進化型ニューラルネットを用いたロボットの順動力学モデル学習

Modeling Forward Dynamics using Neuroevolution Algorithms

○ 森口 博貴 (東大) 本位田 真一 (国立情報研, 東大)

Hiroataka MORIGUCHI, The University of Tokyo, hmori@nii.ac.jp
Shinichi HONIDEN, The University of Tokyo, National Institute of Informatics

Forward dynamics models are the important basis of motion planning of robots. While recent advancement in Robotics has allowed diverse morphologies of robots to take place, there has been little study on identifying the forward models for such robots. In this paper, we propose to use neuroevolution algorithms, which optimize the parameters of neural networks, to learn such models. We show that a neuroevolution algorithm called CMA-NeuroES can learn models of a simulated double pendulum system with higher accuracy and generalization ability compared to conventional regression algorithms, such as Gaussian Process Regression, Support Vector Regression and Linear Regression.

Key Words: Non-linear System Identification, Forward Dynamics, Neural Networks, Evolutionary Computation

1. 緒言

ロボティクスの発展は多様な形状・機構のロボットを生み出してきたが、一方でそれらのロボットの自律制御の研究は未だ発展途上である。既存の自律制御手法の多くは、ロボットの動力学モデルが事前に与えられることを前提としており、開発者は解析力学の知識を駆使してモデルを導出する必要がある。ところが近年の複雑化したロボットの形状・機構を古典的な力学の枠組みで表すことは難しく、解析的なモデル導出は困難である[5, 6]。機械学習を用いて動力学モデルを学習する方法は、人間によるモデル導出の困難さを回避するための有力な手段として注目を浴びている[9, 10, 12]。

本研究では、ロボットの順動力学モデル（運動方程式）を学習し、ロボットの行動計画に利用する方法について議論する。ロボットなどの離散時間システムにおいて、順動力学モデルは状態の時間発展を記述するために用いられる（図1）。順動力学モデルをステップ間の状態遷移についての制約として用いれば、ロボットの行動計画を制約付き最適化問題として解くことが可能となる。

行動計画に用いられる順動力学モデルは、真の運動方程式を高精度で近似すると同時に、推論時の計算効率が高いことが望ましい。文献[8]において、我々はシンボリック回帰(SR, [7, 11])を用いたモデル学習法と、それを用いた行動計画法を提案し、有効性を示した。シンボリック回帰の有効性は、その計算効率と近似精度の高さに依拠していると考えられる。

本研究では、SRと同様に計算効率と近似精度を両立しうる手法として、進化型ニューラルネット(NE, [14])を用いたモデル学習手法を提案する。進化型ニューラルネットはニューラルネットのパラメータを、進化計算を用いて最適化する手法である。ニューラルネットを用いた関数近似は推論時の計算効率が高いため、十分な近似精度が得られればSRと同様に有効なモデル学習法となることが期待される。

二重振り子系を用いた実験において、NEによるモデル学習の有効性を示す。NEはSRに及ばないまでも、従来の機械学習手法を用いたモデル学習に比べると高い近似精度を実現した。またSRはその特徴として近似されるべき関数が複雑になるにつれて、飛躍的に関数近似が困難になることが知られているため、より複雑なシステム同定においてはNEがSRより有効に働くことも期待できる。

Fig. 1 Illustration of forward dynamics models.

2. 手法 (CMA-NeuroES)

本研究では CMA-NeuroES[4] とよばれる進化型ニューラルネットを用いてモデル学習を行う。CMA-NeuroES は CMA-ES[3] と呼ばれる関数最適化法を用いて、固定数の結合強度パラメータを最適化する。CMA-NeuroES による最適化の手順をアルゴリズム 1 に示す。

CMA-NeuroES の特徴は、探索に用いるパラメータが自己適応する点である。一般的な進化計算手法と同様に、候補解の近傍に複数の新たな個体を生成し、解空間を探索する。その時点での候補解に $\sim \sigma \mathcal{N}(0, C)$ に従う正規乱数を加算することで近傍解が生成されるが、自己適応のある CMA-NeuroES では、正規分布のパラメータである σ と C を、過去に評価値の高い個体

Algorithm 1 CMA-NeuroES

```

 $\langle x \rangle_w \leftarrow 0, C \leftarrow I, \sigma \leftarrow \sigma_{init}$ 
while 終了条件が満たされない do
  for  $i = 1 \rightarrow \lambda$  do
     $x_i \leftarrow \langle x \rangle_w + \sigma z_i (\sim \mathcal{N}(0, C))$ 
  end for
   $x_{i.. \lambda}$  の評価値に基づいて共分散行列  $C$  を更新
   $x_{i.. \lambda}$  の評価値に基づいてステップサイズ  $\sigma$  を更新
   $\langle x \rangle_w \leftarrow \sum_{i=0}^{\mu} w_i x_i$ 
   $\sigma < \sigma_{min}$  の場合  $\sigma = \sigma_{min}$  に調整
end while

```

を生んだステップを再現する確率を上げるように最尤推定に基づいて更新する。これにより、解空間の構造を踏まえた探索の効率化が可能となる。CMA-NeuroES は、倒立振り子ベンチマーク [13, 4] において最も優れた結果を残しており、最先端の NE アルゴリズムである

3. 実験

今回の実験では二重振り子をモデル学習の対象とする。二重振り子の仕様は図 2 に示すとおりであり、各関節にはトルクを発生させるモータが装着されており、最大 $4\text{kgf} \cdot \text{m}$ のトルクを両回転方向に発生させられる。振り子の動作は系の運動方程式を 4 次のルンゲクッタ法と併用してシミュレートした。

Fig. 2 Double pendulum. The origin is fixed.

学習に用いる訓練データには、均衡状態にある振り子をランダムに生成したトルクカーブによって動作させシミュレートした動作を 2000 ステップのあいだ記録したものをを用いる。ここで、学習すべきモデルは各ステップにおける各加速度を近似する、以下の 2 関数である、

$$\ddot{\theta}_1 = f(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2, \tau_1, \tau_2)$$

$$\ddot{\theta}_2 = g(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2, \tau_1, \tau_2)$$

ここで θ は各関節の角加速度、 $\dot{\theta}$ は角速度、 τ は各関節モータの発生させるトルクをそれぞれ表す。

また学習されたモデルの汎化性能を評価するため、以下の範囲からランダムに生成された 2000 セットの状態変数およびトルクに対して、適切な θ を推論し、近似精度を評価した。

今回の実験では、CMA-NeuroES を、ガウシアンプロセス回帰 (GPR)、サポートベクタ回帰 (SVR)、線形回帰 (LR)、シンボリック回帰 (SR) と比較する。GPR および LR には Weka [2] を、SVR には LIBSVM [1] を、SR には Eureka [11] を実装として用いた。

3.1 実験設定

CMA-NeuroES のトポロジとして、入力ノード 7 個 (バイアスとして常に 1 を入力として取るノードが 1 個)、隠れノード n (1-10) 個、出力ノード 2 個を持つ、完全結合リカレント型ネットワークを利用した。

CMA-NeuroES では 2 つの出力を 1 つのネットワークで表現するため、単純に平均絶対誤差 (MAE) や平方二乗誤差では評

価値を与えられない。そこで、実験では以下の 3 種類の適応度の評価基準を用いた。

$$L_1 : \sum_i |\hat{\theta}_1[i] - \bar{\theta}_1[i]| + |\hat{\theta}_2[i] - \bar{\theta}_2[i]|$$

$$L_2 : \sum_i \sqrt{(\hat{\theta}_1[i] - \bar{\theta}_1[i])^2 + (\hat{\theta}_2[i] - \bar{\theta}_2[i])^2}$$

$$L_\infty : \sum_i \max(\hat{\theta}_1[i] - \bar{\theta}_1[i], \hat{\theta}_2[i] - \bar{\theta}_2[i])$$

ここで、 $\dot{\theta}[i]$ は i 番目のデータ点における $\dot{\theta}$ を表し、またハット付きの $\hat{\theta}$ は候補となるニューラルネットによる $\dot{\theta}$ の予測値を表す。

実験では CMA-NeuroES の以下のパラメタについて、パラメタサーベイを行い、最も優れたパラメタセットにおける結果を採用する。

n_h (# of hidden nodes): {1, 2, ..., 10}

σ_{init} (initial value of σ): {0.01, 0.05, 0.1, 0.2, 0.5, 1}

σ_{min} (lower bound of σ): {0.01, 0.05, 0.1, 0.2, 0.5}

実験結果の評価には、80000 のネットワークが評価された時点までに、最も評価値の高かったネットワークによる結果を用いて評価した。計算時間は、隠れノード数にもよるが隠れノード数が 1 の場合は約 1 分、10 の場合は約 3 分を要する。

3.2 実験結果

実験結果を表 1, 2 にまとめる。CMA-NeuroES については、訓練データ上での近似性能が高いもの、およびテストデータ上での汎化性能が高いもの双方を示した。

Table 1 Comparison of learning algorithms in terms of mean absolute error. Parentheses below CMA-NeuroES denotes the parameter sets $\{n_h, \sigma_{init}, \sigma_{min}, \text{evaluation metric}\}$.

Algorithm	訓練データ		テストデータ	
	θ_1	θ_2	θ_1	θ_2
SR	1.2033	2.7536	11.7243	29.8667
GPR	3.9227	4.0309	26.2728	42.7183
SVR	2.1574	2.8607	18.3367	29.7949
LR	8.1712	7.8086	17.6821	30.867
CMA-NeuroES {8, 0.1, 0.01, L_1 }	1.336	2.0632	28.1328	44.156
CMA-NeuroES {2, 0.2, 0.05, L_∞ }	2.8073	2.8139	17.2211	27.7246

Table 2 Comparison of learning algorithms in terms of correlation coefficients.

Algorithm	訓練データ		テストデータ	
	θ_1	θ_2	θ_1	θ_2
SR	0.9837	0.9301	0.7711	0.1436
GPR	0.6976	0.8386	0.2177	0.0658
SVR	0.8188	0.874	0.1075	0.0565
LR	0.4358	0.3392	0.0077	-0.0215
CMA-NeuroES {10, 0.2, 0.01, L_1 }	0.9669	0.9533	0.2701	0.1809
CMA-NeuroES {3, 0.05, 0.01, L_1 }	0.8832	0.907	0.5213	0.4039

以上の結果から、CMA-NeuroES は訓練データ上での近似性能およびテストデータ上での汎化性能について、SR に匹敵する

性能を示すことが分かる。GPR, SVR および LR と比べると、性能の差は歴然である。

行動計画への応用において特に重要となる汎化性能について詳しく見ると、隠れノード数が少ない場合に汎化性能が高くなっていることがわかる。

4. まとめ

本研究では、非線形な順動力学モデルの学習タスクにおける、進化型ニューラルネットの利用を検討した。二重振子を用いた比較実験の結果からは、進化型ニューラルネットの一つである CMA-NeuroES の汎化性能が GPR, SVR および LR といった従来の機械学習手法に比べて高く、有効であることを示した。一方で、CMA-NeuroES の利用は計算効率の観点からも有効であると考えられる。たとえば、1 ステップの動作予測にかかる時間は GPR や SVR に比べて 100 倍程度、高速である。

SR との比較からは、CMA-NeuroES が SR に匹敵する学習性能を示すことが分かった。

5. 今後の課題

本研究では各パラメタセットについて 1 度しか実験を行っていない。進化計算は試行ごとに学習性能に大きな差が生まれることが知られているため、統計的に有意な評価を行うためには試行数を増やす必要がある。

学習されたモデルの汎化性能が高ければ、有効な行動計画が可能になることが期待できる[15]。しかし実際に有効であるかは、行動計画に組み込んだ際の有効性を実験的に評価する必要がある。

本研究における実験では、学習性能の比較評価のために、真の運動方程式が解析的に求まる二重振子を利用した。解析的に運動方程式が求まらないような系の順動力学モデルの学習と、行動計画への応用において、NE と SR のどちらがより有効であるかを調べることも、今後の重要な課題である。

文 献

- [1] C. Chang and C. Lin. LIBSVM: a library for support vector machines, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [2] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten. The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.
- [3] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2):159–195, 2001.
- [4] V. Heidrich-Meisner and C. Igel. Neuroevolution strategies for episodic reinforcement learning. *Journal of Algorithms-Cognition Informatics and Logic*, 64(4):152–168, 2009.
- [5] J. Hiller and H. Lipson. Evolving amorphous robots. In *procs. of International Conference on Artificial Life (ALIFE)*, 2010.
- [6] M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, and S. Schaal. Learning, planning, and control for quadruped locomotion over challenging terrain. *The International Journal of Robotics Research (IJRR)*, 30(2):236, 2011.
- [7] J. Koza. *Genetic Programming, On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [8] H. Moriguchi and H. Lipson. Learning symbolic forward models for robotic motion planning and control. In *procs. of European Conference of Artificial Life (ECAL)*, 2011.
- [9] D. Nguyen-tuong and J. Peters. Model learning for robotics: a survey. *Cognitive Processing*, 12(4), 2011.
- [10] D. Nguyen-Tuong and J. Peters. Local gaussian process regression for real-time modelbased robot control. In *procs. IROS*, pages 380–385. IEEE, 2008.
- [11] M. Schmidt and H. Lipson. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85, 2009.
- [12] J. Sturm, C. Plagemann, and W. Burgard. Unsupervised body scheme learning through selfperception. In *procs. of ICRA*, pages 3328–3333.

IEEE, 2008.

- [13] R. S. Sutton and A. G. Barto. *Reinforcement learning*. The MIT Press, 1998.
- [14] X. Yao. Evolving artificial neural networks. *Proceedings of the IEEE*, 87(9):1423–1447, 1999.
- [15] 森口博貴, 本位田真一, "進化計算を用いたロボットの順動力学モデル学習と行動計画・制御への応用", 進化計算シンポジウム 2011 予稿集, 2011.