

Introduction

B(E)3M33MRS — Aerial Multi-Robot Systems

Tomáš Báča, Ph.D.
©2023

Ing. Michal Werner, Ing. Martin Jiroušek
©2025

Multi-Robot Systems group, Faculty of Electrical Engineering
Czech Technical University in Prague



FACULTY
OF ELECTRICAL
ENGINEERING
CTU IN PRAGUE



MULTI-ROBOT
SYSTEMS
GROUP

Lab Tasks

Lab 1: Introduction

Tomáš Bába

Tasks scoring

Tasks' description

Computer requirements

Linux terminal

Git

C++

MRS System

MRS Cheatsheet

Singularity

How to start

Automatic eval.

01 Control

- Pass = 10 points

02 Formation

- Pass = 10 points
- + up to 15 points

03 Swarm

- Pass = 10 points
- + up to 15 points

Exam + Labs = Final mark, 100 points = A, ECTS

Max 60 points from the lab towards your exam.

14 lab weeks

- No holiday on Wednesday this year :-).
- Deadlines are listed at <https://cw.fel.cvut.cz/b251/courses/mrs/start>

01 Control

You will solve:

- UAV control,
- UAV state estimation.

You will learn:

- PID control,
- UAV dynamics,
- Linear Kalman Filter.

You will practice:

- Programming in C++,
- Simulating in Gazebo,
- Visualizing in Rviz.

02 Formation

You will solve:

- Formation control,
- Mission state machine,
- Multilateration.

You will learn:

- multi-UAV planning,
- UAV coordination,
- multilateration.

You will practice:

- More programming
- Simulating in Gazebo,
- Visualizing in Rviz.

03 Swarm

You will solve:

- Swarm control,
- Consensus.

You will learn:

- swarming and flocking,
- consensus.

You will practice:

- Even more programming,
- Simulating in Gazebo
- Visualizing in Rviz

1. Own Linux laptop

- “any” Linux OS
- $\approx$ 5 GB of free space
- + install *Singularity*
- **uninstall older singularity first**

2. Own Windows 11 laptop

- WSL 2.0 (Windows Subsystem for Linux).
- Works well on Windows 11, is trouble on Windows 10 and lower
- Instructions on slide 13.

3. Computer in the lab (physically)

- Works out of the box.
- Setup your own login: <https://cyber.felk.cvut.cz/study/computer-labs/>
- Alternatively, user: *guest*, password: *xxx*.

4. Computer in the lab (remotely)

- might work, might not
- Instructions: <https://cw.fel.cvut.cz/b241/courses/mrs/tutorials/start>
- You need to use your own login.

Work in Linux terminal

Lab 1: Introduction

Tomáš
Báča

Tasks
scoring

Tasks'
description

Computer
requirements

Linux
terminal

Git

C++

MRS
System

MRS
Cheatsheet

Singularity

How to
start

Automatic
eval.

- `cd <path>` — changes directory to path
- `cd ..` — goes one directory higher
- `cd` — goes to HOME
- `./script.sh` — will execute `script.sh` in current directory
- `nano file.txt` — edits `file.txt` using nano
- `vim file.txt` — edits `file.txt` using vim
- `exit` — exits shell session
- `zip -r archive.zip folder` — zips folder to `archive.zip`
- `unzip archive.zip` — unzips the archive
- `tar -xvzf archive.tar.gz` — unpack the `archive.tar.gz`

GIT — Version control system of the Open Source world

Lab 1: Introduction

Tomáš Bába

Tasks scoring

Tasks' description

Computer requirements

Linux terminal

Git

C++

MRS System

MRS Cheatsheet

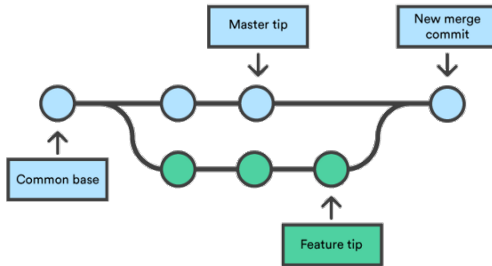
Singularity

How to start

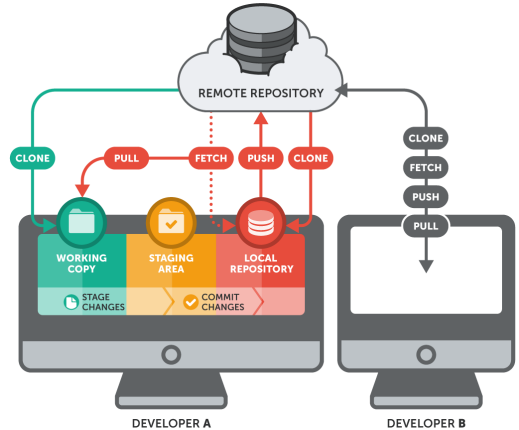
Automatic eval.

- Ability to synchronize changes between computers.
- Keeping track of own changes.
- Maintaining full history.

Git commits organized in branches



Decentralized software version control system



Git hosting servers — where to host your projects

- <http://github.com> — free public and private repositories
- <http://gitlab.com> — free public and private repositories
- <http://gitlab.fel.cvut.cz> — study-related projects

Tips for setting up

- Setup SSH keys to push/pull without password: (<https://docs.github.com/en/authentication/connecting-to-github-with-ssh/adding-a-new-ssh-key-to-your-github-account>)
- Make your Lab task repository private, to avoid others from stealing your unique solution ;-)
- For the start, use Github's desktop client.

GIT — Version control system of the Open Source world

Lab 1: Introduction

Tomáš Bába

Tasks
scoring

Tasks'
description

Computer
requirements

Linux
terminal

Git

C++

MRS
System

MRS
Cheatsheet

Singularity

How to
start

Automatic
eval.

Often-used git commands

- `git clone <url>` — clones repository from <url>
- `git status` — shows the status of the repository
- `git diff` — shows uncommitted changes in the repository
- `git add file.txt` — adds changes in file.txt to the next commit
- `git add -A` — adds all changes to the next commit
- `git commit -m "fixed bugs"` — commits changes with message "fixed bugs"
- `git commit -am "fixed all bugs"` — commits all changes with message "fixed bugs"
- `git reset --hard` — **reverts all changes (dangerous)**
- `git clean -fd` — **removes new uncommitted files (dangerous)**
- `git pull` — pulls current branch
- `git push` — pushes current branch

- Vanilla C++17
- Standard libraries
- Eigen library

Visual Studio Code

- provided within the container
- code completion should work

SublimeText

- provided within the container
- code completion might work

Vim

- provided within the container
- code completion will work

Standard libraries

Output:

- `std::iostream`
- `std::stringstream`

Utilities:

- `std::pair`
- `std::tuple`

Containers:

- `std::vector`
- `std::list`

Eigen library

- `Eigen::Matrix`
- `Eigen::Vector`

MRS UAV System on Robot Operating System

Lab 1: Introduction

Tomáš Bába

Tasks scoring

Tasks' description

Computer requirements

Linux terminal

Git

C++

MRS System

MRS Cheatsheet

Singularity

How to start

Automatic eval.

Do NOT install it!

- Used by the MRS lab to do research.
- Students use to do Bc/Msc theses.
- Contained within a Singularity container for the lab assignments

What does it do?

- UAV control
- State estimation
- UAV deployment
- Motion planning
- Sensor processing

MRS UAV System

The screenshot shows the GitHub repository for the MRS UAV System. The repository is owned by the 'Multi-robot Systems (MRS) group at Czech Technical University in Prague'. It has 113 repositories, 3 projects, 4 teams, 30 people, and settings. The pinned repositories are:

- mrs_uav_system** (Public): The entry point to the MRS UAV system. 193 stars, 55 forks.
- uav_core** (Public): The main integrator of MRS UAV packages in ROS, part of the 'mrs_uav_system'. 36 stars, 14 forks.
- simulation** (Public): The main MRS simulation Gazebo/ROS package. Part of the 'mrs_uav_system'. 28 stars, 12 forks.
- mrs_singularity** (Public): Singularity definitions, scripts and resources for the MRS UAV System. 9 stars, 8 forks.
- example_ros_packages** (Public): Integrates other example packages. Part of the 'mrs_uav_system'. 3 stars, 4 forks.
- mrs_cheatsheet** (Public): Cheatsheet for the 'mrs_uav_system', ROS, Linux, Tmux, Vim and more. 8 stars, 0 forks.

On the right side, there is a 'View as: Public' dropdown, a note about viewing the README, and a 'People' section showing a grid of user avatars.

https://github.com/ctu-mrs/mrs_uav_system

Robot Operating System?

Lab 1: Introduction

Tomáš Bába

Tasks scoring

Tasks' description

Computer requirements

Linux terminal

Git

C++

MRS System

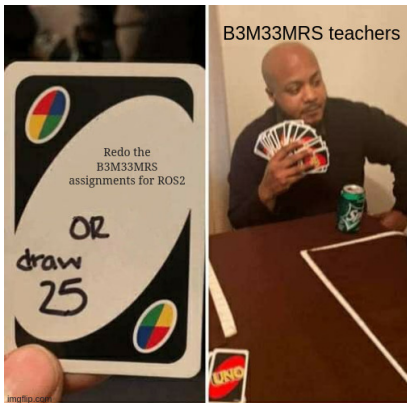
MRS Cheatsheet

Singularity

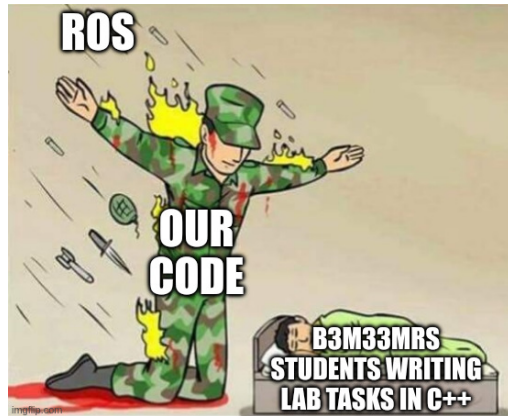
How to start

Automatic eval.

ROS 1 or ROS 2?



Does it matter? No...



MRS Cheat Sheet

Lab 1: Introduction

Tomáš Bába

Tasks scoring

Tasks' description

Computer requirements

Linux terminal

Git

C++

MRS System

MRS Cheatsheet

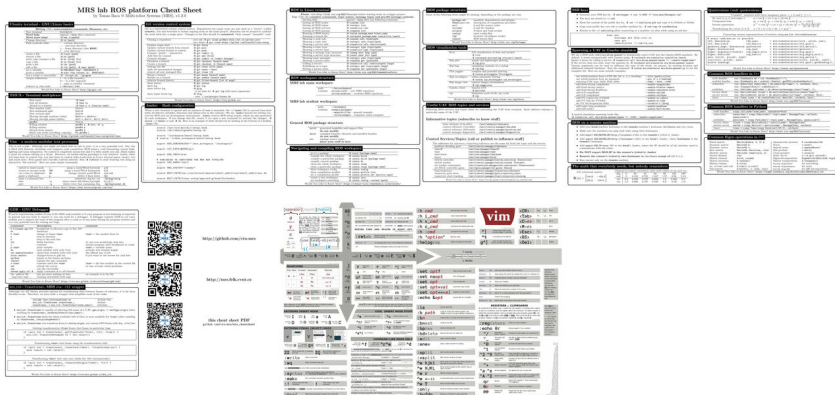
Singularity

How to start

Automatic eval.

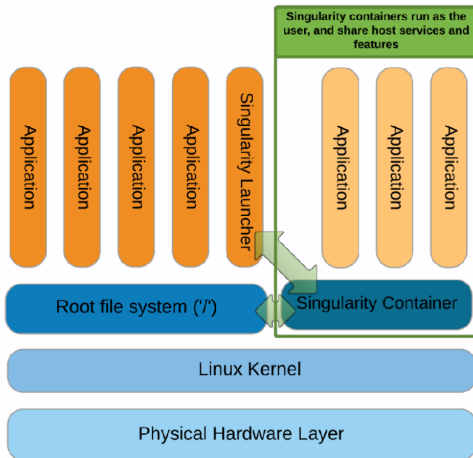
https://github.com/ctu-mrs/mrs_cheatsheet

5 pages of tips for Linux, bash, ROS, C++, Python, Vim and more



Singularity Container

- Minimal-invasive option for personal PC
- ≈ 4 GB file docker image
- Includes all build- and run- time dependencies
- “Guaranteed” compatibility during the semester
- **No need to learn singularity**
- Students are shielded from it by our scripts



On the lab's computer

Lab 1: Introduction

Tomáš Bába

Tasks scoring

Tasks' description

Computer requirements

Linux terminal

Git

C++

MRS System

MRS Cheatsheet

Singularity

How to start

Automatic eval.

1. Go to <https://cw.felk.cvut.cz/brute/>
2. Login as a student
3. Locate the Task 01 Control

Assignment	control - Task 01 Control	
Mandatory	This assignment has to be uploaded or evaluated. Required minimum score is 1.	
Deadline	24.10.2023 20:00	
Penalty	Every 168 hours will cost you 2 points down. But no more than 14 points.	
Latest upload	Uploaded at 26.9.2023 11:47  	
Required time	2	Time spent on solution (in hours). Required item: fill to allow upload file.
Upload new version	Select file... 	
Allowed types	['zip', 'tar', 'gz', 'bz2', 'tgz']	
Allowed file size	350000 kB	
Score	10 (+ AE: 10) (min: 1 max: 10)	
Number of uploads	1 Note: Only 100 uploads are allowed	
Results	AE result	

4. Download the zip file
5. Unpack the archive file (`tar -xvzf task_01_control.tar.gz`)
6. Follow the instructions in the `assignment.pdf` (section 2.3 Starting the simulation)
 - **skip the step with running `install_singularity.sh`**

1. Go to `https://cw.felk.cvut.cz/brute/`
2. Login as a student
3. Locate the Task 01 Control
4. Download the zip file
5. Unpack the archive file (`tar -xvzf task_01_control.tar.gz`)
6. Follow the instruction in the `assignment.pdf`
7. Follow the instruction in the `assignment.pdf` (section 2.3 *Starting the simulation*)
 - **For Ubuntu-compatible OS:** follow the step with running `install_singularity.sh`
Uninstall your previous installation first.
 - **For other Linux OS:** Install Singularity according to the official documentation:

`https://docs.sylabs.io/guides/3.0/user-guide/index.html`

On personal computer with Windows

Lab 1: Introduction

Tomáš
Báča

Tasks
scoring

Tasks'
description

Computer
requirements

Linux
terminal

Git

C++

MRS
System

MRS
Cheatsheet

Singularity

How to
start

Automatic
eval.

Windows system on Linux (WSL) on Windows 11

- Allows running Ubuntu “natively” under Windows
- Works 100% of time

1. Make sure your windows are up-to-date.
2. Install WSL (`wsl --install` in power shell)
3. Install Ubuntu 20.04 using the Microsoft Store
4. **Follow the Linux instructions on the previous page**

On personal computer with Windows

Lab 1: Introduction

Tomáš Bába

Tasks scoring

Tasks' description

Computer requirements

Linux terminal

Git

C++

MRS System

MRS Cheatsheet

Singularity

How to start

Automatic eval.

Windows system on Linux (WSL) on Windows 10 or lower

- Allows running Ubuntu “natively” under Windows
- It's great when it works
- Might not work due to your GPU, GPU driver, Windows version, etc.
- Works on average in 66% of time

1. Make sure your windows are up-to-date.
2. Install WSL (`wsl --install` in power shell)
3. Install Ubuntu 20.04 using the Microsoft Store
4. Download and install VcXsrv (an X-server client for Windows used to see the GUI from within Ubuntu)
5. Start the `vcxsrv.exe` by running: `vcxsrv.exe -ac -multiwindow`
6. Run the Terminal from the Start menu and launch a new terminal with Ubuntu 20.04.
7. Run `sudo apt update && sudo apt upgrade && sudo apt install gedit`
8. Run `gedit` to verify that GUI will show up
9. If the GUI does not show up, add the following line to the `$HOME/.bashrc` file in the Linux subsystem:

```
export DISPLAY=$(grep -m 1 nameserver /etc/resolv.conf — awk '{print $2}'):0.0
```
10. **Follow the Linux instructions on the previous page**

- It is possible, e.g., using VMWare, VirtualBox.
- Will be slow.
- Will need approx. 20 GB of space.

- We offer limited support on MAC.
- The default software will not work with Apple silicon due to different CPU architecture.
- However, we managed to build our system for ARM.
- If you want to use macOS, let us know.

BRUTE server

- All points earned through BRUTE
- Simulation in BRUTE is serialized
- Be patient, it might take time.
- Approx. 3 minutes for task 01, if there is no one in the queue.
- Approx. 8 minutes for task 02, if there is no one in the queue.

On local machine

- The same evaluators as in BRUTE
- Follow the instructions in the assignment pdf

Info result page in BRUTE

MRS — Task 01 Controller — bacatoma (1)

Submission

Login:	bacatoma
Year:	2023Z
Submission no.:	1
Seconds after deadline:	0

Results

Score: 10.0

Control test: **PASSED**, position RMSE 0.35 (limit 0.50), tilt RMSE 0.11 (limit 0.50), additional info: finished

Kalman test: **PASSED**, position RMSE 0.02 (limit 0.05), velocity RMSE 0.03 (limit 0.05)

Control test log:

```
source /tmp/ros_workspace/devel/setup.bash && roslaunch task_01_evaluation controller_test.launch file:=/tmp/tmpxxo2a5_6
... logging to /home/student/.ros/log/b25d2e28-5c52-11ee-ada9-3b9b7c372382/roslaunch-c03518d54a3d-3405.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is ~1GB.
[2:/tmp/ros_workspace/src/ros_packages/evaluation/launch/controller_test.launch
started roslaunch server http://127.0.0.1:42387/

SUMMARY
=====
PARAMETERS
 * /roscppdistro: noetic
 * /rosversion: 1.15.14
 * /uav1/controller_test/file: /tmp/tmpxxo2a5_6
 * /uav1/controller_test/main_timer/rate: 100
 * /uav1/controller_test/path: [0.0, 0.0, 1.5, 0.0]
 * /uav1/controller_test/relax_heading: True
 * /uav1/controller_test/requirements/rmse_position: 0.5
 * /uav1/controller_test/requirements/rmse_tilt: 0.5
 * /uav1/controller_test/start_timer/rate: 1
 * /uav1/controller_test/stop_at_waypoints: False
 * /uav1/controller_test/use_heading: True
 * /uav1/controller_test/version: 1.0.0

NODES
  /uav1/
    controller_test (nodelet/nodelet)

ROS_MASTER_URI=http://localhost:11311
```

MS Teams

- You should be added to the B3M33MRS course.
- Post general questions rather than sending personal messages.

Frequently asked questions

Lab 1: Introduction

Tomáš Bába

Tasks scoring

Tasks' description

Computer requirements

Linux terminal

Git

C++

MRS System

MRS Cheatsheet

Singularity

How to start

Automatic eval.

Student survey: "...the homework tasks and their corresponding PDF descriptions are, in my humble opinion, unnecessarily long and over-complicated."

We know the descriptions may seem long, but that's the trade-off for having tasks that are more interesting and self-contained. The detailed documentation is there so you don't need to guess or ask about missing details. We believe this is better than having shorter (or incomplete) descriptions for less engaging tasks.

Are the lab sessions mandatory?

No.

What are the learning objectives of the labs?*

- To **offer hands-on experience in transforming theory into practice** by implementing methods from the lectures.
- To **recap/extend** some (exam) topics from the lectures
- To **encourage students' creativity and problem-solving skills**. We tried to design the assignments in a way that there are multiple ways how to achieve the required functionality.

**Yes, you are right – no one has ever asked this question. But what if...*

Frequently asked questions no. 2

Lab 1: Introduction

Tomáš
Báča

Tasks
scoring

Tasks'
description

Computer
requirements

Linux
terminal

Git

C++

MRS
System

MRS
Cheatsheet

Singularity

How to
start

Automatic
eval.

I am stuck, my code/simulation/BRUTE does not work and I don't know why (the task description did not help). What should I do?

Ask us! ;-)

References

Lab 1: Introduction

**Tomáš
Báča**

Tasks
scoring

Tasks'
description

Computer
requirements

Linux
terminal

Git

C++

MRS
System

MRS
Cheatsheet

Singularity

How to
start

Automatic
eval.