

NEATにおける振る舞いに基づく多様性維持

Sustaining Behavioral Diversity in NEAT

森口博貴^{*1}
Hiroataka Moriguchi本位田真一^{*1*2}
Shinichi Honiden^{*1} 東京大学
The University of Tokyo^{*2} 国立情報学研究所
National Institute of Informatics

Niching schemes, which sustains population diversity and let an evolutionary population avoid premature convergence, have been extensively studied in the research field of evolutionary algorithms. Neuroevolutionary algorithms, such as NEAT, have also benefitted from niching. However, the latest research indicates that the typical use of genotype- or phenotype-similarity-based niching schemes will result in poor performance. In this paper, we propose a novel niching scheme that takes into consideration both the phenotypic and *behavioral* similarity, and then integrate it with NEAT. An experimental analysis revealed that the proposed algorithm outperforms the original NEAT for various problem settings. More interestingly, it performs especially well for problems with a high noise level and large state space.

1. はじめに

進化型神経回路網は、進化的アルゴリズムを用いてニューラルネットワークを最適化する手法であり、多くの学習問題に有効であることが知られている [Yao 99]. とくに、強化学習問題におけるポリシー探索手法としての応用に多くの成功例が見られる [Floreano 08].

しかしながら、進化型神経回路網によるニューラルネットワークの探索は、しばしば初期収束という現象に悩まされる. これは一般的な進化的アルゴリズムにも共通する課題で、解集団が大域最適解を発見する前に局所最適解へ収束する現象を指す. 初期収束は最適解の発見を阻害するため、進化的アルゴリズムの研究者は Niching という技術を用いてこれを回避する方法を研究してきた [Sareni 98]. 典型的な Niching は、集団内の遺伝的多様性を維持する仕組みを導入することで初期収束を回避する. しかしながら、進化型神経回路網では遺伝的な多様性の維持が必ずしも初期収束の回避につながらないことが近年の研究で示されている [Lehman 08]. 進化型神経回路網の場合、一般的な進化的アルゴリズムと異なり各個体の適応度は遺伝的な情報そのものではなく環境下での振る舞いをもとに測定される. そのため振る舞いの多様性維持が効果的であると考えられ、いくつかの既存研究でその有効性が示されている [Mouret 09, Gomez 09].

本研究では進化型神経回路網の中でも NEAT [Stanley 02] を取り上げ、Niching の性能改善を図る. NEAT はニューラルネットワークのトポロジと結合ウェイトを同時に最適化する種類の進化型神経回路網であり、また最も広く認知されている進化型神経回路網の一つでもある. NEAT は個体の表現型ニューラルネットワークのトポロジに着目した Niching を導入しており、Niching を利用しない場合に比べ探索性能が向上することが示されている. 本研究では、トポロジと振る舞いの双方を考慮した Niching 手法を提案し、その手法が NEAT の探索性能をさらに向上させることを示す. 我々の知るかぎりでは、進化型神経回路網における振る舞いとトポロジの双方を考慮した Niching 手法の提案はこれが初めてである.

また、我々は Kalyanakrishnan らの提案したベンチマーク問題 [Kalyanakrishnan 09] を用いて提案手法の性能を評価した. このベンチマークでは、強化学習問題に内在するマルコフ決定過程の特性をパラメータを介して制御することで、多種多様な学習問題をシミュレートできる. 我々はこれを用いた実験を通じて、提案手法が広範な問題設定で NEAT の性能を上回り、特に知覚ノイズと問題規模が大きい場合に大幅な性能改善が見られることを示す. 現実的に進化型神経回路網の応用が期待されるような問題の多くでは、知覚ノイズや問題規模が大きくなることから、提案手法は進化型神経回路網の応用的な価値を高めると考えられる.

2. 関連研究

いくつかの既存研究で、進化型神経回路網における Niching に個体の振る舞い情報を利用する手法が提案されている [Mouret 09, Gomez 09]. 進化型神経回路網においては、適応度は個体の遺伝子型や表現型から直接導かれるわけではなく、実行時の振る舞いをもとに算出される. また振る舞いは個体と環境との複雑な相互作用によって生み出されるので、遺伝子型・表現型から振る舞いを予測するのは困難である. そのため、Niching では遺伝子型・表現型よりも振る舞いに着目すると都合がよいと考えられている.

例えば、Gomez [Gomez 09] は振る舞いに基づく Niching を単純な進化型神経回路網—ニューラルネットワークのトポロジは固定で結合ウェイトのみを最適化する—に、導入した. その結果、遺伝子型や表現型、あるいは適応度に着目した Niching に比べ、高い性能を導くことが示されている.

また、Trujillo ら [Trujillo 08] は NEAT における種分化に振る舞い情報を用いる手法を提案した. この手法では NEAT で利用されるトポロジ類似度尺度を、単純に振る舞いに基づく尺度で代用した. 彼らの研究では、この手法を用いることで探索される解集団が多様な振る舞いを獲得することが確認された. しかしながら、探索性能の向上には至っていない.

我々の研究では、Trujillo らと同様、NEAT を基盤となるアルゴリズムとして選択した. しかし、Trujillo らが振る舞いのみに着目した Niching を用いたのに対し、我々はトポロジと振る舞いの双方を考慮する手法を提案する. 我々の知る限り、

連絡先: 森口博貴, 東京大学情報理工学系研究科コンピュータ科学専攻, hmori@nii.ac.jp

トポロジと振る舞いの双方を考慮した Niching の提案は本研究が初めてである。また、Trujillo らの手法では NEAT に対する性能の向上が見られなかったのに対し、本研究では提案手法の性能が様々な問題設定において NEAT の性能を上回ることを示す。

3. 手法

この節では、まず NEAT で利用される Niching の詳細を説明する。その上で、我々の提案手法の詳細を示す。

3.1 NEAT の Niching

NEAT の Niching は二種類の既存技術—種分化と適応度共有—を組み合わせて利用する。前者は革新的なトポロジの保護が、後者はトポロジの多様性維持が目的である。

種分化は集団内に種を生成し、トポロジの類似する個体同士が同じ種に属するように個体をそれぞれの種に振り分ける。種をまたいだ交配は認められないというルールを持ち、個体間の生存競争はそれぞれの種の内部に限定される。そのため、革新的なトポロジを持つ個体は自身を含む新たな種を形成し、他の種に属する個体との競争を避けることができる。その結果、革新的なトポロジが保護されるというわけだ。

個体の属すべき種の決定にはトポロジの類似度を定量的に測る尺度を利用する。NEAT においてはニューラルネットのノードと結合がそれぞれひとつの遺伝子で表現されるため、二つの個体 i, j 間のトポロジ類似度 $\delta_T(i, j)$ は、に個体に共有しない遺伝子の超過数 E 、不足数 D 、及び共通する結合遺伝子の平均結合強度差 $\bar{W}$ を用いて、

$$\delta_T(i, j) = \frac{c_1 \cdot E}{N} + \frac{c_2 \cdot D}{N} + c_3 \cdot \bar{W}, \quad (1)$$

として定式化される。ここで c_1, c_2, c_3 は各変数の寄与度を調節する係数で、 N は正規化項である。本研究における評価では、 N の値を慣例的に 1 に設定した [Stanley 02]。このように定量化された類似度が、事前に決められた閾値 δ_t より小さければ二つの個体が同じ種に、そうでなければ別の種に属するように振り分けられる。

これに対し、適応度共有では個体の適応度をその個体の属する種のサイズで割ることで、多数派の種に対してより大きな適応度のペナルティを課す。その結果、小規模な種に属する個体はより多くの子を生む機会が与えられ、集団全体での多様性が維持される。個体 i の修正後の適応度 f'_i は、式 1 で定義したトポロジ類似度尺度 δ_T を用いて以下の修正式で求められる。

$$f'_i = \frac{f_i}{\sum_{j=1}^n sh(\delta_T(i, j))}, \quad (2)$$

ここで、共有関数 sh は以下のように定義された関数である。

$$sh(\delta_T(i, j)) = \begin{cases} 0 & (\delta_T(i, j) > \delta_t) \\ 1 & (\text{else}). \end{cases}$$

3.2 提案手法：振る舞いに基づく適応度共有

本研究で提案する Niching は、NEAT と同様にトポロジ類似度尺度に基づく種分化を行う一方で、振る舞い類似度尺度に基づく適応度共有を行う。これは、NEAT における適応度の修正式 (式 2) の共有関数 sh を、振る舞い類似度を考慮した共有関数 sh_b に置き換えることで可能である。我々は、 sh_b を振る舞い類似度尺度 δ_B を用いて以下のように定義した。

$$sh_b(\delta_B(i, j)) = \begin{cases} 0 & (\delta_B(i, j) > \delta_{tb}) \\ 1 - (\frac{\delta_B(i, j)}{\delta_{tb}})^\alpha & (\text{else}), \end{cases} \quad (3)$$

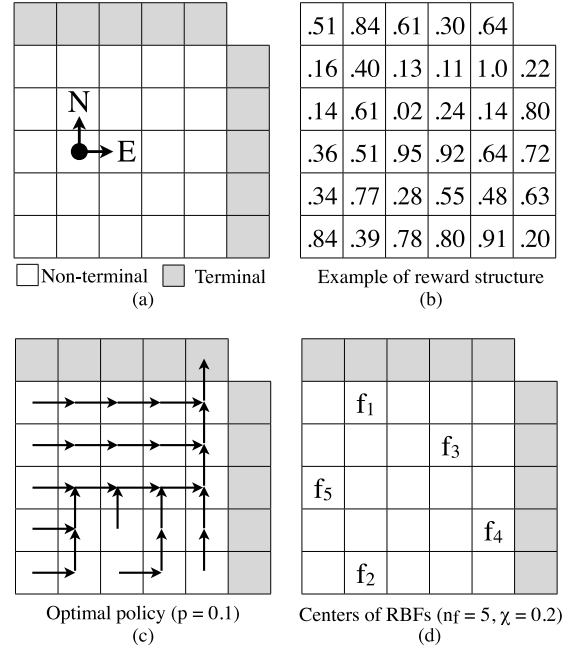


図 1: ベンチマーク問題の概要. (a) グリッドマップ及びエージェント配置例 ($s = 6$), (b) 各セルへの報酬割り当て例, (c) (b) に基づく最適ポリシー例 ($p = 0.1$), (d) RBF の配置例 ($\chi = 0.2, n_f = 5$).

ここで α は共有関数の形状を調整する定数、 δ_{tb} は閾値定数である。我々の提案手法は、この共有関数を用いることでトポロジ類似度に基づく種分化と振る舞い類似度に基づく適応度共有の両立を可能とする。

4. 実験

この節では提案手法を評価する実験環境の詳細と、そこで得られた結果を示す。われわれは学習タスクとして Kalyanakrishnan ら [Kalyanakrishnan 09] の提案したベンチマーク問題を利用する。これは単純な強化学習問題であるが、三つの好ましい性質を持っている。第一に、強化学習問題に内在するマルコフ決定過程のいくつかの特性—入力情報量、知覚・行動ノイズ、問題サイズ—をパラメータを介して制御できること。第二に、最適解と最劣解が動的プログラミングを用いて容易に計算できること。第三に、一試行に要する時間が短いながらも、十分に学習が困難であること。これらの性質は、多彩な問題設定における、学習アルゴリズムの定量的な性能評価と普遍的な結論の導出を可能とする。

4.1 ベンチマーク問題の詳細

図 1 はベンチマーク問題の概要を表している。実行環境は図 1(a) に示すようなグリッド状のマップと、そこに配置された一体のエージェントで構成される。初めにエージェントはマップ上のランダムな場所に配置され、また各セルには報酬が $[0:1]$ の範囲からランダムに与えられる。エージェントはセルを訪れることで対応する報酬を獲得でき、それを最大化することがタスクの目的である。エージェントは北 (N) か東 (E) のどちらかにしか動くことができないうえ、マップの北端と東端には達した時点で試行が終了となるような終端セルが配置されているため、エージェントは現在の位置とマップ内の報酬配

置に基づいて、報酬の期待値がなるべく大きくなる経路を選ぶ必要がある。図 1(b, c) は報酬配置の例と、それに基づく最適ポリシーの例である。最適なポリシーは動的プログラミングを用いれば容易に計算できるが、学習問題を十分に困難なものとするため、エージェントの知覚できる情報には制限が設けられている。具体的には図 1(d) に示したような、いくつかの非終端セルを原点とした放射基底関数 (RBF) を用意し、それらの出力値のリストのみがエージェントには与えられる。エージェントのポリシーであるニューラルネットは、各 RBF の出力値を入力信号として受け取り、その情報をもとに適切な行動を選択しなければならない。

タスクの特性は、 s, χ, p, d の四つのパラメータで決定づけられる。 s はグリッドマップのサイズを決定し、 $(s-1)^2$ 個の非終端セルと $2 \cdot (s-1)$ 個の終端セルを含むマップを構築する。 χ は RBF の数 n_f を $n_f = \chi \cdot (s-1)^2$ 個となるよう制御する。図 1(d) に示したマップは $\chi = 0.2, s = 6$ の設定で構築した例で、 $0.2 \cdot 5^2 = 5$ 個の RBF を持つ。 p はエージェントの状態遷移の不確実性を制御し、例えばエージェントが北へ進む行動 (N) を選択した場合、実際に北に移動できる確率は $1-p$ で、 p の確率で誤って東に移動する。また、 d は RBF の出力値に $[-d : d]$ の一様分布から得られるノイズを追加することでエージェントの視覚ノイズを制御する。これら四つのパラメータを制御することで、多様な学習問題をシミュレートし、広範な問題設定において学習アルゴリズムを評価することが可能となるのだ。

あるマップ上での個体の性能は、複数回の試行で獲得された報酬を平均化すれば評価できる。しかし、学習アルゴリズムが発見した解の定量的な最適性を評価するためには、期待報酬を正規化できることが望ましい。そこで、エージェントが各セルから試行を開始した場合の最適 (最劣) 期待報酬を動的プログラミングで計算し、最適値と最悪値がそれぞれ 1, 0 を表すように期待報酬を正規化し個体の性能とする。平均期待報酬は、各個体につき 1,000 試行を行い算出する。

4.1.1 振る舞い類似度尺度の定義

振る舞い類似度に基づく適応度共有を行うため、我々は個体 i, j 間の振る舞い類似度尺度 $\delta_B(i, j)$ を、

$$\begin{aligned}\delta_B(i, j) &= \frac{1}{|C|} \sum_{c \in C} |P_N(c, i) - P_N(c, j)| \\ &= \frac{1}{|C|} \sum_{c \in C} |P_E(c, i) - P_E(c, j)|,\end{aligned}$$

として定義した。ここで、 C は非終端セルの集合、 $P_N(c, i)$ と $P_E(c, i)$ はそれぞれ個体 i がセル c において行動 N と E を選択した確率であり、1,000 試行にわたる位置-行動のログから計算される。

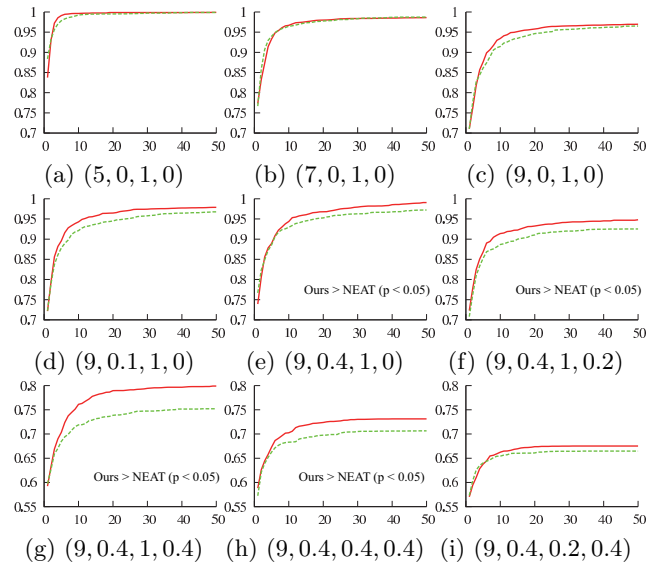
4.1.2 NEAT パラメータ

実験では、NEAT^{*1} の個体数は 50 に設定し、50 世代までの進化を行った。トポロジ類似度尺度 (式 2) の係数 c_1, c_2, c_3 はそれぞれ 1.0, 1.0, 2.0 に設定した。

トポロジ類似度尺度の閾値 δ_t は 3.0 に設定した。一方で、振る舞い類似度尺度の閾値 δ_b は 0.7 に設定した。また、式 3 のパラメータ α は慣例的に 1.0 に設定した [Stanley 02]。これらの設定はすべての実験において共通である。

4.2 実験結果

提案手法と NEAT の各世代における性能の算出には、その世代で最も性能の高い個体を追加の 10,000 試行に適用し得ら



Key:

x-axis: Elapsed Generations — Ours
y-axis: Normalized Performance - - - NEAT

図 2: 代表的な問題設定における提案手法と NEAT の学習曲線。各ラベルは評価に用いた問題設定 (s, p, χ, d) を示す。

れた期待報酬を利用する。また同一のパラメータで構築される 25 の異なるマップにおいて、それぞれ 50 世代にわたる性能を評価し、その平均を各アルゴリズムの性能とする。ただし、それぞれのアルゴリズムの評価には、同一の 25 マップのセットを利用した。学習エージェントがおかれる多様な環境をシミュレートするため、問題パラメータは下記の値をとる。

- 行動ノイズ: $p = \{0, 0.1, 0.2, 0.3, 0.4\}$,
- 知覚ノイズ: $d = \{0, 0.1, 0.2, 0.3, 0.4\}$,
- 問題サイズ: $s = \{5, 7, 9, 11\}$,
- 入力情報量: $\chi = \{0.2, 0.4, 0.6, 0.8, 1\}$.

合計で $5 \times 5 \times 4 \times 5 = 500$ の問題設定となる。

まず、図 2 に提案手法と NEAT を九つの代表的な問題設定において実行して得た学習曲線を示す。50 世代経過後の性能の違いが統計的に有意な差 ($p < 0.05$) を持つ場合には、グラフの中にその旨を表記した。図 2(a-c) はパラメータ p, χ, d を 0, 1, 0 に固定し、マップサイズ s を 5, 7, 9 と変化させた結果を示している。これらの図から、 s が低い場合にはそれぞれのアルゴリズムが最適に近い解を発見したのに対し、 s が高い場合には収束が若干遅まり、解の性能が低下したことがわかる。とはいえ、これら三つの問題設定では、提案手法と NEAT は同程度の性能を示し、50 世代経過後でも性能に有意な差は見られない。図 2(c-e) では行動ノイズ p を 0 から 0.4 まで段階的に変動させた。 p の値が大きくなるに従って、両アルゴリズムの探索の収束が遅まると同時に性能差が拡大した。とくに、 $p = 0.4$ となる場合では提案手法の性能が NEAT の性能を統計的な有意差をもって上回った。図 2(e-g) では知覚ノイズ d を増加させた。 d の増加に従い双方ともに性能が低下するが、その落ち込みは NEAT の場合の方が大きく、 $d = 0.4$ のよう

*1 NEAT implementation is based on rtNEAT version 1.0, <http://userweb.cs.utexas.edu/users/nn/keyword?rtneat>

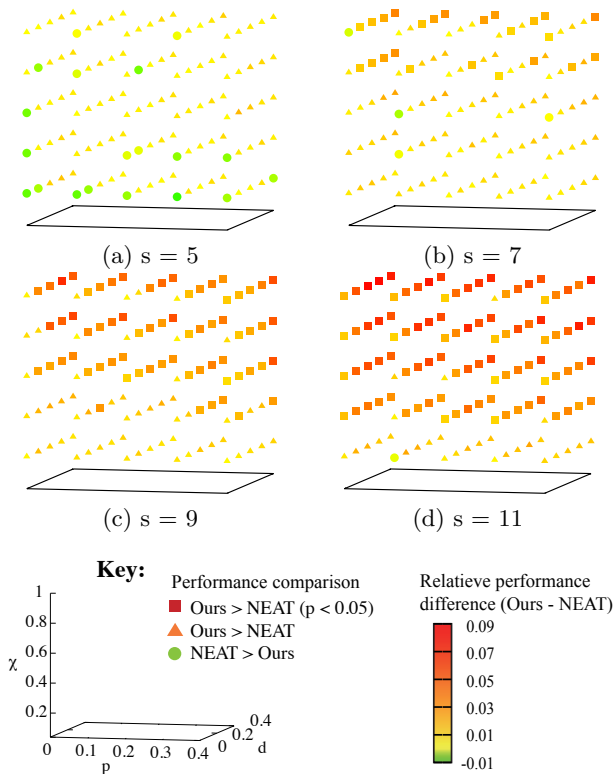


図 3: 50 世代経過後の提案手法と NEAT の比較.

に高いノイズレベルの下では提案手法に比べ大幅に性能の低い局所解に収束した。これらの結果は提案手法が NEAT に比べ、視覚ノイズに対して堅牢であることを示唆している。図 2(g-i) では入力情報量 χ を減少させた。 χ が減少するにつれ提案手法でも最適解には程遠い解しか探索できなくなる。しかしながら、依然として NEAT より高い性能を示した。以上九つすべての問題設定において、学習初期には両アルゴリズムが漸近的な性能向上を示す。しかしながら、多くの場合に NEAT が先に収束を始め、最終的に性能の低い解に収束した。この事実から、提案手法は NEAT に比べ効果的に初期収束を回避したと考えられる。

図 3 は、500 すべての問題設定での 50 世代経過後の性能を比較した結果を表している。図 3(a-d) はそれぞれ問題サイズ $s = \{5, 7, 9, 11\}$ の場合を示しており、各点の色 (グレースケール) は一つの設定における提案手法と NEAT の 50 世代経過後の性能差を表す。プロットされた点の形状は、四角、三角、丸がそれぞれ、提案手法の性能が NEAT の性能を、統計的に有意な差で上回った ($p < 0.05$)、上回ったが有意差はない、下回ったが有意差はない、ことを示している。どの設定においても、NEAT が提案手法の性能を統計的な有意差を持って上回ることにはなかった。図 3 の結果からは、NEAT が提案手法の性能を上回るのは入力ノイズや問題サイズが小さい場合においてのみであることがわかる。NEAT は 500 の設定の中で、24 の設定でのみ提案手法の性能を上回るが、そのうち 19 の設定では問題サイズが小さく ($s = 5$)、また 18 の設定では入力ノイズが無い ($d = 0$)。一方で提案手法は多くの設定—とくに χ と s が高い場合—において統計的に有意な差をもって NEAT の性能を上回った。とくに $s = \{9, 11\}$, $\chi = \{0.8, 1\}$ の場合には、ほとんどの設定において NEAT を有意な差で上回った。また、 d が増加するにつれて、性能差が拡大することも確認さ

れた。これらの結果は提案手法の導入が、数少ない例外を除いた幅広い問題設定において一貫した性能向上を導き、とくに視覚ノイズや問題サイズが大きい場合にその向上が顕著であることを示唆している。

5. まとめ

本研究では振る舞いと表現型を考慮した Niching 手法を提案し、NEAT に統合する方法を示した。ベンチマーク問題を用いた性能評価から、提案手法が幅広い問題設定において NEAT の性能を向上させることが確認された。特に知覚ノイズや問題サイズが大きい場合に提案手法がより大きな効果を発揮するということが示された。現実的に進化型神経回路網の応用が期待される問題はこれら二つの性質をもっていることが多いという事実とこれらの結果から、本研究の提案が NEAT の応用的価値を高めることが期待される。

参考文献

- [Floresano 08] Floresano, D. Dürer, P. and Mattiussi, C: Neuroevolution: from architectures to learning, Evolutionary Intelligence (2008)
- [Gomez 09] Gomez, F.: Sustaining diversity using behavioral information distance, GECCO '09: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation (2009)
- [Kalyanakrishnan 09] Kalyanakrishnan, S. and Stone, P.: An empirical analysis of value function-based and policy search reinforcement learning, AAMAS '09: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems (2009)
- [Lehman 08] Lehman, J. and Stanley, K.: Exploiting open-endedness to solve problems through the search for novelty, ALIFE XI: Proceedings of the 11th International Conference on Artificial Life (2008)
- [Mouret 09] Mouret, J.-B. and Doncieux, S.: Using behavioral exploration objectives to solve deceptive problems in neuro-evolution, GECCO '09: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation (2009)
- [Sareni 98] Sareni, B. and Krahenbuhl, L.: Fitness sharing and niching methods revisited, IEEE Transactions on Evolutionary Computation (1998)
- [Stanley 02] Stanley, K. and Miikkulainen, R.: Evolving neural networks through augmenting topologies, Evolutionary Computation (2002)
- [Trujillo 08] Trujillo, L., Olague, G., Lutton, E. and de Vega, F.: Discovering several robot behaviors through speciation, Applications of Evolutionary Computing, LNCS:4974 (2008)
- [Yao 99] Yao, X.: Evolving artificial neural networks, Proceedings of the IEEE (1999)