Accelerated Robot Training through Simulation with ROS and Gazebo

Risto Kojcev, Alejandro Hernández Cordero, Asier Bilbao Calvo, Irati Zamalloa Ugarte, Yue Leire Erro Nuin, Víctor Mayoral Vilches
We investigate the state of the art deep-learning methods, and apply them to modular robots.
STRATEGIES FOR BUILDING ROBOTS

- **TRADITIONAL**
  1. Buy component
  2. Integration of components
  3. Robot assembly
  4. Programming the robot (to perform a task)
  5. Test and adapt
  6. Deploy: ship it!

- **MODULAR**
  1. Buy module
  2. Robot assembly (plug modules)
  3. Programming the robot (to perform a task)
  4. Test and adapt
  5. Deploy: ship it!

- **MODULAR and RECONFIGURABLE**
  1. Buy module
  2. Define task
  3. Robot assembly (plug modules)
  4. Automatic training
     - Train and re-train robot
     - Test and adapt
  5. Deploy: ship it!

Source: Erle Robotics
Two unlinked worlds

Real Experiment Setup

Robot view (used for Robot Inference)
Query Object

3rd Person view (used for Visualization)
Camera viewpoint changes in different experiments

Sadeghi, Fereshteh, et al. CVPR 2018

ROS

OpenAI

FINGER PIVOTING
SLIDING
FINGER GAITING
Two unlinked worlds

Real Experiment Setup

Robot view (used for Robot Inference)

Query Object

Camera viewpoint changes in different experiments

3rd Person view (used for Visualization)

ROS

OpenAI

gym_gazebo

FINGER PIVOTING
SLIDING
FINGER GAITING
GYM_GAZEBO: a Link between AI and Robotic Methods

Simulator

RL Algorithm

Robot toolbox and libraries

environment API

GYM_GAZEBO is Open Source. Try it!
https://github.com/erlerobot/gym-gazebo

Github stats

<table>
<thead>
<tr>
<th>Watching</th>
<th>Stars</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>395</td>
</tr>
</tbody>
</table>
GYM_GAZEBO: Community Projects

Nicola Ristè
https://youtu.be/h84IkUCze44

Institut de Robòtica i Informàtica
An “end to end” solution for modular robots
Reinforcement Learning requires long training time...

...this is me waiting for results!
ROBOT_GYM: accelerating training
ROBOT_GYM: workflow

Orchestrate Workers

Initialize global policy ($\pi$)

Worker 1 generates rollouts
Worker 2 generates rollouts
Worker 3 generates rollouts
... 

Update policy ($\pi_\theta$)

Merge policy ($\pi_\theta$)
ROBOT_GYM: workflow

Orchestrate Workers

Initialize global policy ($\pi_\theta$)

Worker 1 generates rollouts
Worker 2 generates rollouts
Worker 3 generates rollouts
...

Update policy ($\pi_\theta$)

Merge policy ($\pi_\theta$)

Kubernetes Cloud Network

VPN bridge server

container hub

Local workers

Internet

replica container
**ROBOT_GYM**: workflow

1. Initialize global policy ($\pi_\theta$)
2. Orchestrate Workers
3. Worker 1 generates rollouts
4. Worker 2 generates rollouts
5. Worker 3 generates rollouts
6. …
7. Merge policy ($\pi_\theta$)
8. Update policy ($\pi_\theta$)

**Worker**
- Environment API
- RL Algorithm
- Robot toolbox and libraries
- Simulator

**Container**
Results 3DoF: SCARA

<table>
<thead>
<tr>
<th>SCARA</th>
<th>Accuracy (mm)</th>
<th>Repeatability (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 Worker</td>
<td>8 Workers</td>
</tr>
<tr>
<td></td>
<td>1 Worker</td>
<td>8 Worker</td>
</tr>
<tr>
<td>Simulated</td>
<td>2.80</td>
<td>1.37</td>
</tr>
<tr>
<td>Real</td>
<td>26.14</td>
<td>12.89</td>
</tr>
</tbody>
</table>
Results 6DoF: MARA

![Graphs showing iterations vs mean reward and time vs mean reward for different worker counts. Each graph has lines indicating 1 worker, 2 workers, 4 workers, 8 workers, and 16 workers.]

<table>
<thead>
<tr>
<th>MARA</th>
<th>Accuracy (mm)</th>
<th>Repeatability (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 Worker</td>
<td>8 Workers</td>
</tr>
<tr>
<td>Simulated</td>
<td>0.05</td>
<td>0.08</td>
</tr>
<tr>
<td></td>
<td>1 Worker</td>
<td>8 Worker</td>
</tr>
<tr>
<td></td>
<td>0.02</td>
<td>0.02</td>
</tr>
</tbody>
</table>
a framework to accelerate robot-training using Gazebo and ROS in the cloud

- 12 replicas in parallel
- 1603 hours of cloud computing: 0.134 €/hour per instance, or 1.606 €/hour for all the replicas running at the same time

evaluated 2 different modular robots

training-time can be reduced by more than 33% while maintaining similar levels of accuracy

The total costs for all evaluations was 214 Euros
Perspectives and Future Work

Robot trajectories under different levels of noise when reaching a target

- std 0.05 rad = 2.86 deg per joint
- std 0.1 rad = 5.73 deg per joint
Published Work

- **robot_gym**: accelerated robot training through simulation in the cloud with ROS and Gazebo

- **Towards self-adaptable robots: from programming to training machines**

- **Hierarchical Learning for Modular Robots**

- **Evaluation of Deep Reinforcement Learning Methods for Modular Robots**

- **gym_gazebo**: a toolkit for reinforcement learning using ROS and Gazebo