

# 可飽和吸収体の利用を前提としたナノフォトニック・ニューラルアクセラレータ向け活性化関数の評価

磯部 聖<sup>1,a)</sup> 川上 哲志<sup>2</sup> 小野 貴継<sup>2</sup> 井上 弘士<sup>2</sup> 納富 雅也<sup>3,4</sup>

**概要:** 我々はナノフォトニクスを用いたニューラルネットワーク用アクセラレータの実現を目指している。活性化関数を実現するナノフォトニックデバイスとして可飽和吸収体が候補に挙げられるが、その実用性は明らかではない。そこで本稿では、可飽和吸収体のデバイスパラメータを様々な値に設定し、画像識別問題における識別精度を指標として、可飽和吸収体による活性化関数を評価した。評価の結果、情報の媒体として光電解振幅を用いる場合、代表的な活性化関数である  $\tanh$  に対して識別精度の劣化が MNIST において 0.06 %、CIFAR-10 においての 0.3 % に留まり、一般的な活性化関数と同程度の識別精度を実現可能であることが分かった。

## 1. はじめに

近年、情報処理の一手段として人間の脳を模した機械学習モデルであるニューラルネットワークの活用が注目を集めている。その応用はデータセンターから組み込みシステムまで多岐に渡り、これまでに高い電力性能効率を達成するための専用アクセラレータが提案されてきた [1][2][3][4][5]。今後、より電力制約の厳しい環境への適用や、レイテンシ制約下におけるリアルタイム処理応用を鑑みた場合、更なる高性能化と低消費電力化が必要となる。

このような要求を満たすべく、我々は、ナノフォトニックデバイスを用いたニューラルネットワーク用アクセラレータの実現を目指している [6]。ニューラルネットワークの主要な処理は入力信号と結合荷重の積和演算であり、マツハツェンダ干渉器を用いた光ベクトル行列演算器を直接適用することができる。一般に、積和演算の結果は sigmoid 関数などの活性化関数を通して、非線形処理が行われる。活性化関数はニューラルネットワークの識別精度に影響を与える [7]。しかしながら、ナノフォトニックデバイスで実現できる非線形処理は限られており、現在のニューラル

ネットワークで想定される活性化関数を用いることができない。したがって、ナノフォトニックデバイスでの活性化関数の実用性を明かにする必要がある。

そこで本稿では、非線形性を有するナノフォトニックデバイスとして可飽和吸収体に着目し、その有用性を検討する。具体的には、可飽和吸収体の入出力特性を決定するデバイスパラメータ（光の吸収係数、飽和強度、飽和光電界振幅、デバイス長）を様々な値に設定し、可飽和吸収体の入出力特性を活性化関数として用いた場合の識別精度と、一般的な活性化関数を用いた場合の識別精度を比較する。

本稿の構成は以下の通りである。まず、第 2 節では本稿で着目するニューラルネットワークを示し、第 3 節で可飽和吸収体の入出力特性を説明する。そして、第 4 節で評価実験の詳細を報告し、第 5 節で設定すべきデバイスパラメータについて考察する。最後に、第 6 節でまとめと今後の課題を述べる。

## 2. ニューラルネットワーク

本節では、本稿で対象とするニューラルネットワークの構成と処理を示す。また、活性化関数が識別精度に与える影響について述べる。

### 2.1 対象とするニューラルネットワーク

ニューラルネットワークの構成は、画像認識や音声認識などその応用により様々である。本稿では、図 1 で示す基本的な構成である 3 層フィードフォワード型ニューラルネットワークに着目する。フィードフォワード型ニューラルネットワークは、ノードであるニューロンを層状に単

<sup>1</sup> 九州大学 大学院システム情報科学府  
Graduate School of Information Science and Electrical Engineering, Kyushu University  
<sup>2</sup> 九州大学 大学院システム情報科学研究院  
Faculty of Information Science and Electrical Engineering, Kyushu University  
<sup>3</sup> NTT ナノフォトニクスセンタ  
NTT Nanophotonics Center  
<sup>4</sup> NTT 物性科学基礎研究所  
NTT Basic Research Laboratories  
a) satoshi.isobe@cpc.ait.kyushu-u.ac.jp

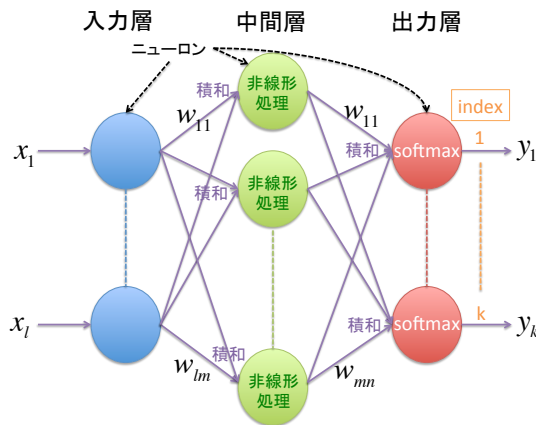


図 1 3層フィードフォワード型ニューラルネットワーク

一方向に結合させた構成を採る。初段を入力層，最終段を出力層，これらの間を中間層と呼び，入力層のニューロンはニューラルネットワークの入力をそのまま中間層に受け渡す。

## 2.2 ニューラルネットワークの処理

各ニューロンでは，式 (1) で示すように入力ベクトルの内積演算（実際には積和演算）を行い，その結果に非線形性を有する活性化関数  $f$  を適用する。

$$y = f(\vec{w} \cdot \vec{x}) \quad (1)$$

ここで， $\vec{x}$  はニューロンの入力， $\vec{w}$  は学習によって調節される結合過重， $y$  はニューロンの出力である。一般的に用いられる活性化関数として，式 (2) に示す sigmoid 関数，式 (3) に示す tanh(hyperbolic tangent)，式 (4) に示す ReLU(Rectifier Linear Unit) などが挙げられる。図 2 にそれぞれの関数のグラフを示す。

$$y = \frac{1}{1 + e^{-x}} \quad (2)$$

$$y = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (3)$$

$$y = \begin{cases} X & (X \geq 0) \\ 0 & (Otherwise) \end{cases} \quad (4)$$

ここで， $X$  は活性化関数の入力  $\vec{w} \cdot \vec{x}$  である。

多クラス識別問題に着目した場合，出力層はクラス数  $K$  と同数のニューロンで構成され，活性化関数として式 (5) に示す softmax 関数が用いられる。

$$y_k = \frac{\exp X_k}{\sum_{j=1}^K \exp X_j} \quad (5)$$

ここで， $y$  と  $X$  の添字は出力層のニューロンのインデックスであり，出力層において最大の出力値を有するもの（すなわち，最も大きな softmax 関数の出力値を有するニュー

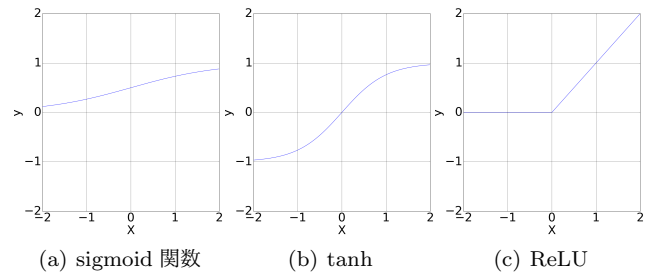


図 2 一般的な活性化関数

表 1 活性化関数による識別精度の違い

活性化関数	識別誤差率 [%]
sigmoid 関数	1.842948718
tanh	1.812900641
ReLU	1.772836538

ロン) に対応するクラスが識別推定結果となる。

ニューラルネットワーク・モデルは結合荷重が調節されることで構成される。この結合荷重の調節を学習と呼ぶ。学習には確率的勾配降下法を用いており，式 (6) に示す誤差関数の値  $l$  が減少するように，ある結合荷重の初期値から繰り返し式 (7) を適用し，結合荷重の値を調節する。

$$l = - \sum_{i=0}^{|D|} \log y_c^{(i)} \quad (6)$$

$$w \leftarrow w - \alpha \frac{\partial l}{\partial w} \quad (7)$$

ここで， $w$  は各結合荷重， $\alpha$  は学習係数， $|D|$  はバッチサイズ， $y_c^{(i)}$  は  $i$  番目のデータの正解のクラスに対応した出力層のニューロンの出力である。バッチとは結合荷重の更新一回に用いられるデータの集合であり，学習に用いられるデータを均等に分割することで作成される。また，1つのバッチに含まれるテストデータの総数をバッチサイズと呼ぶ。結合荷重の調節は全てのバッチを順次用いて行われ，この処理の単位をエポックと呼ぶ。

## 2.3 活性化関数が識別精度に与える影響

活性化関数はニューラルネットワークによる識別精度に影響を与える。中間層のニューロン数が 500 である 3 層フィードフォワード型ニューラルネットワークに対し，一般的に用いられる sigmoid 関数，tanh，ReLU を適用して MNIST[8] の識別を行った際の識別誤差率を表 1 に示す。識別誤差率とは，全てのテストデータのうち，誤って識別したデータの割合である。表 1 から，適用する活性化関数によって識別精度が異なることが分かる。また，図 3(a) の様な，ReLU において定義域が負の領域の傾き  $\beta$  を変化した活性化関数 (8) を適用して，同様の識別を行った際の識別精度を図 3(b) に示す。図 3(b) を見ると， $\beta$  が 0.5 以上になると急激に識別誤差率が上昇していることが分かる。これは，用いる活性化関数によっては著しく識別精

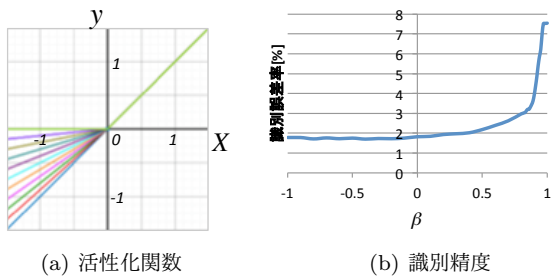


図 3 ReLU の負の部分の傾きを変えた活性化関数による識別

度が劣化することを意味する。しかしながら、ナノフォトニックデバイスでは任意の活性化関数を実現することが難しく、活性化関数は非線形性を有するナノフォトニックデバイスの入出力特性に制限される。

$$y = \begin{cases} X & (X \geq 0) \\ \beta X & (\text{Otherwise}) \end{cases} \quad (8)$$

### 3. 可飽和吸収体による活性化関数

本節では、非線形な入出力特性を示す代表的なナノフォトニックデバイスであり、活性化関数を実現する素子として有力な候補である可飽和吸収体について説明する。この可飽和吸収体は文献 [9][10] でも着目されているものである。

#### 3.1 情報の媒体

可飽和吸収体とは、強度が弱い入射光に対しては吸収体として、また、強度が強い場合には透明体としての性質を備える物質である。この様な可飽和吸収現象は、光によって物質中の電子が励起準位に遷移した結果、遷移先の準位が飽和してしまい、それ以上光を吸収できなくなることで生じる。本稿では、この可飽和吸収体の入出力特性を活性化関数として用いる。

同じ可飽和吸収現象であっても、用いる情報の媒体として光強度と光電界振幅のいずれを選択するかによって可飽和吸収体の入出力特性は異なる。光強度  $I$  は単位面積当たり単位時間で通過する光のエネルギー、光電界振幅  $E$  は入力光電界の振幅であり、これらの間には比例関係が成り立つ。

光強度はエネルギー量であり必ず正の値となる。そのため、可飽和吸収体の入出力強度の関係を活性化関数として用いる場合、活性化関数の入力を正の値に制限しなければならない。そこで本稿では、光強度を用いた場合、式 (7) に示す一般的な確率的勾配降下法に制約を加えた式 (9) を用いて結合荷重の調節を行う。

$$w \leftarrow \begin{cases} w - \alpha \frac{\partial L}{\partial w} & (w - \alpha \frac{\partial L}{\partial w} \geq 0) \\ 0 & (\text{Otherwise}) \end{cases} \quad (9)$$

この結合荷重の調整方法を用いると、結合荷重の初期値を

正に設定した場合、学習の結果得られる値は必ず正となる。ニューラルネットワークの入力が常に正（例えば画像のピクセル値）であると仮定すると、式 (1) より活性化関数の入力は必ず正の値となることが保証される。一方、光電界振幅を情報の媒体として利用する場合には位相制御が必要となり、光強度に比べてナノフォトニックデバイスによる制御が複雑となる。しかしながら、符号に関係なく値を表現できるため、光強度を用いる場合と比較してより自由度の高い学習を実施できるといった利点がある。

#### 3.2 可飽和吸収体の入出力特性

情報の媒体として光強度を用いる場合の可飽和吸収体の入出力特性は、物質による光の吸収に関するランベルト・ベールの法則に基づく式 (10) から導かれる。

$$I_{out} = I_{in} e^{-aL} \quad (10)$$

ここで、 $a$  は吸収係数、 $L$  は物質中を光が通過する距離であり可飽和吸収体のデバイス長に等しいと仮定する。また、 $I_{in}$  は物質への入力強度、 $I_{out}$  は入力光が物質中を  $L$  進んだ後に出力される光強度である。可飽和吸収体の吸収係数は入力強度  $I_{in}$  に依存する。可飽和吸収体が表面の反射の無い薄い単純な薄膜であると仮定すると、可飽和吸収体の吸収係数  $a$  は式 (11) の様に表すことができる [11]。

$$a = \frac{a_0}{1 + I_{in}/I_s} \quad (11)$$

ここで、 $a_0$  は  $I_{in} = 0$  の時の吸収係数、 $I_s$  は飽和強度と呼ばれる励起準位が飽和し始める光強度であり、いずれも物質特性によって定まる値である。式 (10) および式 (11) より、デバイス長  $L$  の可飽和吸収体の光強度の入出力特性は、式 (12) の様に表される。

$$I_{out} = I_{in} \cdot \exp\left(\frac{-a_0 L}{1 + I_{in}/I_s}\right) \quad (12)$$

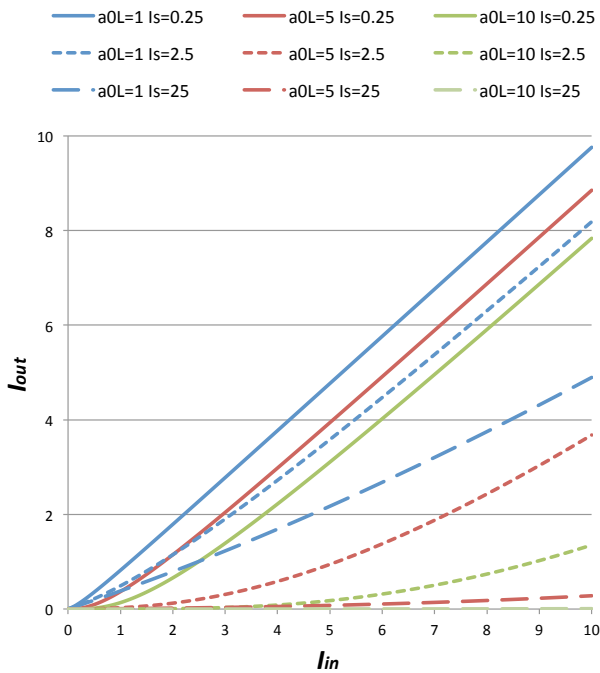
一方、光電界振幅を情報の媒体として用いる場合の入出力特性は、式 (12) および光強度と光電界振幅の比例関係から式 (13) となる。

$$E_{out} = E_{in} \cdot \exp\left(\frac{-a_0 L}{2(1 + |E_{in}/E_s|^2)}\right) \quad (13)$$

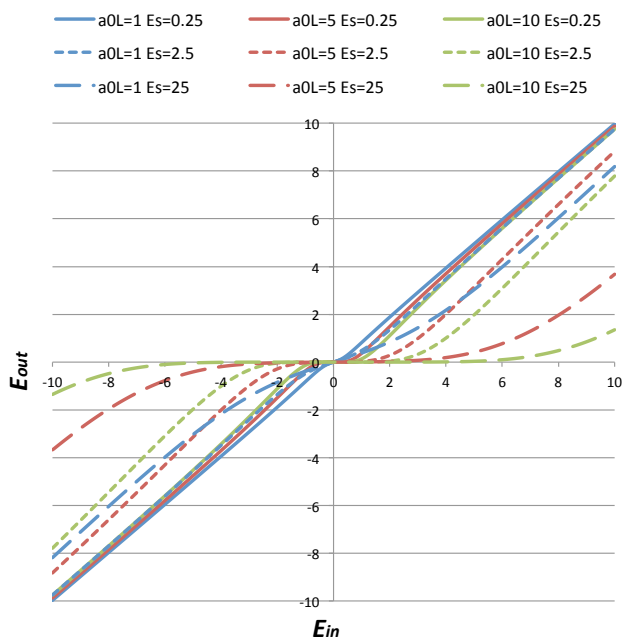
ここで、 $E_{in}$  は入力される光電界振幅、 $E_{out}$  は入力光が物質中を  $L$  進んだ後に出力される出力電界振幅、 $E_s$  は飽和光電界振幅である。図 4 に、式 (12) ならびに式 (13) で示した可飽和吸収体の入出力特性を示す。本稿では、光強度または光電界振幅それぞれを情報の媒体として用いた場合に関して識別精度を解析する。

#### 4. 可飽和吸収体による活性化関数の評価

本節では、可飽和吸収体による活性化関数の評価について述べる。



(a) 光強度を用いた場合



(b) 光電界振幅を用いた場合

図 4 可飽和吸収体の入出力特性

#### 4.1 評価方法

本稿では、複数の画像識別問題に対して、活性化関数のみ異なるニューラルネットワークを同じ条件下で学習させる。識別対象の画像データセットとして、MNIST [8] ならびに CIFAR-10 [12] を用い、識別精度を解析する。これらの画像識別問題では、画像の各ピクセル値を入力として、その画像に描画された物体の名前に対応したクラスに識別する。ピクセル値として、256 階調を 0 から 1 の範囲にプ

ロットした値を用いる。MNIST の学習用画像データの数は 60,000、識別用画像データの数は 10,000 である。また、CIFAR-10 の学習用画像データの枚数は 50,000、識別用画像データの数は 10,000 である。

本評価では、中間層に 500 個のニューロンを持つ 3 層フィードフォワード型ニューラルネットワークを対象にする。これは、Theano[13] を用いたニューラルネットワークのチュートリアルにおいて MNIST の識別に用いられた構成を参考に決定した [13]。活性化関数には、式 (12) ならびに式 (13) で示した可飽和吸収体の入出力特性を用い、比較対象は代表的な活性化関数である sigmoid, tanh, ReLU とした。ここで、吸収係数とデバイス長の積  $a_0L$ 、飽和強度  $I_s$ 、飽和光電界振幅  $E_s$  といった素材やデバイス構造に依存するパラメータが存在する。そこで、本評価ではこれらのパラメータを様々な値に設定して学習させたニューラルネットワークの識別精度を解析する。本評価ではデバイスの実現可能性を考慮して、 $1 \leq a_0L \leq 10$ 、 $0.25 \leq I_s \leq 25$ 、 $0.5 \leq E_s \leq 5$  とした。式 (12) ならびに式 (13) を活性化関数とした場合においても、sigmoid 関数などの代表的な活性化関数を用いた場合と同程度の識別精度を達成できれば、CMOS を用いたニューラルネットワーク用アクセラレータと同程度の応用可能性が期待できる。

また、非線形性を有する活性化関数の必要性を確認するために、恒等関数を用いた場合との比較も実施する。ここで、恒等関数を活性化関数として用いた 3 層フィードフォワード型ニューラルネットワークと中間層の無いフィードフォワード型ニューラルネットワークは等価であることが知られている。そこで、本稿では中間層の無いフィードフォワード型ニューラルネットワークを用いて識別を行い、その識別精度を恒等関数の識別精度として比較する。

本評価における学習条件を表 2 に示す。学習係数  $\alpha$  は誤差関数の値が発散しないように事前に調節した値である。また、バッチサイズは論文 [14] 中で推奨されている 32 を用いた。本評価における学習は、1,000 エポック繰り返しを行い、10 エポック毎に行われるテストデータの識別において最も識別精度が高くなる学習結果（各ニューロンに与えられる結合荷重の値）を評価に用いる。これは、過学習が発生しない理想的な学習を実施できた場合を想定することを意味する。なお、エポック数は、活性化関数に tanh を用いる場合を想定して学習した場合に誤差関数の値が十分収束する値を採用した。各結合荷重の初期値  $w_{ini}$  は、文献 [14] にて推奨されている方法を採用し、式 (14) に示す一様分布  $p(w_{ini})$  を母集団分布として標本抽出した。

$$p(w_{ini}) = \begin{cases} \frac{1}{2r} & (-r \leq w_{ini} \leq r) \\ 0 & (\text{Otherwise}) \end{cases}$$

$$r = k \sqrt{\frac{6}{f_{in} + f_{out}}} \quad (14)$$

表 2 学習の条件

パラメータ	値
MNIST の学習係数	0.01
CIFAR-10 の学習係数	0.001
バッチサイズ	32
結合荷重の調整回数 (エポック)	1,000

ここで,  $f_{in}$ ,  $f_{out}$  はニューラルネットワークの入力数および出力数を表す. また,  $k$  は係数であり, sigmoid 関数を用いる場合 4, それ以外の場合は 1 である. 光強度を用いる場合においては, 結合荷重の初期値を 0 以上とするため, 式 (15) に示す一様分布を母集団分布として標本抽出した.

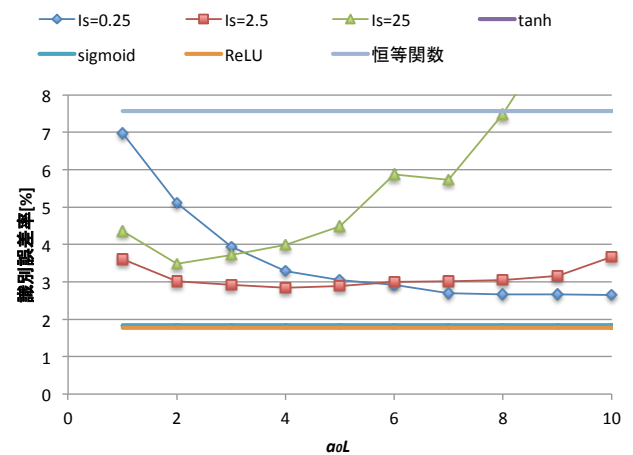
$$p(w_{ini}) = \begin{cases} \frac{1}{r} & (0 \leq w_{ini} \leq r) \\ 0 & (Otherwise) \end{cases} \quad (15)$$

ニューラルネットワークは Theano[13] を用いて実装した.

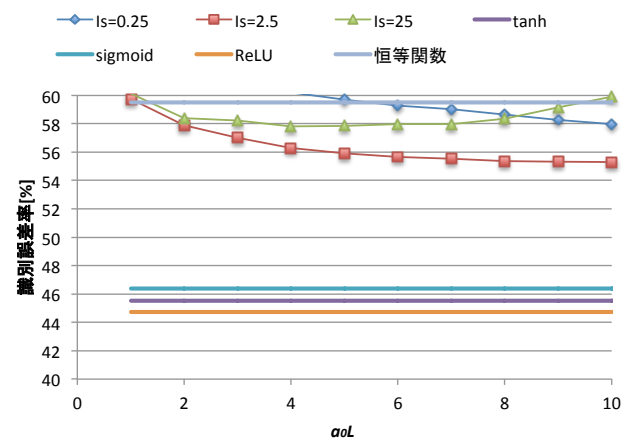
#### 4.2 評価結果

まず, デバイスパラメータ  $a_0L$  が識別精度に与える影響を議論する. 図 5 に, 光強度を用いた場合の可飽和吸収体のパラメータと識別精度の関係を示す. 横軸はパラメータ  $a_0L$ , 縦軸は過学習が発生しない理想的な学習を実施できた場合の識別誤差率である. 光強度を情報の媒体とした活性化関数は, 多くの場合, 中間層がないニューラルネットワークに比べて高い識別精度を達成している. しかしながら, 一般的な活性化関数を用いたニューラルネットワークに比べると識別精度は劣化している. これは, 活性化関数の定義域が 0 以上に制約されたことが原因であると考えられる. 一方, 光電界振幅を用いた場合, 図 6 に示すように, デバイスパラメータを  $a_0L \geq 5$ ,  $E_s = 0.5$  と設定した場合に一般的な活性化関数と同等の識別精度を達成した. 以上より, いずれの場合においても活性化関数として価値があると判断できるが, 識別精度を最優先する場合には光電界振幅を情報の媒体として用いることが妥当であると判断できる.

次に, 光強度ならびに光電界振幅を用いた場合の個別デバイスパラメータ  $I_s$  ならびに  $E_s$  と, 双方に共通するパラメータ  $a_0L$  の依存性を分析する. 図 5 ならびに図 6 より,  $I_s$  や  $E_s$  が大きな値となる場合,  $a_0L$  の増大に伴い識別精度が低くなっていることが分かる. これは, これら 3 種のパラメータが大きな値をとる場合, 学習初期における活性化関数の出力がほぼ 0 となり, 学習が進まないためであると考えられる. 例えば光強度を用いる場合, 図 4(a) から,  $I_s = 25$ ,  $a_0L = 5, 10$  といった大きい値をとり, かつ, 可飽和吸収体の入力強度が  $0 \leq I_{in} \leq 4$  と小さい場合には, 出力強度はほぼ 0 になることが分かる. また, 学習初期の活性化関数への入力  $X$  は, 式 (1), 式 (14), ならびに, ニューラルネットワークへの入力  $x$  が  $0 \leq x \leq 1$  に正規化されていることから, 必然的に小さな値となり, 最終的に



(a) MNIST



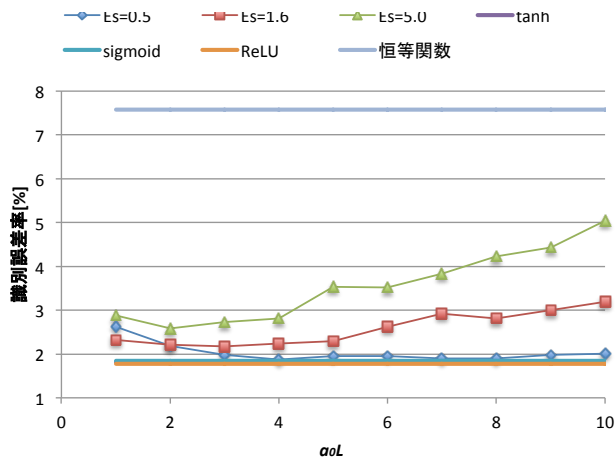
(b) CIFAR-10

図 5 光強度を用いた場合の識別精度

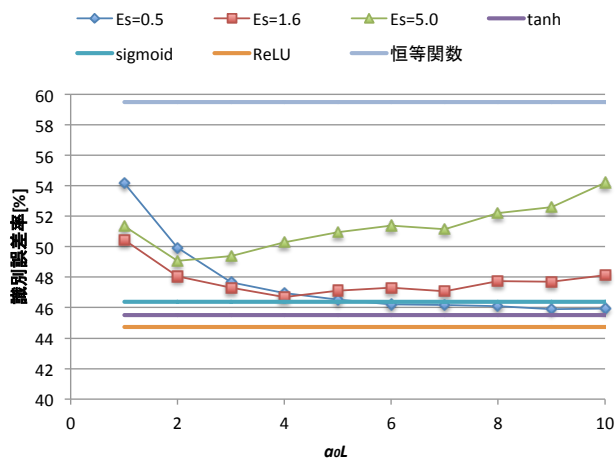
学習初期における活性化関数の出力がほぼ 0 となる現象が発生すると推察する. なお, 学習初期の活性化関数への入力  $X$  は結合荷重の初期値に依存する. そのため, 結合荷重の初期値を変更することで, 大きな  $I_s$  や  $E_s$  を用いた場合の識別精度を向上できる可能性がある. 一方, 図 5 ならびに図 6 より  $I_s$  や  $E_s$  が小さい場合,  $a_0L$  の増加に伴い識別精度が向上する傾向が見られる. 例えば, 図 4(a) において  $I_s = 0.25$  の曲線を見ると,  $a_0L$  が大きい程曲率が大きい. このことから, 曲率が大きいほど識別精度が高くなることが推察される.

#### 5. デバイスパラメータの考察

第 4.2 節で示したように, 可飽和吸収体のデバイスパラメータ  $I_s$  や  $E_s$  が小さな値を採る場合,  $a_0L$  の増加に伴い識別精度が向上する. 一方で,  $L$  が小さくなるほど可飽和吸収体を光が通過するレイテンシは短くなり, 素子サイズも縮小できる. そのため, 高い識別精度と小さい  $L$  を両立可能な可飽和吸収体の利用が望ましいと言える. ここで, 図 5 ならびに図 6 から,  $a_0L$  があるを超えると識別誤差率が一定の値に収束することが分かる. 例えば, 図 5(a) で



(a) MNIST



(b) CIFAR-10

図 6 光電界振幅を用いた場合の識別精度

表 3 好ましい可飽和吸収体のパラメータ

媒体	タスク	パラメータ	値
光の強度	MNIST の識別	$a_0L$	7
		$I_s$	0.25
光の強度	CIFAR-10 の識別	$a_0L$	8
		$I_s$	2.5
光電界 振幅	MNIST の識別	$a_0L$	3
		$E_s$	0.5
光電界 振幅	CIFAR-10 の識別	$a_0L$	6
		$E_s$	0.5

は、 $a_0L = 7$  を超えると識別誤差率が約 2.7% に収束している。以上を総合し、可飽和吸収体のデバイスパラメータとしては表 3 で示す値に設定することが好ましいと考える。

## 6. おわりに

本稿では可飽和吸収体による活性化関数を評価し、デバイスパラメータが識別精度に与える影響を解析した。評価の結果、情報の媒体として光電界振幅を用いる場合、一般的な活性化関数と同等の識別精度を達成できることが分かった。しかしながら、光電界振幅は光強度に比べナノフォトニックデバイスでの扱いが難しいといった特徴があ

る。そのため、ナノフォトニックデバイスでの扱いが容易な光強度を用いて高い識別精度を達成できることが望ましい。ここで、光強度を用いた場合に高い精度を達成できない原因は、活性化関数の定義域が正の制約されることであると考えられる。本評価では、式 (9) の様な簡易的な方法で制約を満たしたが、より高い精度を達成する学習方法が存在する可能性がある。例えば、光電界振幅を用いた行列演算により負の値を利用可能とし、その出力を光強度へと変換した上で、可飽和吸収体へと入力するといった方法が考えられる。そのため、今後は光の強度を情報の媒体とした場合の識別精度向上方法を検討する。一方で、光電界振幅は光強度に比べて制約がなく、高い識別精度を達成できることが分かった。そのため、光電界振幅を情報の媒体としてアクセラレータを実装していく方法についても今後検討していく予定である。

また、デバイスパラメータが識別精度に与える影響の解析から、各ベンチマークにおける好ましいデバイスパラメータを提示した。結果として、この好ましいパラメータは各ベンチマーク毎に異なった。しかしながら、可飽和吸収体のデバイスパラメータは物質やデバイスの設計に依存し、一度設定すると変更することができないため、多くのタスクにおいて汎用的なデバイスパラメータを検討する必要がある。そのため、今後はより多くのタスクに対して、デバイスパラメータと識別精度の関係を解析する。

また、本評価では可飽和吸収体のみ焦点を当てたが、他にも非線形な入出力特性を持つ素子は存在する。そのため今後は、可飽和吸収体以外の非線形素子によって実装される活性化関数の実用性についても検討する。

謝辞 日頃から議論頂く NTT 研究所 新家昭彦氏、京都大学 石原亨氏をはじめ、本研究プロジェクトメンバー各位に感謝します。なお、本研究は一部、JST CREST の支援による。

## 参考文献

- [1] Cavigelli, L., Gschwend, D., Mayer, C., Willi, S., Muheim, B. and Benini, L.: Origami: A convolutional network accelerator, *Proceedings of the 25th edition on Great Lakes Symposium on VLSI*, pp. 199–204 (2015).
- [2] Chen, T., Du, Z., Sun, N., Wang, J., Wu, C., Chen, Y. and Temam, O.: DianNao: A Small-footprint High-throughput Accelerator for Ubiquitous Machine-learning, *Proceedings of the 19th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pp. 269–284 (2014).
- [3] Ovtcharov, K., Ruwase, O., Kim, J.-Y., Fowers, J., Strauss, K. and Chung, E. S.: Accelerating deep convolutional neural networks using specialized hardware, *Microsoft Research Whitepaper*, Vol. 2 (2015).
- [4] Peemen, M., Setio, A. A., Mesman, B. and Corporaal, H.: Memory-centric accelerator design for convolutional neural networks, *2013 IEEE 31st International Confer-*

- ence on *Computer Design (ICCD)*, pp. 13–19 (2013).
- [5] Shafiee, A., Nag, A., Muralimanohar, N., Balasubramonian, R., Strachan, J. P., Hu, M., Williams, R. S. and Srikumar, V.: ISAAC: A Convolutional Neural Network Accelerator with In-Situ Analog Arithmetic in Crossbars, *Proceedings of the 43th Annual International Symposium on Computer Architecture (ISCA)*, pp. 14–26 (2016).
- [6] 川上哲志, 磯部 聖, 浅井里奈, 小野貴継, 本田宏明, 井上弘士, 納富雅也: ナノフォトニック・ニューラルアクセラレーション構想, 情報処理学会研究報告 (デザインガイア), Vol. 2016-ARC (2016).
- [7] Sibi, P., Jones, S. A. and Siddarth, P.: Analysis of different activation functions using back propagation neural networks, *Journal of Theoretical and Applied Information Technology*, Vol. 47, No. 3, pp. 1264–1268 (2013).
- [8] LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P.: Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, Vol. 86, No. 11, pp. 2278–2324 (1998).
- [9] Shen, Y., Skirlo, S., Soljacic, M., Englund, D. R. and Harris, N.: On-Chip Optical Neuromorphic Computing, *CLEO: Science and Innovations*, Optical Society of America, pp. SM3E–2 (2016).
- [10] Shen, Y., Harris, N. C., Skirlo, S., Prabhu, M., Baehr-Jones, T., Hochberg, M., Sun, X., Zhao, S., Larochelle, H., Englund, D. et al.: Deep Learning with Coherent Nanophotonic Circuits, *arXiv preprint arXiv:1610.02365* (2016).
- [11] Smith, P. W., Duguay, M. and Ippen, E.: Mode-locking of lasers, *Progress in Quantum Electronics*, Vol. 3, pp. 107–229 (1974).
- [12] Krizhevsky, A. and Hinton, G.: Learning multiple layers of features from tiny images (2009).
- [13] Team, T. D.: Theano: A Python framework for fast computation of mathematical expressions, *arXiv e-prints*, Vol. abs/1605.02688 (online), available from <http://arxiv.org/abs/1605.02688> (2016).
- [14] Bengio, Y.: Practical recommendations for gradient-based training of deep architectures, *Neural Networks: Tricks of the Trade*, Springer, pp. 437–478 (2012).