Markov Random Fields for Stochastic Trajectory Optimization and Learning with Constraints

Mrinal Kalakrishnan*, Alexander Herzog†, Ludovic Righetti*†, and Stefan Schaal*†
kalakris@usc.edu, alexander.herzog@tue.mpg.de, ludovic.righetti@tue.mpg.de, sschaal@usc.edu
*CLMC Lab, University of Southern California, Los Angeles CA 90089
†Max Planck Institute for Intelligent Systems, Tübingen, Germany 72076

I. Introduction

The use of parameterized policies has become prevalent in reinforcement learning on high-dimensional robotic systems due to difficulties in representing and approximating the value function. Many state-of-the-art reinforcement learning algorithms work by iteratively sampling trajectories from a parameterized policy and updating its parameters to minimize the expected cost. However, explicit representation and satisfaction of constraints along trajectories has not been satisfactorily addressed in such sampling-based learning algorithms.

In this abstract, we present a stochastic policy parameterization for sampling and learning trajectories that satisfy constraints. We model the trajectory as a Markov Random Field (MRF) with pair-wise potentials that represent a smoothness cost. The resulting joint distribution is then conditioned on linear or locally linear constraints, which allows sampling of trajectories directly on the constraint manifold. Sparsity in the graph structure can be exploited to make this sampling process efficient. We show preliminary results from applying a sampling-based trajectory optimizer on a 100-DOF simulated planar robot with end-effector constraints.

II. Markov Random Field Trajectory Model

We represent a robot trajectory in a d-dimensional configuration space, discretized into T timesteps, resulting in a trajectory vector $\mathbf{x} \in \mathbb{R}^{Td}$. Each element of this parameter vector is considered a node in the MRF. The smoothness of this trajectory is measured as the sum of squared accelerations along each dimension of the trajectory:

$$Q_R(\mathbf{x}) = \sum_{j=1}^{d} (\mathbf{D}_{2,j}\mathbf{x})^{\mathrm{T}} (\mathbf{D}_{2,j}\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{R}\mathbf{x} \qquad (1)$$

where $\mathbf{D}_{2,j}$ is a second order finite differencing matrix for joint j, and $\mathbf{R} = \sum_{j=1}^{d} \mathbf{D}_{2,j}^{\mathrm{T}}\mathbf{D}_{2,j}$. The stochastic policy is then defined as:

$$P(\mathbf{x}|\boldsymbol{\theta}) = \exp\left(-(\mathbf{x}-\boldsymbol{\theta})^{\mathrm{T}}\mathbf{R}(\mathbf{x}-\boldsymbol{\theta})\right) \qquad (2)$$

where $\boldsymbol{\theta} \in \mathbb{R}^{Td}$ are the policy parameters which represent the mean trajectory. Samples from this policy lie in the vicinity of the mean, and deviations from the mean are smooth, since distances are measured with respect to the metric $\mathbf{R}$. This joint distribution can be seen as a Gaussian MRF, where each non-zero element of $\mathbf{R}$ corresponds to a pair-wise Gaussian potential between two nodes. Note that $\mathbf{R}$ has a sparse, banded-diagonal structure and only requires $\mathcal{O}(Td)$ of memory.

III. Sampling from the conditional distribution

Sampling from the multivariate normal joint distribution (2) is achieved by first computing the cholesky factorization of $\mathbf{R} = \mathbf{L}\mathbf{L}^{\mathrm{T}}$. Each sample $\mathbf{x}$ is then drawn by solving $\mathbf{L}^{\mathrm{T}}\mathbf{x} = \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} = \mathcal{N}(0, \mathbf{I})$. Linear constraints $\mathbf{A}\mathbf{x} = \mathbf{b}$ ($\mathbf{A} \in \mathbb{R}^{m \times Td}, b \in \mathbb{R}^m$) can be incorporated in this sampling process by conditioning the joint distribution (2) on the constraints: $P(\mathbf{x}|\boldsymbol{\theta}, \mathbf{A}\mathbf{x} = \mathbf{b})$, which is also multivariate normal. Although the resulting covariance matrix could be explicitly computed, it is computationally more efficient to first draw an unconditioned sample x followed by an optimal projection to obtain a conditioned sample x' as follows [2]:

$$x' = x - \mathbf{R}^{-1}\mathbf{A}^{\mathrm{T}}(\mathbf{A}\mathbf{R}^{-1}\mathbf{A}^{\mathrm{T}})^{-1}(\mathbf{A}\mathbf{x} - \mathbf{b}) \qquad (3)$$

The complexity of computing the projector (due to sparsity) is $\mathcal{O}(m^3 + mTd)$, which only needs to be computed once for an entire set of samples. Sampling is then $\mathcal{O}(mTd)$.

Conditioning on linear constraints provides a compact and general way to represent many typical scenarios, such as fixing the start and goal positions, velocities, and accelerations, passing through via points, and respecting kinematic loop closure constraints. Constraints that are globally linear can be satisfied exactly by this method. Non-linear kinematic constraints are locally linearized using the Jacobian, and can hence only be approximately satisifed during sampling, but typically converge after multiple iterations of the optimizer.

IV. Optimization and evaluation

Apart from the *intrinsic* trajectory cost (1) and constraints, we define an *extrinsic* cost $J(\mathbf{x})$, which is task-dependent and typically includes terms related to avoiding collisions and achieving the desired task. We use the STOMP algorithm [1] in conjunction with our constrained sampling procedure to optimize the parameters of the MRF policy to minimize the expected trajectory cost. We conducted a preliminary evaluation on a simulated planar robot, with varying degrees of freedom up to 100, and 100 time-steps per trajectory. Sampling and optimization time was observed to scale linearly with number of DOFs. We are currently working on applying this method for planning dynamic maneuvers with switching contact conditions on a 31-DOF humanoid robot, and expect to have results from these experiments before the workshop.

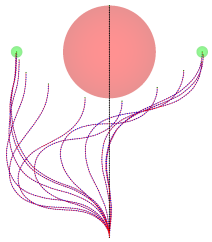

Fig. 1. A 100-DOF planar robot moves from point to point while keeping its end-effector upright and avoiding the red obstacle. The black vertical line shows the initial (stationary) trajectory.

References

[1] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal. STOMP: Stochastic Trajectory Optimization for Motion Planning. In *IEEE Intl Conf. on Robotics and Automation*, pages 4569–4574, 2011.

[2] Håvard Rue. Fast sampling of Gaussian Markov random fields. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63 (2):325–338, 2001.