R2/Ishank Juneja/16D070012

January 18, 2020

Compared to simulated agent-environment interactions, real-world robotic control tasks pose unique challenges in terms of their complex and noisy dynamics. Helicopter-Flight in particular is a difficult control problem, not only due to the numerous degrees of freedom, but also because of the simultaneous interacting actions required. Controlling a helicopter can be modelled as a sequential decision making task where decisions have to be made about control inputs every fraction of a second. The paper, by Andrew Ng and others, looks at a reinforcement learning based approach to control a helicopter's trajectory.

In a robotic control task, creating an accurate training model for vehicle dynamics is crucial since it is impractical (and in this case unsafe) to learn online in the real world. The dynamical model used for training should incorporate as much expert knowledge as possible about the underlying task so that learning becomes tractable. In this paper, the authors exploit environmental symmetries and incorporate prior knowledge into their model so that the onus of learning these relationships does not fall onto the algorithm. Through diligent analysis the authors create an accurate model of the helicopter which they use to come up with good policies.

The authors use Monte-Carlo simulations to get estimate of the value functions associated with various policies and then perform policy search to arrive at an optimal-policy for the task of hovering the helicopter at a fixed location in space. To perform policy search the authors used both *Gradient-Ascent* and *Random-Walk* based approaches. As pointed out in the paper both perform comparably well. Through this training procedure, the authors are successful in determining a policy that performs significantly better at the hovering task than a trained human pilot.

After being successful at the hovering task, the authors take it a step further by attempting to train the helicopter to follow arbitrary trajectories. In principle, once hovering at a fixed location is perfected, any trajectory can be broken down into a sequence of points to "hover" at for fractions of a second. However, as the authors point out, the reward, or rather the penalty, scheme needs to be modified slightly when we plan to undertake continuing motion. This includes not incentivizing aggressive manoeuvrers and providing shaping rewards for making forward progress along the trajectory.

After reading and understanding the motives of the paper, a question that I have is,

- In the problem tackled by this paper, the state and policy spaces were continuous, so the authors used policy-search methods to learn good policies. In general, in Reinforcement Learning tasks are there situations with a finite/discrete state and action space where we would use policy-search over the control approach of estimating action-value functions?