Guided Policy Search using Sequential Convex Programming
For Initialization of Trajectory Optimization Algorithms

STUDENTS: Taewan Kim with Purnanand Elango and Behcet Acikmese

Motivation
- Initial trajectory guess is a key input of trajectory optimization algorithms
- It can impact the speed of convergence and reliability of final trajectory
- We generate the initial trajectory guess by training neural net policy

Problem Formulation
- Optimal control problem input "u" comes from neural net.
  \[ \min \sum_{t=1}^{T} \mathcal{L}(x_t, u_t) \]
  \[ x_{t+1} = f(x_t, u_t), \quad x_0 = x_0 \]
- After optimizing, we can generate a new trajectory by propagating dynamics with the trained policy
  \[ x(t) = x_0, \quad u(t) = \theta(t), \quad t \in [0, T] \]

How to train neural net policy?: Guided policy search
- Imitation learning (IL) by trajectory optimization (TO) algorithms
- Generate multiple trajectories by TO as data, then train neural net policy
- However, sometimes the trajectory data is too hard to learn directly
- Guided policy search (GPS) is motivated from the problem of IL

Background: Sequential Convex Programming (SCP)
- SCP is trajectory optimization based on convex optimization
- Through linearization and discretization, it approximates the problem to convex sub-problem
- It repeats this convex optimization problem

Guided Policy Search via SCP
- We develop a new GPS method by employing the idea of SCP
  - First, we solve single iteration of SCP
  - Second, around the solution of the first step, we generate trajectories with LQR gain
  - Third, the trajectory data in second step is used to train the policy
- Repeat the above steps until the convergence of neural net policy

Detail in trajectory update via convex optimization
- Problem formulation
  \[ \min_{x_{t+1}, u_t} \sum_{t=1}^{T} \mathcal{L}(x_t, u_t) + \lambda(x_{t+1}) + \lambda(u_t) \]
  \[ x_{t+1} = f(x_t, u_t), \quad x_0 = x_0 \]
  \[ u_t = \theta_t(x_t) \]
- Similar with the sub-problem of SCP
- But, it has an additional penalty \[ \lambda(u_t) \]
- This penalty enforces the solution is closed to the policy
- So, trajectory update adapts to the neural network

Detail in policy update via supervised learning
- Policy training step
  - With data, the policy is trained by supervised learning
  - Any neural network optimizer can be used

Application to minimum-fuel powered descent guidance
- 6-dof powered descent guidance problem for a reusable rocket
- Has the following constraints:
  - 6-dof rocket dynamics, mass, glide angle, angle of attack, thrust, gimbal angle
- Performance comparison

Future Work, References, and Acknowledgments
- Future works
  - Obstacle avoidance problems
  - Final time-free problem
  - Policy with partial information of state
- Faculty: Behcet Acikmese
- Students: Taewan Kim, Purnanand Elango