请注意：Version 4.0.0 将是 OmniIsaacGymEnvs 的最后一个版本。之后，OmniIsaacGymEnvs 将合并到 IsaacLab （https://github.com/isaac-sim/IsaacLab）。所有未来更新都将作为 IsaacLab 仓库的一部分提供。

有关迁移到 IsaacLab 的教程，请访问：

<https://isaac-sim.github.io/IsaacLab/source/migration/migrating_from_omniisaacgymenvs.html>

**系统要求**

建议至少有 32GB内存和至少12GB显存的GPU。

详细的系统要求请访问 Isaac Sim System Requirements这个文档，文档链接如下：

https://docs.omniverse.nvidia.com/isaacsim/latest/installation/requirements.html#system-requirements

**安装**

本仓库中的示例依赖于最新 Isaac Sim 版本的功能。请确保将任何现有的 Isaac Sim 构建更新到 4.0.0，以确保示例按预期工作。

安装完成后，这个仓库可以作为一个 python 模块 omniisaacgymenvs 使用，需配合 Isaac Sim 提供的 python 可执行文件。

要安装 omniisaacgymenvs，首先克隆这个仓库：

git clone https://github.com/NVIDIA-Omniverse/OmniIsaacGymEnvs.git

克隆完成后，找到 Isaac Sim 中的 python 可执行文件。默认情况下应该是 python.sh。我们将这个路径称为 PYTHON\_PATH。

要在终端中设置链接到 python 可执行文件的 PYTHON\_PATH 变量，我们可以运行类似以下的命令。确保更新为你的本地路径。

For Linux:

alias PYTHON\_PATH=~/.local/share/ov/pkg/isaac\_sim-\*/python.sh

For Windows:

doskey PYTHON\_PATH=C:\Users\user\AppData\Local\ov\pkg\isaac\_sim-\*\python.bat $\*

For IsaacSim Docker:

alias PYTHON\_PATH=/isaac-sim/python.sh

为 PYTHON\_PATH 安装 omniisaacgymenvs 作为 python 模块：

PYTHON\_PATH -m pip install -e .

在初始安装过程中可能会出现以下错误。这个错误是无害的，可以忽略。

ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the following dependency conflicts.

**运行示例**

注意：所有命令都应该在 OmniIsaacGymEnvs/omniisaacgymenvs 目录下执行。

要训练你的第一个策略，运行：

PYTHON\_PATH scripts/rlgames\_train.py task=Cartpole

此时应该会启动一个 Isaac Sim 应用窗口。一旦 Isaac Sim 初始化完成，Cartpole 场景将被构建，模拟将自动开始运行。训练完成后，进程将终止。

注意，默认情况下，我们显示带有渲染的 Viewport 窗口，这会降低训练速度。你可以选择在训练期间关闭 Viewport 窗口以获得更好的性能。可以通过从顶部菜单栏选择 Window > Viewport 重新启用 Viewport 窗口。

要获得最大性能，可以在 headless 模式下启动训练：

PYTHON\_PATH scripts/rlgames\_train.py task=Ant headless=True

关于模拟启动时间的说明

某些示例可能需要几分钟才能加载，因为启动时间会根据环境数量进行缩放。启动时间将在未来版本中持续优化。

扩展工作流程

扩展工作流程提供了一个简单的用户界面来创建和启动 RL 任务。要为扩展工作流程启动 Isaac Sim，运行：

./<isaac\_sim\_root>/isaac-sim.gym.sh --ext-folder </parent/directory/to/OIGE>

注意：isaac\_sim\_root 应该与 python.sh 位于同一目录中。

可以通过导航顶部菜单栏从 Isaac Examples > RL Examples 激活 UI 窗口。有关扩展工作流程的更多详细信息，请参阅文档。

文档链接：

<https://github.com/isaac-sim/OmniIsaacGymEnvs/blob/main/docs/framework/extension_workflow.md>

如果你遇到 Isaac Examples 菜单缺失的问题，请按如下方式修改 OmniIsaacGymEnvs/omniisaacgymenvs/\_\_init\_\_.py 以显示潜在的错误：

import traceback

try:

from .extension import RLExtension, get\_instance

except Exception:

print(traceback.format\_exc())

如果出现 hydra 的 ModuleNotFoundError，请检查你的 C:\Users\user\AppData\Roaming\Python\Python310 目录，删除可能包含 hydra 包的任何 site-packages 目录。然后，重新运行 OmniIsaacGymEnvs 的 `pip install -e .` 命令。

**加载训练模型 // 检查点（checkpoints）**

检查点保存在 runs/EXPERIMENT\_NAME/nn 文件夹中，其中 EXPERIMENT\_NAME 默认为任务名称，但也可以通过 experiment 参数覆盖。

要加载训练好的检查点并继续训练，使用 checkpoint 参数：

PYTHON\_PATH scripts/rlgames\_train.py task=Ant checkpoint=runs/Ant/nn/Ant.pth

要加载训练好的检查点并只执行推理（不训练），传递 test=True 作为参数，以及检查点名称。为了避免渲染开销，你可能还想使用 num\_envs=64 运行较少的环境：

PYTHON\_PATH scripts/rlgames\_train.py task=Ant checkpoint=runs/Ant/nn/Ant.pth test=True num\_envs=64

注意，如果检查点名称中有特殊字符如 [ 或 =，你需要转义它们并在字符串周围加上引号。例如：

checkpoint="runs/Ant/nn/last\_Antep\=501rew\[5981.31\].pth

我们在 Nucleus 服务器的 Assets/Isaac/4.0/Isaac/Samples/OmniIsaacGymEnvs/Checkpoints 下提供预训练的检查点。运行以下命令以启动预训练检查点的推理：

Localhost（要设置 localhost，请参考 Isaac Sim 安装指南）：

安装指南链接：

https://docs.omniverse.nvidia.com/isaacsim/latest/installation/install\_workstation.html

PYTHON\_PATH scripts/rlgames\_train.py task=Ant \

checkpoint=omniverse://localhost/NVIDIA/Assets/Isaac/4.0/Isaac/Samples/OmniIsaacGymEnvs/Checkpoints/ant.pth test=True num\_envs=64

Production服务器：

PYTHON\_PATH scripts/rlgames\_train.py task=Ant \

checkpoint=http://omniverse-content-production.s3-us-west-2.amazonaws.com/Assets/Isaac/4.0/Isaac/Samples/OmniIsaacGymEnvs/Checkpoints/ant.pth test=True num\_envs=64

首次使用预训练检查点运行时，我们会自动将检查点文件下载到 omniisaacgymenvs/checkpoints。对于后续运行，我们将重复使用已下载的文件，并且不会覆盖 checkpoints 文件夹中具有相同名称的现有检查点。