4J1-1 ガウス過程回帰の信頼度に基づく確率的ロードマップを用いた 動作計画

Motion planning method using a probabilistic roadmap based on the confidence of a Gaussian process regression

岡留 有哉^{*1} 中村 泰^{*1} 石黒 浩^{*1} Yuya Okadome Yutaka Nakamura Hiroshi Ishiguro

*1大阪大学基礎工学研究科システム創成専攻

Department of Systems Innovation, Graduate School of Engineering Science, Osaka University

The development of a motion planning method for a robot to work in real environments becomes important research issue. For such an application, sampling-based motion planning methods have been successfully used in the last decade. In a sampling-based motion planning method, the state space is configured and a graph is constructed on the space, which indicates the reachability between node pairs. The motion planning is done by using the graph. It requires an explicit model of a controlled target to get the graph, but these model is difficult to obtain a model for a robot working in a real environment due to the complexity. In this research, we propose a framework of a motion planning method in which a system model is approximated by Gaussian process regression. We apply our method to the swinging up of a simple-pendulum and investigate the property of our motion planning method.

1. はじめに

近年, ロボットは実環境で動作できるようになることが期待 されている.しかしながら, そのような実環境には様々な外 乱が存在するため, ロボットが活動することは困難である.特 に, 動作計画を行う際には, センサノイズや物理的な接触など が原因となる複雑な制約条件を扱うことが必要となる.この ような問題を解決するため, サンプリングに基づく動作計画法 [9][7][10][1]が発展してきており, 近年では実環境における自 動走行車の制御への応用 [8] も行われている.

サンプリングに基づく動作計画として rapidly-exploring random trees (RRT)[2][9] と probabilistic roadmap (PRM)[12][7] の 2 つの代表的な手法がある. これらの手法で は,はじめに状態空間を設定し,グラフ構造を構築する. グラ フの各ノードは空間の点 (状態) であり,2 つの近接したノー ドのペアが到達可能であればエッジにより接続する. グラフ を構築した後,ダイクストラ法 [4] や A*[3] アルゴリズムによ り経路計画を行う. これらの手法はグラフを構築するために 制御対象のフォワードダイナミクスやインバースダイナミク スなどのモデルを必要とするが,基本的には厳密なモデルが与 えられるものとして研究が行われてきた. しかし,実問題にお けるシステムモデルは不確実な要因により影響をうけるため, そのようなモデルを前もって得ることは困難である.

本研究では、システムモデルとしてガウス過程回帰 (GPR)[13] を用いた、サンプルに基づく動作計画を提案する. GPR はノンパラメトリックベイズ法の1つで多くのタスクに 適用されてきた. GPR はベイズ推定の枠組みであるため、出 力の推定値だけでなく、信頼区間も計算することができること から、本研究では、この信頼区間を2つのノードの到達可能性 と得られた経路の信頼性の評価に利用する.

GPR の精度はサンプルサイズに依存するが, サンプルサイ ズが増加するに従い, 計算コストも増大してしまう. サンプリ ングに基づく動作計画法では, 多数のノードペアの接続性を評 価する必要があるため, GPR のように正確だが計算コストが 大きい手法を適用することは難しい. そのため,本研究では, 我々の手法が以前提案した,高速な GPR の近似手法 [11] を 利用する. この手法では,高速化と精度のバランスをとるため に, locality-sensitive hashing (LSH)[6] と product of experts model (PoEs)[5] を組み合わせて用いる.

本動作計画法を倒立振子の振り上げタスクに利用し,特性を 調べた.また,経路を接続する際のコスト関数の違いが,経路 計画に及ぼす影響を調べた.

2. Sampling based motion planning

本研究では制御対象をマルコフ性を持つ離散時間システム と仮定する. すなわち, ロボットなどの物理システムのダイナ ミクスは p(s'|s, a) と表現できると仮定する. ここで, s, a, s'はそれぞれ状態, 行動, 次の状態である.

動作計画の目標は、与えられた初期状態から目標状態への 実行可能な経路(動作)を見つけることである.サンプリング ベース動作計画では、状態空間を最初に定義する.例えば、ロ ボットの動作制御を行う場合、状態空間は関節角、関節角速度 の空間と定義できる.そして、状態空間全体を覆うグラフを到 達可能なノード同士を接続することにより構築する.ここで、 グラフはシステムの制約を満たすようなノードとエッジから構 築されるため、グラフを利用して得られた経路はそのような制 約条件を満たしていると期待できる.ここでは、2つの有名な サンプリングベース動作計画のアルゴリズム、RRT[9] および PRM[7] を紹介する.

RRT RRT では, ルートノードを初期状態をした木構造グラ フが生成される.木構造は以下の手順を繰り返すことで拡張し ていく: i) 状態空間から1点サンプリングする. ii) サンプル 点に最も近いノードを選択する. iii) サンプル点に向かう, 制 御信号を計算する. iv) 上記の制御信号を入力した時の, 最近 傍ノードからの遷移先の点を計算する. v) 新しい葉ノードと して新たな点を木に追加する. サンプル点は状態空間全体から 選択されるため, 木で覆われる領域は高速に拡大していく傾向 にあり, 近年, 広大な状態空間を扱う必要のある実問題への応 用が盛んに行われている.

連絡先: 560-8531 大阪府豊中市待兼山町 1-3 大阪大学基礎工 学研究科システム創成専攻 { okadome.yuya, nakamura, ishiguro}@irl.sys.es.osaka-u.ac.jp



 \boxtimes 1: Connection by using GPR

PRM PRM では, 任意の2ノード間の経路を探索するために 利用するロードマップを作成する. ロードマップはグラフの1 種であり, 近傍同士のノードペアが到達可能であれば, それら をエッジにより接続する. 図1にロードマップの簡単な作成法 を示す. i) 多数の点を状態空間全域から取得する. ii) ローカル コントローラに従い, ノードペアの接続性を評価する. iii) 到 達可能なノードペアをエッジにより接続する. ロードマップを 作成する際に, 中継点を加える場合もある.

これらの手法は強力であり,注目を集めている.しかしなが ら,実環境において前もってシステムモデルを得ることは困難 であるため,これらの手法をそのまま実空間における動作計画 に適用することは難しい.そこで,本研究では,未知のシステ ムをノンパラメトリックベイズモデルを用いて近似するロード マップの作成法を提案する.

3. GPR を用いたロードマップ構築

我々の手法では,動作データは任意の制御信号を制御対象に 入力することで収集する. Dをセンサ入力(状態)と制御信 号(行動){s(1), a(1), s(2), ...}からなるデータセットとする. 動作データに含まれるすべての点をグラフのノードとして用 い,それぞれの点における次の状態をエッジにより接続し,初 期ロードマップとする.

本手法では, 到達可能性とエッジのコストは GPR により計 算された信頼区間にしたがって決定する. この信頼区間のおか げで, 制御対象についての知識の量 (すなわち, データセット のサンプルサイズ) に基づいた経路の信頼度を考慮した経路計 画を実行することができる. しかしながら, GPR の計算コス トはデータセットのサンプルサイズが増加するにつれ, 劇的に 増加してしまう. そのため, ロードマップを現実的な時間で作 成するために, GPR の高速な近似手法を適用した.

3.1 ロードマップ作成

本研究では、動作データは状態と行動の時系列として与えら れていると仮定する (図 2). 観測された状態と、その状態遷移 は初期ロードマップのノードとエッジとして扱われる. 動作系 列は状態空間中でもつれ合っていると考えられるため、軌跡を 横断するようにエッジを加えることでロードマップを成長させ る. そのため、あるノードの次の状態に近い、|K| 個の仮ノード を取り出し、あるノードとそれぞれの仮ノードの間で到達可能 性の評価を行う. K は仮ノードの集合である. 到達可能であれ ば確信を持って制御信号を決定することができるため、評価に は、'逆動力学'、すなわち s から s' への状態遷移を起こすため の制御信号を利用する.

本研究では,各エッジの接続性の評価に用られる逆動力学



⊠ 2: Connection of our roadmap

 $p(a|s, \Delta s) \in \text{GPR } \text{icl} b$,

$$p(a|s_i, \Delta s_{ij}) = \mathcal{GP}(\mu(a), \sigma^2(a)).$$
(1)

と計算する. ここで, $\Delta s = s' - s$ である. $\mu(a) \geq \sigma^2(a)$ は それぞれ行動 a の期待値と信頼区間である. s から s' への到 達可能性は式 (1) で計算される信頼区間により評価される. 信 頼区間が大きい場合, 逆動力学の推定が正確ではないというこ とを示している. そのため, $\sigma^2(a) < T_c$ である場合には 2 つ のノード間にエッジを生成する. T_c は接続をリジェクトする かどうかを決定する閾値である. エッジが生成された場合, コ スト関数 $c(\sigma^2(a))$ に従い, エッジコストが割り当てられる. こ の処理をすべての近傍ノードペアに対して行うことで, 有向の ロードマップを生成することができる. 初期ロードマップは収 集した動作でしかないため, 追加のエッジは 'ショートカット' の役割を果たす.

ロードマップを生成した後,経路の探索を行う.経路探索は 従来の RRT や PRM と同様の方法で行う.動作の始点 s_{init} とゴール s_{goal} が与えられたとする.エッジコストに基づき, Dijkstra 法により経路を生成する.

3.2 GPR の高速な近似手法

本手法では,式(1)はGPRを用いて計算する. そのため,デー タセット $\mathcal{D} = \{s_1, a_1, \dots, s_N, a_N\}$ を入力集合 $\mathcal{X} = \{x_i = (s_i, \Delta s_i = s_{i+1} - s_i) | i = 1, \dots, N - 1\}$ と出力集合 $\mathcal{A} = \{a_i | i = 1, \dots, N - 1\}$ に変形する. データセットの *i* 番目の要素 $x_i \in \mathcal{X}, a_i \in \mathcal{A}$ は GPR の *i* 番目の訓練データとなる.

GPR への入力ベクトル $x_* = (s, s' - s)$ が与えられた時, 出力は

$$\mathcal{GP}(\mu(a_*), \sigma^2(a_*)) = P(a_* | \boldsymbol{x}_*, (\mathcal{X}, \mathcal{A}))$$
(2)

$$= \frac{1}{Z} \exp\left(-\frac{1}{2} \frac{(\hat{a} - \mu(a_*))^2}{\sigma(a_*)}\right), (3)$$

$$\mu(a_*) = \mathbf{k}^\top \mathbf{C}^{-1} \mathbf{a}, \qquad (4)$$

$$\sigma^2(a_*) = K(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}^\top \boldsymbol{C}^{-1} \boldsymbol{k}.$$
 (5)

と計算できる. ただし, *C* は要素 $C_{lm} = K(\boldsymbol{x}_l, \boldsymbol{x}_m) + \delta_{lm} \sigma_n^2$ を持つ共分散行列である. $K(\cdot, \cdot) \geq \sigma_n^2$ はカーネル関数とノ イズである. また, $\boldsymbol{k} = (K(\boldsymbol{x}_1, \boldsymbol{x}_*), \dots, K(\boldsymbol{x}_N, \boldsymbol{x}_*))$ である. $\boldsymbol{s}' = \boldsymbol{s}_j, \, \boldsymbol{s} = \boldsymbol{s}_i$ とした時, *i* から *j* へのエッジを評価すること ができる.

式 (3)-(5) は共分散行列の逆行列を求める際の計算コストが $O(N^3)$, 予測分布を求める際にかかる計算コストが $O(N^2)$ で ある. そのため, サンプルサイズの増加に伴う計算コスト増加 が問題となる. この問題を解決するために, 我々は以前, データセットを分 割する手法である LSH[6] と確率分布を統合する手法である PoEs[5] を用いた高速な GPR の近似手法である LSH-GPR を 提案した [11]. LSH では, *B* 個のハッシュ関数からなる *B* ビッ トのバイナリハッシュキーを用いることでデータセットを 2^B 個のサブセットへと分割する. ハッシュキー値の計算はサンプ ルサイズに依存せず, '分解' された各 GPR はサブセットのみ を用いて計算することができるため, 計算コストが劇的に減少 する. 一方, サブセットの境界付近において, 分解された GPR の精度は低下してしまうが, この弱点は *L* 個の異なる分解され た GPR を PoEs を用いて統合することにより回避する.

4. 実験

本動作計画法の特性を調べるため、単振り子の振り上げタス クに適用した. この問題は、状態-行動空間は小さいが、非線 形ダイナミクスを持つシステムの制御課題である. 実験では、 CPU:Core i5 3.6GHz, Memory: 4GB の PC を用いて動作計 画及び以下に記載するフィードバックの計算を行った.

フィードフォワード制御, すなわち制御器が現在の状態を考慮 せずに計画された行動のみを出力する場合では, ノイズやモデル エラーのために観測された軌跡が計画された経路から離れてし まう可能性がある. それゆえ, 次ステップでの目標状態に基づく フィードバック制御を行う. s(t)を現在, 観測された状態とする と, $GP(\mu(a_{FB}), \sigma^2(a_{FB})) = P(a_{FB}|(s(t), s_i - s(t)), (\mathcal{X}, \mathcal{A}))$ のように計算できる. また, 制御器が出力する行動を $a_* = \alpha a_{FF} + (1 - \alpha)a_{FB}$, すなわちフィードフォワードとフィード バックを一定の比率で混合したものとした. a_{FF} と a_{FB} はそ れぞれフィードフォワードでの行動とフィードバックでの行動 である. また, α は 0 < α < 1 であるような混合比である.

4.1 Simple pendulum



(a) Simple-pendulum

(b) Dataset

図 3: 振り上げタスクに用いたデータセット. データセットの サンプルサイズは 2000.

単振り子の状態は,振り子の位置 $(\sin\theta, \cos\theta)$ および振り子 の角速度 θ' (図 3(a)) とする. θ は振り子の角度である. 振り 子に対する行動は回転中心から与えられるトルク τ とする. 単 振り子のダイナミクスは以下のように計算することができる.

$$\ddot{\theta} = -\frac{mg\sin\theta}{ml} - \frac{\mu\dot{\theta}}{m} + \frac{f}{m}.$$
 (6)

ここで, *l*, *m*, *g*, および μ はそれぞれ振り子の長さ (1.0 [m]), 重さ (1.0 [kg]), 重力加速度 (9.8 [m/s²]), 摩擦 (0.10) である. 本実験では,入力トルクは [-5,5] [Nm] の連続値であり,振り 子が真下で静止した状態を初期状態 ($\theta = \pi, \dot{\theta} = 0$) とする. 図 3(b) にデータセットに含まれるサンプルの分布を示す. 異なるコスト関数を持つ 2 つの動作計画器 CRM_C, CRM_T を 比較し, コスト関数が動作計画に与える影響を調べた. CRM_C のコスト関数は $c(\sigma^2(a_*)) = \sigma^2(a_*)$ とし, CRM_T のコスト関 数は $c(\sigma^2(a)) = \beta\sigma^2(a_*) + t_c$ とした. $\beta \ge t_c = 0.1$ [s] はそれ ぞれ正の定数と制御周期である. すなわち, CRM_C では最も 信頼できる経路を探し出し, CRM_T ではより短い経路を探し 出すようなコスト関数になっている. 到達可能性を調べる候補 の数は経験的に $|\mathcal{K}| = 5 \ge 0.t.$ また, LSH-GPR のビット数 及びハッシュキーの数は $B = 3, L = 2 \ge 0.t.$

図 4(a), 4(b) の動作計画器 CRM_C および CRM_T を用いた ときに得られた動作を示す.図 4(c),4(d) は動作計画器 CRM_C および CRM_T を用いたときの単振り子の振り上げ制御結果で ある.図 4 の赤点と青点は計画された経路と制御を行った際に 得られた観測値である.

CRM_C と CRM_T はともに振り上げタスクを達成すること ができた. CRM_C と CRM_T で得られた経路の経路長はそれ ぞれ, 110 step と 78 step で CRM_T のほうが短くなっている. コスト関数に制御周期の項を加える事で, 短い経路を生成する ことができた. しかし, CRM_T では CRM_C と比べ, 目標状態 ($\theta = 0, \dot{\theta} = 0$) での誤差が大きくなっている. CRM_T で得ら れた経路には, 信頼区間 $\sigma^2(a_*)$ が大きい, すなわち逆動力学の 推定誤差が大きいような動作も含まれているからだと考えら れる.

CRM_C と CRM_T のロードマップを作成するためにかかった 計算時間は約 73 秒で、ダイクストラ法の計算時間は約 0.15 秒 であった. 高速化を行っていない通常の GPR を用いて動作計 画を行った場合では、ロードマップの作成時間は約 1581 秒で、 ダイクストラ法の計算時間は約 0.15 秒であった. LSH-GPR を用いることで、現実的な時間でロードアップを作成すること ができた.

5. 結論

本研究では、GPR を用いた動作計画法を提案した.そして、 本手法を単振り子の振り上げタスクに適用し、非線形ダイナミ クスを持つシステムの動作計画が可能であることを示した.ま た、高速な GPR の近似手法である、LSH-GPR を用いること で、現実的な時間で動作計画を行う事ができた.

本手法では,障害物については考慮していないが,実際のロ ボットに適用していく場合には障害物の考慮は必要になると考 えられる.障害物を回避するような動作計画を行う手法の開発 を行い,実際のロボットに本手法を適用することが,今後の課 題である.

参考文献

- [1] Howie M Choset. Principles of robot motion: theory, algorithms, and implementations. MIT press, 2005.
- [2] Sébastien Dalibard, Antonio El Khoury, Florent Lamiraux, Alireza Nakhaei, Michel Taïx, and Jean-Paul Laumond. Dynamic walking and whole-body motion planning for humanoid robots: an integrated approach. *The International Journal of Robotics Research*, 32(9-10):1089–1103, 2013.
- [3] Rina Dechter and Judea Pearl. Generalized best-first search strategies and the optimality of a*. Journal of the ACM (JACM), 32(3):505–536, 1985.



(a) Resulting path obtained by CRM_C (

(b) Resulting path obtained by CRM_T



(c) Control result of swinging up task by CRM_C

(d) Control result of swinging up task by CRM_T



- [4] Edsger W Dijkstra. A note on two problems in connexion with graphs. Numerische mathematik, 1(1):269– 271, 1959.
- [5] Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. In *Neural Computation*, volume 14, pages 1771–1800. MIT Press, 2002.
- [6] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM* symposium on Theory of computing, STOC '98, pages 604–613, New York, NY, USA, 1998. ACM.
- [7] Lydia E Kavraki, Petr Svestka, J-C Latombe, and Mark H Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *Robotics and Automation, IEEE Transactions on*, 12(4):566–580, 1996.
- [8] Yoshiaki Kuwata, Sertac Karaman, J Teo, E Frazzoli, JP How, and G Fiore. Real-time motion planning with applications to autonomous urban driving. *Control Systems Technology, IEEE Transactions on*, 17(5):1105–1118, 2009.
- [9] Steven M LaValle and James J Kuffner. Randomized kinodynamic planning. *The International Journal of Robotics Research*, 20(5):378–400, 2001.

- [10] Steven Michael LaValle. Planning algorithms. Cambridge university press, 2006.
- [11] Yuya Okadome, Yutaka Nakamura, Yumi Shikauchi, Shin Ishii, and Hiroshi Ishiguro. Fast approximation method for gaussian process regression using hash function for non-uniformly distributed data. In Artificial Neural Networks and Machine Learning (ICANN 2013), volume 8131 of Lecture Notes in Computer Science, pages 17–25. Springer Berlin Heidelberg, 2013.
- [12] Samuel Prentice and Nicholas Roy. The belief roadmap: Efficient planning in belief space by factoring the covariance. *The International Journal of Robotics Research*, 28(11-12):1448–1465, 2009.
- [13] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.