何らかの信号を出す一つのシステムであり、個々のニューロンを工学的にみると他のニューロンからの入力を受け取るシナプス結合の機能とこの受け取った出力を出す機能をもっている。また、シナプス結合を介してどれだけの入力を送るかという結合の重みはそれぞれのシナプス結合に依存する。 Y_j が i 番目のニューロンへの入力、そしてその間の結合強さを表すシナプス荷重を W_{ij} とすると i 番目のニューロンが受け取る入力 X_i は

$$X_i = \sum_j W_{ij} Y_j + I_i \quad (1)$$

であたえられる。 I_i は外部からの i 番目のニューロンへの入力であり、入力 X_i を受け取ったニューロンは出力 Y_i をだす。

$$Y_i = f(X_i / \mu_0) \quad (2)$$

ここに関数 f は、 μ_0 の値によって傾きが変わるシグモイド型の単調増加関数である。

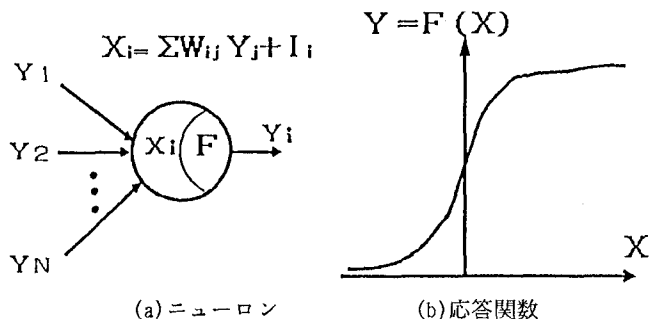


図-1 ニューラルネットワークモデル

Application of Neural Network Models to Optimal Traffic Control Problem (2)

by Shunichi SEKI, Takashi NAKATSUJI and Terutoshi KAKU

2. 1 階層的ネットワーク

図-2は、後に最適制御問題に使用する階層ネットワークの、基本的な構造を示している。ここでは、入力層二つの中間層、出力層をもつ4層のネットワークを用いる。入力層は入力された信号を次の層に伝える役目だけを担う。この階層的ネットワークの学習に用いられるのが逆伝搬法（Back-Propagation）である。

各層間のシナプス荷重の大きさが与えられると、入力層へ入力された信号は、式(1)(2)に基づき、中間層から出力層へと順次伝達される。一般的に中間層の数、あるいはニューロンの大きさは任意であり、経験的に定められる。出力層の結果は既知の教師信号と比較され、誤差の度合いに応じて各層間のシナプス荷重の大きさが補正される。いま、出力層の k 番目のニューロンの出力信号を y_k 、それに対応する教師信号を z_k とすると、逆伝搬法による学習は、誤差の自乗和

$$E = (1/2) \sum (z_k - y_k)^2 \quad (3)$$

を極小化しようとすることによって実現される。いま、式(2)のシグモイド関数 f が次式で定義され、

$$f = 1 / \{ 1 + \exp(-X) \} \quad (4) \quad (\text{図-1, (b)})$$

BC間、CD間のシナプス荷重が W_{ij} 、 W_{jk} と与えられるとき、それらの変化量 δW_{ij} 、 δW_{jk} は

$$\delta W_{jk} = (z_k - y_k) y_k (1 - y_k) y_j \quad (5)$$

$$\delta W_{ij} = \sum_k (z_k - y_k) y_k (1 - y_k) W_{jk} y_i y_j (1 - y_j) \quad (6)$$

として求めることができる。実際の計算においては演算の収束を早めるために

$$\Delta W(n) = \varepsilon \delta(n) + \alpha \Delta W(n-1)$$

$$W(n+1) = W(n) + \Delta W(n) \quad (7)$$

としている。ここで ε は学習定数といわれる十分小さな正の値であり、 α は安定係数といわれる0から1の間の値である。

2. 2 ステップワイズ法

ローカルミニマムへの収束は、ニューラルネットワークモデルにおいて大きな問題である。これを避けるために、ニューロンの出力を確率的なものにしたボルツマンマシンやガウシアンマシン等の方法が提案されている。Szuは、グローバルミニマム（最適極小値）に収束する確率的方法、コーシーマシンを開発した。コーシーマシンは、微少なランダムノイズを発生させることによってシナプス荷重を調整する。目的関数が減少したときにはノイズによって荷重を変化させ、それ以外の場合には荷重の大きさを前値に戻し、その繰り返しによって目的関数を減少させていく方法である。Wassermanは、コーシーマシンと逆伝搬法を組み合わせた方法を提案した。それによると「シナプス荷重の値を、逆伝搬法により求められた結果と、コーシーマシンによるランダムステップによる結果との和になるように調整すれば、どちらか一方の方法だけを用いた場合よりも、早く、確実にグローバルミニマムに収束するシステムを形成することができる」のである。ここでは、Wassermanの方法を参考に別の方法を提案する。最初に、ローカルミニマムへの収束を避けるためにコーシーマシンによってシナプス荷重を調整し、その後、逆伝搬法を適用する。この学習アルゴリズムをここではステップワイズ法と名付ける。

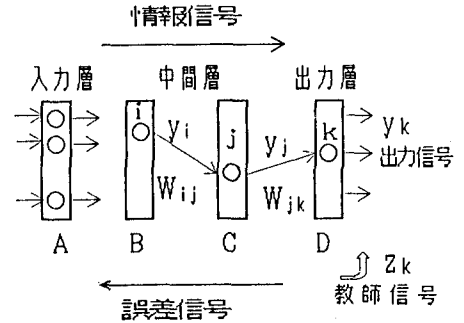


図-2 階層型ニューラルネットワークモデル

3 スプリット最適制御

いくつかの交差点から構成されるネットワークにおいて、各交差点のスプリットを最適化するニューラルネットワークモデルについて説明する。ただしオフセットの最適化は行わないものとする。

3. 1 状態方程式

サイクル長単位に離散化された時間軸を持つ交通ネットワークにおいて、制御変量（スプリット）と状態変量（行列長）の時系列ベクトルを $u(k)$ 、 $y(k)$ 、 $(k=1, 2, \dots, n)$ とする。固定した時間区間 $0 < t < n \delta t$ において制約条件

$$0 \leq y(k) \leq Y_{\max} \quad (8)$$

$$U_{\min} \leq u(k) \leq U_{\max} \quad (9)$$

のもとで目的関数

$$J = \sum \|y(k)\|^2 \quad (10)$$

を最小とするような制御変量 $u(k)$ を求める問題を考える。このとき、状態変量 $y(k)$ は、 $u(k)$ や流入交通量 $q(k)$ を説明変数と状態方程式によって記述される。例えば、各流入リンクにおける行列長を状態変量とするときには、状態方程式は、

$$y(k+1) = y(k) + B_0 u(k) + B_1 u(k-1) + \dots + B_m u(k-m) + q(k) \quad (11)$$

と表現され、初期値を与えることによって行列長の時系列値を算出することができる。ここで、 B_j は制御重み行列と言われ、各流入部における飽和交通流率により定義される。

3. 2 最適スプリット推定モデル

最適なスプリットの時系列値を推定するためのニューラルネットワークモデルを図-3に示す。四層の階層構造とした。A（入力）層、D（出力）層のニューロンの個数の決め方は、スプリットをサイクルごとの時系列値とする場合と、どのサイクルでもスプリット一定とする場合とがある。まず前者の決定方法は、A層からは制御変数、スプリット $u(k)$ の時系列値を入力する。サイクル数を N 個とし、 r 番目交差点の現示数を p_r とするとA層は全部で $\sum p_r N$ 個のニューロンから構成される。B層においては情報の統合は行わずA層からのデータをそのままC層へ伝える。D層からは状態変量 $y(k)$ の時系列値を出力する。 r 番目交差点の流入リンク数が n_r であるときには、D層は全部で $\sum n_r N$ のニューロンを持つ。後者の場合には、現示数に等しい。

本システムは、二つの過程から構成される。一つが各交差点における交通流の動的挙動を表現するための学習過程であり、他方がスプリットを最適化するための最適化過程である。学習過程においてはまず、入力変量（スプリットの時系列値）を乱数によって初期設定する。次に逆伝搬法を用いて、出力層からの出力変量 $y(k)$ が学習信号 $z(k)$ に一致するようシナプス荷重の大きさを調整する。このとき学習信号 $z(k)$ は状態方程式に基づいて算出される。スプリットの初期値を変えて学習を繰り返すことによって安定した入出力関係を形成することができる。

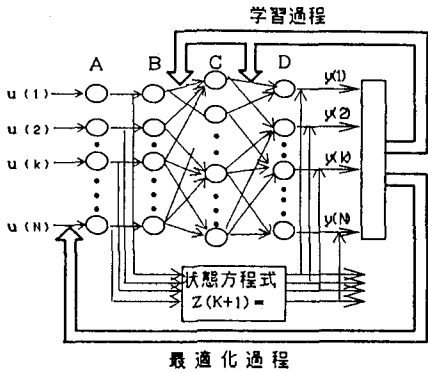


図-3 最適スプリット推定モデル

こうした学習過程を形成することにより単純な積和演算のみで状態変量を推定することができるようになる。巨視的交通流モデルにおいては、制御変量の微調整による状態変量の再計算に多くの演算時間を費やしている。本モデルを交通流モデルに組み込むことにより演算時間の短縮を計ることができる。

3. 3 最適化過程

最適化過程においては、与えられた制限条件下で目的関数を最小とするよう制御変量を調整する。この調整は、シナプス荷重の調整と同じように目的関数、式 (10) を制御変量 $u_i(k)$ で偏微分することによって求めることができる。すなわち、

$$\delta u_i = \eta \sum_k y_k^2 (1 - y_k) \sum_j W_{ij} W_{jk} (1 - y_j) \tag{12}$$

ここで、 y_i 、 y_j 、および y_k は、図-2 に示すように、それぞれ B、C、D 層における出力値である。また、 η は学習定数と呼ばれる定数である。ただし、この調整は、シナプス荷重の調整における逆伝搬法とは異なり直接 A 層にフィードバックされる。ここでは、この算法をフィードバック法と呼ぶ。このように制御変量の調整量をシナプス荷重を用いて表現することによって、シナプス荷重の変化に連動して、制御変量もその推定係数を変化させることができる。すなわち、自己組織化能力を持つことを表している。これが、ニューラルネットワークモデルによるスプリット最適化の特色である。

4 数値計算例

4. 1 学習精度の比較

中間層、ニューロンの数は、経験的に定められると述べたが、その解析結果について示す。

図-4 は、最も単純な 4 枝交差点を対象として、入力層のニューロン数を現示の数と一致させて、入力を与えた時の、(a) ニューロンの数、(b) 中間層の数、(c) 学習時に与えるスプリットパターンを変えた場合について逆伝搬法による推定誤差を表したものである。

(a) は、四層構造として、図-2 における C 層のニューロンの数を 2、3、4、・・・と変えた場合の比較である。(b) は、C 層の数を 2、3、4 つと変え、それらの層のニューロンの総数を 8 に固定した場合である。(a)、(b) とともに学習時に与えるスプリットパターン数は 20 として学習させ、未学習スプリットパターンに対する精度を比較した。この二つについて共通していえることは、出力層の前の層のニューロンの数が 2 の場合に推定誤差が大きくなり有効ではないことである。これは、出力層の前の層のニューロンの数 (M) と、出力層のニューロンの数 (R) は次式の関係を満たさなければならないことによるものである。

$$M \geq \log_2 R \tag{13}$$

今回は出力層のニューロンの数が 8 であるため、出力層の前の層のニューロンの数が 2 では有効な精度が得られない結果となった。それ以外は、学習時に与えたスプリットパターンに対し、未学習のスプリットパターンでの誤差は大きくはなるが、有効範囲内となっている。また、それらを比較すると誤差の値にそれほど優劣

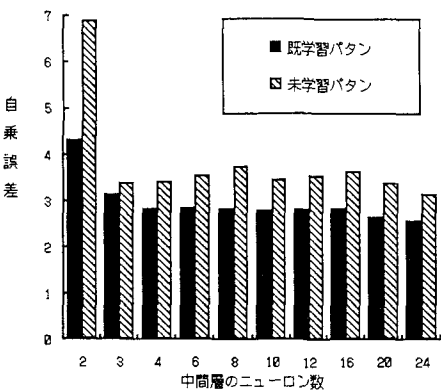


図-4 (a) 学習精度の比較

がみられない。中間層の数、ニューロンの数による精度への影響はそれほど大きくはない。(c)においては、四層構造で、C層のニューロンの数を8とし、学習時に与えたスプリットパターン数による影響を調べた。これも、5パタンの時を除いては、精度にばらつきはみられない。

これらのことから、単純な構造でも複雑な構造に精度的に劣ることはないと思われるので、なるべく単純な構造として、演算時間を早めた方が有効であろう。

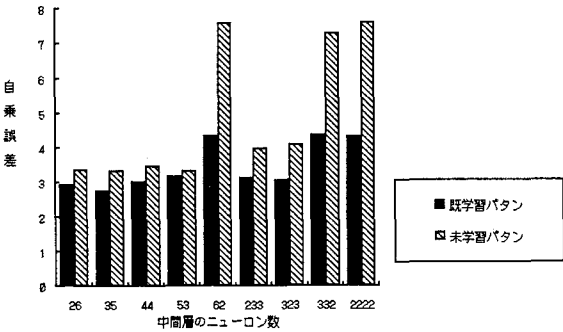


図-4 (b) 学習精度の比較

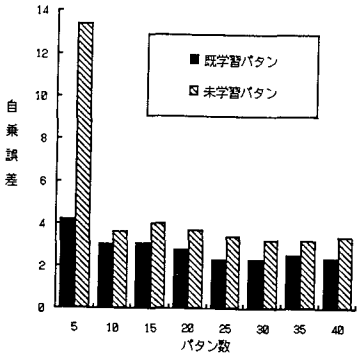


図-4 (c)

4. 2 実交差点への適用

Singhによって与えられた数値計算例を用いて解析解との比較を行った。図-5に交差点の形状、流入リンク、及び各交差点における現示方法を示した。入力層には、スプリットの時系列値をあたえとした。各リンクにおける飽和交通流率等の基本的入力値については文献(10)に詳しい。最適化は行列長最小規範によって行った。実際の計算においては、ローカルミニマムへの収束を避けるとともに演算速度を早めるために、前述のコーシーマシンとフィードバック法を段階的に組み合わせたステップワイズ法を用

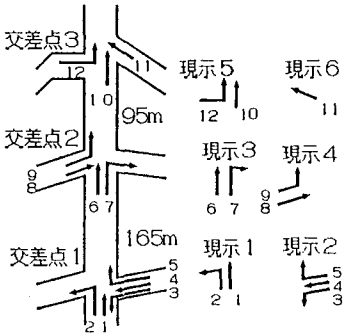


図-5 交差点形状と現示(Singh et al)

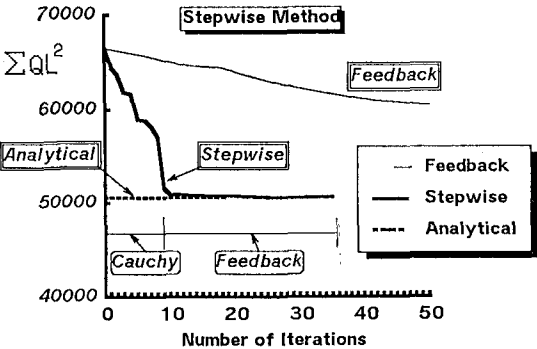


図-6 ステップワイズ法による最適解への収束

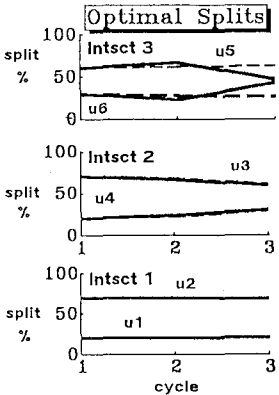


図-7 最適スプリットの推定結果

いた。図-6に最適解への収束状況を、フィードバック法のみでの収束状況と比較して示し、図-7に、推定された最適スプリットの時系列値を示している。 $u_1(k)$ は図-5の現示1にと、それぞれに対応するスプリットの時系列値である。コーシーマシンの導入によりグローバルミニマムへの収束が促進されていること、スプリットの推定値が解析解によく一致していることが示されている。

5 あとがき

ニューラルネットワークモデルの代表的モデルである階層型モデルをスプリット最適制御問題に適用した。これにおいて、従来の単独交差点における行列長最小化規範に基づく最適化問題を多交差点の問題に拡張して数値解析を行い、解析解と一致することを確認した。

しかし、今後に残されている課題も多い。一番大きな問題はオフセットの最適化である。このこと同じ多層型のモデルを導入して交通状況と経験的に結び付けて推定するシステムも考えられるが、組み合わせが膨大となり学習することが困難だと思われる。ホップフィールド型のモデルの適用を検討していきたい。

ニューラルネットワークモデルを組み込んだ交通シミュレーションプログラムは多くの優れた機能を有する。それは、簡単な積和演算による状態量の推定能力であり、ローカルミニマムの回避能力であり、演算速度の向上（並列処理能力）であるが、最も大きなものは過去の経験を取り込む学習能力である。こうした交通流シミュレーションプログラムの実現によって、交通管制官がリアルタイムにシミュレーション解析を行いながら制御パラメータを検討する。そしてその結果を学習機能を通してモデルにフィードバックさせる。そのようなシステムが近い将来作られるものと思われる。

参考文献

- 1) OECD: Dynamic Traffic Management in Urban and Suburban Road Systems, 1987
- 2) 秋山他: ファジィ流入制御モデルを用いた交通制御方法の評価と検討、土木学会論文集、413号/IV-12 1990, pp. 77-86
- 3) R. Sigua et al.: An Expert System for Intersection Design and Control, 土木計画学研究・講演集、No. 12, 1989, pp. 275-282
- 4) P. D. Wasserman: Neural Computing, Van Nostrand, 1988
- 5) D. E. Rumelhart et al.: "Learning internal representations by error propagation, in Parallel Distributed Processing Vol. 1, MIT Press, 1986
- 6) 麻生: 「ニューラルネットワーク情報処理」、産業図書、1988
- 7) 前田他: ヒトの上肢多関節運動を学習・生成する多層神経回路モデル、信学技法、MBE87-133, 1988
- 8) 合原: 「ニューラルコンピュータ」、東京電機大出版、1988
- 9) 武田: 神経回路網と組み合わせ最適化問題、数理科学、No. 289, pp. 14-22, 1987. 7
- 10) M. G. Singh and et al.: Modeling and hierarchical optimization for over-saturated urban road traffic networks, Int. Journ. Control, 20, No. 6, pp. 913-934, 1974
- 11) 今井: 「ニューラルネットワーク - コレクション -」、共立出版、1990
- 12) T. Nakatsuji and et al.: Application of Neural Network Model to Traffic Engineering Problems, 11th Int. Symp. Transp. Traffic Theory, 1990 (in press) pp. 291-306