Study of Path Integral based stochastic optimal control and Implementation of Model Predictive Path Integral for Aggressive Driving

Rajnish Tiwari, Vishnu R, Ameya Wagh

{rtiwari,vrudrasamudram,aywagh} @ wpi.edu

Table of contents

Introduction and Motivation

Introduction

- Optimal Control, a mathematical description of how to act optimally to get future rewards
- Uncertainties and model imperfections
- Stochastic Optimal Control

Motivation

- Autonomous driving
- Imperfect model and uncertain environment
- Operating at vehicle's friction limits - a great challenge
- Apply hierarchical models, get feasible path - Difficult
- Classical optimal controllers - Computationally expensive, No use for on-the-fly maneuvers
- Motivation to learn about new methods
- **Model Predictive Control based on Path Integral Control**

Goals

- Reachable Goals
  - Study of Path Integral based Stochastic Optimal Control and Model Predictive Control
  - Implementation of MPPI Algorithm for aggressive driving
- Stretched Goals
  - A simple generalized ROS Package for MPPI

Path Integral

- To develop Optimal Control Algorithms based on stochastic sampling of trajectories
- Requires no derivatives of either dynamics or cost functions
- Exponential transformation with noise assumptions - Stochastic HJB -¿ Linear PDE
- Linear PDE to Path Integral using Feynman-Kac Lemma
- This takes form of an expectation over trajectories

Methodology

Theory

- Consider stochastic dynamic system which is affine in control

$$dx = \mathbf{F}(x_t,\ \mathbf{u}_t,\ t)dt + \mathbf{B}(x_t,\ t)dw$$

x_t - State at time t
dW - Brownian motion

- Classic Control System [3]

$$\mathbf{u}^*(\cdot) = \underset{\mathbf{u}(\cdot)}{\operatorname{argmin}}\ \mathbb{E}_{\mathbb{Q}}\left[\phi(x_T, T) + \int_{t_0}^{T} \mathcal{L}(x_t,\ \mathbf{u}_t,\ t)\mathrm{d}t\right]$$

$\phi(x_T, T)$ - Final Cost
$\mathcal{L}(x_t,\ \mathbf{u}_t,\ t)$ - Running Cost

Theory

Interpretation of PI in terms of free energy and relative entropy based on following equality:

$$-\lambda\mathcal{F}(S(\tau)) = \inf_{\mathbb{Q}} \left[\mathrm{E}_{\mathbb{Q}}[S(\tau)] + \lambda\mathbb{D}_{KL}(\mathbb{Q}\|\mathbb{P})\right] \tag{1}$$

- $S(\tau)$ - State dependent cost-to-go term
- $\mathcal{F}(S(\tau))$ - Free energy distribution over $S(\tau)$
  $\mathcal{F}(S(\tau)) = \log(\mathbb{E}_{\mathbb{Q}}[\exp(-\frac{1}{\lambda}S(\tau))])$
- $\mathbb{P}$ - Probability measure over the space of trajectories
- $\mathbb{Q}$ - Any Probability measure such that $\mathbb{Q}$ is absolutely continuous with $\mathbb{P}$
- $\mathbb{D}_{KL}(\mathbb{Q}\|\mathbb{P})$ - KL Divergence

Theory

- Infimum of equation (1) w.r.t uncontrolled dynamics

$$\frac{\mathrm{d}\mathbb{Q}^*}{\mathrm{d}\mathbb{P}} = \frac{\exp(-\frac{1}{\lambda}S(\tau))}{\mathbb{E}_{\mathbb{P}}[\exp(-\frac{1}{\lambda}S(\tau))]}$$

- Relative entropy between $\mathbb{Q}^*$ and $\mathbb{Q}(\mathbf{u})$ gives the following minimization problem

$$\mathbf{u}^*(\cdot) = \underset{\mathbf{u}(\cdot)}{\operatorname{argmin}}\ \mathbb{D}_{KL}(\mathbb{Q}^* \| \mathbb{Q}(\mathbf{u}))$$

Algotithm

Given: K: Number of samples;
N: Number of timesteps;
$(\mathbf{u}_0, \mathbf{u}_1, ...\mathbf{u}_{N-1})$: Initial control sequence;
$\Delta t, \mathbf{x}_{t_0}, \mathbf{f}, \mathbf{G}, \mathbf{B}, \mathbf{B}_E$: System/sampling dynamics;
$\phi, q, \mathrm{R}, \lambda$: Cost parameters;
$\mathbf{u}_{\text{init}}$: Value to initialize new controls to;

while *task not completed* **do**
- **for** $k \leftarrow 0$ **to** $K-1$ **do**
  - $\mathbf{x} = \mathbf{x}_{t_0}$;
  - **for** $i \leftarrow 1$ **to** $N-1$ **do**
    - $\mathbf{x}_{i+1} = \mathbf{x}_i + (\mathbf{f} + \mathbf{G}\mathbf{u}_i)\Delta t + \mathbf{B}_E \epsilon_{i,k}\sqrt{\Delta t}$;
    - $\tilde{S}(\tau_k) = \tilde{S}(\tau_k) + \tilde{q}(\mathbf{x}_i, \mathbf{u}_i, \epsilon_{i,k}, t_i)$;
- **for** $i \leftarrow 0$ **to** $N-1$ **do**
  - $\mathbf{u}_i = \mathbf{u}_i + \boldsymbol{\mathcal{H}}^{-1}\boldsymbol{\mathcal{G}}\left[\sum_{k=1}^{K}\left(\frac{\exp\left(-\frac{1}{\lambda}\tilde{S}(\tau_k)\right)\frac{\epsilon_{j,k}}{\sqrt{\Delta t}}}{\sum_{k=1}^{K}\exp\left(-\frac{1}{\lambda}\tilde{S}(\tau_k)\right)}\right)\right]$;
- send to actuators$(\mathbf{u}_0)$;
- **for** $i \leftarrow 0$ **to** $N-2$ **do**
  - $\mathbf{u}_i = \mathbf{u}_{i+1}$;
- $\mathbf{u}_{N-1} = \mathbf{u}_{\text{init}}$
- Update the current state after receiving feedback;
- check for task completion;

Assumptions

- To make the PI as back-in-time PDE it is assumed that

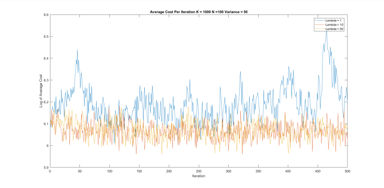
$$BB^T = \lambda G R^{-1} G^T$$

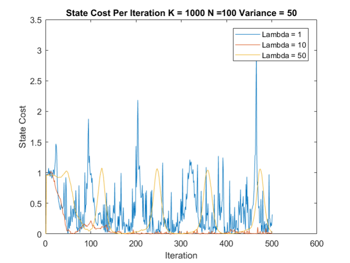
- For simulations, we consider dynamics in the following form

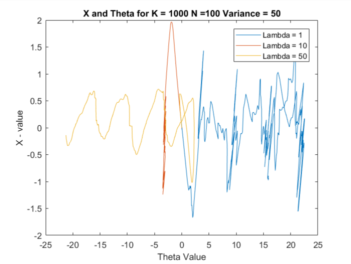
$$dx = f(x_t, t)\Delta t + G(x_t, t)(u(x_t, t)\Delta t + \frac{1}{\sqrt{\rho}})\epsilon\sqrt{\Delta t}$$

So, we have $B_c(x_t, t)\frac{\epsilon}{\sqrt{\Delta t}} = G_c(x_t, t)\delta u$

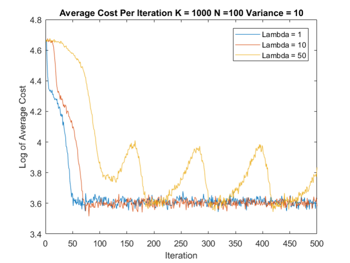
Study on Effect of Parameters λ and υ

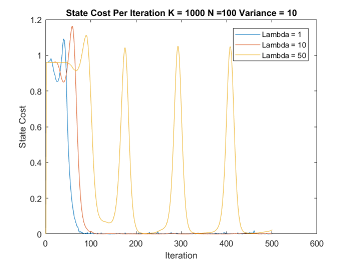


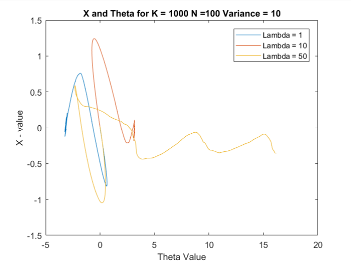




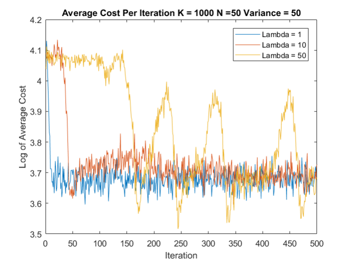
Study on Effect of Parameters λ and υ

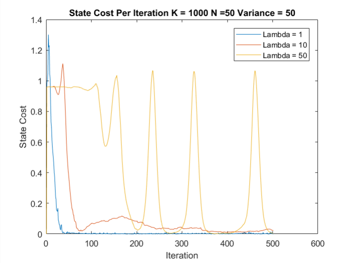


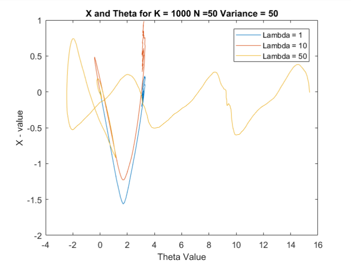




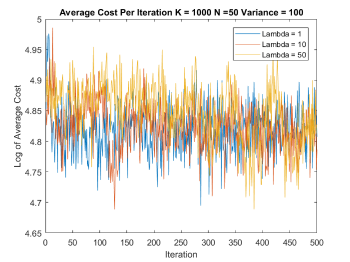
Study on Effect of Parameters λ and υ

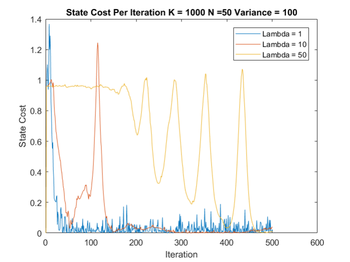


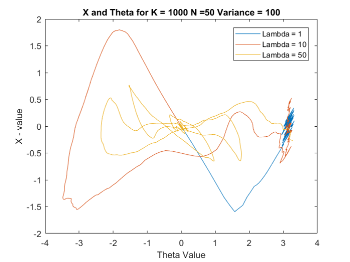




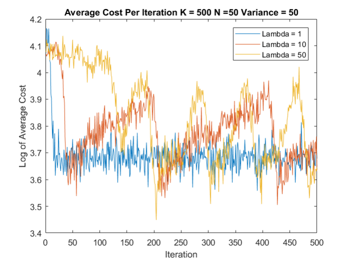
Study on Effect of Parameters λ and υ

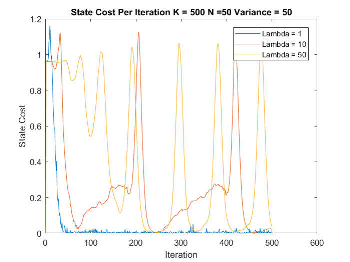


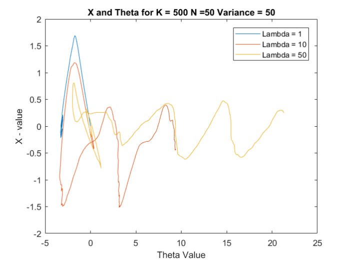




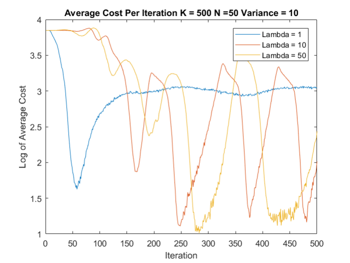
Study on Effect of Parameters λ and υ

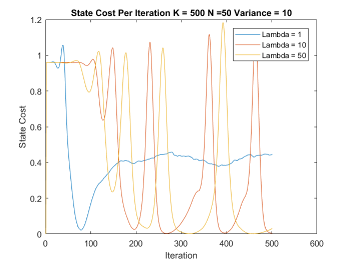


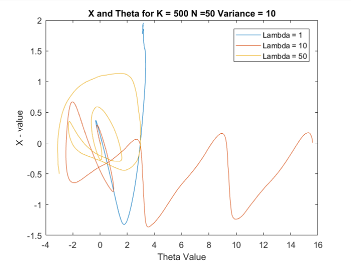




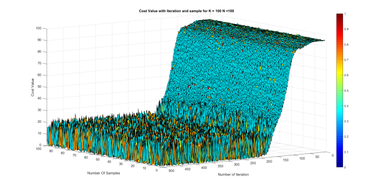
Study on Effect of Parameters λ and υ

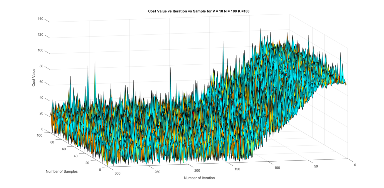


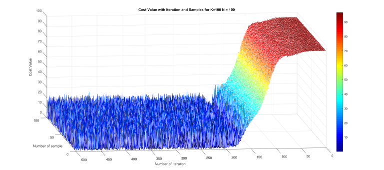


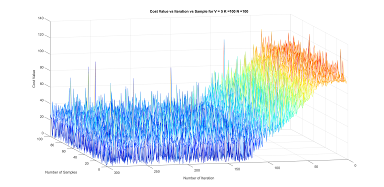


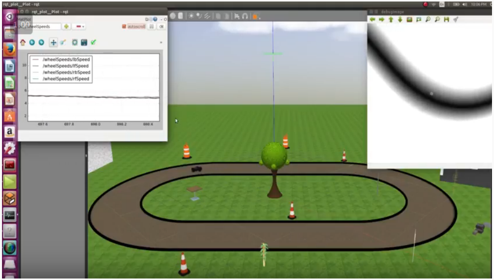
Cost function











States evolution based on cost

Implementation

Dynamic Model

- Cart-pole [2]

$$\ddot{x} = \frac{1}{m_c + m_p sin^2(\theta)} \left(u + m_p sin\theta \left(l\dot{\theta}^2 + gcos\theta \right) \right)$$

$$\ddot{\theta} = \frac{1}{l\left(m_c + m_p sin^2(\theta)\right)} \left(-ucos\theta - m_p l\dot{\theta}^2 cos\theta sin\theta \right)$$

Dynamic Model

- Car[1]

$$\dot{U}_x = \frac{1}{m}\sum F_x = \frac{1}{m}F_{xR}$$
$$\dot{U}_y = \frac{1}{m}\sum F_y - U_x r = \frac{1}{m}(F_{yF} + F_{yR}) - U_x r$$
$$\dot{r} = \frac{1}{I_z}\sum M_z = \frac{1}{I_z}(L_f F_{yF} - L_r F_{yR})$$
$$\dot{X} = U_x \cos\psi - U_y \sin\psi$$
$$\dot{Y} = U_x \sin\psi + U_y \cos\psi$$
$$\dot{\psi} = r$$

$$F_{yF} = C_{\alpha f}\alpha_f$$
$$F_{yR} = C_{\alpha r}\alpha_r$$

$$\alpha_f = \delta - \frac{U_y + L_f r}{U_x}$$
$$\alpha_r = \frac{L_r r - U_y}{U_x}.$$

Cost function

- Cart-pole
  - Running cost

$$6 \times p^2 + 12 \times (1 + \cos\theta)^2 + 0.1 \times \dot{\theta}^2 + 0.1 \times \dot{p}^2$$

- Car - AutoRally
  - Control cost

$$W \times U \times U^T$$

  - Speed cost

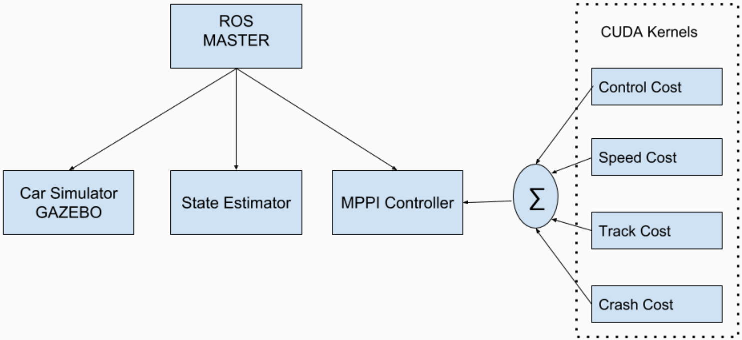
$$(v - v_{desired})^2$$

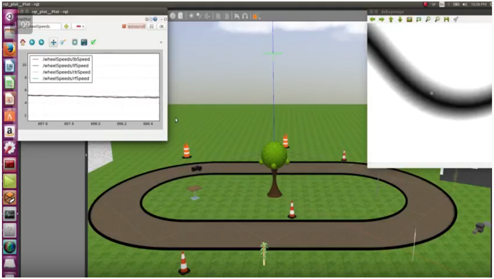
  - Track cost

$$100 \times |(\frac{x}{13})^2 + (\frac{y}{6})^2 - 1|$$

  - Crash cost

$$\begin{cases} 0 & heading \leq \frac{\pi}{2} \\ constant & otherwise \end{cases}$$

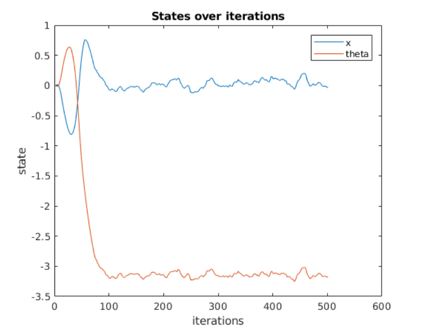

ROS
MASTER
Car Simulator
GAZEBO
State Estimator
MPPI Controller
Σ
CUDA Kernels
Control Cost
Speed Cost
Track Cost
Crash Cost

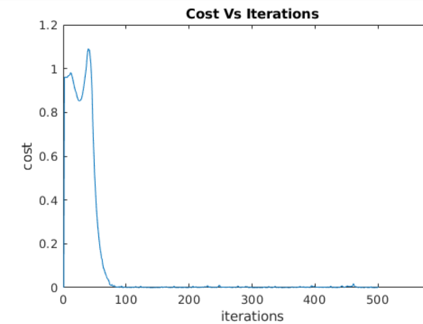


Autorally testing

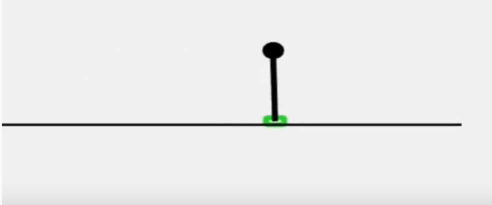
Conclusion

States over time

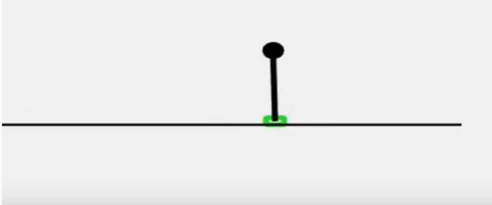




States over time



Inverted Pendulum - Cost function 1



Inverted Pendulum - Cost function 2

Discussions

Work accomplished

- Study of PI based stochastic optimal controller
- Implementation of MPPI
- MPPIController https://github.com/vvrs/MPPIController

Future Work

- Obstacle avoidance by assigning infinite cost to obstacles
- Avoiding local-minima, analysis
- Investigate Monte Carlo methods for MPPI

References I

K. F. B. F.Zhang, J. Gonzales.
Autonomous drift cornering with mixed open-loop and closed-loop control.

G. Williams, A. Aldrich, and E. A. Theodorou.
Model predictive path integral control: From theory to parallel computation.
Journal of Guidance, Control, and Dynamics, 40(2):344–357, Jan 2017.

G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou.
Aggressive driving with model predictive path integral control.
In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1433–1440, May 2016.