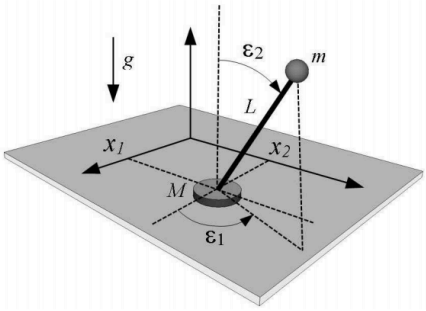
Aydar Akhmetzyanov

MCP HW5



The system has four generalized coordinates: (x_1, x_2) are variables for representing the position of the puck on a plane; $(\varepsilon_1, \varepsilon_2)$ are two angles (precession and nutation) for representing the status of the pendulum.

The dynamics of the system are

$$\frac{d}{dt}\left[\frac{\partial \mathcal{L}}{\partial \dot{\varepsilon}_1}\right] - \frac{\partial \mathcal{L}}{\partial \varepsilon_1} = 0 \qquad \frac{d}{dt}\left[\frac{\partial \mathcal{L}}{\partial \dot{\varepsilon}_2}\right] - \frac{\partial \mathcal{L}}{\partial \varepsilon_2} = 0$$

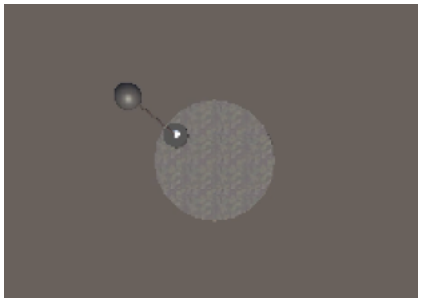
$$\frac{d}{dt}\left[\frac{\partial \mathcal{L}}{\partial \dot{x}_1}\right] - \frac{\partial \mathcal{L}}{\partial x_1} = F_1 \qquad \frac{d}{dt}\left[\frac{\partial \mathcal{L}}{\partial \dot{x}_2}\right] - \frac{\partial \mathcal{L}}{\partial x_2} = F_2$$

with F_1, F_2 being external forces acting on the puck along x_1, x_2 axes

---

The task: to find an induced behavior of the system when the puck is forced to move along of a circle of a radius R while the pendulum stays above the horizontal for all the time.

SOLUTION

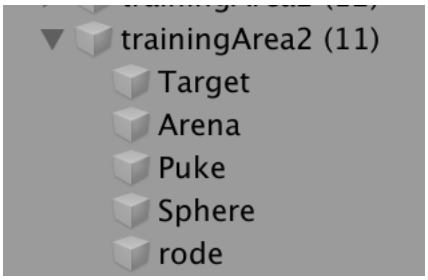


Demo video available here:
https://drive.google.com/open?id=1ezv51f6ls_nEG6qEB90c4iAFDtbJXPuE

To simulate the environment, I used Unity 3d engine with integrated NVIDIA PhysX engine. This product provide ability to model complex scenes with realistic physical simulation which is highly used in robotics as well. Video demonstrates the results.

PROBLEM MODELLING WITH UNITY

SCENE CONTENT



Target is a goal (circle movement) distance to which is calculated as a reward.
Puck is our main agent, rigid body connected with Puck with the center of Sphere object. Puck and Sphere are the only objects with mass.
Control solution. As an experiment I tried Unity ML agent's library. The Unity Machine Learning Agents Toolkit (ML-Agents) is an open-source Unity plugin that enables games and simulations to serve as environments for training intelligent agents. Agents can be trained using reinforcement learning, imitation learning, neuroevolution, or other machine learning methods through a simple-to-use Python API.
RL method in this library based on Proximal Policy Optimization. Full description is available here: https://arxiv.org/abs/1707.06347

MODEL PARAMETERS

```
trainer: ppo
batch_size: 1024
```

```
    beta: 5.0e-3
    buffer_size: 20480
    epsilon: 0.2
    gamma: 0.95
    hidden_units: 256
    lambd: 0.95
    learning_rate: 3.0e-4
    max_steps: 5000000
    memory_size: 256
    normalize: false
    num_epoch: 3
    num_layers: 2
    time_horizon: 128
    sequence_length: 64
    summary_freq: 1000
    use_recurrent:
    use_curiosity: true
    curiosity_strength: 0.01
    curiosity_enc_size: 128
```

ACTION SPACE

Agent control force applied to puck.

```
controlSignal.x = vectorAction[0];
controlSignal.z = vectorAction[1];
rBody.AddForce(controlSignal);
```

OBSERVATION SPACE

```
// Target and Agent positions
AddVectorObs(sphere.transform.localPosition);

AddVectorObs(this.transform.localPosition.x);
AddVectorObs(this.transform.localPosition.z);

AddVectorObs(this.target.transform.localPosition.x);
AddVectorObs(this.target.transform.localPosition.z);


// Agent and target velocity
AddVectorObs(rBody.velocity.x);
AddVectorObs(rBody.velocity.z);
AddVectorObs(rBodySphere.velocity.x);
AddVectorObs(rBodySphere.velocity.z);
```

```
AddVectorObs(Vector3.Distance(this.transform.localPosition,
                                    new Vector3(0, 0, 0)));
```

STOP CONDITION

```
// too close to middle
if (distanceToMiddle < 1f)
{
SetReward(-1.0f);
Done();
return;
}

// Fell off platform
if (distanceToMiddle > 2.4f)
{
SetReward(-1.0f);
Done();
return;
}

//ball felt
if ( sphere.transform.localPosition.y < 2f)
{
SetReward(-1.0f);
Done();
return;
}
```

REWARD FUNCTION

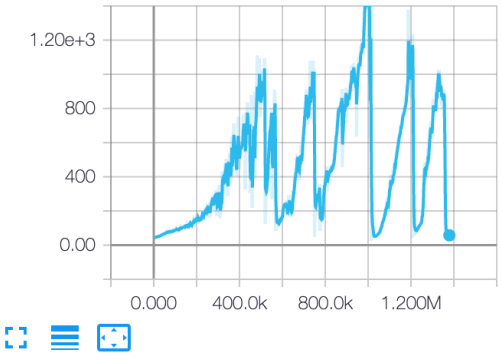
Distance to target + y position of sphere

```
SetReward((1 / (0.1f + Vector3.Distance(this.target.transfo
rm.localPosition, this.transform.localPosition)))/10 +
(this.sphere.transform.localPosition.y/6));
```

TRAINING PROCEDURE

It is interesting that training agent faces problems when it reaches 90-degree milestone.

Environment/Cumulative Reward



Environment/Episode Length