

[3.1.2021]

## [Policies Modulating Trajectory Generators]

### Summary

This paper thinks, to learn the behavior of robots, we need to extract two layers of knowledge. One layer is that they called trajectory generator  $\rightarrow$  some essential parameters are rather stable and important to define the trajectory, we should leverage this kind of information instead let policy learning do the whole work. (prior matters).

One interpretation is that TG provides a suboptimal solution, and the policy learned add correctness to that solution.

Another interpretation is that although TG (ideally) should explain some physical meanings, from the perspective of learning process, it is still just a black-box, (we give meaning to the parameters that are fed into and given from the black-box). Therefore, what TG does here is that it plays a role a time-buffer, it saves history information, and this should be helpful to generate correct motions.

- The architecture is the important part, the actual how they implement TG is not that

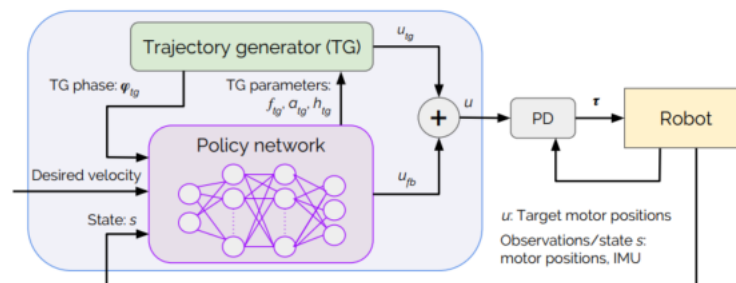


Figure 5: Adaptation of PMTG to the quadruped locomotion problem.

important. And the idea of phase function during a phase is interesting.

- Control signals are motor positions here. (although I am not quite clear what it means)
- TG phase is important. Another test shows only put time into observation and don't use trajectory generator (or other TG parameters to generate  $u_{tg}$ . It still works much

better than not using time at all. But worse than using everything. (doubt the necessity of using everything.... time is more important)

## Comparison

- Delete TG makes it fail to train the policy.
- Trajectory generator is very abstract, in another science robot paper, ANYmal use a different trajectory generator → output is the foot position instead of motor position as in this paper.

## Thoughts

- Learning problem in robotics → prior knowledge matters and should be used.
- Need to check another paper: sim→real
- DO I need to look at PPO again...?