



UgCS DDC

Running SITL version 3.0

User Guide

Table of Contents

Overview 3

Preparing environment 3

Configuring swarm size 3

Configuring drone locations 3

Overview

To run a swarm dance in simulation, Ubuntu 16.04 operating system needs to be used. It is also possible to run it in a virtual environment.

Preparing environment

If you have a 64-bit OS, use sitl64.zip:

1. Unpack the provided archive into your working directory:
 - a. `unzip sitl(64).zip -d SITL`
2. SITL is a directory name where sitl files are to be located.
3. Enter the directory and add execute permissions:
 - a. `cd SITL`
 - b. `chmod +x swarm.bash Tools/autotest/run_in_terminal_window.sh build/sitl/bin/arducopter`
4. Install the following required packages or make sure they are already installed:
 - a. `sudo apt-get install python-pip`
 - b. `sudo pip install future pymavlink MAVProxy`
 - c. `sudo apt-get install gnome-terminal`

Configuring swarm size

Open **swarm.bash** in any text editor and find the following lines:

Range of system IDs to use.

SYSID_FIRST=1

SYSID_LAST=20

The last two lines specify the range of drone IDs and the number of drones. Here, 20 SITL instances will be initialized in a swarm.

Configuring drone locations

Open locations.py in any text editor and find the following lines:

Y_count = 7

X_count = 5

Distance_between_drones = 3 # meters

CNT_LAT = 56.973501 # Latitude of Drone formation center

CNT_LON = 24.071459 # Longitude of Drone formation center

Heading = 0

Altitude = 1

This file specifies initial location for each drone in a swarm to prevent their congestion in a single point.

When you run this script, the following output will be displayed:

S1=56.9734471011,24.0713106617,1,202

S2=56.9734471011,24.0713601078,1,78

S3=56.9734471011,24.0714095539,1,0

...

Insert this output in the text file Tools/autotest/locations.txt

Run the swarm

Execute

./swarm.bash

Wait until all SITL instances are initialized. Press any key in swarm.bash terminal window to terminate all the processes. The first run is for setting up SITL environments for each drone and not for swarm operations.

As a result, a new ddc-cache directory will be created with subdirectories for each drone ID (1, 2, 3, 4, etc.).

Logs, parameters and trajectory files for each drone will be kept in the corresponding subdirectory.

Run the swarm again.

A script will start instances, with each instance being available through a TCP port according to the formula:

$$\text{Port} = 5760 + I \cdot 10 + 2$$

For example:

5772 for the drone with ID = 1

5782 for the drone with ID = 2

5792 for the drone with ID = 3

5802 for the drone with ID = 4

To connect DDC Client to instances, VSM should be configured respectively:

You may use the vsm.py script to automate the configuration process. If SITL instances are running on the host 192.168.1.39 and there are 5 instances in total, then execute the following line:

python vsm.py --host 192.168.1.39 --num 5

Copy the output and insert it in your **vsm-arducopter.config** file.

N 1

connection.tcp_out.1.port = 5772

connection.tcp_out.1.address = 192.168.1.39

N 2

connection.tcp_out.2.port = 5782

connection.tcp_out.2.address = 192.168.1.39

N 3

connection.tcp_out.3.port = 5792

connection.tcp_out.3.address = 192.168.1.39

N 4

connection.tcp_out.4.port = 5802

connection.tcp_out.4.address = 192.168.1.39