R06/Sarthak Mittal/200050129

February 8, 2023

The task of **locomotion of legged robots** involves several complexities, such as coordination, stability, friction and optimization. This paper proposes an approach to optimize the motion of a **quadrupedal** using **policy gradient** reinforcement learning (RL). The testing is conducted using **Sony Aibo** robot, and it **performed better** than existing solutions. The existence of competitions such as **RoboCup** serves as an incentive to improve the speed of robots. The past methods used hand-tuning of parameters of the "gait", which is time-consuming and required human experts. There have been recent methods that used machine learning as well.

The gait of Aibo is determined by a series of positions for the **joints** in its legs. Some of the recent approaches also tried using joint velocities, body positions, and trajectories of the feet. The approaches differed in the parametrization of the **"loci"** used by the feet, and the method used to learn or tune them.

This paper uses a previous hand-tuned gait having a **half-elliptical loci** as the starting point. The complete model uses **12 parameters** - front and rear loci (3 parameters each), locus length, skew multiplier, heights (front and rear), time for foot movement through locus, and fraction of time spent by a foot on ground. The objective that they considered was **forward speed**.

They consider each possible assignment as defining an open-loop policy. With the assumption that the policy is differentiable with respect to all parameters, they follow the **gradient** to reach a **local optimum**. The lack of simulators mandates the use of real-world learning. To create an efficient method, they consider that the only effect of sensory input is adjustment for **noise compensation**. The trade-off is that this method only converges to a **local optimum**.

The **iteration** of their approach starts with an initial parameter vector, estimates the partial derivatives of the objective (by evaluating randomly generated policies near the original), and computes a **score** that serves as the **"measure"** of speed. Following this, they group the policies into 3 sets for each dimension, compute the average score and construct an **adjustment vector**, which is added to the initial policy (after normalization).

The learning process was conducted using three Aibos walking (learning in parallel) and them sending the results back after reaching their landmarks, preparing for the next set of parameters. They evaluated **15 policies** in each

iteration, and each policy was averaged over **3 traversals** to reduce noise (45 traversals per iteration).

Their algorithm was able to achieve the **fastest known gait** for Aibo in just over **1000 traversals** (in around **3 hours**). The large variations in the learning could be due to **noise**, or the **large step size** that was taken to adjust for the small values of adjustment ϵ.

They attribute their overall success to (1) policy gradient algorithm being particularly **effective** (2) gait implementation is **superior**. They acknowledged that the **starting point** could also have played a role. The best results were obtained when the initial parameters were roughly hand-tuned (but not overly-tuned) and gave a reasonable but slow gait.

The **advantages** of using automated optimization are (1) less bias which leads to discovering settings that humans would otherwise not try (2) there is a discrepancy between ideal or requested loci and the real loci, which could make hand-tuning difficult, but the learning process is not dependent on the semantics of the parameters (3) if repeated surface switching is needed, hand-tuning would need much more human intervention (4) this algorithm does not require a central controller over policies.

Their approach has a few **drawbacks** - (1) it requires a reasonable starting policy (2) policies are not tuned for any particular robot (wear and tear can lead to significant differences) (3) omni-directional gaits could be considered.

All things considered, their algorithm enables quadrupedal robots to fine-tune for speed **without human intervention**.