

修士論文

SpikeProp ネットワークに対し
入出力関係を滑らかにする
ウェイト・ディケイ導入効果の検討

平成 23 年度

三重大学大学院 地域イノベーション学研究科

博士前期課程 地域イノベーション学専攻

吉川 雄也

目次

Abstract	1
第1章 はじめに	2
第2章 SpikeProp	4
2.1 スパイクングニューロン	4
2.2 SpikeProp の構造	6
2.3 SpikeProp の学習	6
2.4 SpikeProp の入出力関係の問題	8
第3章 提案	10
3.1 不連続な入出力関係の要因	10
3.2 改善手段	12
3.3 ウェイト・ディケイ (WD)	12
第4章 WD 導入の実験	14
4.1 実験条件	14
4.1.1 対象教師パターン	14
4.1.2 ネットワーク構造	15
4.2 入出力関係の調査	15
4.2.1 入出力空間全体における入出力関係の調査	15
4.2.2 学習した点近傍における入出力関係の調査	19
4.3 結合荷重と内部電位の調査	22
4.3.1 結合荷重の調査	22
4.3.2 内部電位の調査	26
4.3.3 考察	29
第5章 まとめ	30

謝辭	31
参考文献	32
発表論文	35

目次

2.1	スパイク応答関数の形状	5
2.2	スパイクニューロン	5
2.3	SpikeProp のネットワーク構造	6
2.4	SpikeProp での誤差	7
2.5	SpikeProp に XOR 問題を学習させた後の入出力関係の一例	9
2.6	SpikeProp に XOR 問題を学習させた後の理想の入出力関係の一例	9
3.1	不連続な出力変化の要因	11
3.2	学習後のユニットの内部電位の変化	11
3.3	不連続な出力変化を改善するための手段	12
4.1	SpikeProp に XOR 問題を学習させた後の入出力関係 (1)	16
4.2	SpikeProp に XOR 問題を学習させた後の入出力関係 (2)	17
4.3	SpikeProp に XOR 問題を学習させた後の入出力関係 (3)	18
4.4	検査領域	20
4.5	結合荷重の大きさの分布 (1)	23
4.6	結合荷重の大きさの分布 (2)	23
4.7	結合荷重の大きさの分布 (3)	24
4.8	図 4.5 拡大	24
4.9	図 4.6 拡大	25
4.10	図 4.7 拡大	25
4.11	誤出力をした時の学習後のユニットの内部電位の変化	27
4.12	図 4.11 拡大	27
4.13	誤出力をしなかった時の学習後のユニットの内部電位の変化	28
4.14	図 4.13 拡大	28

表 目 次

4.1	時間版 XOR 問題	14
4.2	評価の条件	19
4.3	正常なネットワーク数	21
4.4	パターン毎の正領域数	21
4.5	学習成功率と正領域率	21

Effectiveness of Weight Decay to improve discontinuous output changes in SpikeProp

2011 Yuya Yoshikawa

Abstract

Recently spiking neural networks (SNNs) attract many research attentions in the robot control field. SNNs represent information by spikes. They are good at temporally signal processing such as robot control.

We focus on SpikeProp, which is a model of SNNs. It represents information by timing of spikes. Our final goal is to apply SpikeProp to robot control. SpikeProp is based on complex model, which is an integrate-firing model. It causes complex behaviors: outputs of SpikeProp is often sensitive to weights. It implies that the outputs are also sensitive to inputs. These sensitivities cause unexpected output in a noisy environment. So, our purpose is to reduce discontinuous output changes against a continuous input changes.

To reduce discontinuous output changes, we introduce Weight Decay (WD), which improve input-output characteristic of conventional neural networks, which are constructed with non-spiking neuron such as sigmoidal neurons. WD exactly works to eliminating useless connections by reducing both the output error and a sum of squared weights. We assume that discontinuous output changes are caused by local maxima of activities. Since an activity is a sum of each response for all input stimuli, eliminating useless connections would reduce local maxima of activities. As a result, SpikeProp would resolve the discontinuity problem.

We show the effectiveness of our method by some simple experiments: training the XOR-problem that is a famous benchmark problem. Without WD, SpikeProp show discontinuous output changes even near trained input patterns. By applying WD, discontinuous output changes are reduced around the trained input patterns. We show that the WD works well not only for conventional neural networks but also SpikeProp.

In future, more improvements are required. In particular, it's important to lessen weights which are related to local maxima of activity that degrade input-output characteristics. Improved method would reduce discontinuous output changes. Finally, we can apply SpikeProp to real problems such as robot control.

第1章 はじめに

近年，生体の脳の情報処理機構を模倣したスパイクニューラルネットワークの研究が盛んに行われている [1] [2] [3]．また，介護ロボットや掃除ロボットなど，人間と共生する情報システムとして自律ロボットの研究開発が進められている [4]．スパイクニューラルネットワークでは，時間情報をエンコードせずに使うことができ，時間情報の扱いにたけている [5] といえる．そのため，ロボット制御などの分野で注目を集めている．具体的には，入力にセンサー信号を用いて自律ロボットのコントローラを構築する応用研究 [6] [7] [8] が行われている．また，音声信号処理の分野においても注目されており，ネットワークの入力に音声信号を用いて到達する音声信号のずれから音源の位置の特定を行う応用研究 [9] が行われている．現在，最も広く用いられている従来のシグモイド型ニューラルネットワークではユニット間の情報伝達にアナログ値を用いている．しかし，生体の脳の神経回路網では，ユニット間の情報表現にスパイクを用いている．スパイクニューラルネットワークでは，生体の脳と同様に，スパイク (パルス) を用いていることが，従来のシグモイド型のニューラルネットワークとの大きな違いであり，生体に近いニューラルネットワークとして期待されている．

このスパイクニューラルネットワークには，構成ユニットのモデル化手法，ネットワークの構成方法，学習方法の違いにより，さまざまなモデルが存在する．また，スパイクニューラルネットワークを構成するユニット間でやりとりする情報表現には，スパイクの発火密度を用いる表現方法と，スパイクの発火タイミングを用いる表現方法がある．スパイクの発火密度を用いた代表的なモデルとしてはSTDP(Spike Timing Dependent Plasticity) と呼ばれるモデルが挙げられる [10]．スパイクニューラルネットワークに関する研究の多くが，このSTDPに関する研究であり，スパイクの発火密度を扱うモデルが，研究の主流を占めているのが現状である．しかし，ロボット制御等ではタイミングの情報が重要であるため，情報表現にスパイクの発火タイミングを用いるモデルの需要がないということはない．そこで，本研究では，情報表現にスパイクの発火タイミングを用いる代表的なモデルである SpikeProp [11] [12] に着目する．

SpikeProp は，Bohte らが提案したスパイクニューラルネットワークのモデルの一

つであり，教師あり学習法の階層型ニューラルネットワークである．SpikeProp では，スパイクの発火タイミングにより情報を表現し，スパイクニューロンを用いた特殊な結合のネットワーク構造を持つ．このモデルは，誤差逆伝搬法 [13] に基づいて，ネットワークの出力スパイクの時刻を学習により調整する．この学習法についてはさまざまな改良法が提案されている [14] [15] ．

それに対して，その入出力関係に関する検討は，未だ不十分である．実際，SpikeProp のネットワークは，その結合荷重の微小な変化に対して，その出力が滑らかに変化しないという問題が指摘されている [16] ．入力の微小な変化に対しても同様の現象が発生することが予想されるが，それに関する検討はほとんどされていない．このような現象が発生すると，ノイズ等に起因する入力の微小な変化に対して，ネットワーク出力が不連続に変化してしまうことを意味する．

そこで，本研究では，SpikeProp ネットワークにおいて，その入出力関係について調査し，その出力の入力変化に対する不連続な変化を解消する手法について検討する．具体的には，従来のシグモイド型ニューラルネットワークにおいてその入出力関係を改善するために有効な手法であるウェイト・ディケイ (WD: Weight Decay) を，SpikeProp の学習に適用した際の効果を検討する．そして，将来的には，SpikeProp を自律ロボットに応用することを目的とする．

本論文の構成を以下に示す．第 2 章で本研究で扱う SpikeProp について述べ，第 3 章では，第 2 章で示した問題を解決するための提案について述べる．そして，第 4 章では，提案手法について実験を行い，最後に第 5 章で本研究をまとめる．

第2章 SpikeProp

SpikeProp とは、Bohte らが提案したスパイクニューラルネットワークのモデルの一つであり、スパイクニューロンを用いた特殊なネットワーク構造をもつ。2.1 節でスパイクニューロンについて述べ、2.2 節で SpikeProp の構造について、2.3 節で SpikeProp の学習について、2.4 節では SpikeProp の入出力関係の問題について述べる。

2.1 スパイクニューロン

スパイクニューロン（以下、ユニット）とは脳の神経細胞をモデル化したものである。ユニット o がシナプス前ユニット h からのスパイクを d だけ時間遅れしたうえで受け取る場合を考える。このとき、ユニット o の内部電位 $x_o(t)$ は式 (2.1) で定義されるスパイク応答関数 $\varepsilon(t)$ に結合荷重 w_{ho} を乗じた分だけ変化する。この関係は式 (2.2) で表される。

$$\varepsilon(t) = \begin{cases} 0 & (t < 0) \\ \frac{t}{\tau} e^{1-t/\tau} & (t \geq 0) \end{cases} \quad (2.1)$$

$$x_o(t) = \sum_{h \in \Gamma_o} w_{ho} \varepsilon(t - t_h - d) \quad (2.2)$$

ここで τ は時定数、 Γ_o はユニット o の前ユニット h の集合、 t_h はシナプス前ユニット h の発火時刻である。本研究で扱う SpikeProp では、ユニットは一度しか発火しないという特性を持つ。図 2.2 中、スパイクが①、②、③と順にユニットに伝達した時、伝達するごとに式 (2.2) に応じて内部電位 $x_o(t)$ が変化する。そして内部電位 $x_o(t)$ が最初に出力しきい値 θ を超えたとき ($t=A$) にユニットは発火し、この時刻を出力時刻とする。ただし、SpikeProp では、ユニットは一度しか発火しないため、 $t=B$ では発火しない。

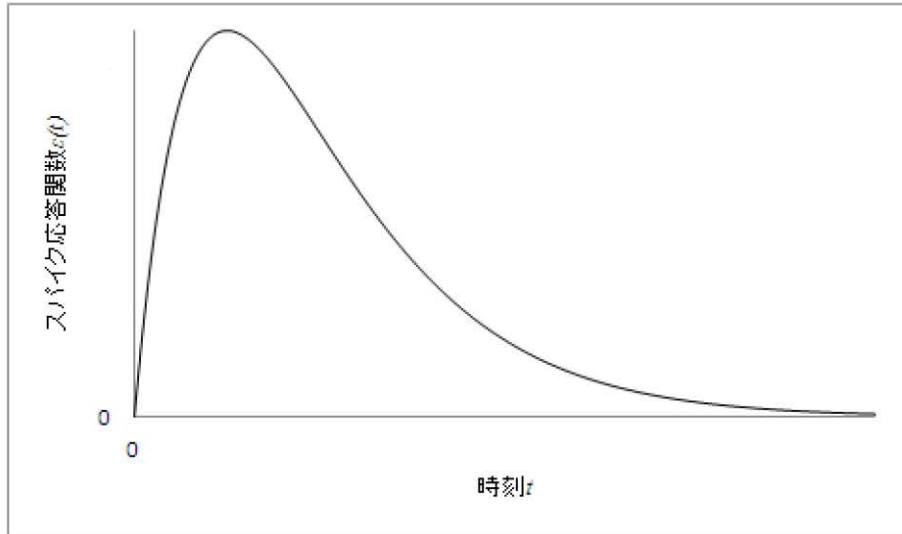
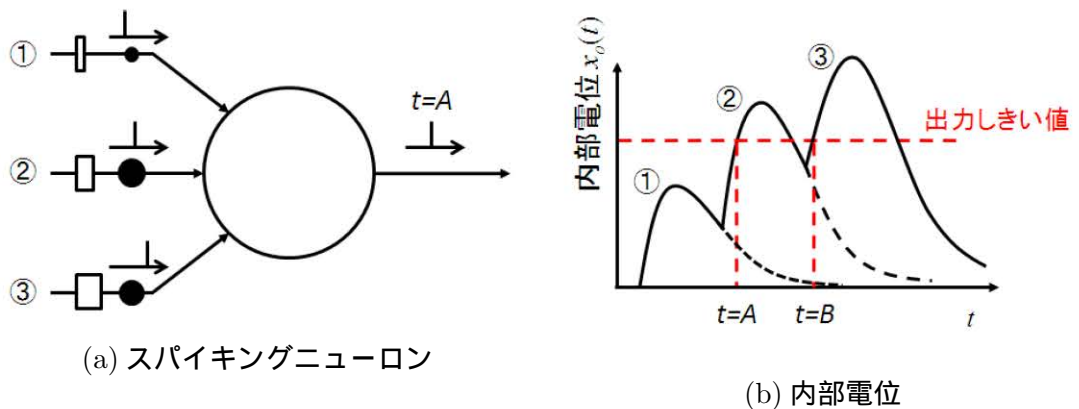


図 2.1: スパイク応答関数の形状



(a) スパイクングニューロン

(b) 内部電位

図 2.2: スパイクングニューロン

- (a): 各スパイクングニューロンは、多入力 1 出力の素子である。各入力信号は、時間遅れ () と結合荷重 (), 2 つのパラメータにより変換される。
- (b): (a) のように, ①, ②, ③ と順に, スパイクがユニットに伝達された場合の, ユニットの時間変化に対する内部電位の変化を表している。

2.2 SpikeProp の構造

SpikeProp では、図 2.3 に示すようなユニット間に特殊な結合を持つ階層型ネットワークを使用する．この結合は、1 本の主結合が m 本の副結合で構成されている．この副結合には、それぞれ、決まった時間遅れ d^k と、可変の結合荷重 w_{ho}^k が定められている．時間遅れには一定間隔 $(1, 2, 3, \dots, m)$ のような定数を用い、すべての主結合において同じ値を用いる．そのため、結合荷重を調整するだけで出力時刻の調整ができるようになり学習を容易に行うことができる．ユニット o の内部電位 $x_o(t)$ は式 (2.3) となる．

$$x_o(t) = \sum_{h \in \Gamma_o} \sum_{k=1}^m w_{ho}^k \varepsilon(t - t_h - d^k) \quad (2.3)$$

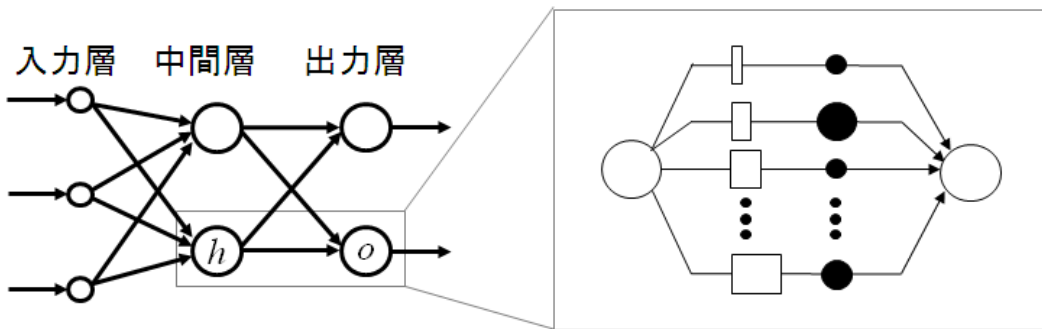


図 2.3: SpikeProp のネットワーク構造

SpikeProp は階層型ニューラルネットワークの構造をしている．ユニット間は、複数の結合で構成されており、図 2.2 と同様に、結合それぞれに、時間遅れと結合荷重が定められている．

2.3 SpikeProp の学習

SpikeProp の学習では、ネットワークの出力ユニットが所望の時刻で発火するように結合荷重を調整する．この際、誤差が最小になるように勾配を用いた降下法 (誤差逆伝搬法) による反復計算を行う．誤差逆伝搬法では誤差関数を定義し、その誤差関数の値が減るように結合荷重を更新する．結合荷重の更新には勾配を用い、勾配の方向と逆向きに少しずつ結合荷重を調整し繰り返すことで誤差を最小化する．この誤差関数には二乗誤差関数を用いる．実際の出力時刻を t_o^p 、希望する出力時刻を \hat{t}_o^p とすると、誤差関数は式 (2.4) で定義される (図 2.4) ．

$$E = \frac{1}{2} \sum_{p \in P} \sum_{o \in O} (t_o^p - \hat{t}_o^p)^2 \quad (2.4)$$

ここで、 O は出力層ユニットの集合、 P は学習パターンの集合を意味する。また、 t_o^p 、 \hat{t}_o^p は、パターン p におけるユニット o の実際の出力時刻および理想の出力時刻をそれぞれ表す。

この誤差に対して勾配 ∇E を求め、式 (2.5) にしたがって、勾配と逆方向に少しずつ結合荷重を繰り返すことで学習する。

$$\mathbf{w}(n) \leftarrow \mathbf{w}(n-1) - \eta \nabla E \quad (2.5)$$

ここで、 η は学習率であり、正の定数とする。また、 $\mathbf{w}(n)$ は n 回目の更新後の結合荷重とする。

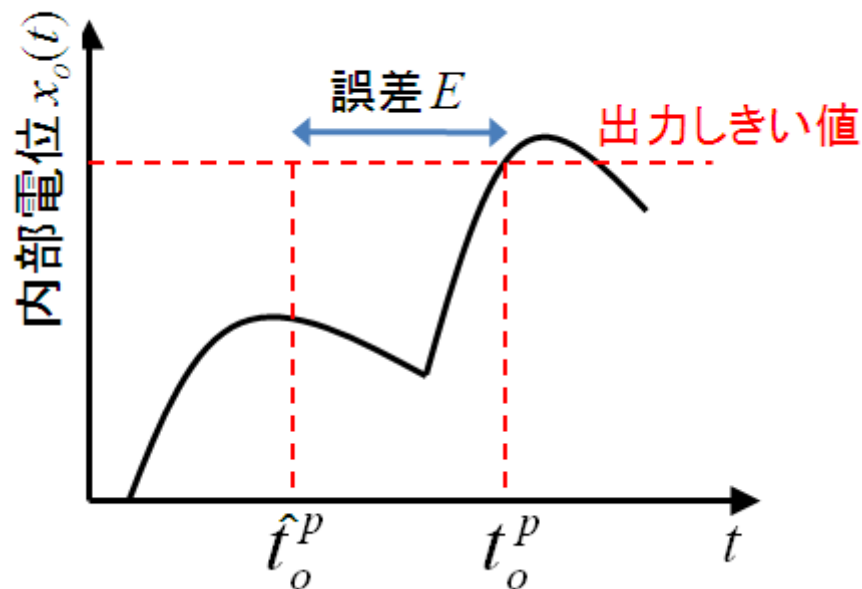


図 2.4: SpikeProp での誤差

t_o^p は実際の出力時刻、 \hat{t}_o^p は希望する出力時刻を表している。これらの時刻の差を、誤差と定義する。

2.4 SpikeProp の入出力関係の問題

SpikeProp はその学習過程で、結合荷重がわずかに変化しただけであるにもかかわらず、ネットワークの出力が大きく変化してしまうことが、指摘されている (3.1 節参照) [16] . ニューラルネットワークにとっては、入力の変化も結合荷重の変化も、パラメータの変化という意味で同じと見なせる . そのため、入力が変わわずかに変化しただけで、ネットワークの出力が大きく変化してしまうことが予想できる .

図 2.5 は、実際に SpikeProp に XOR 問題 (4.1.1 節) を学習させた後の入出力関係の一例である . 学習した 4 点はきちんと学習できている (教師時刻を出力している) もの、それらの間では入力の連続的な変化に対して出力が連続的に変化していない部分がある . 例えば、(1,7) では、教師時刻である 11 を出力しているが、近傍に 17 を出力している部分がある . また、図 2.6 は、入力の連続的な変化に対して、出力が連続的に変化している図の一例である . 図 2.6 のように滑らかな入出力関係をしていないため、このままでは、入力の微小な変化に対して出力が大幅に変化してしまう可能性があり、ネットワークとして好ましくないといえる .

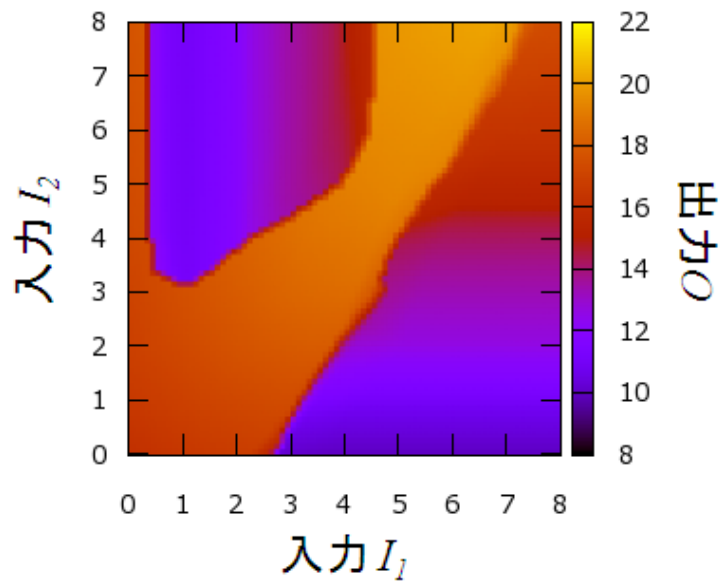


図 2.5: SpikeProp に XOR 問題を学習させた後の入出力関係の一例

横軸は入力 1, 縦軸は入力 2 である. そして, 出力を色の分布で表している. (入力 1, 入力 2)=(1, 1), (1, 7), (7, 1), (7, 7) が学習した点であり, 17, 11, 11, 17 が各点に対する教師時刻である.

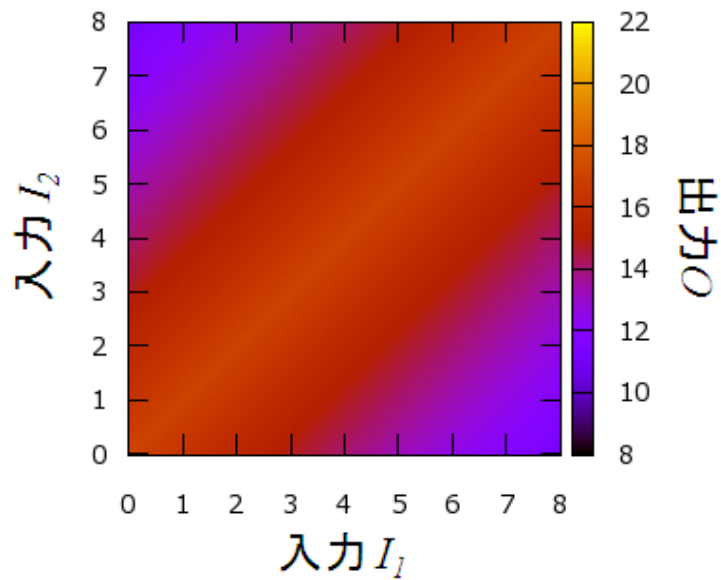


図 2.6: SpikeProp に XOR 問題を学習させた後の理想の入出力関係の一例

第3章 提案

本章では，2.4節で示した問題を引き起こしてしまう要因について述べる．そして，その問題を解決するために，従来のシグモイド型のニューラルネットワークにおいて入出力関係を改善するために有効であるウェイト・ディケイを，SpikeProp に適用する．

3.1 不連続な入出力関係の要因

文献 [16] では，結合荷重の変化に対する不連続な出力変化の要因が報告されている．それらによると，結合荷重の変化に対する不連続な出力変化の要因は，SpikeProp のユニットの発火方法に起因する．ユニットは，式 (2.3) にしたがって，スパイク応答関数に結合荷重を乗じた分だけ内部電位を変化させる．スパイク応答関数は，図 2.1 のように極大を持つ変化をしている．また，SpikeProp には，内部電位を減少させる抑制性シナプス結合がある．これらにより，内部電位に極大が生じる．内部電位に極大が生じると，内部電位がいったん極大値となり再びその極大値を超えるまでの時間では，ユニットは発火できない．そのため，微小な結合荷重の変化に対して，内部電位の形状はわずかにしか変化していないにもかかわらず，出力時刻が大幅に変化する (図 3.1)．これが，結合荷重の変化に対する不連続な出力変化の要因である．

また，入力の場合においても，結合荷重の場合と同様の要因が考えられる．2.4節で述べたように，ニューラルネットワークにとっては，入力の変化も結合荷重の変化も，パラメータの変化という意味で同じと見なせる．式 (2.3) より，結合荷重の場合，内部電位の形状は縦方向に変化するが，入力の場合は，横方向 (時刻方向) に変化する．結合荷重と入力では，内部電位の形状の変化の仕方が異なっているが，入力が少し変化する場合も，内部電位の形状が少し変化することが分かる．よって，近い入力値であったとしても，出力時刻が大幅に変化してしまい，不連続な入出力関係となってしまうのではないか．実際に SpikeProp を学習させた後の内部電位を観測すると，図 3.2 のような変化が見られる (4.3.2 節参照)．

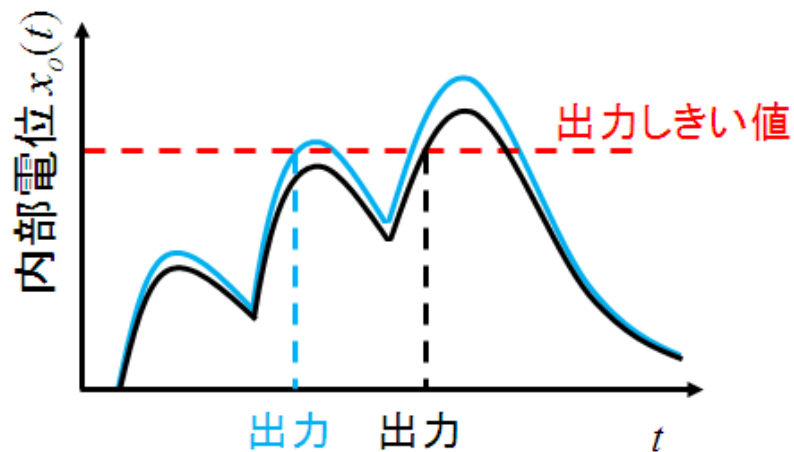


図 3.1: 不連続な出力変化の要因

SpikeProp の不連続な出力変化の要因の図である．横軸が時刻，縦軸がユニットの内部電位の大きさを表している．黒線と青線は，結合荷重が微小に変化した時の内部電位の形状の変化を表している．内部電位の形状は似ているが，内部電位の 2 つ目の極大が出力しきい値を，超えるか超えないかで出力時刻が大幅に変化してしまう．

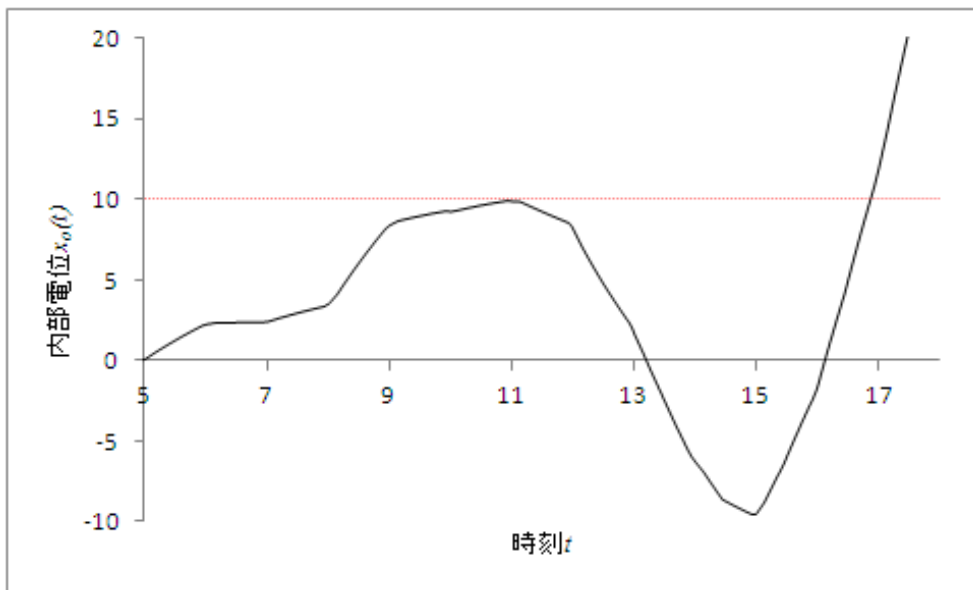


図 3.2: 学習後のユニットの内部電位の変化

XOR 問題を学習した SpikeProp ネットワークに，入力 (1, 1) を入力した時の，出力層ユニットの内部電位の変化である．横軸は時刻，縦軸は内部電位の大きさを，また，赤の点線は出力しきい値を示す．

3.2 改善手段

3.1 節により, 2.4 節で示した SpikeProp の入出力関係の問題が, SpikeProp のユニットの発火方法に起因するのではないかと考えられる. 出力しきい値付近での内部電位の極大が原因で, 不連続な出力変化を引き起こすのなら, 出力しきい値付近での内部電位の極大を抑制することで, 問題を解決できると考える.

極大を抑制するためには, 結合荷重の値を 0 にすることが必要である. 式 (2.3) より, 内部電位は多数のスパイク応答関数と結合荷重の積の重ね合わせである. SpikeProp において, スパイク応答関数は必要不可欠なものであり, 失くすことができない. そのため, 学習するうえで必要でない結合荷重の値を 0 にすることで, 内部電位の極大を抑制する. 結合荷重の値が 0 になると, その結合荷重に対応する内部電位の変化がなくなる. その結果, 内部電位の形状が, 図 3.3 のようにシンプルなものになり不連続な出力変化を抑制することができるのではないかと考える.

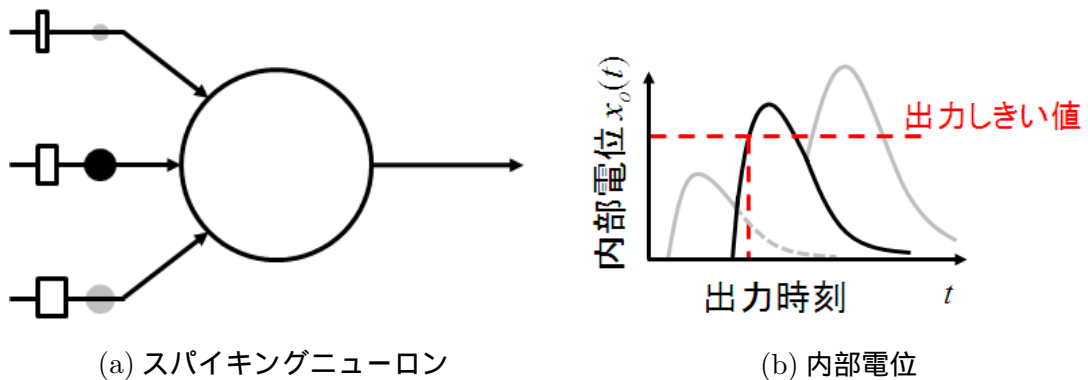


図 3.3: 不連続な出力変化を改善するための手段

(a): 図 2.2 において, 学習に不要な結合荷重を 0 にした場合のスパイクングニューロンである.

(b): (a) のように, 学習に不要な結合荷重を 0 にすると, その結合荷重に対応する内部電位の変化がなくなる.

3.3 ウェイト・ディケイ (WD)

3.2 節で説明した改善手段に, 本研究では, WD [17] [18] [19] という手法を利用する.

従来のシグモイド型のニューラルネットワークでは, 過学習などに起因する不適切な入出力関係を改善するための手法として, WD をその学習に適用する方法がしばしば用いられる. この手法は, 誤差逆伝搬法で学習を行う際に, 式 (2.5) ではなく, 式 (3.1) を結合荷

重の更新式として用いる方法である．

$$\mathbf{w}(n) \leftarrow \mathbf{w}(n-1) - \eta \nabla E' \quad (3.1)$$

これは，学習の評価関数に式 (2.4) の E ではなく，ペナルティ項を付加した式 (3.2) の E' を用いることを意味している．

$$E' = E + \beta \sum_w |w^n| \quad (3.2)$$

この E' を最小化することで，出力の誤差を小さくするとともに，誤差を減少させるために必要でない結合を除くことができる．

今回，実験では，式 (3.2) において， $n = 2$ または， $n = 1$ のペナルティ項を使用する． $n = 2$ の場合，式 (3.1) は，式 (3.3) のようになる．

$$\mathbf{w}(n) \leftarrow \mathbf{w}(n-1) - \eta(\nabla E + 2\beta\mathbf{w}) \quad (3.3)$$

$n = 1$ の場合，式 (3.1) は，式 (3.4) のようになる．

$$\mathbf{w}(n) \leftarrow \begin{cases} \mathbf{w}(n-1) - \eta(\nabla E + \beta) & (\mathbf{w} > 0) \\ \mathbf{w}(n-1) - \eta(\nabla E - \beta) & (\mathbf{w} < 0) \\ \mathbf{w}(n-1) - \eta(\nabla E) & (\mathbf{w} = 0) \end{cases} \quad (3.4)$$

$n = 2$ のペナルティ項 [17] は，WD の中で最も一般的なものである．また， $n = 2$ よりも $n = 1$ の方が，WD の効果が高い (重みを減衰させる働きが強い)， $n = 1$ では， $n = 2$ と違い， w が 0 に近づいても忘却量 (重みを減衰させる量) が減少せず， w が大きくなっても忘却量が増大せず，良い働きが得られるとされている [18]．しかし， β の値が過大であれば非零であるべき結合荷重も消滅し，学習到達度が大幅に劣化する．逆に β が過小であれば多くの結合荷重が生き残り，忘却量が過小になる恐れがあるため，適切な β の値を選ばなければ，良い効果が得られなくなる．そのため，次章で，比較の意味合いも込め， $n = 2$ 及び， $n = 1$ のペナルティ項を使用し，WD 導入により得られる効果について検証する．

第4章 WD 導入の実験

この章では、実際に WD を SpikeProp に導入し、その効果について調査する。具体的には、入出力、内部電位、結合荷重がどのように変化するか調査する。

4.1 実験条件

本節では、実験で扱う時間版 XOR 問題とネットワーク構造について説明する。

4.1.1 対象教師パターン

SpikeProp を提案した Bohte らの論文 [11] は、対象教師パターンに時間版 XOR 問題 (以下、XOR 問題) を使用した。そのため、本研究においても Bohte らと同様に XOR 問題を学習対象とした。XOR 問題とは、線形分離不可能なパターン分類問題の中で、最も簡単な問題である。多くの複雑な問題は、XOR 問題を部分問題として内包するとされていることから、学習対象としてよく用いられている。SpikeProp では、シグモイド型のニューラルネットワークと違い、ユニットの入出力にスパイクの発火タイミングを用いるため、XOR 問題を時間情報でコーディングした時間版 XOR 問題を使用した。本研究で扱う XOR 問題は、実験の都合上、Bohte らの実験で使用された XOR 問題の入出力信号での時刻をそれぞれ 1 増やしたものを使用した。それぞれの入力層ユニットが同時にスパイクを出力した場合は出力層ユニットは時刻 17 にスパイクを出力し、入力層ユニットが違う時刻にスパイクを入力した場合は出力層ユニットは時刻 11 にスパイクを出力する (表 4.1)。

表 4.1: 時間版 XOR 問題

入力時刻 1	入力時刻 2		出力時刻
1	1	→	17
1	7	→	11
7	1	→	11
7	7	→	17

4.1.2 ネットワーク構造

実験は、基本的に SpikeProp を提案した Bohte らの実験 [11] に準じた条件で行った。主要な条件を以下に示す。

- ネットワーク構造は、入力層ユニットが3つ(1つはバイアス)、中間層ユニットが5つ、出力層ユニットが1つの階層型とする(入力層のバイアスとは、時刻0で発火し、時刻の基準となるものである)。
- 副結合は、各種結合に対して16本とし、時間遅れは $1, 2, \dots, 16$ とした。
- 抑制性ユニットは設けず、結合荷重は正負ともに扱った。
- 結合荷重の初期化値は-2から8の一様乱数を用いた。
- スパイク応答関数の時定数 $\tau = 7$ 。
- ユニットの出力しきい値 $\theta = 10.0$ (ユニットが発火しなかった時は、自動的に出力しきい値 θ は下がる)。
- 学習率 $\eta = 1.0 \times 10^{-2}$ 。

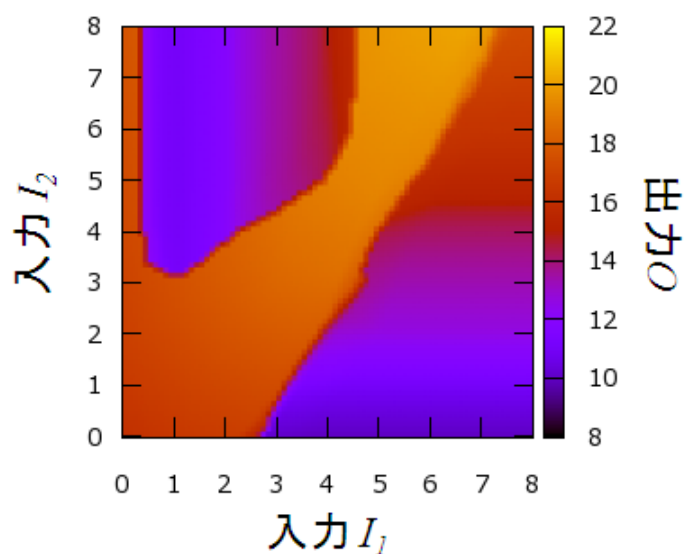
4.2 入出力関係の調査

本節では、WDの効果について確認するため、入出力関係について調査する。

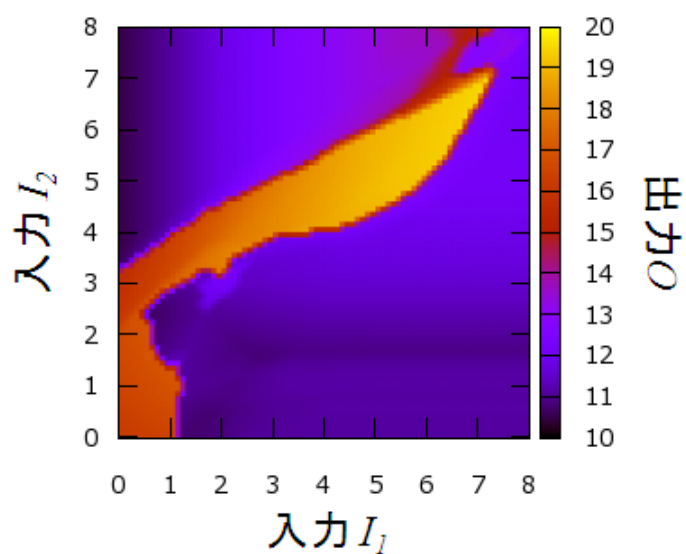
4.2.1 入出力空間全体における入出力関係の調査

SpikeProp に XOR 問題を学習させた後の入出力関係について、WD 導入の有無でどのように変化するか調査した。WD なしの場合はオリジナルの SpikeProp で学習した結果を、それ以外の場合は WD を適用した結果を意味する。なお、さまざまな β を試した結果、 $n = 2$ の場合は $\beta = 1.0 \times 10^{-2}$ 、 $n = 1$ の場合は $\beta = 5.0 \times 10^{-2}$ の時に、最も WD の効果が高かった。よって、以下の結果はこれらの β の値を用いた場合のものを示す。図 4.1 は異なる初期荷重から学習した WD なしの入出力の図である。図 4.2 は WD あり ($n = 2$)、図 4.3 は WD あり ($n = 1$) の図で、それぞれ2つの図は異なる初期荷重から学習した図である。これらの図を比較すると、不連続な出力変化は WD を導入しても存在しているが、WD なし、WD あり ($n = 2$)、WD あり ($n = 1$) の順に出力 11 付近と 17 付近の境目のラ

インがシンプルになった．つまり入出力関係がシンプルになっており，出力が不連続に変化してしまう可能性が減ったといえる．



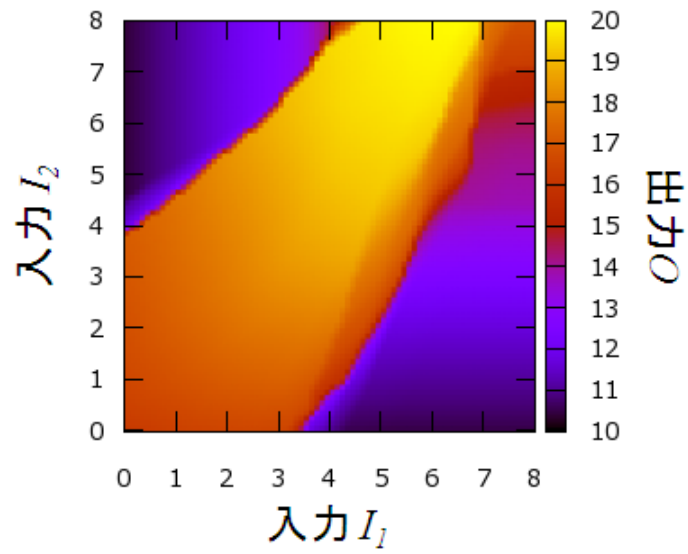
(a)



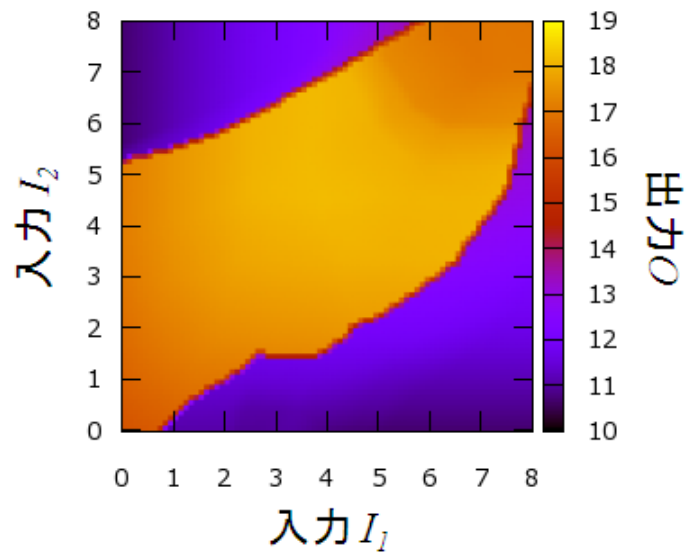
(b)

図 4.1: SpikeProp に XOR 問題を学習させた後の入出力関係 (1)

WD なし．オリジナルの SpikeProp に XOR 問題を学習させた後の入出力関係．(a) と (b) は，異なる初期荷重から学習したネットワークである．



(a)

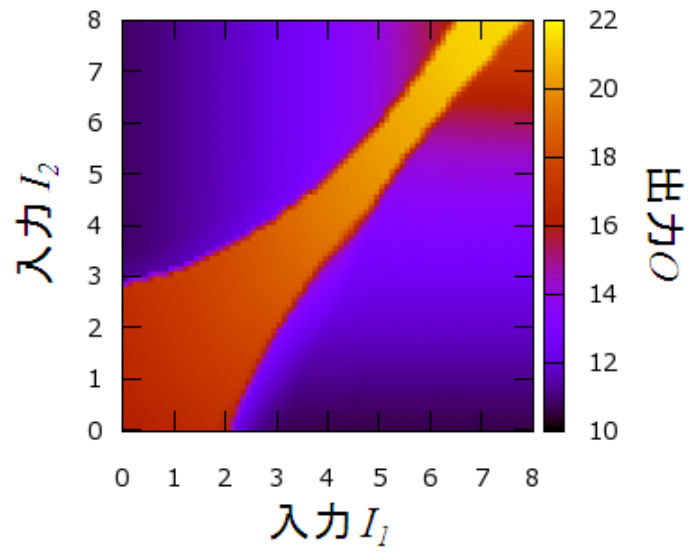


(b)

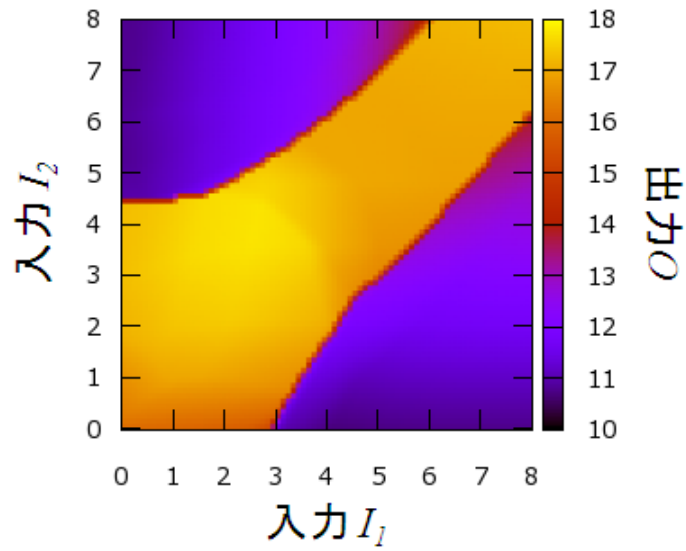
図 4.2: SpikeProp に XOR 問題を学習させた後の入出力関係 (2)

WD の種類を指定する定数 $n = 2$.

WD を導入した SpikeProp に XOR 問題を学習させた後の入出力関係 . (a) と (b) は , 異なる初期荷重から学習したネットワークである .



(a)



(b)

図 4.3: SpikeProp に XOR 問題を学習させた後の入出力関係 (3)

WD の種類を指定する定数 $n = 1$.

WD を導入した SpikeProp に XOR 問題を学習させた後の入出力関係 . (a) と (b) は , 異なる初期荷重から学習したネットワークである .

4.2.2 学習した点近傍における入出力関係の調査

4.2.1 節の評価はあくまで主観的な評価であるため、定量的な評価を行った。しかし、入出力空間全体において、不連続な出力変化が減少したのが定量的に評価することが難しい。そのため、検査領域を、入出力空間全体ではなく、学習した点の近傍に設定した。これは、ニューラルネットワークは、その学習後に学習した入力に近い値が入力されることが多いと考えたためである。そして、検査領域内において、どの程度出力が不連続に変化しているのか(入出力関係の妥当性)について調査し、WD の効果を検証した。

評価方法

XOR 問題を学習した SpikeProp ネットワークに対して、各教師入力パターンの近傍に 2×2 の領域を設定し(図 4.4)、その中に格子状にとった 400 点それぞれに対する出力を計算した。計算した出力が、教師出力と大きく異なる出力(誤出力)が含まれているかどうかにより、入出力関係の妥当性を判定した。具体的な数値を、表 4.2 に示す。検査領域内に、誤出力と設定した出力が 1 つでも含まれていれば、その教師パターン近傍において、ネットワークは誤出力をしたものと判断した。誤出力がなければ、出力が誤って判定されるほどの大きな出力の変化が、学習した入力の近傍では生じていない、つまり、連続的な出力変化をしていると判断できる。ここで、検査領域内で誤出力をしていない場合、その検査領域を正領域と呼ぶ。そして、学習パターンが 4 パターン全てに対して正領域なら、正常なネットワークであると判断する。

表 4.2: 評価の条件

教師パターン	検査領域	誤出力
(1, 1)→17	([0, 2), [0, 2))	出力 < 14, 20 < 出力
(1, 7)→11	([0, 2), (6, 8])	出力 < 8, 14 < 出力
(7, 1)→11	((6, 8], [0, 2))	出力 < 8, 14 < 出力
(7, 7)→17	((6, 8], (6, 8])	出力 < 14, 20 < 出力

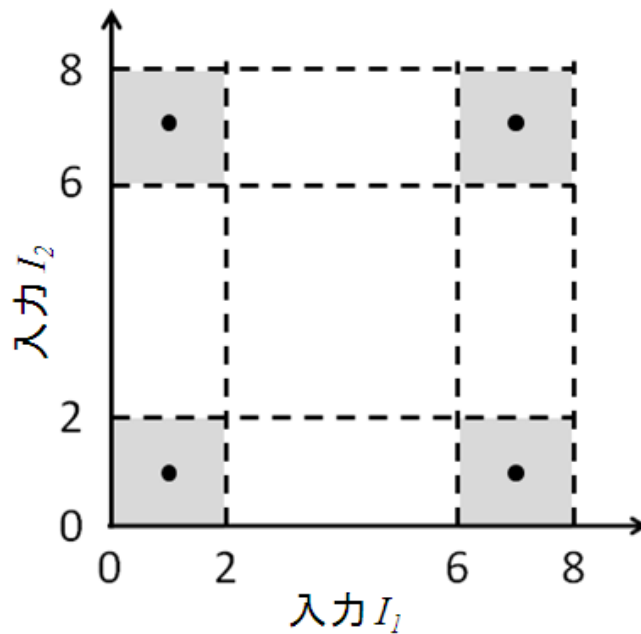


図 4.4: 検査領域

(入力 1, 入力 2)=(1, 1), (1, 7), (7, 1), (7, 7) の 4 つの黒点は学習地点である．
そして，学習地点を中心に 2×2 の領域を設定し，その領域を検査領域とする．

実験結果

表 4.3 に，式 (3.3) または (3.4) に従い学習し，学習に成功したネットワークのうち，正常なネットワーク数を示す．この結果は，異なる初期荷重から学習したネットワーク 500 個を対象としている．オリジナルの SpikeProp では 32 個が正常なネットワークであった．WD を導入すると， $n = 2$ で 36 個， $n = 1$ で 122 個が正常なネットワークであった．WD を導入することで正常なネットワーク数が増加したことが分かる．

表 4.4 は，学習成功数，ネットワークのパターン毎の正領域数を示す．オリジナルの SpikeProp では 2000 パターン中 1223 パターンが正領域であった．WD を導入すると， $n = 2$ で 1280 パターン， $n = 1$ で 1296 パターンが正領域であった．最も正領域数が多かったものは， $n = 1$ であった．

表 4.5 は，学習成功率，正領域率 (学習全て)，正領域率 (学習成功時) を示す．正領域率 (学習全て) は，学習したネットワーク 500 個のうちの正領域率を表している．正領域率 (学習成功時) は，学習が成功したネットワークのうちの正領域率を表している．WD を導入することにより，学習成功率が減少していた．表 4.5 には，WD なしと他の手法との間で正領域率の差について有意差検定を行った結果も付している．各手法間の分散が等しいか

どうかを F 検定により判定した後，分散が等しい場合には t 検定を，そうでない場合にはウェルチの検定を行った．その結果， $p < 0.01$ の場合については有意な差があったものとして ** を， $0.01 \leq p < 0.05$ の場合については * を付した．なお，正領域率 (学習全て) の場合には $n = 2$ の場合のみ，正領域率 (学習成功時) の場合には $n = 1$ の場合のみを分散が等しくない ($p < 0.01$) と判定した．正領域率 (学習全て) では，WD の有無であまり顕著な差は見られなかった．しかし，正領域率 (学習成功時) は，WD 導入ありの $n = 1$ と $n = 2$ の場合両方で，WD なしの場合よりも高い．つまり，WD を導入すると，学習が成功すれば，学習した点近傍で誤出力をしなくなる割合が高くなったことを意味している．

表 4.3: 正常なネットワーク数

	正常なネットワーク数
WD なし	32
$n = 2$	36
$n = 1$	122

表 4.4: パターン毎の正領域数

	(1, 1)	(1, 7)	(7, 1)	(7, 7)	計
WD なし	111	436	433	243	1223
$n = 2$	119	465	472	224	1280
$n = 1$	212	388	386	310	1296

表 4.5: 学習成功率と正領域率

	学習成功率	正領域率 (学習全て)	正領域率 (学習成功時)
WD なし	99.8	61.2	61.3
$n = 2$	98.8	64.0*	64.8**
$n = 1$	83.8	64.8*	77.3**

4.3 結合荷重と内部電位の調査

本節では，WD の効果を確認するために，WD 導入有無の内部状態について調査した．具体的には結合荷重と内部電位について調査し検討を行った．

4.3.1 結合荷重の調査

WD の効果は，出力を維持したまま，不要な結合荷重を 0 にすることである．その効果について確認するため，結合荷重の大きさの分布をヒストグラムにて表す (図 4.5, 図 4.6, 図 4.7)．図 4.5 が WD なし，図 4.6 が WD あり ($n = 2$)，図 4.7 が WD あり ($n = 1$) に対するヒストグラムである．縦軸は結合荷重の割合を，横軸は結合荷重の大きさを示している．WD を導入することで，大きな値の結合荷重が少なくなった．特に， $n = 1$ では，約 75 % が $-1 < w < 1$ の範囲に分布し，WD なしと比較すると，結合荷重の総和が大幅に減少しているのが分かった．4.2.2 節と同様に，WD なしと WD ありとの間で結合荷重の差について有意差検定を行った．F 検定により， $n = 2$ と $n = 1$ の共に，分散が等しくないと判定した．よって，ウェルチの検定を行った結果， $n = 2$ と $n = 1$ ，共に有意水準 $p < 0.01$ で有意差があった．

また，図 4.5，図 4.6，図 4.7 では，不要な結合荷重が 0 になったのか判別できないため，それぞれ $-1 < w < 1$ の範囲を拡大した．それが，図 4.8，図 4.9，図 4.10 である．Weight Decey を導入しても 0 になった結合荷重は存在しなかったため，これは，期待していた効果とは異なっていた．これらのことから，WD を導入することで，大きな値の結合荷重を小さくし，結合荷重の総和は減少したが，結合荷重を 0 にすることはできなかったことが分かった．

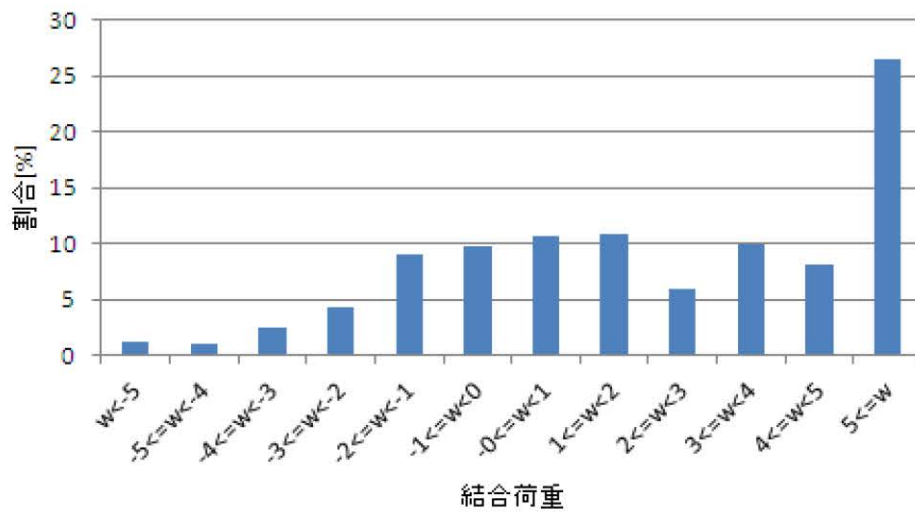


図 4.5: 結合荷重の大きさの分布 (1)

WD なし．オリジナルの SpikeProp に，XOR 問題を学習させた後の結合荷重の大きさを，ヒストグラム化したものである．横軸が結合荷重の大きさを表しており，縦軸が全体に占める割合を表している．

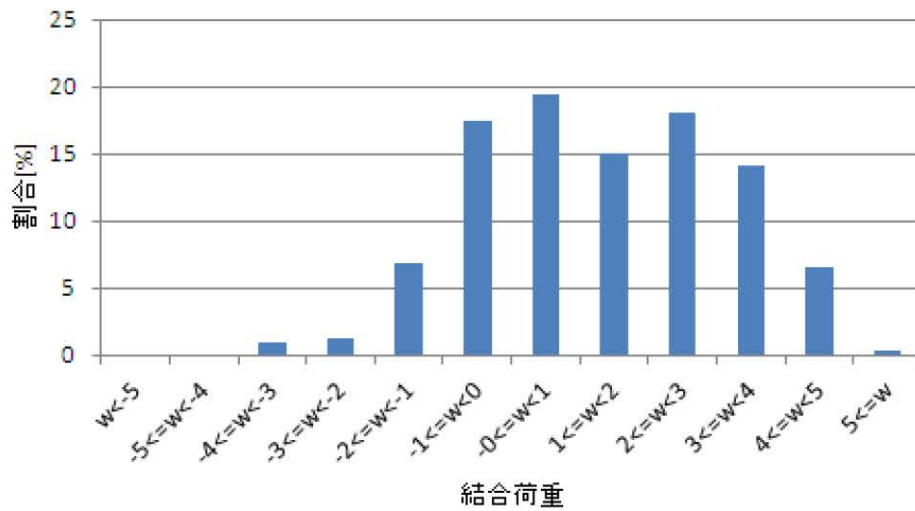


図 4.6: 結合荷重の大きさの分布 (2)

WD の種類を指定する定数 $n = 2$.

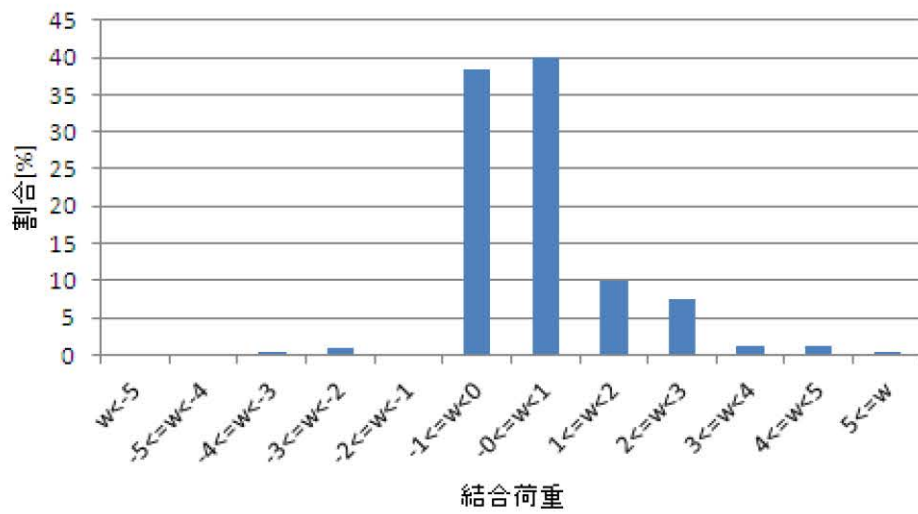


図 4.7: 結合荷重の大きさの分布 (3)

WD の種類を指定する定数 $n = 2$.

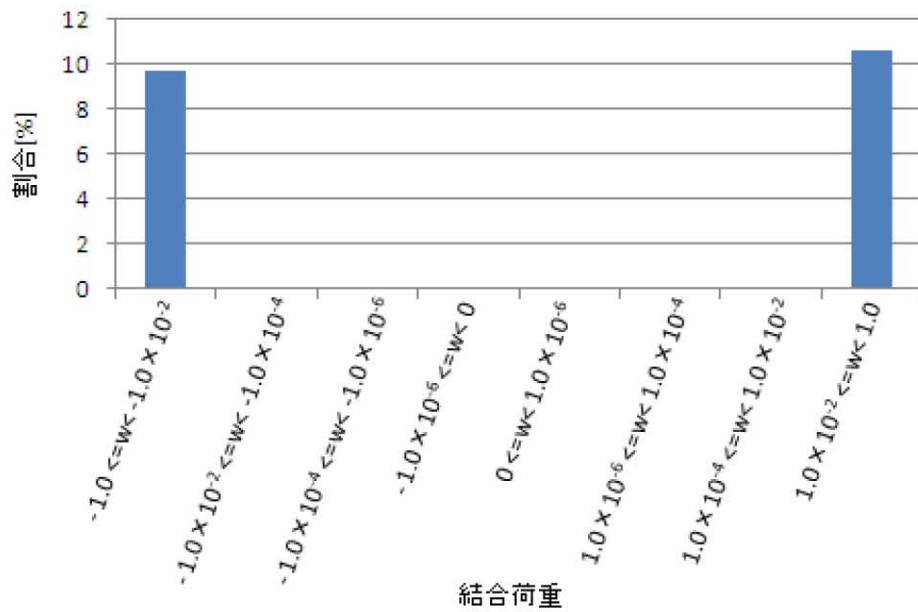


図 4.8: 図 4.5 拡大

図 4.5 において, $-1 < w < 1$ の範囲の結合荷重の大きさの分布を細かくしたものである .

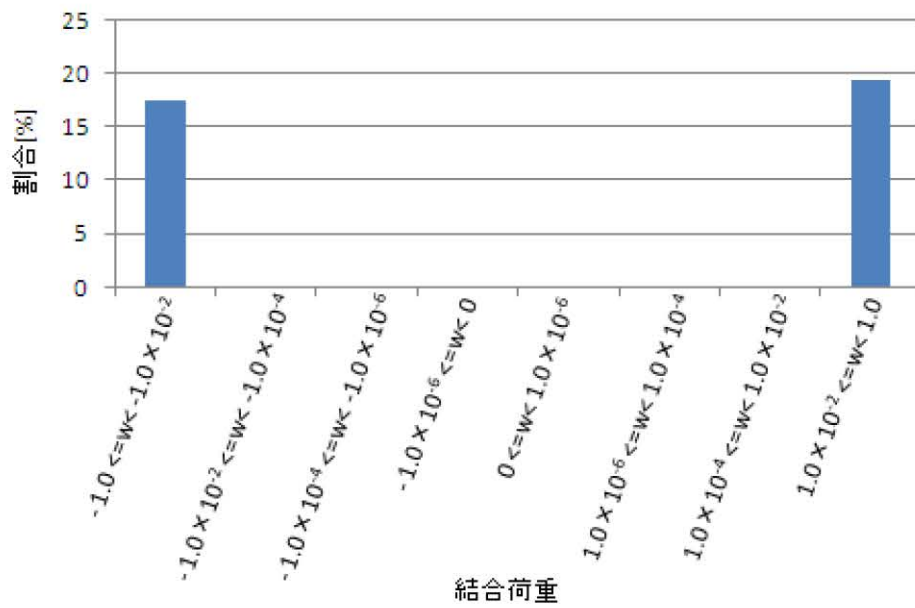


図 4.9: 図 4.6 拡大

図 4.6 において， $-1 < w < 1$ の範囲の結合荷重の大きさの分布を細かくしたものである．

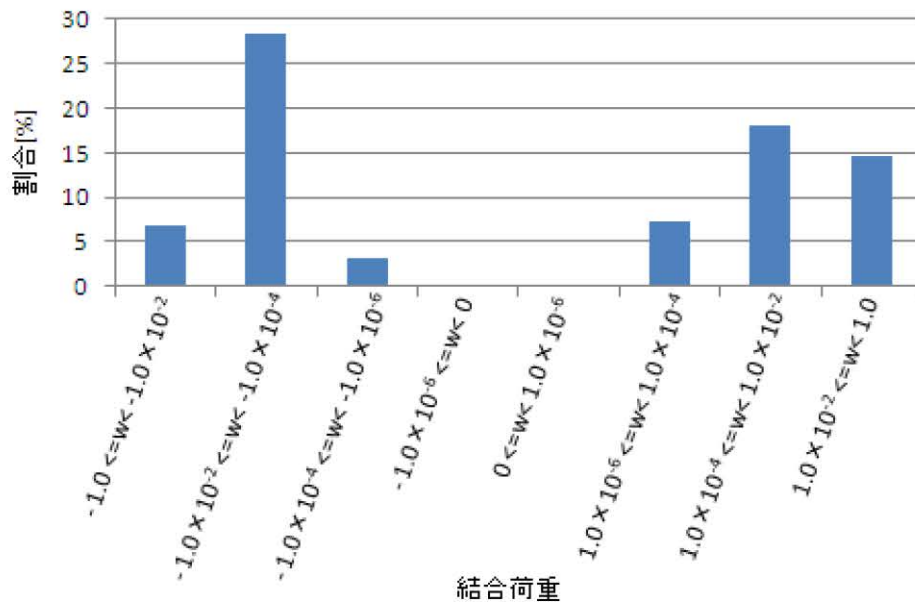


図 4.10: 図 4.7 拡大

図 4.7 において， $-1 < w < 1$ の範囲の結合荷重の大きさの分布を細かくしたものである．

4.3.2 内部電位の調査

3.2節で説明した，WD 導入による予想効果について実際に内部電位を調査した．図 4.11 は，誤出力をした場合 (WD なし) の学習した点 (入力 1, 入力 2)=(1, 1) と，学習した点近傍 (入力 1, 入力 2)=(1.2, 1), (入力 1, 入力 2)=(1.3, 1) の内部電位の図である．横軸が時刻 t で，縦軸が内部電位 $x_o(t)$ の大きさである．3.1 節で述べたような凹凸のある形状をしていた．また，(1.2, 1) と (1.3, 1) 共に，(1, 1) とよく似た形状をしていた．これは，これらの入力の違いが，ネットワークにおいて，わずかな違いであったことを意味している．しかし，時刻 11 付近を拡大してみると (図 4.12)，(1, 1)，(1.2, 1) では，時刻 11 付近で出力しきい値 $\theta = 10.0$ を超えていないのに対し，(1.3, 1) では，時刻 11 付近で極大の部分が出力しきい値 $\theta = 10.0$ を超えているため，近い入力であるにも関わらず出力時刻が大幅に変わっていた．

次に誤出力をしなかった場合の内部電位について観測する．図 4.13 は WD あり ($n = 1$) の誤出力をしなかった場合の学習した点近傍の内部電位の図である．誤出力をした場合と同様，(入力 1, 入力 2)=(1, 1)，(1.2, 1)，(1.3, 1) の内部電位について観測した．凹凸のある形状をしているが，図 4.11 と比較すると凹凸が緩やかになっていた．また，図 4.14 は，図 4.13 において，発火時刻より前の最も出力しきい値との距離が近い極大を拡大したものである．図 4.14 より，入力変化 0.1 に対する出力変化と比較し，出力しきい値と極大との距離が離れていた．そのため，入力が少し変わっただけでは，内部電位の形状の変化も少ないため，出力しきい値を超えることがない．

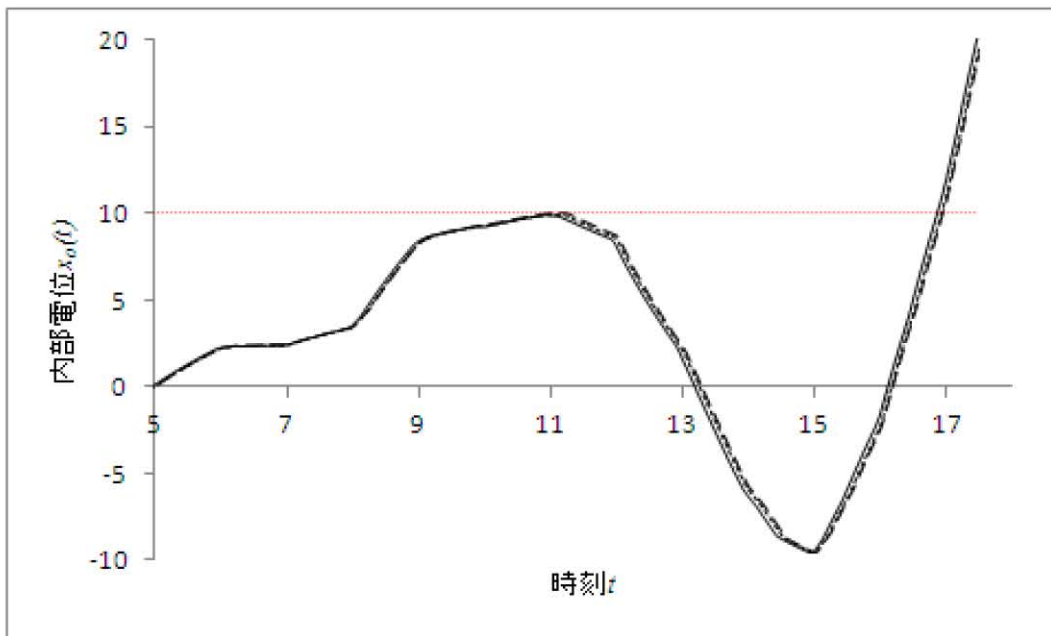


図 4.11: 誤出力をした時の学習後のユニットの内部電位の変化

SpikeProp ネットワークに XOR 問題を学習させ、誤出力をした時の、出力層ユニットの内部電位の変化である。横軸は時刻、縦軸は内部電位の大きさを、また、赤の点線は出力しきい値を示す。また、入力は、実線が (1, 1)、点線が (1.2, 1)、破線が (1.3, 1) である。

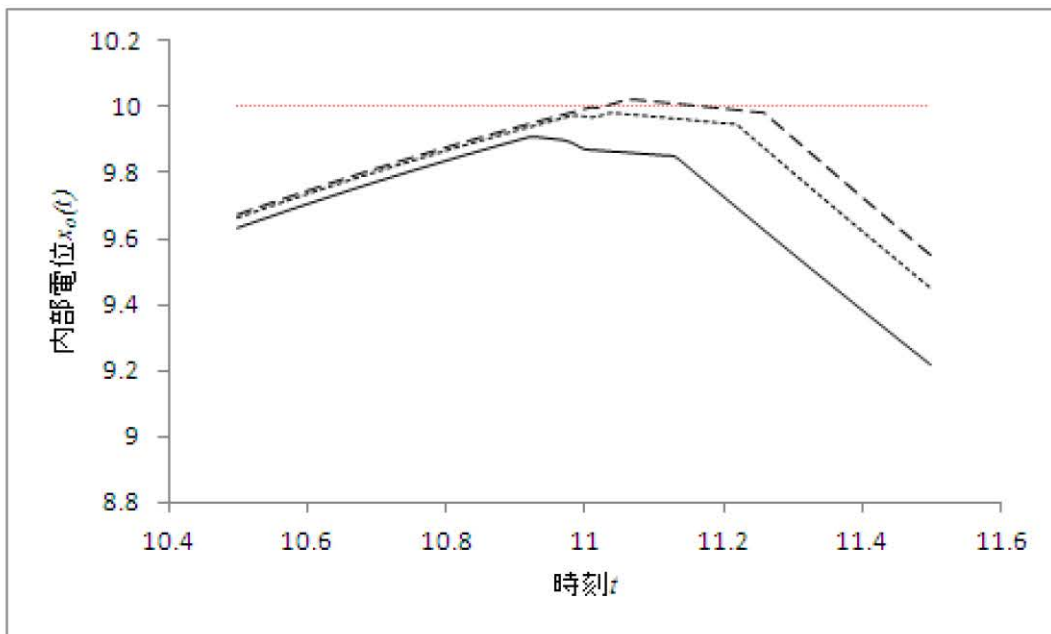


図 4.12: 図 4.11 拡大

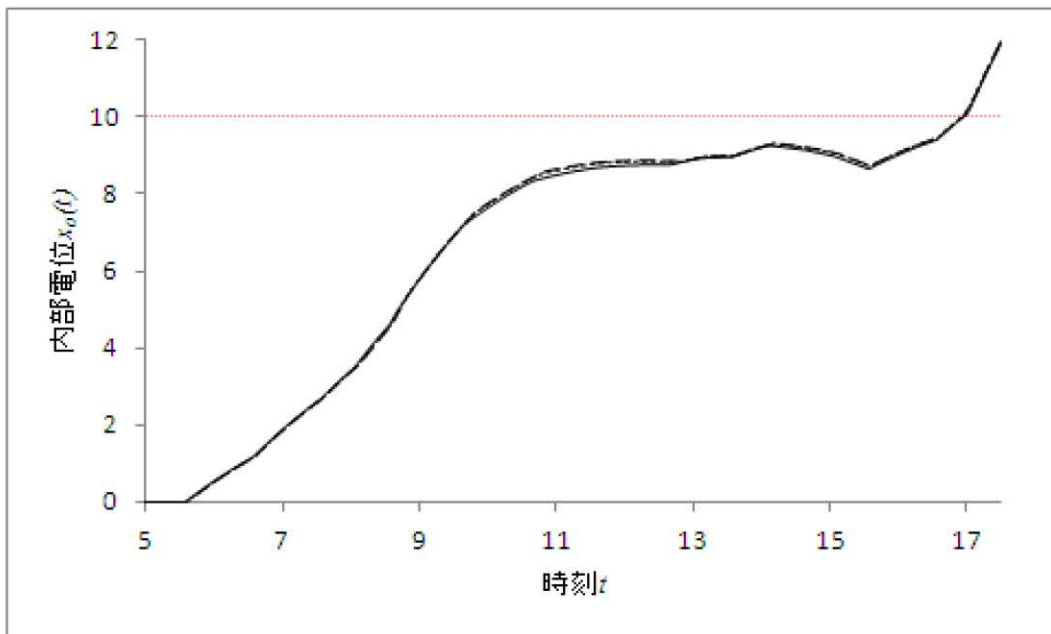


図 4.13: 誤出力をしなかった時の学習後のユニットの内部電位の変化

SpikeProp ネットワークに XOR 問題を学習させ、誤出力をしなかった時の、出力層ユニットの内部電位の変化である。横軸は時刻、縦軸は内部電位の大きさを、また、赤の点線は出力しきい値を示す。また、入力は、実線が (1, 1)、点線が (1.2, 1)、破線が (1.3, 1) である。

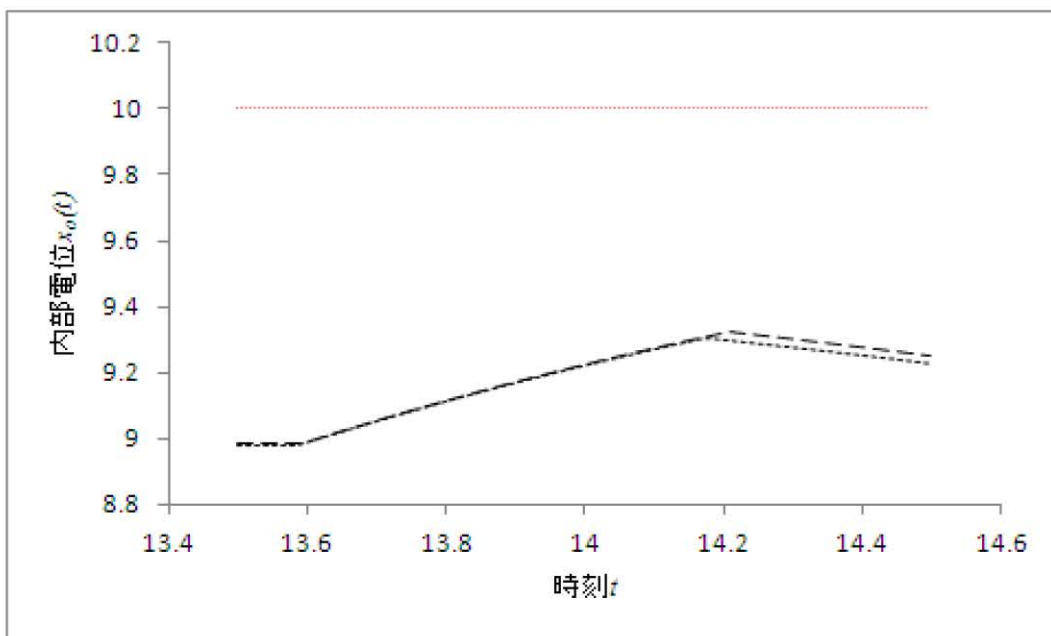


図 4.14: 図 4.13 拡大

4.3.3 考察

これまでの実験より，WDを導入することで，正常なネットワーク数，学習全ての正領域率，学習成功時の正領域率，全て良くなることが示された．また，文献 [18] のとおり， $n = 2$ よりも $n = 1$ の方が WD の効果が高いことも確認できた． $n = 1$ の場合，学習成功率が減少してしまう問題があるが，それでも 80 % 以上あり，許容範囲であると考えられる．また，学習成功時の正領域率が高いため，学習が成功すれば，誤出力をする可能性を減らすことができる．これは，学習終了時に良いネットワークと悪いネットワークを区別できるということになる．そのため，学習終了後に出力が不連続に変化してしまうネットワークを使用するよりも良いので，あまり深刻な問題ではないと考えられる．これらのことを考慮すると，WD を導入することで WD なしの場合よりも良い結果が得られたといえる．

また，4.3 節より，WD を導入することで，結合荷重の値を 0 にすることはできなかったが，総和を大幅に減少させることができた．大きな結合荷重の値を少なくできたことにより，内部電位の形状の変化を緩やかにすることができた．その結果，出力しきい値付近に内部電位の極大が存在しても，入力変化 0.1 に対する出力変化と比較し，出力しきい値と極大との距離が離れていたため，出力しきい値を超えることが少なくなったと考えられる．

ただし，完全に不連続な出力変化を解消できたわけではない．現在，WD では，小さくしたい結合荷重を指定しているわけではないため，出力しきい値付近の極大と出力しきい値の距離が近い場合もあった．そのため，出力しきい値付近の極大をもたらす結合荷重を，重点的に小さくすることができれば，出力しきい値と極大の距離が大きくなるため，今よりも更に誤出力を減らすことができるのではないかと考えられる．

当初，不要な結合荷重が 0 になると予想していたが，予想に反し，大きな結合荷重の値を小さくする結果となった．ただし，上記のように，WD 導入の効果は表れているので，深刻な問題ではないと考える．結合荷重を 0 にすべきか，小さくすれば良いのかについては，更に解析する必要がある．表 4.4 より，パターン (1, 1) の正領域数が他のパターンと比べ少ないことが分かる．そのため，パターン (1, 1) の正領域数を増加させることが，正常なネットワーク数の増加に繋がると考えられる．具体的には，他のペナルティ項を使用する・小さくしたい結合荷重を狙えるようにするなど改良していくことで，更に良い結果が得られるのではないかと考える．

第5章 まとめ

本研究では、学習終了後の SpikeProp ネットワークにおける、入力の微小な変化に対する不連続な出力変化を解消することを目的とした。SpikeProp では、学習終了後に入力の微小な変化に対して出力が不連続に変化してしまう問題がある。このような現象が起こると、ネットワーク出力が、ノイズ等に起因する入力の微小な変化に対して、出力が不連続に変化してしまい、ネットワークとして好ましくないものになってしまう。

そのため、解決の手段として、従来のシグモイド型のニューラルネットワークにおいて、不適切な入出力関係を改善するための手法として使われるウェイト・ディケイ (WD) を SpikeProp に導入した。その結果、WD を導入することで、不連続な出力変化を減少させることができた。完全には失くすことはできていないが、WD が SpikeProp の不連続な出力変化の問題に対しても効果があることが分かった。完全に不連続な出力変化を失くすことができなかった原因として、WD を導入しても、出力しきい値付近の極大を完全に失くすことができなかったことが考えられる。

現在は、不連続な出力変化が、特定のパターン近傍にかたまっている。これを解決することで、正常なネットワーク数の増加に繋がると考えられる。そのため、WD を改良していく必要がある。具体的には、出力しきい値付近の極大をもたらす結合荷重を、重点的に小さくすることが重要である。その結果、出力しきい値付近の変化が緩やかになるため、今よりも更に誤出力を減らすことができるのではないかと考えられる。そして、学習した点近傍の不連続な出力変化がなくなれば、SpikeProp をロボット制御等に应用することも可能になるのではないかと考える。

謝辞

本論文は、著者が三重大学大学院地域イノベーション学研究科博士前期課程時に行った研究をまとめたものである。本研究の遂行及び修士論文の作成にあたり、日ごろ丁寧なご指導とご助言を頂きました本学工学研究科電気電子工学専攻の高瀬治彦准教授，ならびに本学地域イノベーション学研究科の坂内正明教授，鶴岡信治教授，萩原克幸准教授，本学工学研究科電気電子工学専攻の川中普晴助教に深く感謝致します。また，貴重な時間を割いて頂き，論文を査読して頂いた本学地域イノベーション学研究科の奥村晴彦教授，久保雅敬教授，矢野竹男教授，三島隆准教授に深く感謝致します。また，日頃熱心に討論して頂いた情報処理研究室の皆様方にお礼申し上げます。最後となりましたが，日頃お世話になりました本学地域イノベーション学研究科と情報処理研究室の皆様，本論文をまとめるにあたり，助言，討論，その他お世話になりました全ての方々に感謝致します。

参考文献

- [1] Wolfgang Mass and Christopher M. Bishop: Pulsed Neural Networks, The MIT Press, pp.3–85 (2001)
- [2] 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, システム制御情報学会誌, Vol.48, No.2, pp.57–62 (2004)
- [3] 川人光男: 小脳内部モデルとモザイクの計算論的神経科学, 人口知能学会誌, Vol.17, No.6, pp.720–731 (2002)
- [4] Takenori Obo, Naoyuki Kubota, and Beom Hee Lee: Localization of Human in Informationally Structured Space Based on Sensor Networks, Proc. of 2010 IEEE World Congress on Computational Intelligence (WCCI2010), pp.2215–2221 (2010)
- [5] 瀧田航一郎: パルスニューラルネットワークによる時系列情報処理に関する研究, 慶応義塾大学博士論文 (2005)
- [6] 元木誠, 小塚成一, 平田廣則: ニューラルネットワークの違いに着目した自律移動ロボットコントローラの性能比較, 電子情報通信学会技術研究報告, Vol.105, No.457, pp.25–30 (2005)
- [7] Hani Hagnas, Anthony Pounds-Cornish, Martin Colley, Victor Callaghan and Graham Clarke: Evolving Spiking Neural Network Controllers for Autonomous Robots, IEEE International Conference on Robotics and Automation (ICRA2004), pp.4620–4626 (2004)
- [8] Alnajjar F, Murase K: Sensor-fusion in spiking neural network that generates autonomous behavior in real mobile robot, Proceedings of International Joint Conference on Neural Networks 2008, pp.2200–2206 (2008)

- [9] 黒柳奨, 奥村享司, 岩田彰: パルスニューロンモデルを用いた音源定位聴覚系モデル — 位相情報抽出のためのニューロン層の構築 —, 電子情報通信学会技術研究報告, Vol.96, No.134, pp.163–169 (1997)
- [10] Anthony N. Burkitt, Hamish Meffin, and David B. Grayden: Spike-Timing-Dependent Plasticity: The Relationship to Rate-Based Learning for Models with Weight Dynamics Determined by a Stable Fixed Point, *Neural Computation* 16, pp.885–940 (2004)
- [11] Sander M. Bohte, Joost N. Kok, and Han La Poutré: Error-backpropagation in temporally encoded networks of spiking neurons, *Neurocomputing*, Vol. 48, pp.17–37 (2002)
- [12] Simon Christian Moore: Back-propagation in Spiking Neural Networks, Master’s thesis, University of Bath (2002)
- [13] 吉富康成: シリーズ非線形科学入門 ニューラルネットワーク, 朝倉出版, pp.27–49 (2002)
- [14] Sam McKennoch, Dingding Liu, and Linda G. Bushnell: Fast Modifications of the SpikeProp Algorithm, *Proceedings of 2006 International Joint Conference on Neural Networks*, pp.7701–7708 (2006)
- [15] Benjamin Schrauwen, Jan Van Campenhout: Extending SpikeProp, *Proceedings of International Joint Conference on Neural Networks 2004*, pp.471–475 (2004)
- [16] H. Takase, M. Fujita, H. Kawanaka, S. Tsuruoka, H. Kita, and T. Hayashi: Obstacle to Training SpikeProp Networks — Cause of Surges in Training Process —, *Proceedings of International Joint Conference on Neural Networks 2009*, pp.3062–3066 (2009)
- [17] David C. Plaut, Steven J. Nowlan, and Geoffrey E. Hinton: Experiments on Learning by Back Propagation, Technical Report CMU-CS-86-126, Pittsburgh, PA (1986)
- [18] 石川真澄: 忘却を用いたコネクショニストモデルの構造学習アルゴリズム, *人工知能学会誌*, Vol.5, No.5, pp.595–603 (1990)

- [19] Shinji Fukuda: The effects of weight decay backpropagation on the fuzzy neural network model-based habitat evaluation for Japanese medaka (*Oryzias latipes*), Bulletin of the Institute of Tropical Agriculture, Kyushu University, Vol.33, pp.83-93 (2010)

発表論文

- [1] 吉川雄也, 高瀬治彦, 川中普晴, 鶴岡信治: SpikeProp ネットワークの安定性 — 結合荷重の変化に対する出力変化 —, 第 8 回情報学ワークショップ (WiNF2010) 論文集, pp.99–102 (2010)
- [2] 吉川雄也, 高瀬治彦, 川中普晴, 鶴岡信治: SpikeProp ネットワークに対する Weight Decay 導入の効果の検討, 第 27 回ファジィシステムシンポジウム (FSS2011) 論文集, pp.793–796 (2011)
- [3] Yuya Yoshikawa, Haruhiko Takase, Hiroharu Kawanaka , and Shinji Tsuruoka: Effectiveness of Weight Decay to improve discontinuous output in SpikeProp, Proc. of Second International Workshop on Regional Innovation Studies 2011 (IWRIS 2011), pp.65–68 (2011)