# RoboSim5G - User Documentation

## Version: 1.0

## Date: March 2026

## Project: RoboSim5G - 5G Network Simulation for Robotics Research

## Organization: phine.tech GmbH

## Table of Contents


1. What is RoboSim5G?
2. Who is it for? / Who does it help and how?
3. How does it work? - User Perspective
4. Installation
5. Quick Start Guide
6. Detailed Usage Instructions
7. Configuring Your Simulation
8. Troubleshooting
9. Frequently Asked Questions
10. Getting Help


---

# 1. What is RoboSim5G?

RoboSim5G is an **easy-to-use simulation platform** that allows you to test robotic applications with realistic 5G network connectivity. Instead of needing expensive 5G hardware and infrastructure, you can simulate complete 5G networks on your computer using standard simulation tools.

**In simple terms:**

- Run robot simulations in Gazebo (a popular robotics simulator)
- Experience realistic 5G network behavior (connection, disconnection, delays)
- Test how your robots perform under different network conditions

- No need for physical 5G equipment or deep networking knowledge

  **What you can simulate:**
- 5G base stations (gNodeB/gNB) providing wireless coverage
- Robots with 5G modems (User Equipment/UE) connecting to the network
- Remote control of robots over 5G
- Network failures and recovery scenarios
- Multiple robots sharing 5G infrastructure

---

# 2. Who is it for? / Who does it help and how?

## Target Users

**Robotics Researchers & Students:**

- Test autonomous robots that depend on wireless connectivity
- Study how network outages affect robot behavior
- Publish research on 5G-enabled robotics applications
- Learn about 5G integration without complex setup

  **Robot Application Developers:**
- Validate cloud-connected robot applications
- Test remote teleoperation scenarios
- Develop multi-robot coordination systems
- Prototype before deploying to real 5G networks

  **Educators:**
- Teach 5G networking concepts through practical examples
- Demonstrate end-to-end 5G system integration
- Provide hands-on learning without expensive equipment

  **Industry Professionals:**
- Evaluate 5G-enabled robot solutions before production
- Test edge computing scenarios
- Validate network requirements for deployments

  For detailed documentation, refer to:
- **Main Project README:** README.md
- **Core Network Comparison:** doc/5G_CORE_NETWORKS.md
- **Developer Documentation:** DEVELOPER_DOCUMENTATION.md

---

# 3. How does it work? - User Perspective

## System Overview

RoboSim5G creates a complete 5G ecosystem on your computer with Docker containers running the 5G core network, Gazebo simulating the physical environment, and ROS 2 handling robot control.

## Choosing the Right Mode

**Demo Mode** ( `demo/` folder):

- Gazebo runs natively on your computer
- Better performance and GPU access
- Requires Gazebo Ignition Fortress installed locally
  **Containerized Mode** ( `demo_containerized/` folder):
- Everything runs in containers including Gazebo
- Fully reproducible environment
- No local Gazebo installation needed

## Choosing a 5G Core Network

Three options supported:

- **OAI** - High performance, research-focused
- **free5GC** - Easy to use with web interface (recommended for beginners)
- **Open5GS** - Mature, production-ready
  📖 **For detailed comparison:** doc/5G_CORE_NETWORKS.md

---

# 4. Installation

## System Requirements

- **OS:** Ubuntu 22.04 (recommended)
- **RAM:** 16 GB recommended (8 GB minimum)
- **Disk:** 20 GB free space
- **CPU:** 4+ cores recommended

## Software Prerequisites

**For Demo Mode:**

```
sudo apt install ros-humble-desktop-full ros-humble-ros-gz
sudo apt install docker.io docker-compose-v2
sudo usermod -aG docker $USER  # Logout/login required after
```

**For Containerized Mode:**

```
sudo apt install docker.io docker-compose-v2
sudo usermod -aG docker $USER
```

### Download and Build

```
# Clone repository
git clone https://github.com/phinetech/RoboSim5G.git
cd RoboSim5G
# Build (choose your mode)
cd demo               # or demo_containerized
./build_images.sh     # Takes 10-30 minutes first time
```

---

## 5. Quick Start Guide

### Your First Simulation (Demo Mode with free5GC)

```
cd RoboSim5G/demo
# Launch simulation
source launch_robot_5G_free5gc.sh
# Enter password when prompted
```

**Wait for startup (~40 seconds, first run ~5 minutes):**

- Gazebo opens with robot and environment
- RViz opens with map interface

  **Control the robot:**

1. In RViz, click "Nav2 Goal" button
2. Click destination on map

3. Watch robot navigate!

   **Test 5G effects:**
4. In Gazebo, click three dots (⋮) in top right
5. Search "UE_Power_Plugin"
6. Click button to disable/enable 5G
7. Observe robot behavior changes

   **Stop simulation:**

```
./kill_all_free5gc.sh
```

### Using Other Core Networks

**OAI:**

```
source launch_robot_5G.sh
./kill_all.sh
```

**Open5GS:**

```
source launch_robot_5G_open5gs.sh
./kill_all_open5gs.sh
```

---

## 6. Detailed Usage Instructions

### Verifying Successful Startup

**Check Core Network:**

```
cd demo/free5gc_setup  # or oai_setup
docker compose ps  # All should show "running"
```

**Check gNB connected:**

```
docker logs oai-amf | grep "gNBs' Information"
```

**Check UE registered:**

```
docker logs oai-amf | grep "UEs' Information"
```

**Check 5G tunnel:**

```
docker exec -it ue_turtlebot ip addr show oaitun_ue1
```

## Controlling the Robot

**Send navigation goals via RViz:**

- Click "Nav2 Goal" button
- Click destination and drag for orientation

  **Simulate network failures:**
- Use gNB_Power_Plugin or UE_Power_Plugin in Gazebo
- Click to toggle connection on/off
- Observe robot behavior under network loss

## Monitoring

```
# View robot position
ros2 topic echo /robot_pose
# View gNB position
ros2 topic echo /pose_of_gNB1
# Check container logs
docker logs <container_name>
```

---

# 7. Configuring Your Simulation

## Changing Core Network Type

1. **Edit world file selection:**

```
# demo/gazebo_launch/src/ign_turtlebot/launch/gazebo_launch.launch.py
# Line ~68, change world file name:
world = "world_only_free5gc.sdf"  # or world_only.sdf, world_only_open5gs
```

2. **Rebuild:**

```
cd demo/gazebo_launch
colcon build
source install/setup.bash
```

3. **Use matching launch script**

## Avoiding Multi-User Conflicts

Edit launch script:

```
# Change ROS domain ID (0-101)
export ROS_DOMAIN_ID=42
# Change Gazebo partition name
export IGN_PARTITION=robosim_myname
```

## Adjusting Positions

Edit world file (e.g., `demo/gazebo_launch/src/ign_turtlebot/worlds/world_only_free5gc.sdf`):

**Robot position:**

```
<pose>0 0 0 0 0 0</pose>  <!-- X Y Z Roll Pitch Yaw -->
```

**gNB position:**

```
<pose>-6.8 1 7 0 0 0</pose>
```

---

# 8. Troubleshooting

## Common Issues

**Docker permission denied:**

```
sudo usermod -aG docker $USER
# Logout and login again
```

**First launch fails/timeout:**

- Normal - robot container is building (up to 5 minutes)
- Run launch script again after build completes

  **Gazebo opens but robot doesn't move:**
- Check UE registration: `docker logs oai-amf | grep UE`
- Verify RViz shows "Nav2: Ready"
- Check navigation stack: `docker exec -it ue_turtlebot ros2 node list`

  **Can't see Gazebo (containerized mode):**

```
xhost +local:docker
source launch_robot_5G_docker_free5gc.sh
```

**5G tunnel not created:**

- Check UE registration in AMF logs
- Verify IMSI/key/OPC match in world file and core config
- Check UE logs: `docker logs ue_turtlebot`

### Complete Reset

```
cd demo  # or demo_containerized
./kill_all_free5gc.sh
# Remove containers and networks
docker compose -f free5gc_setup/docker-compose.yml down -v
docker compose -f free5gc_setup/docker-compose-gNB.yml down -v
docker compose -f free5gc_setup/docker-compose-ue.yml down -v
docker network rm ros_gz_net phine-net demo-n3 2>/dev/null || true
# Rebuild
./build_images.sh
```

---

## 9. Getting Help

### Documentation

📖 **Main README:** README.md
📖 **Core Networks Guide:** doc/5G_CORE_NETWORKS.md
📖 **Developer Guide:** DEVELOPER_DOCUMENTATION.md
📖 **Setup READMEs:** `demo/[oai|free5gc|open5gs]_setup/README.md`

## Community Support

💬 **Slack:** Join RoboSim Workspace
🐛 **GitHub Issues:** RoboSim5G Issues
📧 **Email:** riccardo.belletti@phine.tech

## Contributing

📝 **Contributing Guide:** CONTRIBUTING.md
📜 **Code of Conduct:** CODE_OF_CONDUCT.md
⚙️ **Coding Guidelines:** CODING_GUIDELINES.md

## Quick Reference

| Core Network | Launch | Kill |
|---|---|---|
| OAI | `source launch_robot_5G.sh` | `./kill_all.sh` |
| free5GC | `source launch_robot_5G_free5gc.sh` | `./kill_all_free5gc.sh` |
| Open5GS | `source launch_robot_5G_open5gs.sh` | `./kill_all_open5gs.sh` |
| **Essential Commands:** | | |

```
docker ps -a                          # List containers
docker logs <container_name>          # View logs
docker exec -it <container> bash      # Enter container
ros2 topic list                       # List ROS topics
ros2 node list                        # List ROS nodes
```

---

**Document Version:** 1.0
**Last Updated:** March 2026
**Maintainer:** phine.tech GmbH
**Happy Simulating!** 🤖📡