

# Neuroevolution による複数タスク学習 Multitask Learning by Neuroevolution

八尾泰洋<sup>1</sup> 橋本康弘<sup>1</sup> 陳昱<sup>1</sup> 大橋弘忠<sup>1</sup>

Yasuhiro Yao<sup>1</sup>, Yasuhiro Hashimoto<sup>1</sup>, Yu Chen<sup>1</sup>, and Hirotada Ohashi<sup>1</sup>

<sup>1</sup>東京大学システム創成学専攻

<sup>1</sup>Department of Systems Innovation, School of Engineering, The University of Tokyo

**Abstract:** We applied neuroevolution (NE) to multitask learning, and showed that NE can take advantage of learning multiple tasks. Also, we compared multitask learning by two evolutionary methods, standard NE and Neuroevolution of Augmenting Topologies (NEAT), to discuss about suitable evolutionary methods to multitask learning.

## 1. 研究の目的

本研究はニューラルネットワーク（以後、NN）の複数タスクの学習[1]に Neuroevolution（以下、NE）を適用する研究である。本研究は2つの主要目的を持つ。

第1の目的は、複数タスク学習に対してNEを適用することにより、進化的手法によって複数タスク学習の恩恵を受けることができることを示すことである。複数タスク学習については、教師あり学習を用いた研究報告が数多くなされているが、NEなどの強化学習によって複数タスク学習の相乗効果を示した研究は見られない。よって、本研究によりNEによる複数タスク学習の単一タスク学習に対する優位性が示されれば、進化計算の分野だけではなく強化学習の分野においても注目すべき結果となると考えている。

第2の目的は、複数タスク学習に適した進化計算の手法を提示することである。NNの進化的学習手法の分野では、Stanleyらによって Neuroevolution of Augmenting Topologies（以下、NEAT）という手法が提案され、大きな成功を収めている[2]。よって本研究においては、従来からの Standard Neuroevolution(SNE)だけではなく、NEATを複数タスク学習に適用し、

どちらの進化的手法が複数タスク学習に効果を発揮するかを検討した。

以上のような2つの研究目的のため、複数タスクの学習に進化計算手法のアプローチを適用し、研究を行った。

## 2. 複数タスク学習の単一タスク学習に対する優位性

計算機実験により、複数タスク学習の単一タスク学習に対する優位性を示す。3×4マスからなるフィールドを塗りつぶすゲームを課した(Figure 1)。

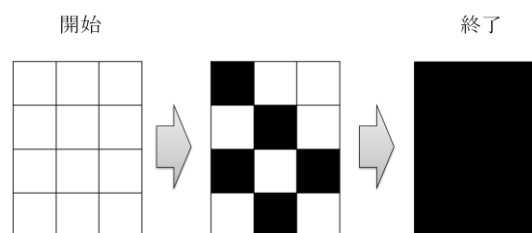


Figure 1: ゲームの概要

このゲームにおいて、NNは少ない手数でフィールドを塗り終えるほど、高い評価を得る。NNは2種類のスプレーを与えられ、スプレーの選択と、座標の選択という2つのタスクを課される。NNは2種類のスプレーを適切に使い分けることで最高の適応度を得る。学習したNNは2種類で、複数タスク NN(Figure 2)と単一タスク NN(Figure 3)のペアである。

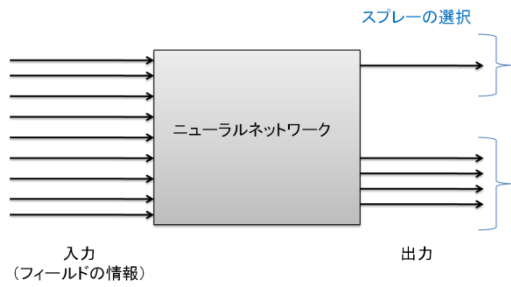


Figure 2: 複数タスク NN

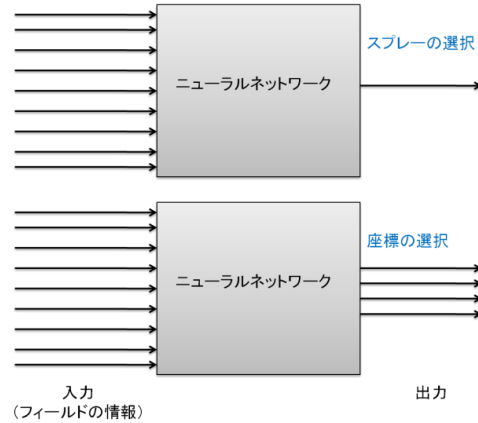


Figure 3: 単一タスク NN ペア

学習はいずれの NN に対しても SNE によって行い、突然変異率などのパラメータは共通とした。これにより、NN の進化に差が現れれば、それは進化の方法ではなく、NN の複数タスク学習の方法に起因すると考えることができる。以下、様々なノード数の NN に 5000 世代の進化を 20 回行った結果を示す。以後、NN(X)とは中間ノード数が X の複数タスク NN を、NN(X,Y)とは中間ノード数が X の NN と、Y の NN の単一タスク NN のペアを表わす。

Figure 4 に 5000 世代を経た時点ですべてのマス目を塗り終えることができる NN が出現なかった進化の数を示す。

Figure 4 から、NN(10)と NN(9)の複数タスク NN の方が他の単一タスク NN ペアよりも、すべてのマス目を塗り終えることができるような戦略が進化しやすいことが分かる。

Figure 5 に、1000 世代以内にすべてのマス目を塗り終えることができるような NN が出現した進化の数を示す。

Figure 5 から、この指標では NN(10)や NN(9)に匹敵するのは単一タスク NN ペアでは NN ペア(9,9)程度であることがわかる。ここで、NN ペア(9,9)は中間ノードが 9 個の NN を 2 つもつことに注目しなければならない。このペアと同程度の成績を収めることに関して、複数タスク NN では中間ノードが 9 の NN を 1 つもつことで実現できている。

Figure 6 に、5000 世代を経た時点で最小手数 of 8 手ですべてのマス目を塗り終えることができるような手順が出現した進化の数を示す。

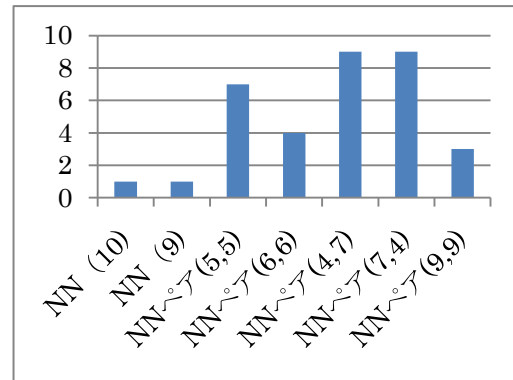


Figure 4: 5000 世代ですべてのマス目を塗り終えられなかった進化の数

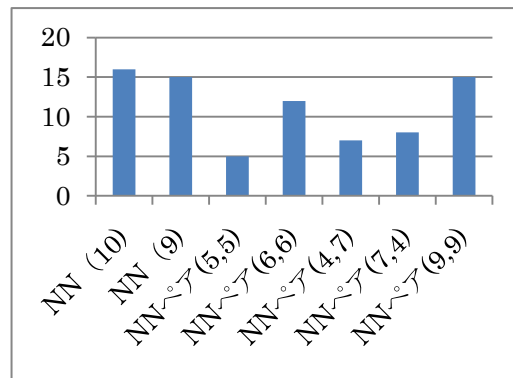


Figure 5: 1000 世代以内にすべてのマス目を塗り終えることができた進化の数

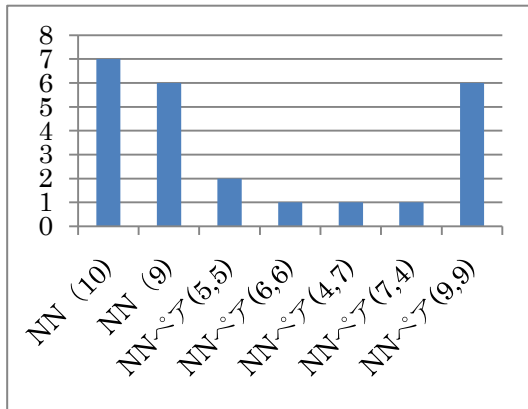


Figure 6: 5000 世代で最短の手数でマス目を塗り終えることができた進化の数

Figure 6 から、この指標では NN(10)や NN(9)に匹敵するのは単一タスク NN ペアでは NN ペア(9,9)程度であることがわかる。

複数タスク NN は単一タスク NN ペアよりも、同じノード数、同じ結合数程度の場合は優れた成績を示すことがわかる。単一タスク NN ペアは、合計で 2 倍程度のノードを持っているような場合で、複数タスク NN と匹敵するような成績を示した。しかしそれでもなお、5000 世代を経た時点ですべてのマス目を塗り終えるような個体が進化しているかという指標で見れば、複数タスク NN の方が優れた結果を示した。

以上のように、本計算では複数タスク NN が同時に複数のタスクを学習することによってその恩恵をうけ、単一タスク NN を凌駕する成績を示した。複数タスクの学習では、1 つのタスクの学習が他のタスクの学習に有利に働き、このような結果が得られたと考えられる。

### 3. NEAT の複数タスクへの応用

NEAT は近年活発な研究がされている NN の進化的な学習手法である。本研究では NEAT と SNE を用いて NN の学習を行

い、その結果を比較した。さらに、本計算では 2 種類の NN について学習を行った。1 種類目は出力をタスクごとに分割する通常の複数タスク NN(以後、NN1)(Figure 7)、2 種類目は入力によってタスクに合わせて出力を変える複数タスク NN(以後、NN2)(Figure 8)である。

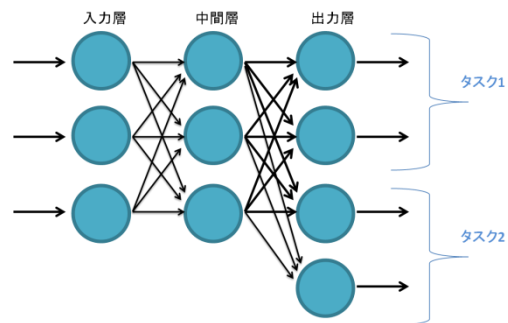


Figure 7: NN1

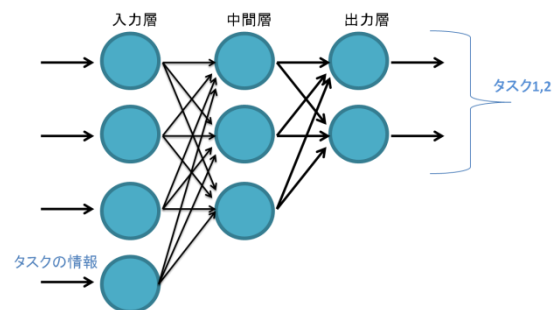


Figure 8: NN2

NN2 は生物がするように自身で行うべきタスクを判断して行うモデルであり、また、出力でタスクを限定していないので、使い方によれば後からタスクを追加していくことができるようなモデルである。

NN に課したタスクは、フィールド上の赤いトークンを集めるタスク、または青いトークンを集めるタスクである。NN は目的のトークンを獲得することで得点し、目的でないトークンを獲得することで減点される。NN はフィールド上を隣接する縦横斜めに隣接するマスの情報（加えて NN2 ではどちらのタスクを行うのかという情報）

から判断して上下左右のいずれかの方向に1マスずつ動くことでトークンを獲得していく (Figure 9)。フィールドは毎世代ランダムで生成され、偶然あるフィールドに適した個体が出現して全体に広がることを阻止した。

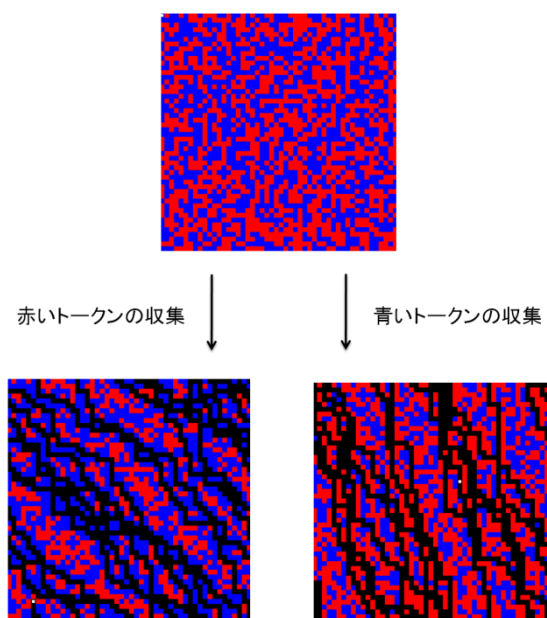


Figure 9: ゲームの様相

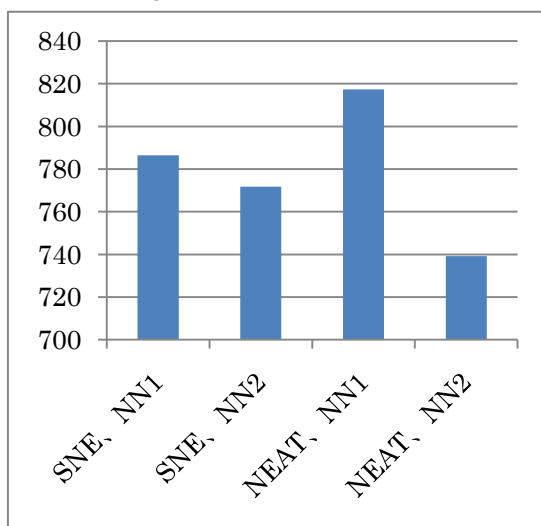


Figure 10: 進化の結果

計算機実験では NN1、NN2 を NEAT と SNE によって 3000 世代にわたって進化させた。2000 世代以降の最高得点の平均値を Figure 10 に示す。なお、得点は目的のト

クンと目的でないトークンの獲得数の差を 2 つのタスクについて合計した値である。

これらの結果から、NEAT は SNE に対して NN1 では優れた結果を示すが、NN2 での結果は劣っていることが分かる。これは、NEAT は一般的に SNE よりも優れた手法であり、タスクごとに出力を分割した場合の複数タスクの学習ではその威力を発揮したが、2 つのタスクで出力を共有した場合には、タスクの情報の変化という全体からみればわずかな入力の変化で出力を大きく変化させる必要があり、そのような振る舞いの変更に必要な構造が NEAT では進化の過程で獲得されなければならないためであると考えられる。構造を初期状態から決めている SNE ではある程度最適な構造を持っていることが保証されているので、最適な構造を生み出すことが難しい NN2 では NEAT よりも SNE が良い結果を示したと考えられる。

#### 4. 結論

本研究では、複数タスク学習による恩恵は NE によって享受できること、また適用法次第であるが NEAT は複数タスク学習の 1 つの選択肢として有効であることを示した。

#### 参考文献

- [1] Rich Caruana (1997), Multitask Learning, *Machine Learning*, 28, 41-75.
- [2] Kenneth O. Stanley and Risto Miikkulainen (2002). Evolving Neural Networks through Augmenting Topologies. *Evolutionary Computation*, 10(2): 99-127.