

Welcome to Robobulls!

Started in March 2014, this project contains the software that powers the University of South Florida's autonomous Small Size League Robocup team.

To get started in the project, you should familiarize yourself in the following programs & concepts:

- Linux fundamentals (Bash, common commands, etc)
- Git, Github, fundamentals of version control
- Basics of soccer, especially as they pertain to the [SSL-Robocup Ruleset](#)
- All of the documentation pertaining to this project (see [Additional Resources](#))

If you plan on working with the Robobulls project outside of the lab, you will need a machine that runs Linux (we have had limited success running virtual machines of Xubuntu 18.04 and Linux Mint 18.3) and has substantial CPU power. In particular, simulating two Robobulls teams against one another requires considerable processing capability.

There are many parts to this project, including several external programs. The highest-level parts of the Robobulls project consist of these programs: the Robobulls Dashboard, grSim, SSL-Refbox, and the SSL-Vision system. Below is a brief overview of each component.

The Robobulls Dashboard

The Robobulls Dashboard is the Robobulls AI program which controls the team. It includes a GUI dashboard, which displays the status of the field in the center, and the status of each robot on the left panel. In the below image, there are four blue team bots visible, one of which (Robot #2) is overridden (see the checkbox in the top left). This bot will not move autonomously and can be operated manually, regardless of whether the game is simulated or being run with real, physical bots. This allows you to interact with the bots in the simulation more easily than using the options in grSim, or allows you to position a physical bot without having to move into the game field.

All other bots are not overridden and will move autonomously according to the current game state. It is possible to override multiple bots, or the entire team simultaneously.

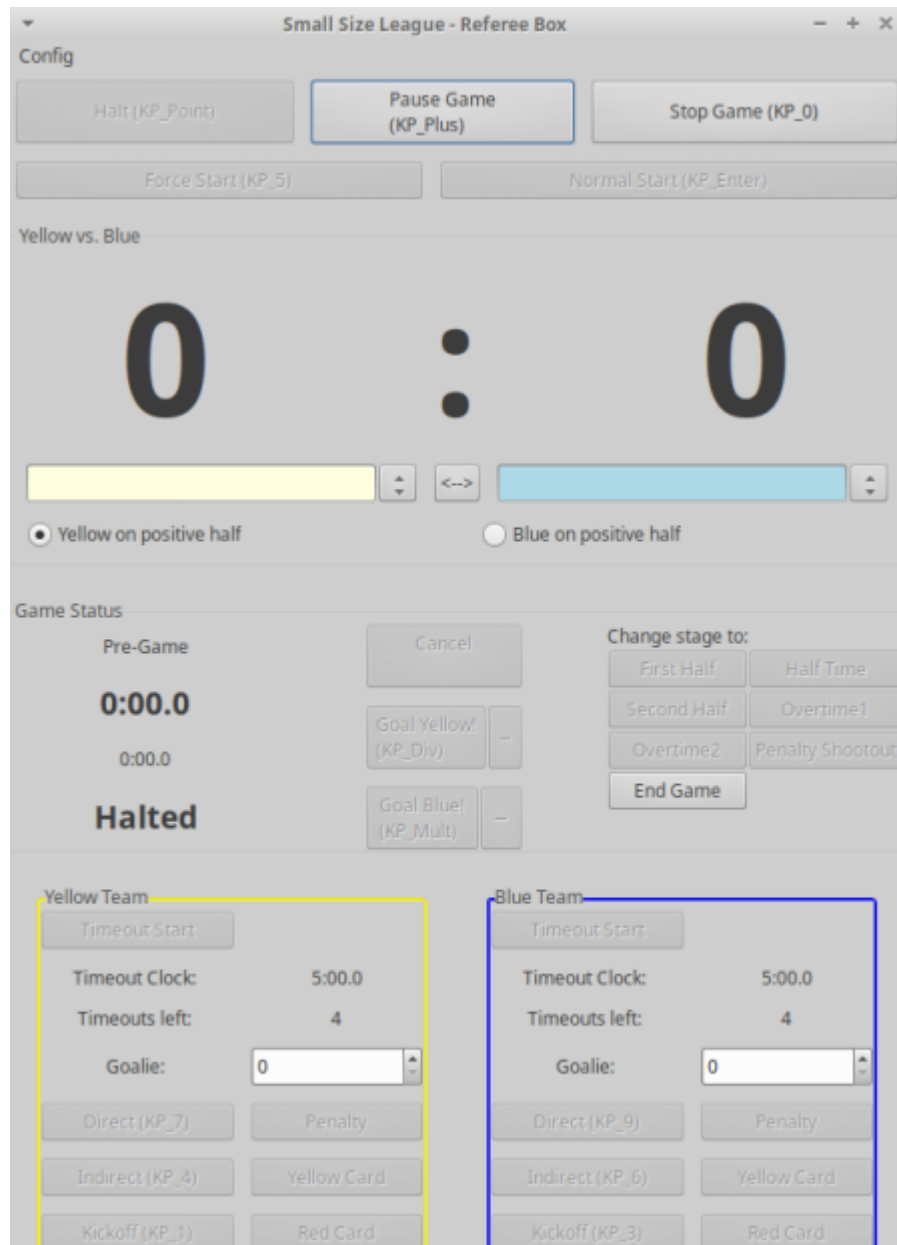


SSL-Refbox

[SSL-Refbox](#) is a simple program that lets a neutral third-party operator (a member of the officiators, not one of the teams) send orders of the referee to the teams' software over network.

Since approximately 2012-2013, SSL-Refbox has implemented a [new communication protocol](#) (see the documentation link above) which sends much more game information than the legacy protocol. Currently, Robobulls is using the [legacy implementation](#).

In the lab, use a copy of Refbox to control the team's behaviors for testing. Note that you must name the teams in refbox, or the software will not work properly.



SSL-Vision System

The [SSL-Vision System](#) is a shared camera/vision system used by all SSL Robocup teams, introduced in 2010 to eliminate the issues caused by individual camera systems & setup. This is the system that allows the teams' software to perceive the field and state of the game.

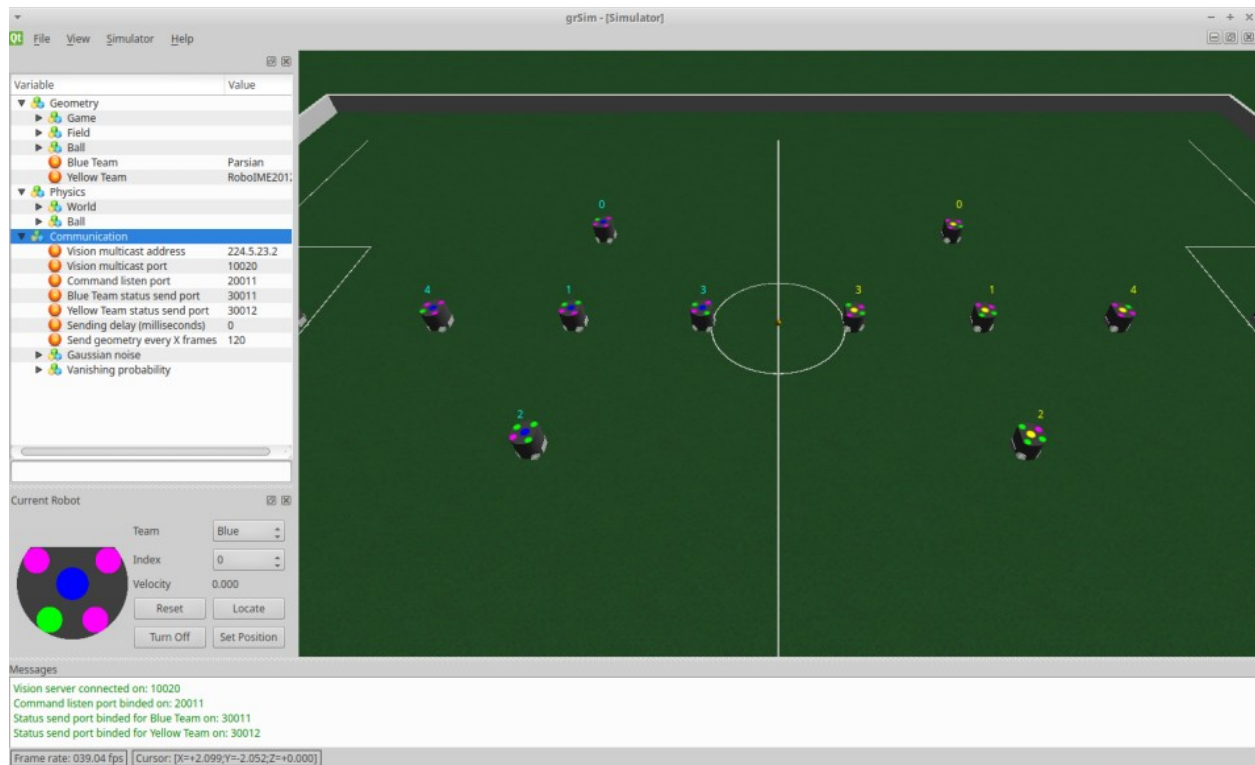
The field is covered by two or more cameras, one over each half of the field. A software package is provided which calibrates the cameras, detects the robots, and then continually sends the robot/field information over the network to the team's software for use.

For further information on acquiring, setting up, and using the vision system, see the documentation links below.

grSim

Similar to the SSL-Vision System, grSim provides software for allow the team's software to perceive the game field. The difference between grSim and SSL-Vision is that grSim's field is completely simulated - generated by grSim's software.

This allows the Robocup teams to simulate games virtually, by communicating their team actions to grSim, and receiving game state/field info from grSim, rather than SSL-Vision. grSim also provides options for testing different divisions, fields, and the ability relocate, disable, and reset individual bots.



Additional Resources

- [Robobulls Readme](#)
- [Robobulls Installation Guide](#)
- [Robobulls Operation Guide](#)
- [Robobulls Development Goals](#)
- [SSL-Robocup Ruleset](#)
- [SSL-Refbox Documentation](#)
- [SSL-Vision Official Documentation](#)
- [SSL-Vision Robobulls Operation Guide](#)
- [Yisibot Manual](#)
- [Robobulls GUI Honors Thesis \(Overview of Robobulls\)](#)

Also, see the Robobulls pre-2016 Doxygen HTML documentation, found by opening the Documentation.html file located in documents of the main project folder.

Contact Info

After reading all of the above documentation, if you need additional clarification on an issue detailed above and the current lab admins are not available or are unfamiliar with your problem, you can contact one of the previous team members for more information.

May-Aug 2018 Team Members:

nherbert2@mail.usf.edu

- Familiar with Yisirobot movement code, out-of-lab installation. Authored readme, devgoals, installation guide, operation guide. (8/03/18)

kellerc@mail.usf.edu