Particle Swarm Optimizationによるリカレント スパイキングニューラルネットワークの学習法

山本 昌弘*・黒江 康明*・飯間 等*

Learning Method of Recurrent Spiking Neural Networks
by Using Particle Swarm Optimization

Masahiro YAMAMOTO*, Yasuaki KUROE* and Hitoshi IIMA*

In this paper, we propose a learning method of recurrent spiking neural networks by using particle swarm optimization (PSO). The existing learning method by using the gradient based method sometimes falls into local minima. The proposed method aims to find the global optimum regardless of initial solutions. Since PSO can treat optimization problems in which the objective function is a non-differentiable function, in this paper we formulate learning problems with such objective functions and propose a learning method based on PSO to solve them. The proposed method makes it possible to treat not only the learning problem in terms of firing instants which the conventional method treats, but also one in terms of the number and frequency of firings.

Key Words: spiking neural network, particle swarm optimization, learning method, recurrent neural network

1. はじめに

スパイキングニューラルネットワーク (以下 SNN) は、スパイクの伝達により情報を伝達し、情報はスパイク列の発生時刻あるいは時間間隔に埋め込まれるという点でより生体に近いニューラルネットワークのモデルとして、近年活発に研究されている¹⁾。SNN は連続信号と不連続なスパイク信号が混在し、スパイクの発生は事象駆動として扱う必要があるためハイブリッドシステムとなり、設計・解析が非常に困難である。そのため学習法の研究もあまり多くないが、これまでいくつかの方法が提案されている。まず、教師なし学習として階層型の SNN に対し、ヘップ学習則の考え方に基づく学習法²⁾が提案された。また、教師付き学習として、リカレント SNN に対して勾配法に基づく学習法³⁾が提案された。その後、階層型の SNN に対し、バックプロパゲーション法と同様の勾配法に基づく学習法⁴⁾が提案された。

ところが、勾配法に基づく方法^{3), 4)}ではシナプス結合荷重の初期値を適切に選択しなければ局所的最適解に陥り、大域的最適解が得られないことがある。また、これらの方法では、発生するスパイク列のそれぞれのスパイクの発生時刻を望みの時刻に一致させる問題として学習問題を定式化していた。

ところが、SNN の学習問題においてはスパイクの発生時刻そのものよりその発生頻度が問題となることがあり、望みのスパイク発生頻度を実現するための学習法を開発する必要がある。

本論文では、非線形最適化問題を解くための有力な手法の 1 つとして知られている Particle Swarm Optimization⁵⁾ (以下 PSO) を学習に用いた、大域的最適解を得るためのリカレント SNN の学習法を提案する。PSO は目的関数の連続性や勾配を必要としないために、さまざまな目的関数を用いることができる。そこで、スパイクの発生時刻を望みの時刻に一致させる問題に対して、いくつかの目的関数を設定した PSO に基づく学習法を提案する。また、本論文ではスパイク列の発生頻度を実現する問題を扱う。具体的には、従来の学習問題に加えて、スパイクの発生時刻は問題とせずに発生回数を一致させること、および発生回数の上限値がある望みの値とすることを新たに取り入れた学習問題を設定し、この学習問題に対する PSO に基づく学習法を提案する。

SNN の応用例に、リカレント SNN を用いて視覚系の特徴抽出機構のモデルを構築する研究⁶⁾がある。このモデルでは、物体の動きに対する光刺激にニューロンが応答することで運動方向選択性や動的視差を実現しており、ニューロンの応答の違いはスパイク列の発生頻度により表現される。文献 6) では学習法として勾配法を用いているため、スパイク列の発生時刻を与えることにより発生頻度を実現している。本論文で提案している方法により発生頻度を直接扱うことができる。このような学習問題は多くあり、本論文の成果により SNN

* 京都工芸繊維大学 京都市左京区松ヶ崎御所海道町

* Kyoto Institute of Technology, Goshokaido-cho, Matsugasaki, Sakyo-ku, Kyoto

(Received February 12, 2010)

(Revised September 15, 2010)

Fig. 1 Integrate-and-fire model

の応用が広がるものと期待される。

2. リカレントスパイクングニューラルネットワーク

本論文で対象とするリカレント SNN について説明を行なう^{3), 7)}。

2.1 ニューロンの発火機構のモデル

本論文ではスパイクングニューロン (以下 SN) のモデルとして代表的である **Fig. 1** に示す integrate-and-fire モデルを対象とする。このモデルは入力刺激を一定期間蓄積する積分機構とスパイクを発する機構、内部状態をリセットする機構により構成される。SNN の i 番目のスパイクングニューロン SN_i ($i = 1, 2, \dots, M$; M はニューロン数) において、発火を促す入力 $e_i(t)$ が積分機構 (一次遅れ要素 $\frac{1}{s+c_i}$; c_i は定数) に入力され、内部状態 $p_i(t)$ の絶対値がしきい値 $s_i (> 0)$ に達すると SN_i はスパイクを発し、同時に内部状態を零にリセットする。このとき、スパイクの符号は信号 $p_i(t)$ の符号が保存されるとしている。スパイクに Dirac のデルタ関数 $\delta(t)$ を用い、このモデルは次式で表現される。

$$\sigma_i(t) = \sum_{k_i=1}^{K_i} \epsilon_{i,k_i} \delta(t - t_{i,k_i}) \quad (1)$$

$$t_{i,k_i} = \min\{t : t > t_{i,k_i-1}, |p_i| \geq s_i\} \quad (2)$$

$$\epsilon_{i,k_i} = \text{sgn}[p_i(t_{i,k_i}^-)] \quad (3)$$

$$\frac{dp_i(t)}{dt} = -c_i p_i(t) + e_i(t), \quad t_{i,k_i-1}^+ < t < t_{i,k_i}^- \quad (4)$$

$$p_i(t_0) = p_i^0 \quad (5)$$

$$p_i(t_{i,k_i}^+) = 0 \quad (6)$$

ここで、 $\sigma_i(t)$ は SN_i の出力スパイク列、 K_i は開始時刻 ($t = t_0$) から終了時刻 ($t = t_f$) までに SN_i が発した総スパイク数、 t_{i,k_i} は SN_i が k_i ($k_i = 1, 2, \dots, K_i$) 番目のスパイクを発した時刻 (ただし、 $t_{i,0}=0$)、 ϵ_{i,k_i} は k_i 番目のスパイクの符号、 p_i^0 は SN_i の内部状態の初期値である。

2.2 ネットワークのモデル

Fig. 2 に示すように、2.1 節の SN を用いたリカレント SNN モデルを対象とする。各スパイクングニューロン SN_j からの出力 $u_{i,j}(t) = \sigma_j(t)$ は、遅延を表わす伝達要素 $g_{i,j}(s)$

Fig. 2 Recurrent SNN model

Fig. 3 Spike-response function

を通り、出力 $y_{i,j}(t)$ がシナプス結合荷重 $w_{i,j}$ により重み付けされる。さらに外部からの入力 $v_i(t)$ が加算され、 SN_i の入力 $e_i(t)$ となる。SNN モデルを次式に示す。

$$e_i(t) = \sum_{j=1}^M w_{i,j} y_{i,j}(t) + v_i(t) \quad (7)$$

$$u_{i,j}(t) = \sigma_j(t) \quad (8)$$

$$\frac{d\mathbf{x}_{i,j}(t)}{dt} = \mathbf{A}_{i,j} \mathbf{x}_{i,j}(t) + \mathbf{b}_{i,j} u_{i,j}(t) \quad (9)$$

$$y_{i,j}(t) = \mathbf{c}_{i,j} \mathbf{x}_{i,j}(t) \quad (10)$$

$$\mathbf{x}_{i,j}(0) = \mathbf{x}_{i,j}^0 \quad (11)$$

$$g_{i,j}(s) = \mathbf{c}_{i,j} (s\mathbf{I} - \mathbf{A}_{i,j})^{-1} \mathbf{b}_{i,j} \quad (12)$$

$g_{i,j}(s)$ は状態空間表現として表わしており、 $\mathbf{x}_{i,j}(t)$ は N 次元状態ベクトル、 $\mathbf{A}_{i,j}$ 、 $\mathbf{b}_{i,j}$ 、 $\mathbf{c}_{i,j}$ は $N \times N$ 行列、 $N \times 1$ ベクトル、 $1 \times N$ ベクトル、 $\mathbf{I}$ は $N \times N$ の単位行列である。 $g_{i,j}(s)$ は遅延を表わすと同時に **Fig. 3** のように伝達するスパイクの波形を表わす spike-response function と呼ばれる関数を決める役割を果たす⁷⁾。

以上のように、SNN モデルは連続信号と不連続なスパイク信号が混在し、スパイクの発生は事象駆動として扱う必要があるためハイブリッドシステムとなり、設計・解析が非常に困難である。

3. 学習問題

以前に扱っていた学習問題³⁾は、2 章の SNN の SN_i が発生するスパイクの発生回数 K_i および発生時刻 t_{i,k_i} が、与えられた望みの発生回数 K_i^d および発生時刻 t_{i,k_i}^d に一致するようにシナプス結合荷重 $w_{i,j}$ を決定する問題とし、これを最適化問題として定式化していた。このとき、最適化問題の目的関数は微分可能である。ところが、本論文で提案する PSO に基づく学習法では微分不可能な目的関数を取り扱うことができる。そこで本論文では、従来の学習問題に加えて、スパイクの発生時刻は問題とせず発生回数を一致させること、お

および発生回数の上限值をある望みの値とすることを新たに取り入れたつぎの学習問題を設定する.

学習問題

任意のニューロン SN_i に

- a) スパイクの発生回数と発生時刻
- b) スパイクの発生回数
- c) スパイクの発生回数の上限值

のいずれかを指定する望みのスパイク列を発生するようにシナプス結合荷重 $w_{i,j}$ を決定する.

望みのスパイク発生回数 K_i^d と発生時刻 t_{i,k_i}^d の両方, 望みのスパイク発生回数 K_i^d のみ, 望みのスパイク発生回数の上限值 K_i^{du} を与えるニューロンのインデックスの集合をそれぞれ O_a, O_b, O_c とし, 全ニューロンのインデックスの集合を $O = \{1, \dots, M\}$ とする. このとき, 学習問題は SN_i のスパイクの発生回数 K_i が K_i^d に等しくなること, および K_i^{du} 以下となることを制約条件とし, SN_i のスパイクの発生時刻 t_{i,k_i} と望みのスパイクの発生時刻 t_{i,k_i}^d との差をとる関数を目的関数 J_o とした最適化問題として次式のように定式化できる. J_o の具体的な関数は後に示す.

$$\text{Minimize } J_o(t_{i,k_i}) \quad (\forall i \in O_a) \quad (13)$$

$$\text{w.r.t. } w_{i,j} \quad (\forall i, j \in O)$$

$$\text{subject to } K_i = K_i^d \quad (\forall i \in O_a \cup O_b)$$

$$K_i \leq K_i^{du} \quad (\forall i \in O_c)$$

4. PSO を用いた SNN の学習法

3 章の SNN の学習問題を解くために, PSO を用いた学習法を提案する. PSO は目的関数が微分可能である必要がないので, 多種多様な目的関数を設定できる. そこで, 本論文ではいくつかの目的関数を提案し, 数値実験でそれらの有効性を比較する.

4.1 PSO の概要

PSO は生物の群行動を基に考えられたメタヒューリスティクス⁵⁾で, Particle と呼ばれる複数の探索点 (解候補) が過去に発見した最良解の情報を共有しながら解空間を探索し, 最適解を発見しようとする. 目的関数の連続性や勾配を必要とせずに目的関数値のみを用いるので, 非常に拡張性が高く, 目的関数が多峰構造になるような複雑な問題にも適用可能である.

4.2 目的関数の設定

PSO は基本的に制約なしの問題に対する方法であるので, 最適化問題 (13) の集合 O_a, O_b, O_c に属するニューロンのスパイク発生回数 K_i に関する制約条件をそれぞれペナルティを与える目的関数 J_a, J_b, J_c に置き換えて, 次式の目的関数 J を有する制約なしの最適化問題に変換する.

$$J = J_o(t_{i,k_i}) + \mu_a J_a(K_i) + \mu_b J_b(K_j) + \mu_c J_c(K_\ell) \\ i \in O_a, j \in O_b, \ell \in O_c \quad (14)$$

ここで, μ_a, μ_b, μ_c はペナルティ項の重みである.

従来の学習法^{3), 7)}では勾配法を用いるために, スパイクの発生回数と発生時刻を学習する問題しか扱えなかった. それに対して, PSO は目的関数の連続性や勾配を必要とせずに目的関数値のみを用いるので, さまざまな目的関数を設定でき, 発生回数のみを一致させること, および発生回数の上限值をある望みの値とすることを新たに取り入れた問題を扱うことができる. また, スパイクの発生回数と発生時刻を学習する問題に対しても微分不可能な関数を設定できる. 以降では目的関数 (14) の各項 J_o, J_a, J_b, J_c の種々の定義を提案する.

4.2.1 発生時刻を一致させるための目的関数 J_o

(14) 式第 1 項の目的関数 J_o はスパイクの発生時刻が所望の時刻に近いほど値が小さくなるように設定すればよい. ここではつぎの 2 種類の関数 J_{o1}, J_{o2} を設定する. J_{o1} は SN_i における望みのスパイク発生時刻 t_{i,k_i}^d と実際に発生する時刻 t_{i,k_i} との 2 乗誤差とし, J_{o2} は差そのもの (の絶対値) とする.

$$J_{o1} = \sum_{i \in O_a} \sum_{k_i=1}^{K_i} (t_{i,k_i}^d - t_{i,k_i})^2 \quad (15)$$

$$J_{o2} = \sum_{i \in O_a} \sum_{k_i=1}^{K_i} |t_{i,k_i}^d - t_{i,k_i}| \quad (16)$$

4.2.2 発生回数を一致させるための目的関数 J_a

発生回数に関する制約条件 $K_i = K_i^d$ ($\forall i \in O_a$) を満たさない場合, (14) 式の第 2 項 J_a に対して, 3 種類の関数 J_{a1}, J_{a2}, J_{a3} を設定する.

J_{a1} では, $K_i > K_i^d$ の場合, ネットワークの終了時刻 t_f よりも十分大きい値の時刻 t_{i,k_i}^p を教師信号とし, 過剰に発生したスパイクの発生時刻との 2 乗誤差をペナルティ項とする. また, $K_i < K_i^d$ の場合, 実際に発生する最後のスパイクの発生時刻 t_{i,K_i} に, 不足している $K_i^d - K_i$ 個のスパイクが同時に発生したと考え, 望みのスパイクの発生時刻との 2 乗誤差をペナルティ項とする. 以上より, J_{a1} を次式で与える.

$$J_{a1} = \begin{cases} \sum_{i \in O_a} \sum_{k_i=K_i^d+1}^{K_i} (t_{i,k_i}^p - t_{i,k_i})^2 & (K_i > K_i^d) \\ \sum_{i \in O_a} \sum_{k_i=K_i+1}^{K_i^d} (t_{i,k_i}^d - t_{i,K_i})^2 & (K_i < K_i^d) \\ 0 & (K_i = K_i^d) \end{cases} \quad (17)$$

J_{a2} では望みのスパイク発生回数 K_i^d と実際に発生する回数 K_i との 2 乗誤差をペナルティ項とし, 次式で与える.

$$J_{a2} = \sum_{i \in O_a} (K_i^d - K_i)^2 \quad (18)$$

J_{a3} では望みのスパイク発生回数 K_i^d と実際に発生する回数 K_i との誤差の絶対値をペナルティ項とし, 次式で与える.

$$J_{a3} = \sum_{i \in O_a} |K_i^d - K_i| \quad (19)$$

4.2.3 発生回数のみを一致させるための目的関数 J_b

発生回数に関する制約条件 $K_j = K_j^d$ ($\forall j \in O_b$) を満たさない場合, J_b は望みのスパイク発生回数 K_j^d と実際に発生する回数 K_j との誤差の絶対値をペナルティ項とし, 次式で与える.

$$J_b = \sum_{j \in O_b} |K_j^d - K_j| \quad (20)$$

4.2.4 発生回数の上限値をある望みの値とするための目的関数 J_c

発生回数に関する制約条件 $K_\ell \leq K_\ell^{du}$ ($\forall \ell \in O_c$) を満たさない場合, J_c は望みのスパイクの発生回数の上限値 K_ℓ^{du} を越えた回数に応じたペナルティを次式で与える.

$$J_c = \sum_{\ell \in O_c} \max(K_\ell - K_\ell^{du}, 0) \quad (21)$$

4.3 解候補の更新式

Particle p ($p = 1, 2, \dots, P$: P は Particle の個数) の解候補とするシナプス結合荷重 $\mathbf{w}^p = (w_{1,1}^p, w_{1,2}^p, \dots, w_{i,j}^p, \dots, w_{M,M}^p)$ を繰り返し更新して最適解を探索する. 解候補の更新式は PSO の更新式に基づいて以下のように表わされる.

$$w_{i,j}^{p,k+1} = w_{i,j}^{p,k} + \Delta w_{i,j}^{p,k+1} \quad (22)$$

$$\begin{aligned} \Delta w_{i,j}^{p,k+1} = & W \Delta w_{i,j}^{p,k} \\ & + C_1 \text{rand}_1()_{i,j}^{p,k} (pbest_{i,j}^{p,k} - w_{i,j}^{p,k}) \\ & + C_2 \text{rand}_2()_{i,j}^{p,k} (gbest_{i,j}^{p,k} - w_{i,j}^{p,k}) \\ & (i, j = 1, 2, \dots, M) \end{aligned} \quad (23)$$

ここで, k は学習回数, $\Delta \mathbf{w}^{p,k} = (\Delta w_{1,1}^{p,k}, \Delta w_{1,2}^{p,k}, \dots, \Delta w_{i,j}^{p,k}, \dots, \Delta w_{M,M}^{p,k})$ は解候補の変化量, $\text{rand}_1()_{i,j}^{p,k}$, $\text{rand}_2()_{i,j}^{p,k}$ は 0 から 1 の間に分布する一様乱数, W, C_1, C_2 は重み係数である. また, $\mathbf{pbest}^{p,k} = (pbest_{1,1}^{p,k}, pbest_{1,2}^{p,k}, \dots, pbest_{i,j}^{p,k}, \dots, pbest_{M,M}^{p,k})$ は Particle p が k 回目までに発見した最良解, $\mathbf{gbest}^k = (gbest_{1,1}^k, gbest_{1,2}^k, \dots, gbest_{i,j}^k, \dots, gbest_{M,M}^k)$ は群れ全体が k 回目までに発見した最良解であり, 次式で与えられる.

$$pbest_{i,j}^{p,k} = w_{i,j}^{p,k*} \quad (24)$$

$$gbest_{i,j}^k = pbest_{i,j}^{p*,k} \quad (25)$$

ここで,

$$k^* = \arg \min_{1 \leq k' \leq k} J(\mathbf{w}^{p,k'}) \quad (26)$$

$$p^* = \arg \min_{1 \leq p \leq P} J(\mathbf{pbest}^{p,k}) \quad (27)$$

とする.

4.4 学習アルゴリズム

学習法のアルゴリズムを以下に示す.

Step1:[初期化]

各 Particle の初期解候補 $\mathbf{w}^{p,0}$, 初期変化量 $\Delta \mathbf{w}^{p,0}$ をある

Fig. 4 Recurrent SNN with 5 SNs and SN_{input}

Fig. 5 Recurrent SNN with 10 SNs and SN_{input}

範囲内でランダムに与え, 目的関数値 $J(\mathbf{w}^{p,0})$ を求める. 目的関数値は後述する Step3 の方法で求める. $pbest_{i,j}^{p,k}$ と $gbest_{i,j}^k$ の初期値を以下のように求め, $k = 0$ とする.

$$\begin{aligned} pbest_{i,j}^{p,0} &= w_{i,j}^{p,0} \\ gbest_{i,j}^0 &= pbest_{i,j}^{p*,0} \end{aligned}$$

Step2:[速度と位置の更新]

$w_{i,j}^{p,k+1}$ と $\Delta w_{i,j}^{p,k+1}$ を (22), (23) 式で更新する.

Step3:[目的関数値の計算]

$\mathbf{w}^{p,k+1}$ を結合荷重とする SNN の微分方程式 (1)–(12) を解いて各 SN のスパイクの発生時刻 t_{i,k_i} と発生回数 K_i を求め, それらから各 Particle p の目的関数値 $J(\mathbf{w}^{p,k+1})$ を求める.

Step4:[$pbest_{i,j}^{p,k}$ と $gbest_{i,j}^k$ の更新]

$pbest_{i,j}^{p,k+1}$, $gbest_{i,j}^{k+1}$ を (24), (25) 式で更新する.

Step5:[終了判定]

$k = T_{max}$ ならば, 最適解を $\mathbf{gbest}^{k+1}$, 最適値を $J(\mathbf{gbest}^{k+1})$ として終了する.

さもなければ, $k = k + 1$ として Step2 へ戻る. ここで, T_{max} は総学習回数である.

5. 数値実験

提案する学習法の有効性を確認するために数値実験を行ない, 性能を評価する.

5.1 実験条件

学習問題には以下の 3 つの例題を用意した。

exA

5 個の SN_i ($i = 1, \dots, 5$) と、発火を促すための入力 $v_i(t)$ に用いる SN_{input} が結合して構成された **Fig. 4** に示すリカレント SNN を対象として、各ニューロンに **Table 1** に示すような回数 (K_i^d) でスパイクを発生させ、それぞれの時刻を t_{i,k_i}^d に一致させるように学習させる。

exB

exA と同じ構成のリカレント SNN を対象として、各ニューロンに **Table 2** に示すような回数 (K_i^d) および回数の上限値 (K_i^{du}) でスパイクを発生させ、発生時刻を指定したニューロンにはそれぞれの時刻を t_{i,k_i}^d に一致させるように学習させる。

exC

10 個の SN_i ($i = 1, \dots, 10$) と SN_{input} が結合して構成された **Fig. 5** に示すリカレント SNN を対象として、exA と同じ **Table 1** の条件で学習させる。

exA は従来の問題設定³⁾で用いていたスパイクの発生回数と発生時刻を学習させる問題である。exB は従来の学習問題に加えてスパイクの発生時刻は問題とせずに発生回数を一致させること、および発生回数の上限値がある望みの値とすることを新たに取り入れた学習問題である。exC は exA, B よりも規模の大きなモデルを対象にしている。

SNN モデルのパラメータはすべての例題で、 $c_i = 0.2$, $s_i = 1.0$ とし、 $g_{i,j}(s)$ は指数的に減衰する 2 つの関数の差がよく用いられることから $N = 2$ とし、 $A_{i,j} = \begin{pmatrix} -3.0 & 0.0 \\ 0.0 & -6.0 \end{pmatrix}$, $b_{i,j} = \begin{pmatrix} 1.0 \\ 1.0 \end{pmatrix}$, $c_{i,j} = (1.0, -1.0)$ とした。

PSO はさまざまな目的関数を設定できるため、つぎの 3 通りの目的関数を用いて結果を比較する。

$$\text{PSO1: } J = J_{o1} + \mu_a J_{a1} + \mu_b J_b + \mu_c J_c$$

$$\text{PSO2: } J = J_{o1} + \mu_a J_{a2} + \mu_b J_b + \mu_c J_c$$

$$\text{PSO3: } J = J_{o2} + \mu_a J_{a3} + \mu_b J_b + \mu_c J_c$$

また、目的関数のペナルティ項の重みは予備実験を元につぎのように与えた。

$$\text{PSO1: } \mu_a = \begin{cases} 5 & (K_i > K_i^d) \\ 10 & (K_i < K_i^d) \end{cases}, \mu_b = \mu_c = 1$$

$$\text{PSO2: } \mu_a = 4, \mu_b = \mu_c = 1$$

$$\text{PSO3: } \mu_a = 4, \mu_b = \mu_c = 1$$

PSO の解候補更新式のパラメータを $(W, C_1, C_2) = (0.7, 1.4, 1.4)$, Particle の数を $P = 100$ とし、学習回数を $T_{max} = 10000$ とし、解候補を T_{max} 回更新することを終了条件とした。各 Particle の初期解候補 $w_{i,j}^{p,0}$, 初期変化量 $\Delta w_{i,j}^{p,0}$ は $[-10, 10]$ の範囲でランダムに決定した。

従来の勾配法による学習には微分可能な目的関数 $J = J_{o1} + \mu_a J_{a1}$ を設定し、PSO1 と同じ μ_a とした。勾配法には共役勾配法を用い、勾配の導出には感度方程式を用いた³⁾。

Table 1 The desired spike train (exA, C)

(K_i^d : the desired number of spikes, t_{i,k_i}^d : desired firing instants)

SN_i	K_i^d	t_{i,k_i}^d
SN_1	3	1.0 1.5 2.0
SN_3	2	1.5 3.0

Table 2 The desired spike train (exB)

SN_i	K_i^d	K_i^{du}	t_{i,k_i}^d
SN_1	4	-	1.0 4.0 7.0 10.0
SN_2	4	-	not specified
SN_3	1	-	5.0
SN_4	-	6	not specified

Table 3 The realized best spike train among 20 learning trials (exA)

SN_i	K_i	t_{i,k_i}
SN_1	3	1.000020 1.500005 2.000015
SN_3	2	1.500005 2.999965

Table 4 The realized best spike train among 20 learning trials (exB)

SN_i	K_i	t_{i,k_i}
SN_1	4	1.000052 4.000008 6.999994 9.999995
SN_2	4	1.440334 4.731712 5.422345 9.385837
SN_3	1	5.000014
SN_4	6	3.741611 4.473035 6.739319 7.469212 9.545622 10.836106

Table 5 The realized best spike train among 20 learning trials (exC)

SN_i	K_i	t_{i,k_i}
SN_1	3	0.999983 1.500008 1.999995
SN_3	2	1.500008 3.000004

最大学習回数を 500 回とし、勾配のノルムが 10^{-20} 以下となることを終了条件とした。

5.2 実験結果

各例題に対して、乱数列を変更して PSO1, PSO2, PSO3 を 20 回ずつ実行した。各例題に対して計 60 回実行した結果の中で最もよい解でのスパイク発生時刻、発生回数を **Table 3**~**Table 5** に示す。ただし、**Table 3** は PSO3, **Table 4** は PSO2, **Table 5** は PSO2 を用いて実行した結果である。どの例題に対してもスパイクの発生回数に関する制約条件を満たしており、かつスパイクの発生時刻を指定したニューロンでは望みの時刻に近い時刻を得ることができていることがわかり、正しく学習が行われたといえる。

exA は微分可能な目的関数 $J = J_{o1} + \mu_a J_{a1}$ により従来の勾配法による学習が可能であり、ある初期値からの勾配法による学習前と学習後のスパイク発生時刻を **Table 6**, **Table 7** に示す。**Table 6** は大域的最適解に比較的近い解を初期値とし、**Table 7** は大域的最適解に比較的離れた解を初期値とした学習結果である。**Table 6** のように設定した初期解から望

Table 6 The spike trains before and after learning by the gradient based method
(for an initial solution near the optimum solution)

SN_i	before learning		
SN_1	0.999998	1.500029	1.999985
SN_3	1.499998	2.999993	
	after learning		
SN_1	1.000002	1.500000	2.000000
SN_3	1.500000	3.000000	

Table 7 The spike trains before and after learning by the gradient based method
(for an initial solution far from the optimum solution)

SN_i	before learning		
SN_1	0.670528	2.063770	2.365858
SN_3	0.436765	2.623721	
	after learning		
SN_1	0.672480	2.032816	2.325544
SN_3	0.438649	2.710548	

Table 8 Mean and variance of squared error between the desired and realized firing instants (exA)

PSO	mean	variance
PSO1	4.00×10^{-4}	3.09×10^{-6}
PSO2	5.52×10^{-4}	8.93×10^{-6}
PSO3	1.69×10^{-3}	5.33×10^{-5}

Table 9 Mean and variance of squared error between the desired and realized firing instants (exB)

PSO	mean	variance
PSO1	1.44×10^{-2}	3.66×10^{-3}
PSO2	2.77×10^{-3}	1.22×10^{-4}
PSO3	1.97×10^{-2}	6.56×10^{-3}

Table 10 Mean and variance of squared error between the desired and realized firing instants (exC)

PSO	mean	variance
PSO1	2.49×10^{-2}	3.25×10^{-3}
PSO2	3.77×10^{-2}	7.58×10^{-3}
PSO3	2.43×10^{-1}	3.51×10^{-1}

みの時刻に近い時刻を得られる場合もあれば、Table 7 のようにスパイクの発生回数は一致している初期解からでも、途中で局所的最適解に陥ってしまい、望みの時刻に近い時刻を得られない場合もある。一方で、提案する PSO による学習では、Table 3 に示されるように、初期解を複数用意しそれらをランダムに与えることによって大域的探索を実現し、大域的最適解を発見することが確認できた。exB は望みのスパイクの発生時刻と発生回数だけでなく、望みのスパイク発生回数、望みのスパイク発生回数の上限値を与えるニューロンが混在する、より一般的な問題となっており、Table 4 に示されるように、このような問題に対しても PSO によって正しく学習が行なわれることが確認できた。exC はより規模の大きな SNN を扱ったが、Table 5 に示されるように、exA, exB の SNN と同様に正しく学習が行なわれることが確認で

Fig. 6 The number of particles satisfying constraints (13) (exA)

Fig. 7 The number of particles satisfying constraints (13) (exB)

Fig. 8 The number of particles satisfying constraints (13) (exC)

きた。

また、どの程度平均的に精度のよい学習が行なわれるかを確認するために、学習後に得られた解でのスパイク発生時刻から算出した 20 試行分の 2 乗誤差の平均と 2 乗誤差の分散を Table 8~Table 10 に示す。なお、スパイクの発生回数に関する制約条件はどの学習の実行結果においても満たされ

ていた。exA は Table 8 を見ると、2 乗誤差の平均、分散が共に小さく、平均的に精度のよい学習を行なうことができたといえる。exB, exC は Table 9, Table 10 を見ると、exA に比べて 2 乗誤差の平均、分散が共に大きくなっているものの、平均的にある程度精度のよい学習を行なうことができたといえる。これは exB, exC は制約条件が多い、あるいはニューロン数が多いので問題が複雑になるからである。各例題において、PSO1~PSO3 の結果を比較すると、PSO1, PSO2 に比べると PSO3 の 2 乗誤差の平均、分散が大きいが、どの PSO も平均的に精度のよい学習を行なうことができた。

最後に、各 Particle がどの程度制約条件を満たしながら最適解を探索しているのかに着目し、PSO1~PSO3 を用いた学習において、Particle 100 個のうち (13) 式の制約条件を満たす Particle の個数の推移を Fig. 6~Fig. 8 に示す。ただし、20 試行分の平均値を取っている。Fig. 6~Fig. 8 を見ると、学習を重ねるごとに制約条件を満たす Particle の個数が増えている。また、Fig. 6 に示すように、exA では PSO1 が PSO2 や PSO3 に比べて多かった。一方、Fig. 7 や Fig. 8 に示すように、exB, exC では PSO2 や PSO3 が PSO1 に比べて多かった。これらの結果から、exA のような簡単な問題ならば、時刻の誤差をペナルティとする関数 J_{a1} を取り入れた PSO1 がスパイクの発生回数に関する制約条件を満たしやすいといえる。一方で、exB や exC のように複雑な問題に対しては、回数の誤差をペナルティとする関数 J_{a2} , J_{a3} を取り入れた PSO2, PSO3 のほうがスパイクの発生回数に関する制約条件を満たしやすいという傾向があるといえる。

以上の数値実験により、exA, exB, exC どの例題に対しても、今回用意した 3 通りの目的関数 PSO1, PSO2, PSO3 を用いると、どの関数でも平均的に精度よく学習できることが確認できた。

6. おわりに

本論文では、PSO を学習に用いた、大域的最適解を得るためのリカレント SNN の学習法を提案した。この方法は目的関数の勾配を必要としないので、従来の学習法で扱っていた、スパイクの発生時刻を一致させる学習問題に加えて、スパイクの発生時刻は問題とせずに発生回数を一致させること、および発生回数の上限値をある望みの値とすることを新たに取り入れた学習問題を設定することができた。3 種類の目的関数を設定する学習法を提案し、いくつかの例題を用いた数値実験を通して性能を評価した結果、スパイクの発生回数に関する制約条件を満たし、発生時刻を指定したニューロンでは望みの時刻に近い時刻でスパイクを発生する解が得られ、平均的に精度のよい学習を行なうことができることを確認した。これにより、提案した 3 種類の目的関数を設定する学習法を用いると、SNN の種々の現実の学習問題に対してよい解を発見できることが期待される。SNN の学習問題においてはスパイク列の発生頻度が問題になることがあるので、スパイクの発生時刻だけでなく、発生頻度を実現した本研究により SNN

の応用が広がるものと期待される。

参考文献

- 1) W. Maass and C. Bishop (Eds.): Pulsed Neural Networks, MIT Press (1998)
- 2) W. Gerstner, R. Ritz and J.L. van Hemmen: Why spikes? Hebbian learning and retrieval of time-resolved excitation patterns, *Biological Cybernetics*, **69**, 503/515 (1993)
- 3) K. Selvaratnam, Y. Kuroe and T. Mori: Learning Methods of Recurrent Spiking Neural Networks — Transient and Oscillatory Spike Trains, システム制御情報学会論文誌, **13-3**, 95/104 (2000)
- 4) S.M. Bohte, J.N. Kok and H.L. Poutre: Error-Backpropagation in Temporally Encoded Networks of Spiking Neurons, *Neurocomputing*, **48-1-4**, 17/37 (2002)
- 5) J. Kennedy and R.C. Eberhart: Particle swarm optimization, *Proc. of IEEE International Conference on Neural Networks*, 1942/1948 (1995)
- 6) 木村, 黒江, 小寺, 村田: スパイクニューラルネットワークを用いた視覚系の特徴抽出モデルの構築, 計測自動制御学会論文集, **45-11**, 612/619 (2009)
- 7) 黒江康明: スパイクニューラルネットワーク—学習法を中心として—, システム/制御/情報, **48-2**, 57/62 (2004)

[著者紹介]

山本 昌弘 (学生会員)

2009 年京都工芸繊維大学工学部電子情報工学科卒業。同年京都工芸繊維大学大学院工学科学研究科情報工学専攻入学、現在に至る。ニューラルネットワークに関する研究に従事。

黒江 康明 (正会員)

1982 年神戸大学大学院自然科学研究科博士課程修了。同年神戸大学工学部電気工学科助手、91 年京都工芸繊維大学工学部電子情報工学科助教授、現在、同大学大学院工学科学研究科情報工学部門教授。96 年マサチューセッツ工科大学客員研究員、ニューロコンピューティングと知能情報処理、システム制御理論とその応用、計算機援用解析と設計などの研究に従事。学術博士。IEEE、システム制御情報学会、電気学会、電子情報通信学会、日本ロボット学会、日本応用数理学会などの会員。

飯間 等 (正会員)

1993 年京都工芸繊維大学大学院工学科学研究科電子情報工学専攻博士前期課程修了、95 年同大学院同研究科情報・生産科学専攻博士後期課程中途退学、同年京都工芸繊維大学工学部助手、2007 年同大学大学院工学科学研究科助教授、2008 年同大学大学院工学科学研究科准教授、現在に至る。博士 (工学)。最適化および強化学習に関する研究に従事。システム制御情報学会、電気学会などの会員。