University of Magdeburg

School of Computer Science

Bachelor Thesis

Mixed-reality Simulation of Quadcopter-Swarms

Author:

[Lukas] [Mäurer]

Matrikelnummer: 189073

September 30, 2015

Advisors:

Prof. Sanaz Mostaghim

Department of Intelligent Systems

Christoph Steup

Department of Intelligent Systems

***[Mäurer]*, *[Lukas]*:**
Mixed-reality Simulation of Quadcopter-Swarms
Bachelor Thesis, University of Magdeburg, 2015.

Contents

List of Figures iv

List of Tables v

List of Code Listings vi

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Problem Statement . . . 1
- 1.3 Outline . . . 1

2 Theory 2
- 2.1 Quadcopter Modelling . . . 2
- 2.2 Vrep . . . 2
- 2.3 Communication V-REP-Quadrocopter . . . 2

3 Implementation 5
- 3.1 Simulation Environment . . . 5
- 3.2 Communication Link . . . 5
- 3.3 *[fancy name]* . . . 5
- 3.4 Quadcopter . . . 5

4 Evaluation 6
- 4.1 Speed . . . 6
- 4.2 Accuracy . . . 6
- 4.3 | . . . 6
- 4.4 usability . . . 6

5 Conclusion 7
- 5.1 . . . 7
- 5.2 Future Work . . . 7

Bibliography 8

List of Figures

2.1 communication V-REP - Quadrocopter 3
2.2 Ivy-Bus in Paparazzi Ground Station 4

List of Tables

List of Code Listings

1. Introduction

1.1 Motivation

[project context of this work]

[who needs the results]

[what are the problems to be solved?]

[what are existing solutions, what's different in this approach, what is the improvement]

1.2 Problem Statement

[what are the goals] [how are we going to reach this goals] [what is to be done]

1.3 Outline

[short description of the sections]

2. Theory

2.1 Quadcopter Modelling

[fundamental physics]

[particle simulation]

2.2 Vrep

[connecting visual representation and physical model]

[simulation structure (lua scripts, scene structure]

[lua module structure]

[external interface (signals)]

2.3 Communication V-REP-Quadrocopter

***Goal*:** our mixed reality simulation needs a dependable link of communication between the V-REP simulation environment and the flying quadrocopters. The Quadcopter needs to stream its telemetry data in real-time to the V-REP, and the reverse communication is needed as well. The communication between the V-REP and the quadrocopters passes through several software components, which are depicted on figure Figure 2.1.

V-REP provides several means of communication with an external application. One of them is the Remote API, which allows to control a simulation (or the simulator itself) from an external application or a remote hardware (e.g. real robot, remote computer, etc.). The V-REP remote API is composed by approximately one hundred functions that can be called from a C/C++ application, a Python script, a Java application, a

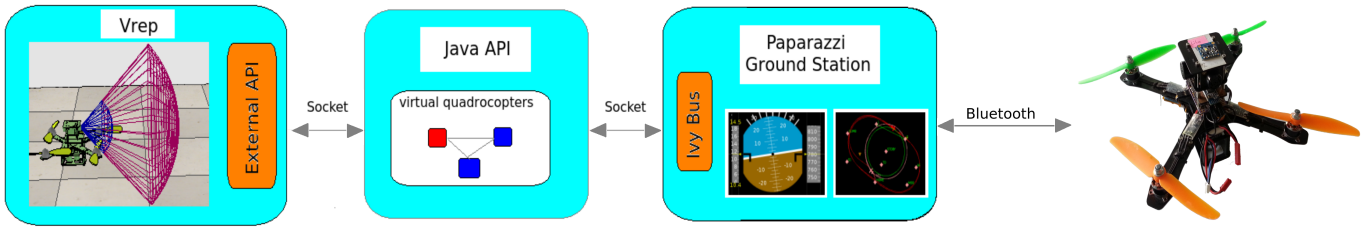

Figure 2.1: communication V-REP - Quadrocopter

Matlab/Octave program, an Urbi script, or a Lua script. The remote API functions are interacting with V-REP via socket communication in a way that reduces lag and network load to a great extent.

Java API is the external program, that communicates with V-REP through the Remote API. We have chosen to implement our external program, communicating with the V-REP, in the Java programming language regarding the following advantages: Java's platform independence allows to run the external program even on different machine with different operating system than the one used for running the V-REP environment. Java is object-orientated which favours the use of design patterns and highly abstraction layers, which allows us to write an API that is modular, reusable and can later be easily extended to support other mixed-reality scenarious. The implementation and architecture of the Java API is duscussed in details in Chapter 3. The purpose of the Java application is to serve as a communicating bridge between the Paparazzi Ground Station and the V-REP. It detects all quadrocopters in the V-REP simulation, builds their virtual representations and feeds the models with real-time data.

Ivy Bus is a simple protocol and a set of open-source (LGPL) libraries and programs that allows applications to broadcast information through text messages, with a subscription mechanism based on regular expressions. Ivy libraries are available in C, C++, Java, Python and Perl, on Windows and Unix boxes and on Macs. The Paparazzi Ground Station uses the Ivy Bus as a means of communication between the different software components. Figure Figure 2.2 depicts the communication structure in the Paparazzi Ground Station, in which the different agents communicate with each other by sending messages on the Ivy-Bus.

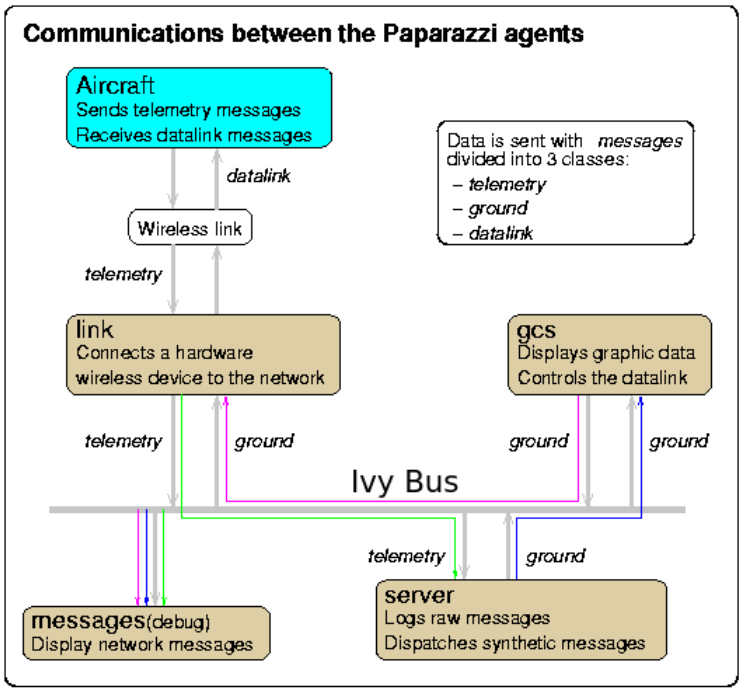

Figure 2.2: Ivy-Bus in Paparazzi Ground Station

3. Implementation

3.1 Simulation Environment

[finken parameter estimation]

[controller tuning]

[simulation parameters]

[script structure]

3.2 Communication Link

[link in Vrep (signals]

3.3 *[fancy name]*

3.4 Quadcopter

4. Evaluation

[how realisitc is the simulation?] [which properties can be modelled well, which can't?]

4.1 Speed

[communication delay] [vrep simulation speed]

4.2 Accuracy

[error]

4.3 |

Stability

4.4 usability

5. Conclusion

5.1

[do the results show that it works?]

5.2 Future Work

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Magdeburg, den 29. November, 2013