

实验流程

- 配置环境

git clone https://github.com/unitreerobotics/unitree_rl_gym.git

- 激活环境进入目录

```
conda activate rl-g1
cd unitree-
```

- 使用默认参数训练

执行以下命令开始训练（默认 10000 次迭代）：

```
python legged_gym/scripts/train.py --task=g1 --experiment_name=g1_default --
run_name= run1 --headless
```

- 查看训练指标

启动 TensorBoard：

```
tensorboard --logdir logs/g1_default
```

- 演示训练效果 Isaac gym:

```
play python legged_gym/scripts/play.py --task=g1 --
experiment_name=g1_default --run_name=run1
```

- 导出策略网络

导出路径自动保存为：

```
logs/g1_default/exported/policies/policy_lstm_1.pt
```

- Sim2Sim 验证（部署到 Mujoco）

配置模型路径：

编辑 deploy/deploy_mujoco/configs/g1.yaml 中：

```
policy_path: logs/g1_default/exported/policies/policy_lstm_1.pt
```

启动模拟：

```
python deploy/deploy_mujoco/deploy_mujoco.py g1.yaml
```

- 结合仿真结果和训练指标分析

- 默认参数 10000 次训练已经能达到稳定行走效果

尝试训练一个新的模型 **g1_flex** 能否稳定行走的情况下更灵活，接近人类

- **修改训练 reward 参数**

修改 G1 的 reward 配置（如 g1_config.py）中的 scales 项

- **使用更新后的参数训练**

执行以下命令开始训练（节省时间设置 5000 次迭代）：

```
python legged_gym/scripts/train.py --task=g1 --experiment_name=g1_flex--  
run_name= g1_flex -headless -max_iteration=5000
```

- **对比两次训练参数指标**

创建对比文件夹

```
mkdir logs_compare
```

```
ln -s ../logs/g1_default logs_compare/default
```

```
ln -s ../logs/g1_flex logs_compare/flex 启动 TensorBoard：
```

查看对比图

```
tensorboard --logdir logs_compare
```

- **演示新的训练效果 Isaac gym:**

```
play python legged_gym/scripts/play.py --task=g1 --  
experiment_name=g1_flex --run_name= g1_flex
```

- **导出新的策略网络**

导出路径自动保存为：

```
logs/g1_flex/exported/policies/policy_lstm_1.pt
```

- **Sim2Sim 验证（部署到 Mujoco）**

配置模型路径：

编辑 deploy/deploy_mujoco/configs/g1.yaml 中：

```
policy_path: logs/g1_flex/exported/policies/policy_lstm_1.pt
```

启动模拟：

```
python deploy/deploy_mujoco/deploy_mujoco.py g1.yaml
```

- **对比分析实验结果，录制视频**

Discussion

Tensorboard 图表见附录

默认参数模型 g1_default

参数选择

```
# default
class scales( LeggedRobotCfg.rewards.scales ):
    tracking_lin_vel = 1.0
    tracking_ang_vel = 0.5
    lin_vel_z = -2.0
    ang_vel_xy = -0.05
    orientation = -1.0
    base_height = -10.0
    dof_acc = -2.5e-7
    dof_vel = -1e-3
    feet_air_time = 0.0
    collision = 0.0
    action_rate = -0.01
    dof_pos_limits = -5.0
    alive = 0.15
    hip_pos = -1.0
    contact_no_vel = -0.2
    feet_swing_height = -20.0
    contact = 0.18
```

结果分析

共进行了 10000 次迭代，从 TensorBoard 曲线来看，策略在第 4000 步左右开始收敛，reward 趋势稳定，loss 下降良好。rev_contact 稳步提升至 0.35，rev_alive 保持在 0.15，而 rev_action_rate、dof_acc 等指标持续下降，说明控制逐渐平稳。在 4000-8000 步间出现短暂波动，是策略探索更自然步态过程中的正常现象，加之未使用学习率衰减，也可能加剧了 loss 的震荡。整体来看，策略最终稳定，表现良好，并且在 Isaac Gym 和 Mujoco 两个平台上均能实现稳定的行走。

但是感觉步态比较僵硬，是否能调整一些参数比如放宽一些限制项，减小惩罚，实现更自然的步态或者是更快收敛。

对训练参数分析：

参数	理解
tracking_lin_vel	奖励线速度与目标一致，鼓励向前 稳定 运动。
tracking_ang_vel	奖励角速度与目标一致，提升 转向 响应能力。
lin_vel_z	惩罚 竖直 方向速度，避免机器人跳起或 抖动 。
ang_vel_xy	惩罚 X/Y 平面角速度，防止 倾斜 翻倒。
orientation	惩罚身体姿态偏差，鼓励保持平衡。
base_height	强烈 惩罚高度偏离，限制重心控制。
dof_acc	惩罚 关节 加速度过大，提升 动作 平滑度。
dof_vel	惩罚 关节 速度过高，减少 震荡 与损耗。
feet_air_time	未启用，用于 奖励 步态节奏协调。
collision	未启用，用于 惩罚 身体与障碍物碰撞。
action_rate	惩罚 动作 变化剧烈，鼓励输出 连续 平滑。
dof_pos_limits	惩罚接近 关节 极限的行为， 保护 结构。
alive	奖励机器人保持站立与存活状态。
hip_pos	惩罚 髋 关节位置异常，提升步态 对称 性。
contact_no_vel	惩罚脚接触地面但无速度。
feet_swing_height	惩罚 摆腿 高度异常，限制脚步自然性。
contact	奖励良好的脚步接触，提高步态 稳定 性。

更新参数模型 g1_flex

参数选择

```
# more flexible
class scales( LeggedRobotCfg.rewards.scales ):
    tracking_lin_vel = 1.0
    tracking_ang_vel = 0.5
    lin_vel_z = -2.0
    ang_vel_xy = -0.05
    orientation = -1.0
    base_height = -3.0
    dof_acc = -2.5e-7
    dof_vel = -1e-3
    feet_air_time = 0.0
    collision = 0.0
    action_rate = -0.01
    dof_pos_limits = -5.0
    alive = 0.15
    hip_pos = -1.0
    contact_no_vel = -0.2
    feet_swing_height = -5.0
    contact = 0.3
```

参数	default	flex	motivation
base_height	-10.0	-3.0	减弱高度偏差惩罚，避免过度束缚重心
feet_swing_height	-20.0	-5.0	放宽摆腿自由
contact	0.18	0.3	提高接触奖励

结果分析

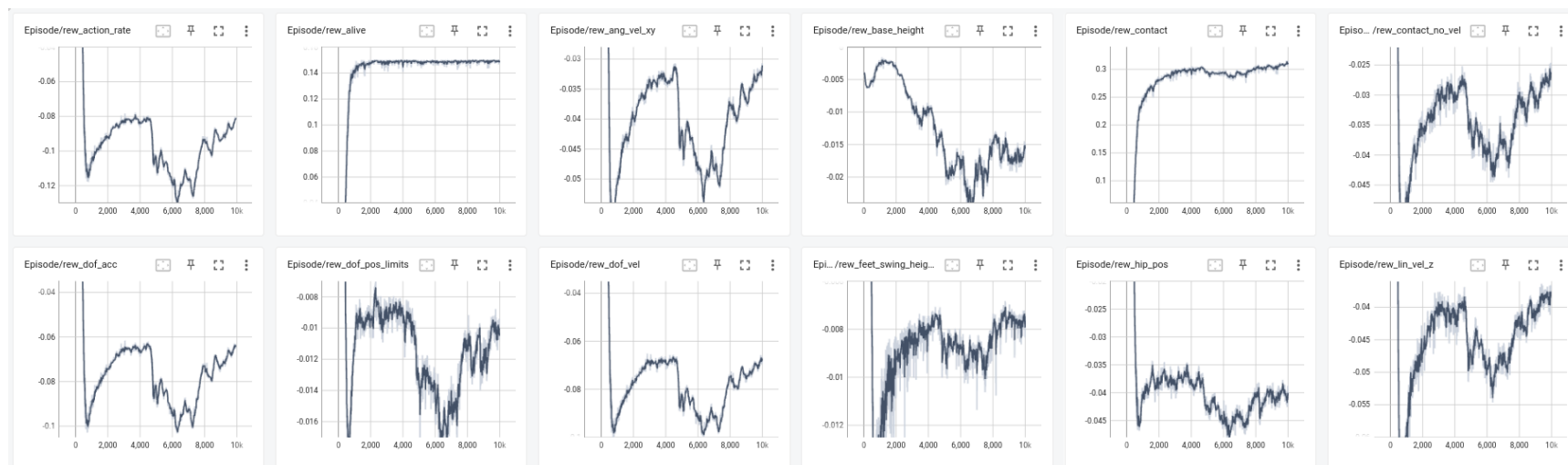
从 reward 曲线来看， g1_flex 的 rev_contact 上升更快，步态接触更充分；rev_contact_no_vel 和 rev_action_rate 下降明显动作更平滑；rev_dof_acc、rev_dof_vel、rev_hip_pos 等控制指标的波动幅度更小，表明整体控制输出更自然，步态更加协调。而 rev_feet_swing_height 的改善也显示出策略摆腿动作更灵活，不再受限。loss 曲线部分 g1_flex 的 Loss/value_function 波动范围较大，但整体保持下降趋势，而 Loss/surrogate 在收敛平台后仍有上升空间，说明策略仍在有效探索，但未造成发散。两者 learning rate 都保持在稳定范围内，未出现训练中断或早停。

但是仿真结果没有肉眼可见差别，两个模型都能在两个环境中稳定行走，可能 g1_flex 会多一些轻微抖动。

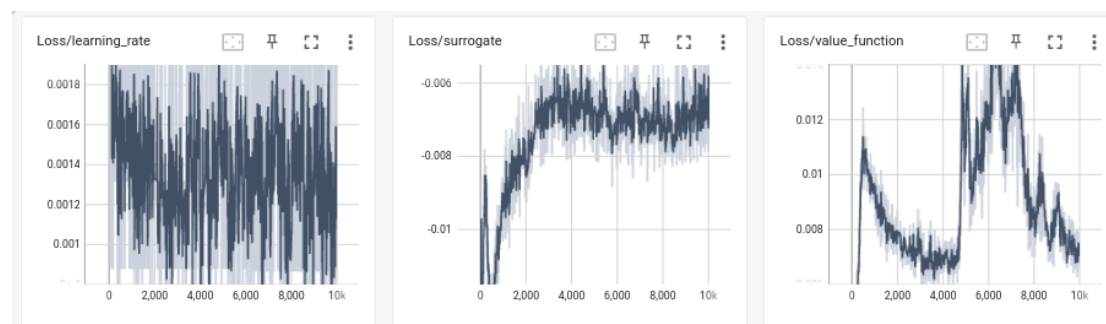
附录

默认参数模型 gl_default 指标

Reward

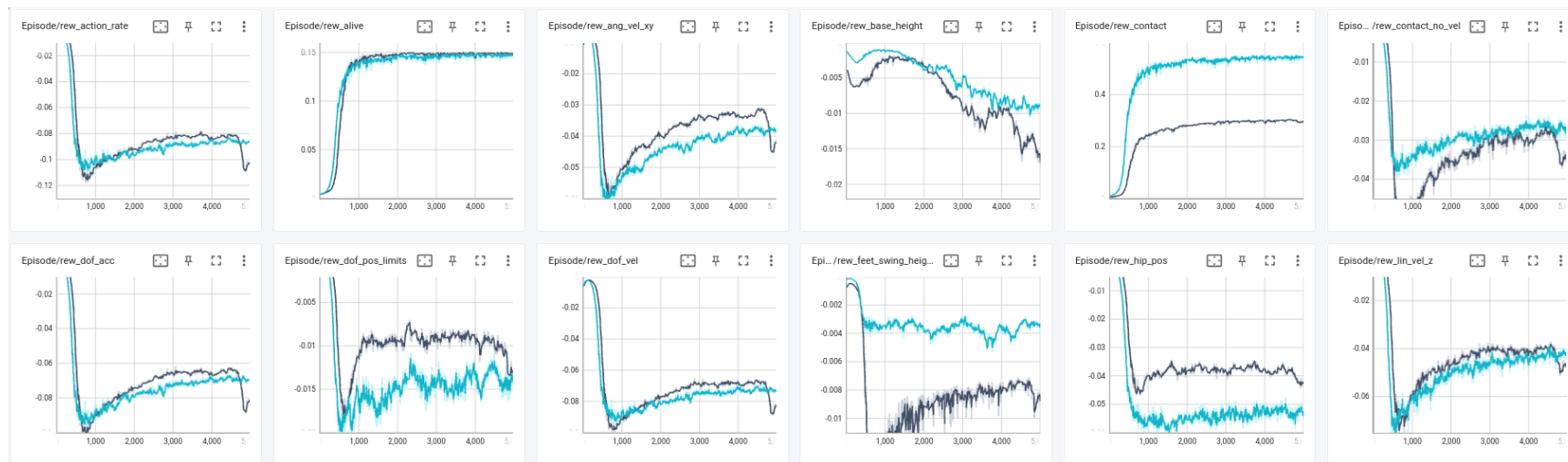


Loss



默认参数模型 `gl_default` (深蓝) 与更新参数模型 `gl_flex` (浅蓝) 指标对比 (5000 次迭代)

Reward



Loss

