

G-008

構造進化型ニューラルネットワークへの強化学習と単純化の導入

Introduction of reinforcement learning and simplification to
NeuroEvolution of Augmenting Topologies

赤池 誠吾*

Seigo Akaike

鳴原 知希†

Tomoki Sigihara

服部 元信‡

Motonobu Hattori

1 はじめに

一般に、ニューラルネットワークはその構造が複雑であれば、より複雑な問題に対応することができるが、重みを最適な値にするのに時間がかかる。そこで、進化的アルゴリズムを用いて最適な構造と、重みを同時に獲得しようとする手法として Neuro Evolution of Augmenting Topologies (NEAT) [1] が提案されている。

NEAT は、1つのネットワークを個体として扱う。様々な個体の集団が存在し、生物の進化と同様に、突然変異や、交叉、淘汰を繰り返して、環境への適応度の高い構造、重みを持つ個体を獲得する。NEAT では突然変異によって重みがランダムに変化する。しかし、試行錯誤的に重みの組み合わせを調べていくため、重みの探索が非効率である。また、接続やノードが付加されるだけなので構造の肥大化が起こるといふ問題点がある。

そこで、本研究では突然変異によってランダムに設定される重みを、より効率的に獲得するために、NEAT に Direct-Vision-Based (DVB) 強化学習を取り入れた手法を提案する (以下、DNEAT)。また、この手法に、重みの小さい接続を削除する単純化 [2] を導入する。これにより、構造の肥大化を抑制しつつ効率良く重みの探索を行うことができるようになることを期待できる。さらに、構造の多様性を強化する組み換え突然変異を提案し、その効果を調べる。

2 NEAT

NEAT は次の3つのランダムな突然変異を用いて、最適な構造、重みを探す。

- 重み変更：重みをランダムに変更
- 接続追加：ノードを繋ぐ新しい接続を追加
- ノード追加：新しいノードを追加

NEAT の特徴として、交叉と多様性の保護がある。交叉は、親の特徴を引き継いだ個体を生み出す進化的操作である。しかし、複雑な異なる構造の個体をどう組み合わせるかという問題がある。NEAT では、同じ時期に同じ場所にできたノードの番号を統一させて、異なる構造の対応関係を把握し、これを容易にしている。多様性は様々な構造や重みを持つ個体が集団に存在するということである。新しくできたばかりの個体は、その後の発展の可能性があるにもかかわらず、適応度が低いことがあり、これをどのように、保護するかという問題がある。NEAT では、似た構造と重みを持つ個体をまとめて1つの種とし、他とは異なる新しい構造や重みを持つ個体が生まれたときには、それを新種として扱い、適応度が低くても淘汰されないように保護している。これにより、個体集団に多くの種が存在することになり、より多様な構造や重みを探索することが可能になる。

3 NEATの改良

本研究では段階的に以下の3つの手法を NEAT に導入する。

3.1 DVB 強化学習

DVB 強化学習 [3] では、観測された状態を直接入力に与え、出力の値を制御に用いる。通常のネットワークに現在の状態評価値を出力するノードを加え、強化学習を用いて、行動前と後の状態から学習の指標となる教師信号を作成し、誤差逆伝搬法でネットワークの出力結果をその教師信号に近づけていく。

効率的な重みの獲得のために、DNEAT では、NEAT に DVB 強化学習を取り入れる。DNEAT ではまず NEAT と同じように構造を獲得する。そして、DVB 強化学習で重みを学習する。

3.2 単純化

0に近い重みは不要である可能性が高いため、単純化アルゴリズム [2] では、その接続を確率的に削除する。不要な接続を削除することで、構造の肥大化防止、適応度の向上、新しい構造を獲得する効果が期待できる。以下、NEAT に単純化を導入した既存手法を NEAT-とする。

DNEAT に単純化を導入した提案手法を DNEAT-とする。DNEAT-では、DNEAT と同じように構造の獲得と、重みの学習を行い、その重みの大きさに応じて接続を削除する。この繰り返しで、世代を重ねていく。既存手法では、重みの獲得はランダムな突然変異で行われた。しかし、本手法では、重みを学習によって獲得するため、0に近い重みは不要である可能性が極めて高いと判断できる。そのため、本手法では、一定の値以下の重みを持つ接続を一括して削除した。

3.3 組み換え突然変異

本研究では、単純化により種の増加が抑制されないように、単純化と構造付加を両立させる手法として、「組み換え突然変異」を提案する。DNEAT-に対して組み換え突然変異を導入したものを DNEAT-C とする。組み換え突然変異では単純化を行った個体に対して、その構造が新種であれば保存する。一方、それが既存の構造であれば、新しい構造が出現するように、新たに接続を付加する。これにより、多様性が強化され、効率的な探索が可能になると期待できる。

4 計算機実験

ここでは提案手法 (DNEAT, DNEAT-, DNEAT-C) と既存手法 (NEAT, NEAT-) の性能を比較するため、シミュレータ上で、ロボットが障害物を回避し光に到達する行動 (以下、障害物回避光到達) と3目並べの学習 [2] を行った。

*山梨大学大学院医学工学総合教育部, 甲府市

†PASOKA, 桐生市

‡山梨大学大学院医学工学総合研究部, 甲府市

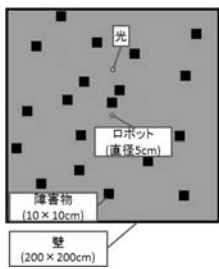


図 1: 障害物マップ

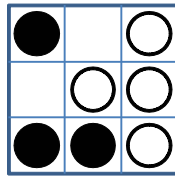


図 2: 3目並べ

表 1: 実験条件

	障害物回避光到達	3目並べ
入力ノード数	16	10
出力ノード数	3	10
初期接続数	48	100
重み変更突然変異確率:DNEAT	10%	10%
重み変更突然変異確率:NEAT	80%	75%
接続追加突然変異確率	30%	10%
ノード追加突然変異確率	10%	5%
世代数	150 世代	50 世代

4.1 障害物回避光到達

この実験では、障害物や壁への衝突を回避しながら、目標である光に到達するタスクをロボットに獲得させた。ロボットは光センサと赤外線センサがそれぞれ前方に6個、後方に2個ついたものを使用した。ネットワークへの入力はそれら16のセンサ値とし、出力は左右の車輪速度と、状態評価値の3つとした。初期ネットワークは入力ノードと出力ノードが全てつながり、接続数は48であった。環境を図1に示す。毎回、ロボットのスタートの向きと、20個の障害物の配置はランダムとした。

実験では、初めにロボットと光の距離を5cmとした。10世代毎に成功率の高い個体を1つ選び、100回テストを行い、60%の確率でゴールできたら、5cmずつ離れた。最終的に距離40cmで成功率80%に達したときにタスク達成とした。適応度と罰、報酬はロボットと障害物、光との距離で設定した。ロボットが障害物に近づくにつれ適応度を大きく減らし、光に近づくほど適応度を大きく増加させた。同様に、障害物に近いほど大きい値の罰を与え、光に近づくほど、大きい値の報酬を与えた。

表2ではタスク達成数以外は10試行の平均値を示す。なお、計算時間は5試行の平均である。最終成功率は、各試行における最終的な成功率を示している。

表 2: 障害物回避光到達実験結果

	NEAT	NEAT-	DNEAT	DNEAT-	DNEAT-C
接続数	103	78	86	71	74
最終成功率 (%)	54	61	68	72	79
タスク達成数	0	2	3	4	6
最終種数	23	12	16	12	21
時間 (秒)	3524	6889	8346	39974	5784

表2より提案手法のDNEAT-Cでは、既存手法に比べ、最終世代での平均接続数が少なかった。成功率は提案手法においてDNET, DNEAT-, DNEAT-Cの順に高くなった。接続数が少なく、成功率が上がっているため、無駄な接続を削除した、コンパクトで有効な学習が行われたと考えられる。また、DNEAT-と比べると、DNEAT-Cの方が種の数が多く、実行時間が短かった。DNEAT-Cが強化学習を用いているにもかかわらずNEATに次ぐ速さ

で学習が終了したのは構造の多様性の強化により、良い個体が生まれたため、素早く光に到達することが可能になったためと考えられる。

4.2 3目並べ

3目並べは3×3の盤上で相手と交互に自分のコマを打ち合い、先に縦横斜めのうち1列に自分のコマを3つ並べた方が勝ちというゲームである。学習は、文献[2]と同様に、5種類の敵と先攻後攻の計10回対戦し、勝敗に応じて適応度を加算し、勝ちを5点、分けを2点、負けを0点とした。強化学習の報酬と罰は、勝ちを10点、分けを5点、負けを-10点とした。ネットワークへの入力は盤面に対応した9つのノードとバイアスノード1つの計10個とし、出力は盤面に対応した9つのノードと、状態評価値のノードの計10個とした。それらが全てつながり、初期ネットワークは100の接続を持つ。最終的に得られた適応度の最も高い個体を、各相手と1万回ずつ対戦させたときの敗北率を表3に示した。実験の結果は5試行の平均である。

表 3: 3目並べ実験結果

	NEAT	NEAT-	DNEAT	DNEAT-	DNEAT-C
敗北率 (%)	50	45	45	43	41
最終種数	3	23	29	26	36
時間 (秒)	670	13495	5622	16027	28037

敗北率は提案手法でDNEAT-Cが最も低くなったが、時間も最もかかった。DNEAT-は次ぐ結果となった。種の増加が必ずしも良い結果につながるわけではないが、DNEATとDNEAT-Cは既存の手法に比べて種の数が多い傾向であった。また、単純化を取り入れた手法が時間がかかる傾向であった。単純化では、全ての接続を見て、接続を削除していくため、結果的に時間がかかった可能性がある。

5 まとめ

本研究では、強化学習を用いた構造進化型ニューラルネットワークの性能向上を目的とし、DVB強化学習と単純化、組み換え突然変異の導入を行った。効率的な重み獲得と構造変化の結果として、2つの計算機実験において、性能の向上が確認できた。

一方で、タスクによっては実行時間の顕著な増加が見られた。このため、今後の研究では構造の付加や削除における時間短縮などを検討する必要がある。

参考文献

- [1] K.O. Stanley and R. Miikkulainen, "Evolving Neural Networks Through Augmenting Topologies," *Evolutionary Computation*, vol.10, no.2, pp.99-127, 2002.
- [2] D. James and P. Tucker, "A comparative analysis of simplification and complexification in the evolution of neural network topologies," *Proceedings of the Genetic and Evolutionary Computation Conference*, 2004.
- [3] 柴田 克成, 岡部 洋一, 伊藤 宏司, "ニューラルネットワークを用いた Direct-Vision-Based 強化学習-センサからモータまで-", *計測自動制御学会論文集*, vol.37, no.2, pp.168-177, 2001.