解 説

ロボットの動作計画における最適化

Optimization in Robot Motion Planning

原 田 研 介* * (独) 産業技術総合研究所

Kensuke Harada* * National Institute of Advanced Industrial Science and Technology

1. 緒 言

従来より、何らかの評価関数に従ってロボットの動作を生成する手法に関して多くの研究が行われている（例えば文献 [1]）。それに対して、本稿ではロボットアームの初期姿勢と目標姿勢が与えられているという条件の下で、ロボットアームが障害物を避けながら初期姿勢から目標姿勢へ到達する軌道を導出する問題に議論を絞り、この問題に関する最適化手法について議論する。このような問題は 80 年代より解かれているクラシカルな研究である。例えば Shiller ら [2] はロボットアームの最短時間動作計画を空間軌道を計画する問題と、軌道に沿った動作の時間軌道を計画する問題に分離することで解く手法を提案した。しかしながら、この手法においては障害物の形状を陽に与える必要があったため、複雑な形状をした環境モデルの問題に適用するのは非常に困難であった。その後、動作計画手法として PRM (Probabilistic Roadmap Method) や RRT (Rapidly-Exploring Random Tree) などのランダムサンプリングに基づく手法 [3] [4] の有効性が認められて広く用いられるようになると、これらの手法の Post Processor として軌道長を短くする操作が行われるようになった [5]~[8]。さらに、近年最適化手法をより直接的に動作計画のアルゴリズムの中に組み込むことで、評価関数を最適化するような軌道を導出する手法が提案されている。特に、RRT など従来の手法を改良することによる最適化手法 [9] [10] や関数最適化に基づく手法 [11]~[15] などが注目を集めている。本稿では、これらの手法を紹介し、また関連した筆者らの研究も同時に紹介する。

なお、例えばコンフィグレーション空間においてロボットアームの軌道の長さを最小化する場合、状態変数が $SE(2)$ や $SE(3)$ を含むと、最小化の定義自体が不明確になる [3]。本稿においては、コンフィグレーション空間 C は $C \subset R^n$ として定義される場合を考える。

2. Post Processor としての最適化

まず、何らかの動作計画手法により、ロボットアームが障害物を避けながら初期姿勢から目標姿勢へ到達する軌道が与えられていると仮定し、この軌道に対する Post Processor としての最適化手法について議論する。ここで、軌道はロボットのコンフィグレーションから構成されるノードと、このノードを直線で繋ぐパスにより構成される場合を考える。PRM や RRT は、このような軌道を出力することが可能であるが、これらのプランナによって生成される軌道は、通常不必要な迂回を多く含んでおり、この軌道を実行することで得られるロボットの動作はぎくしゃくしたものになる。また、これらの手法で得られた軌道は何らかの評価関数を最適化するものではないということが知られている [9]。そこで、PRM や RRT で得られた軌道の長さを短くすることを目的として、Path Pruning や Shortcut と呼ばれる手法が広く用いられている [5]~[8]。この概要を Fig. 1 に示す。ロボットの動作軌道がノードのリスト、 $v_1, \dots, v_n \in V$ とこれらを順番に直線で結ぶパスにより表現されている場合を考える。ここで、各ノードはロボットのコンフィグレーション $q_1, \dots, q_n$ によって定義される。

Path Pruning: ノードのリスト $v_1, \dots, v_n \in V$ から v_i を仮に取り除いたときに、 v_{i-1} と v_{i+1} を結ぶパスに干渉が生じていないなら、実際に v_i を取り除く。この操作を繰り返し実行する。

Shortcut: ロボットの動作軌道上に二つの新たなノード v_j , v_{j+1} をランダムに仮定する。これら二つのノードを結ぶパスに干渉が生じていないなら、 v_j と v_{j+1} の間に含まれるすべてのノードを取り除く。この操作を繰り返し実行する。

一般に Path Pruning と比較して Shortcut のほうがより軌道長の短い軌道を求めることができる反面、軌道に含まれるノードの数が非常に多くなる場合がある。ここで、これら Post Processor によって得られる軌道は、その長さが最小のものであることは保証されていない。また、このア

原稿受付 2014 年 5 月 7 日

キーワード: Motion Planning, Optimization, PRM, RRT

*〒 305-8568 つくば市梅園 1-1-1

*Tsukuba-shi, Ibaraki

図 1 Methods for optimizaing path length

プローチによって得られる動作はノードごとにロボットが静止するものである。さらに、Post Processor への入力として与える軌道が、障害物を大きく迂回する軌道である場合は、Post Processor を実行しても、その軌道は本来軌道長が最短な軌道として得られるべき軌道からは大きく離れたものになる。また、これは Post Processor に限った問題ではないが、軌道長を短くする処理を行った軌道は、多くの場合障害物との干渉をぎりぎりでするものとなる。このような問題意識の下で、Post Processor に関して種々の研究が行われている。まず、より軌道長が短い軌道を得るために、自由度ごとに Shortcut を行う手法が提案されている [5]。また、ノードごとにロボットが静止する問題に対しては、ノード間をスプライン関数で結ぶ手法が提案されている [7] [8]。特に、文献 [8] では Fig. 2 に示すように、軌道上において B-スプライン関数のノードをインクリメンタルに増やす手法を提案している。この手法を用いることにより、ヒューマノイドロボットが滑らかな動作で歩きながら障害物を避ける動作計画を実現している。また、障害物との距離を保ちながら障害物を避ける手法として、例えば Geraerts ら [5] は Path pruning や Shortcut を実行した後のパスに対して、障害物からある程度の距離を確保する手法を提案している。

図 2 Methods for smoothing path by using spline interpolation

3. ランダムサンプリングに基づく手法の改良

次に、RRT や PRM の動作計画手法自体に、評価関数を最小化する作用を含めようという試みが行われており、その一つとして RRT* と呼ばれる手法を紹介する [9] [10]。この手法は Asymptotic Optimality, つまり最適解に収束することが保証されていることが特徴である。

この手法の概要を Fig. 3 に示す。RRT* では、通常の RRT と同様にコンフィグレーション空間内でノード V とエッジ E から構成される木 $T = (V, E)$ をインクリメンタルに構築する。ランダムにコンフィグレーション q_{rand} を生成し、この q_{rand} から一番近い距離にある T に含まれるノードを選択する。この一番近い距離にあるノードのコンフィグレーションを $q_{nearest}$ とし、この q_{rand} と $q_{nearest}$ を結ぶ直線上で $q_{nearest}$ から STEP_SIZE だけ離れた位置に q_{new} を仮定する。ここで、RRT では q_{new} を新しいノードのコンフィグレーションとし、これを $q_{nearest}$ に接続する。それに対して RRT* では q_{new} を中心とする球を仮定し、その球の内部に含まれるノードのコンフィグレーションの集合を Q_{near} と定義する。ここで、 $q_{near} \in Q_{near}$ を q_{new} と直線により接続した場合のコストを以下により定義する。

図 3 Methods for tree construction in RRT* algorithm

$$c = Cost(q_{near}) + c(Line(q_{near}, q_{new})) \quad (1)$$

ここで、 $Cost(q_{near})$ は q_{near} に至ることに関するコストであり、 $c(Line(q_{near}, q_{new}))$ は q_{near} から q_{new} に移動するのにかかるコストである。 Q_{near} の要素の中で、このコストが最小になるものを選び、さらにノードを逆方向に辿った際のコストを考慮することで最終的に Q_{near} の要素の一つに対して q_{new} を接続する。文献 [9] では、RRT* によりサンプル数を増やすに従って初期コンフィグレーションと目標コンフィグレーションを結ぶ軌道が最短距離の軌道に近づく様子が示されている。つまり、本手法ではプランナによって最初に得られる軌道が最適な軌道という訳ではない。また、文献 [10] では、RRT* を実時間軌道計画問題に適用している。

4. 軌道最適化

次に、ロボットの軌道を陽に含んだ形で評価関数を定義し、この評価関数に対して関数空間において勾配を求めることで、評価関数を最小化するロボットの軌道を求める試みが行われている [12]~[14]。本章では、このような試みとして CHOMP (Covariant Hamiltonian Optimization for Motion Planning) [12] [13] を紹介する。ここで、この手法の特徴は、従来から提案されている軌道を仮想的なバネ質量系でモデル化する Elastic Strips Planning [11] などとは異なり、最初に与える軌道に干渉が生じていたとしても、最終的には干渉が生じていない軌道が得られる点である。つまり、最初に与える軌道としては、例えばロボットの初期姿勢と目標姿勢を直線軌道で結んだものを与えることができる。

CHOMP では、ロボットの軌道を $\xi: [0, 1] \rightarrow C \in R^n$ のような時間に関して正規化した滑らかな関数として定義

し、動作の滑らかさの指標と障害物を避けることに関する指標を足し合わせた以下のような評価指標を最小化する。

$$U(\xi) = F_{obj}(\xi) + \lambda F_{smooth}(\xi) \quad (2)$$

ここで、 λ は $F_{smooth}(\xi)$ に対する重みを表す。また、各評価指標は以下のように定義される。

$$F_{smooth}(\xi) = \frac{1}{2} \int_0^1 \left\| \frac{d}{dt} \xi(t) \right\|^2 dt \quad (3)$$

$$F_{obj}(\xi) = \int_0^1 \int_B col(x(\xi(t), u)) \left\| \frac{d}{dt} x(\xi(t), u) \right\| du dt \quad (4)$$

ここで、 $B \in R^3$ はロボットの表面上の点集合であり、 $x: C \times B \rightarrow R^3$ はロボットのコンフィグレーション q とロボット上の点 $u \in B$ の順運動学マッピングである。さらに、 $col: R^3 \rightarrow R$ は干渉していない場合は距離、干渉している場合は干渉の深さに応じて値が変化する関数である。次に、この評価関数の勾配を求め、最急降下法により干渉を避ける滑らかな軌道を求めている。ここで、距離計算のコストを下げるため、ロボットのリンクを直方体、球、あるいは楕円体のような単純な形状の集合体としてモデル化している。また、局所最小解から抜け出すために、HMC (Hamiltonian Monte Carlo) 法を用いている。この手法を用いることで、論文では 7 自由度マニピュレータや 4 足歩行ロボットの動作計画が実現できることを示している [13]。ここで、Ramirez ら [15] は CHOMP を応用することで、柔軟なリング状の部品を円柱に組み付けるための動作計画を行っている。この手法では、対象物の柔軟性を考慮したマニピュレーションが式 (2) の評価関数に柔軟性を考慮した項を追加するだけで実現されている。

なお、このような関数最適化に基づくアプローチは ROS (Robot Operating System) 上で使えるソフトウェアパッケージとして提供されている [16]。近年、このようなアプローチを使った研究が増えてきている背景には、ROS に対応したソフトウェアパッケージが用意されていることがあると考えられる。

5. その他の事例

本稿では、ロボットの初期コンフィグレーションと目標コンフィグレーションが与えられている条件下で、評価関数を最適化しつつ障害物を避ける動作を計画する問題に絞って最近の事例を紹介している。本章では、上記問題に厳密には当てはまらないが関連する研究事例をいくつか紹介する。

まず、現在のコンフィグレーションが与えられているという条件下で、インクリメンタルに障害物を避けるロボットの動作を生成する研究が近年多く行われている [17]~[19]。また、ヒューマノイドロボットの初期手先位置と目標手先位

置が与えられている条件下で評価関数を最適化する全身の動作を計画する研究が行われている [20]. さらに, 障害物を避けるという条件を含めないのであれば, 初期コンフィグレーションと目標コンフィグレーションを結ぶ最適軌道に関して従来から多くの研究が行われている (例えば文献 [21]).

6. 結 言

本稿では, ロボットの動作計画における最適化について議論した. 特に, RRT, PRM などランダムサンプリングに基づく動作計画法の Post Processor としての最適化手法, ランダムサンプリングに基づく動作計画法自体の改良による最適化手法, ならびに関数最適化による動作計画法の三つを主に紹介した.

本稿によって, ロボットの軌道計画の研究分野が着実に進歩を遂げていることがお分かりいただけたのではないかと思います. ロボットが実時間で障害物を厳密に避けつつ, 最適軌道で初期姿勢から目標姿勢へと到達することができるアルゴリズムも, もうすぐ開発されるのではないかと期待させられる. 本稿が, ロボット学会においてロボットの動作を生成することに関する研究を行っている研究者の動作計画の研究に対するモチベーションを高めることに寄与すると幸いです.

謝 辞 本稿を作成するにあたり, 平素より議論を行っている産業技術総合研究所の Ramirez Ixchel 氏, 吉田英一氏, 九州大学の辻徳生氏, ならびにタスクビジョン研究グループのメンバーに感謝します.

参 考 文 献

- [1] O. Khatib: "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," Int. J. Robotics Research, vol.5, no.1, pp.90-98, 1986.
- [2] Z. Shiller and S. Dubowsky: "Global Time Optimal Motions of Robotic Manipulators in the Presence of Obstacles," Proc. of IEEE Int. Conf. on Robotics and Automation, pp.370-375, 1988.
- [3] S.M. Lavalle: Planning Algorithms. Cambridge University Press, 2006.
- [4] H. Choset, K.M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki and S. Thrun: Principles of Robot Motion—Theory, Algorithms and Implementations—. The MIT Press, 2004.
- [5] R. Geraerts and M.H. Overmars: "Creating High-quality Paths for Motion Planning," Int. J. Robotics Research, vol.26, no.8, pp.845-863, 2007.
- [6] L. Kavraki and J. Latombe: 'Probabilistic roadmaps for robot path planning,' Practical Motion Planning in Robotics: Current Approaches and Future Directions. pp.33-53, Wiley, 1998.
- [7] K. Hauser and V. Ng-Thow-Hing: "Fast Smoothing of Manipulator Trajectories using Optimal Bounded-Acceleration Shortcuts," Proc. of IEEE Int. Conf. on Robotics and Automation,

pp.2493-2498, 2010.

- [8] K. Harada, S. Hattori, H. Hirukawa, M. Morisawa, S. Kajita and E. Yoshida: "Two-Stage Time-Parametrized Gait Planning for Humanoid Robots," IEEE/ASME Transactions on Mechatronics, vol.15, no.5, pp.694-703, 2010.
- [9] S. Karaman and E. Frazzoli: "Sampling-based Algorithms for Optimal Motion Planning," Int. J. of Robotics Research, vol.30, no.7, pp.846-894, 2011.
- [10] S.Karaman, M.R. Walter, A. Perez, E. Frazzoli and S. Teller: "Anytime Motion Planning using the RRT*," Proc. of IEEE Int. Conf. on Robotics and Automation, pp.1478-1483, 2011.
- [11] O. Brock and O. Khatib: "Elastic Strips: A Framework for Motion Generation in Human Environments," The International Journal of Robotics Research, vol.21, no.12, p.1031, 2002.
- [12] N. Ratliff, M. Zucker, J.A. Bagnell and S. Srinivasa: "CHOMP: Gradient Optimization Techniques for Efficient Motion Planning," Proc. of IEEE Int. Conf. on Robotics and Automation, pp.489-494, 2009.
- [13] M. Zucker, N. Ratliff, A. Dragan, M. Pivtoraiko, M. Klingensmith, C. Dellin, J.A. Bagnell and S. Srinivasa: "CHOMP: Covariant Hamiltonian Optimization for Motion Planning," Int. J. Robotics Research, vol.32, no.9-10, pp.1164-1193, 2013.
- [14] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor and S. Schaal: "STOMP: Stochastic Trajectory Optimization for Motion Planning," Proc. of IEEE Int. Conf. on Robotics and Automation, pp.4569-4574, 2011.
- [15] I. Ramirez, K. Harada and E. Yoshida: "Motion Planning for the Assembly of Elastic Ring Shaped Objects", 日本機械学会ロボティクスメカトロニクス講演会講演予稿集, 1P1-P04, 2014.
- [16] <http://www.ros.org/news/2009/09/chomp-motion-planner.html>
- [17] 金広, 吉田, Lamiraux, Kanoun, Laumond: "任意の多面体間に適用可能な干渉回避運動生成法", 日本ロボット学会誌, vol.27, no.1, pp.63-70, 2009.
- [18] A. Escande, N. Mansard and P.-B. Wieber: "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," Int. J. Robot. Res., vol.33, no.7, 2014.
- [19] O. Kanoun, F. Lamiraux, P.-B. Wieber, F. Kanehiro, E. Yoshida and J.-P. Laumond: "Prioritizing linear equality and inequality systems: application to local motion planning for redundant robots," Proc. of ICRA, pp.12-17, 2009.
- [20] M. Toussaint, M. Gienger and C. Goerick: "Optimization of Sequential Attractor-based Movement for Compact Behaviour Generation," Proc. of IEEE-RAS Int. Conf. on Humanoid Robots, 2007.
- [21] J.E. Bobrow, B. Martin, G. Sohl, E.C. Wang, F.C. Park and J. Kim: "Optimal robot motions for physical criteria," J. of Robotic systems, vol.18, no.12, pp.785-795, 2001.

原田研介 (Kensuke Harada)

1997 年京都大学大学院工学研究科博士後期課程修了. 同年広島大学工学部助手. 2002 年産業技術総合研究所知能システム研究部門研究員, 2005 年から 1 年間 Stanford 大客員研究員, 2013 年同研究所タスクビジョン研究グループ長, 現在に至る. ロボットハンドによる把持や操りに関する研究に従事. IEEE, 計測自動制御学会, 日本機械学会, システム制御情報学会の会員. 博士 (工学). (日本ロボット学会正会員)