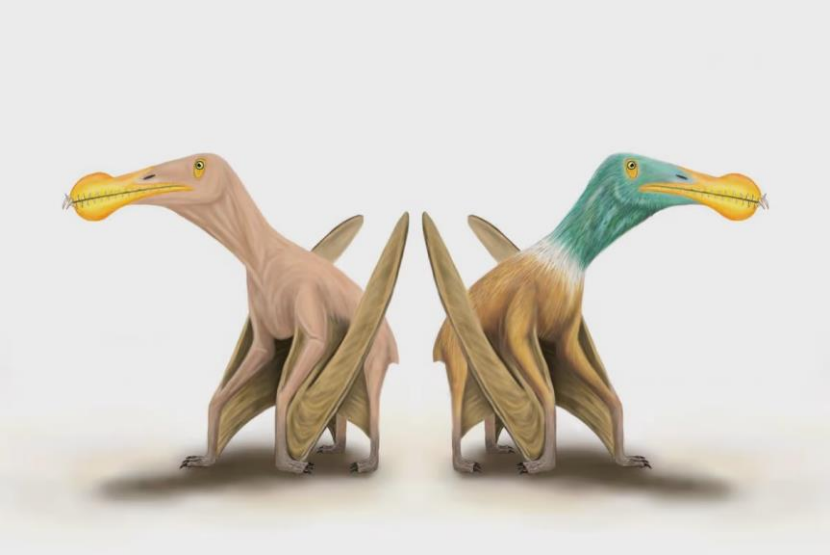
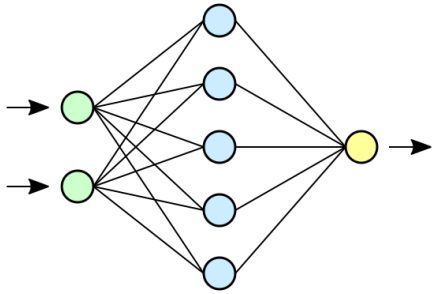
New project



+



= ???

- Unique locomotion – two modes
- Robot that can walk and fly would be useful
- Drive and fly is easy, walk and fly has benefits

Methodology, steps, goal

Methodology:

- Use popular RL libraries: MuJoCo, Brax, Gymnasium, Stable Baselines
- Train on GPU for vastly more training (500x+ faster, 2000+ in parallel)

Steps:

- Build custom model (CAD)
- Setup model for simulation
- Train in batches while tuning hyperparams, architecture, policy, etc.
- Optimize for natural looking and reasonably fast gait

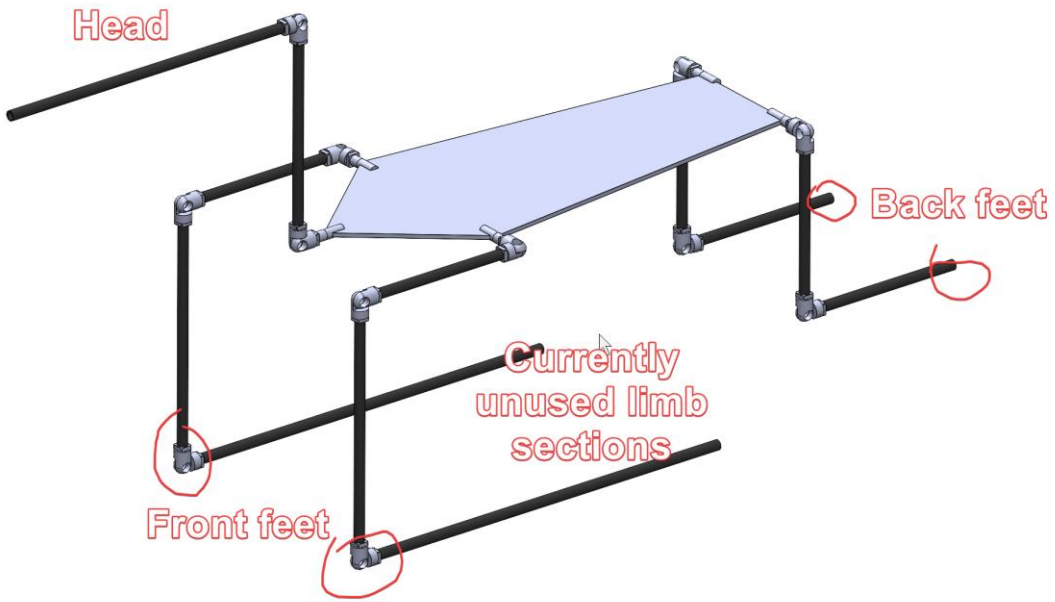
Goal / success definition:

- Sim a pterosaur model walking with a **reasonable-looking gait**
- Eventually (long after this class): build robot, get it walking, then train it to fly!

Model, URDF, MJCF

Got familiar with RL libraries through examples (MuJoCo, Gymnasium, Stable Baselines)

Modified example code: python and MJCF



→

```
<origin
  xyz="0.000645734028304206 8.91528981195044E-09 0.00570370451483394"
  rpy="0 0 0" />
<mass
  value="0.000552160381379268" />
<inertia
  ixx="1.37754406098898E-08"
  ixy="-1.03297312070412E-13"
  ixz="1.40112923189962E-09"
  iyy="1.28003109152368E-08"
  iyz="-4.97237598467118E-14"
  izz="8.09062527386862E-09" />
</inertial>
<visual>
<origin
  xyz="0 0 0"
  rpy="0 0 0" />
<geometry>
<mesh
  filename="package://BA14/meshes/NeckBase.STL" />
</geometry>
<material
  name="">
<color
  rgba="0.792156862745098 0.819607843137255 0.933333333333333 1" />
</material>
</visual>
<collision>
<origin
  xyz="0 0 0"
  rpy="0 0 0" />
<geometry>
<mesh
  filename="package://BA14/meshes/NeckBase.STL" />
</geometry>
</collision>
```

+ meshes

→

```
<worldbody>
<light cutoff="100" diffuse="1 1 1" dir="-0 0 -1.3" directional="true" expone
<geom conaffinity="1" condim="3" material="MatPlane" name="floor" pos="0 0 0"
<!-- <geom type="mesh" rgba="0.792157 0.819608 0.933333 1" mesh="torso"/> -->
<body name="torso" pos="0 0 0.2">
<camera name="side" pos="0 -3 1" xyaxes="1 0 0 0 1 2" mode="trackcom"/>
<joint armature="0" damping="0" limited="false" margin="0.01" name="root" p
<geom type="mesh" rgba="0.792157 0.819608 0.933333 1" mesh="torso"/>
<!-- <inertial pos="0 0 0.000645734" quat="0.707105 0 0 0.707108" mass="0.0
<body name="NeckBase" pos="-0.17145 0 0" quat="0.707105 0 0 0.707108">
<inertial pos="0.000645734 8.91529e-09 0.0057037" quat="0.993453 0 -0.114
<joint name="NeckUpDown" pos="0 0 0" axis="-1 0 0" range="0.5 1.5"/>
<geom type="mesh" rgba="0.792157 0.819608 0.933333 1" mesh="NeckBase"/>
<body name="Neck" pos="0 0 0.0127">
<inertial pos="-0.000127501 1.76034e-09 0.0698158" quat="1 0 -0.0002325
<joint name="NeckYaw" pos="0 0 0" axis="0 0 1" range="-1.3 1.3"/>
<geom type="mesh" rgba="0.792157 0.819608 0.933333 1" mesh="Neck"/>
<body name="Head" pos="0 0 0.127" quat="0.707105 -0.707108 0 0">
<inertial pos="8.60607e-05 1.18819e-09 0.0943231" quat="1 0 -5.67425e
<joint name="HUpDown" pos="0 0 0" axis="1 0 0" range="-1 1"/>
<geom type="mesh" rgba="0.298039 0.298039 0.298039 1" mesh="Head"/>
</body>
</body>
</body>
<body name="LWingBase" pos="-0.0762 -0.06985 0" quat="-2.59734e-06 2.59735e
<inertial pos="-8.91529e-09 -0.000645734 0.0069963" quat="0.702477 0.0807
<joint name="LShouldRot" pos="0 0 0" axis="0 0 1" range="-0.3 1.5"/>
<geom type="mesh" rgba="0.792157 0.819608 0.933333 1" mesh="LWingBase"/>
<body name="LWingProx" pos="0 0 0.0127" quat="0.707105 0 0.707108 0">
<inertial pos="-0.000122961 0.000122961 0.05715" quat="0.923879 0.00024
<joint name="LShouldExt" pos="0 0 0" axis="0 1 0" range="-1.8 0.3"/>
<geom type="mesh" rgba="0.792157 0.819608 0.933333 1" mesh="LWingProx"/
```

Training: Local (CPU)

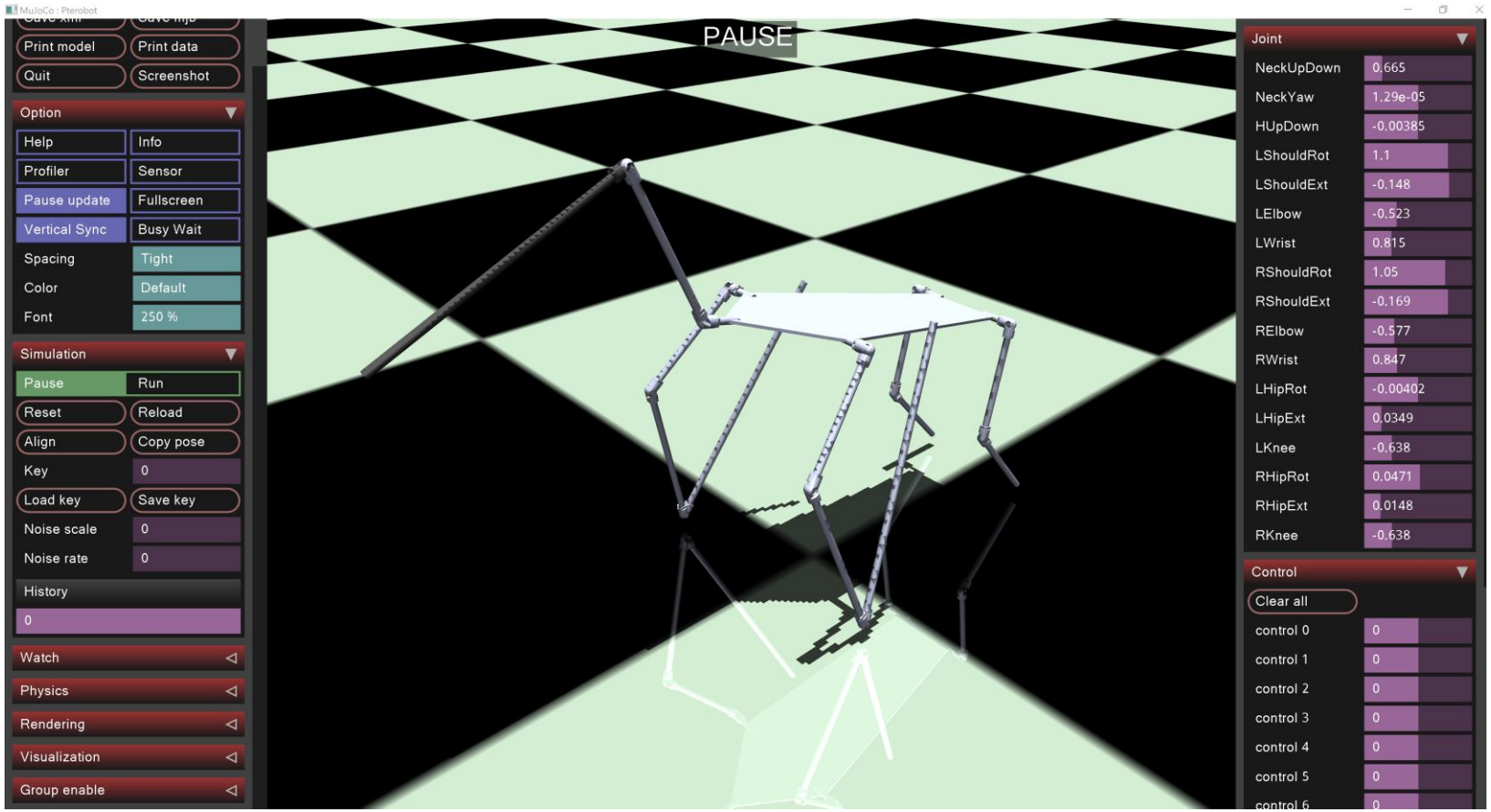
Defined reward function

Set starting pose

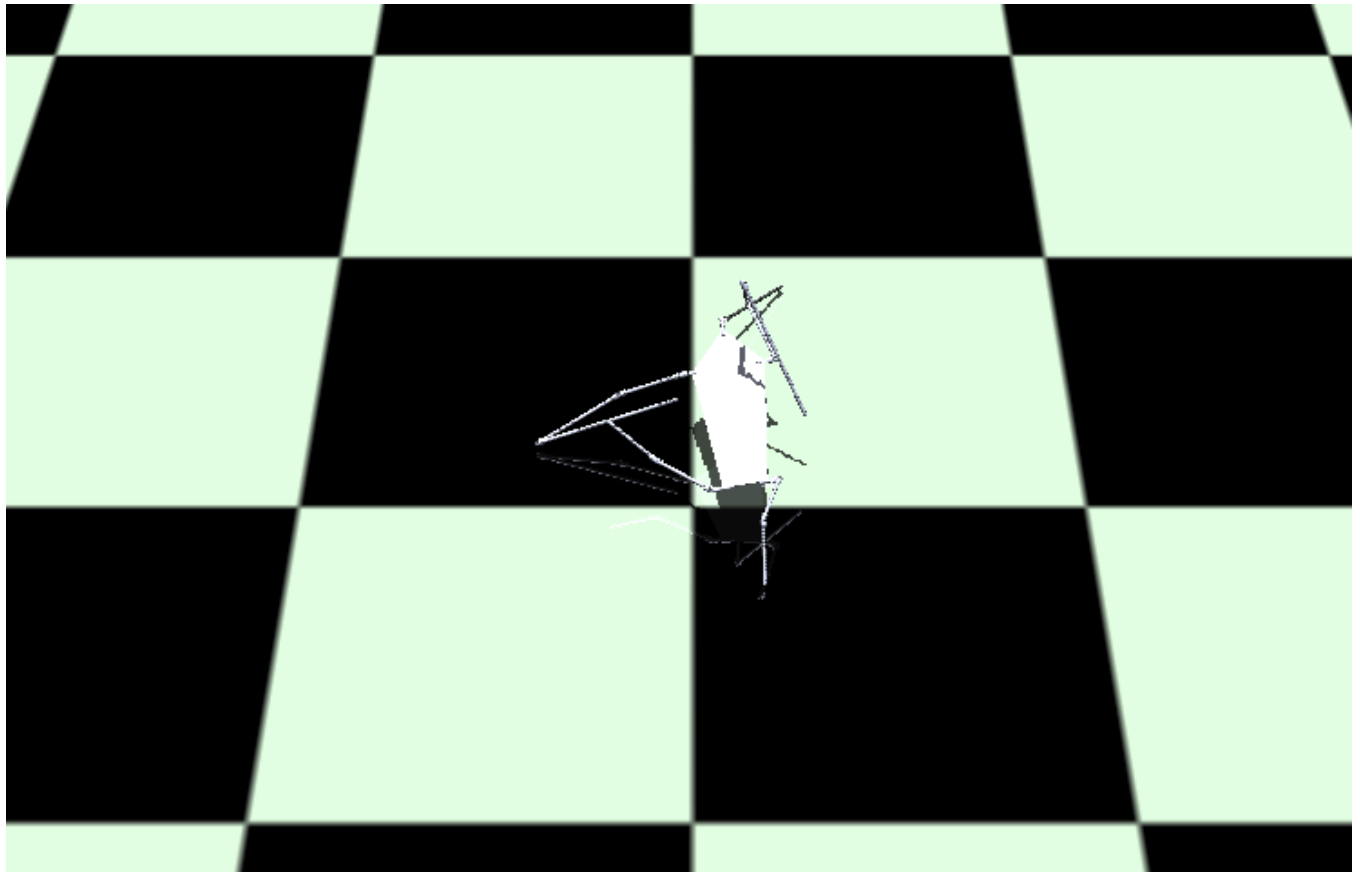
Got more familiar with toolchain

Train!

..very slow training, no noticeable progress. Kinda just flopping around



Minimal training: break dancing



Training: Cloud (GPU)

GPU parallelization enables vastly faster training

Required library swaps because a lot of RL is still on CPU

Developed scripts to interpret policy and training (save all training data)

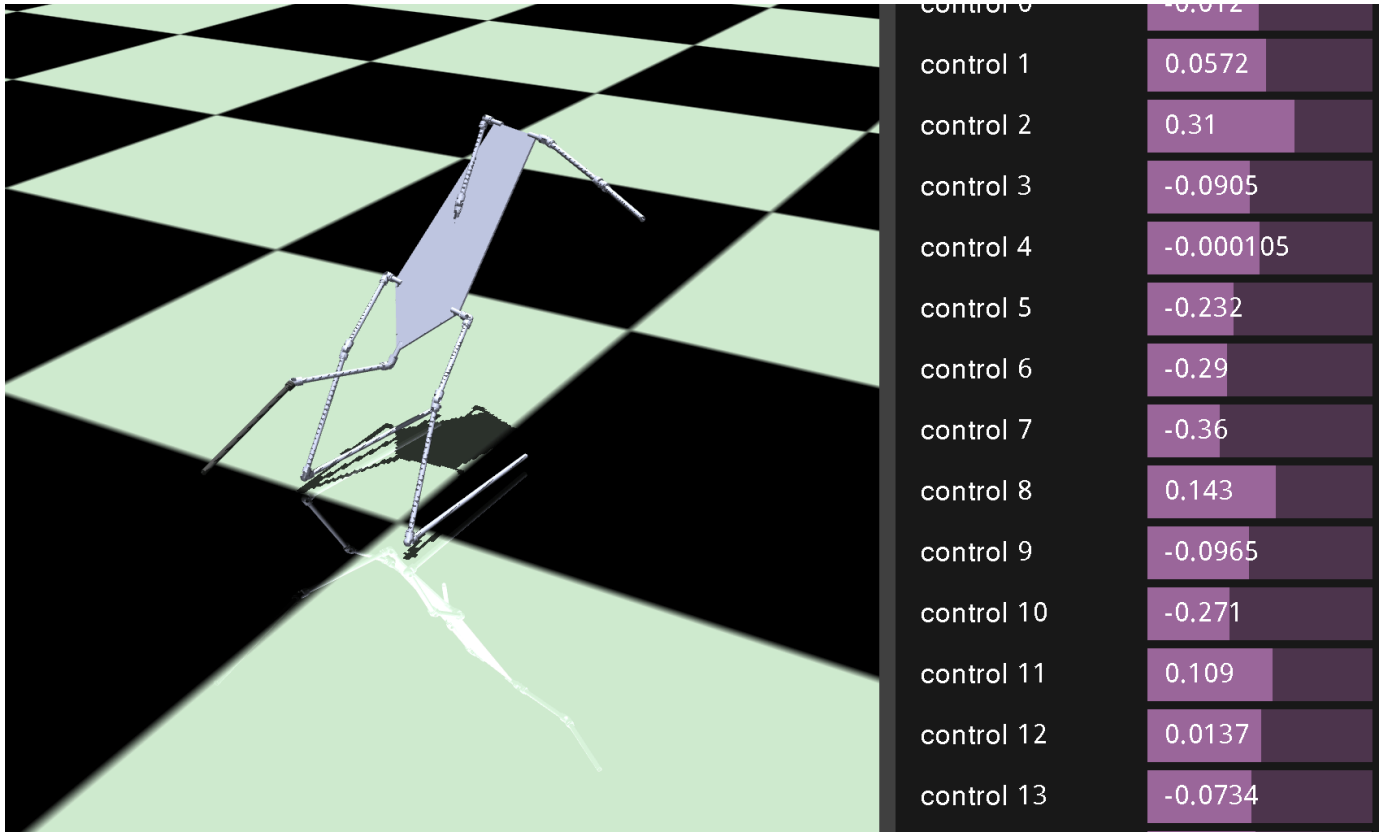
After WAY more training than on CPU, still no progress

Adjusted hyperparams:

- Reward weights
- Reward types
- Restricted some joint motion (wrist)
- Weakened actuators

```
2: brax PPO, converted humanoid env, left settings on default

3:  reward weight changes:
    ctrl_cost_weight=0.5 -> 0.2,
    forward_reward_weight=1.0 -> 2.0,
    vertical_reward_weight=0.5 -> 1.0

4: 30mil timesteps

5: changed vel_x to pos_x in reward, 5mil timesteps

6: revert to vel_x, restrict wrist joints, weaken all actuators 5x (100->20), better starting pose

7: add reward_lowvel: -1/vel (encourage movement)

8: reward_lowvel_weight 1.0 -> 0.1

9: reward_lowvel_weight 0.1 -> 0.01 (still has some effect because it blows up as velocity goes to 0)
```

Some training: Yoga!!



Training analysis

Training analysis

https://synapsomorphy.com/filehost/train2.html
https://synapsomorphy.com/filehost/train6.html
https://synapsomorphy.com/filehost/train7.html

Next steps

Continue tweaking reward

Continue restricting joint motion

More timesteps of training, especially if reward curves are promising

If walking soon: Evaluate different policies (Current: PPO)

Change mesh (potential for large speedup)