

Introduction to Swarm Robotics in ARGoS

ARGoS

ARGoS is a multi-robot simulator developed at IRIDIA. It can simulate large-scale swarms of robots of any kind efficiently. You will use ARGoS to work on the practical exercises of this course, as well as on the final course project.

Getting ARGoS

In this course, we will not be working with the latest version of ARGoS; instead, we will work with version **3.0.0-beta57**. Therefore, to install ARGoS on your computer, you will have to compile it directly from the sources. To do so, please follow the instructions provided below carefully! We will only be able to support installation of ARGoS under **Ubuntu 20.04 LTS**. If you do not have an Ubuntu 20.04 LTS system readily available, we recommend that you use a virtual machine (VirtualBox is a popular solution; see <https://ubuntu.com/tutorials/how-to-run-ubuntu-desktop-on-a-virtual-machine-using-virtualbox#1-overview>). Alternatively, you can dual boot Ubuntu with Windows (for example, see <https://itsfoss.com/install-ubuntu-1404-dual-boot-mode-windows-8-81-uefi/>). Note, however, that the process to get dual booting is much more involved and you risk losing your data, which is why we recommend that you use a virtual machine.

Install ARGoS from sources

The instructions assume that you are working on an installation of Ubuntu 20.04 LTS. They will guide you through a local installation of ARGoS. First, run the following command to install all prerequisites:

```
sudo apt install build-essential cmake git libfreeimage-dev libfreeimageplus-dev \
    qt5-default freeglut3-dev libxi-dev libxmu-dev liblua5.3-dev lua5.3 doxygen \
    graphviz libgraphviz-dev asciidoc
```

You may be asked to restart your machine for the installed packages to take effect. Having installed all prerequisites, you will decide where you want to install ARGoS. Usually, you would install it in your `$HOME` directory. Run the following command in a terminal to set the installation path:

```
$ export ARGOS_INSTALL_PATH=$HOME
```

Now run the commands below in the same terminal to download and install ARGoS from the sources.

```
$ cd $ARGOS_INSTALL_PATH
$ git clone https://github.com/ilpincy/argos3.git argos3
$ cd argos3
$ git checkout 3.0.0-beta57
$ mkdir build && cd build
$ cmake -DCMAKE_INSTALL_PREFIX=$ARGOS_INSTALL_PATH/argos3-dist \
    -DCMAKE_BUILD_TYPE=Release -DARGOS_INSTALL_LDSOCONF=OFF \
    -DARGOS_DOCUMENTATION=OFF ../src
$ make install
```

Back in your `$HOME` directory, create a file called `setup-argos.sh` and add the following lines to the file.

```
export ARGOS_INSTALL_PATH=$HOME
export ARGOS_PLUGIN_PATH=$ARGOS_INSTALL_PATH/argos3-dist/lib/argos3
export PKG_CONFIG_PATH=$ARGOS_INSTALL_PATH/argos3-dist/lib/pkgconfig
export LD_LIBRARY_PATH=$ARGOS_PLUGIN_PATH:$LD_LIBRARY_PATH
export PATH=$ARGOS_INSTALL_PATH/argos3-dist/bin/:$PATH
```

You will need to source this file every time you open a new terminal. If you do not want to do that manually, you can add the following line to the end of your `.bashrc` file:

```
source setup-argos.sh
```

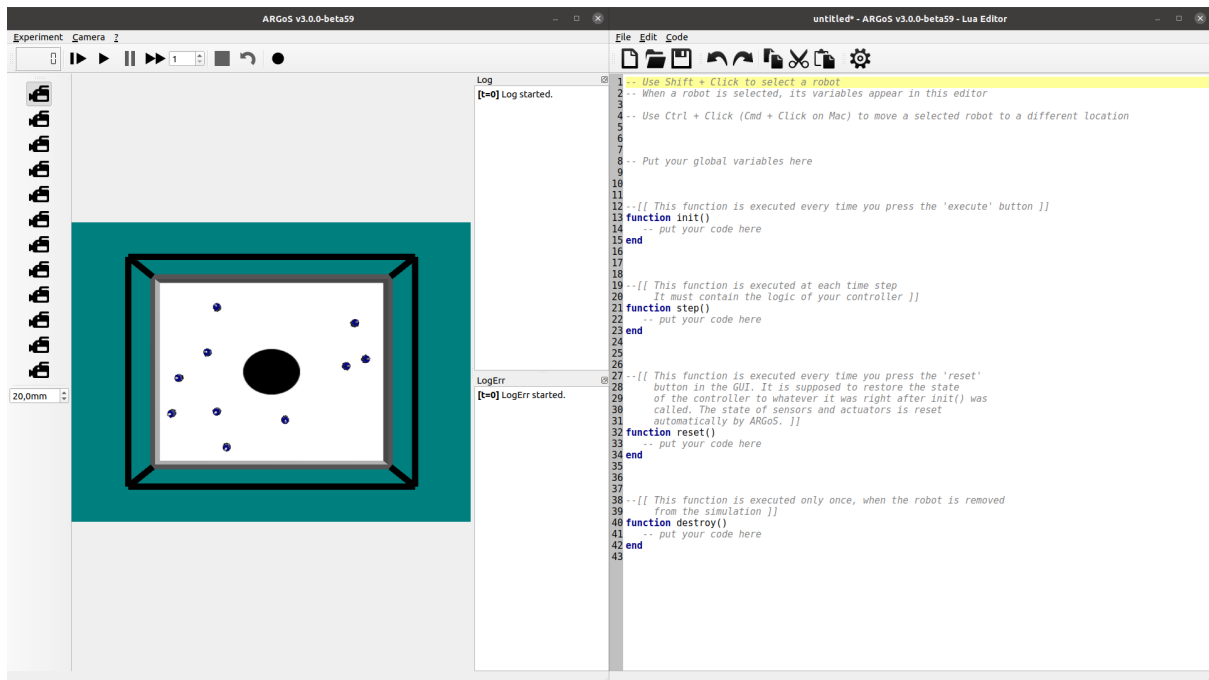


Figure 1: The simulation window (left) and the Lua editor (right) of ARGoS.

Running ARGoS

To run ARGoS, you need a file with the extension `.argos` that configures the program for a specific experiment. We will provide you each week with the `.argos` file for the practical exercise of that week. For the first session, we provide you with the `sandbox.argos` file (see the accompanying `sandbox.pdf` for more information). After installing ARGoS, you can run the simulator by opening a terminal and entering the following command:

```
$ argos3 -c sandbox.argos
```

This will open the two windows of ARGoS: a simulation window and a Lua code editor.

ARGoS interface

The code editor opens a template of a Lua script with a few empty functions where you will develop your controller; each function has comments that explain what it does. In this course you will work only with homogeneous robot swarms—that is, all robots in the swarm are identical and independently execute the same control software. The control software that you develop in the editor is ported to all robots in the swarm.

To execute your script on the robots, follow these steps. First, click on the “Save the current file” button to store the script in a file with the extension `.lua`. Then, upload the control software to the robots by clicking on the “Execute code” button. Finally, start running the experiment by clicking on the “Play experiment” button. Note that the template script is empty by default. It is therefore normal that the robots do not do anything when the script is executed as it is.

If you wish to open a previously saved script, click on the “Open a file” button. The interface will only look for files with the extension `.lua`. Therefore, make sure that you store your controllers with that extension.