

ITAS 0.99
Manual

Tarek Hassan
(thassan@connectmail.carleton.ca)

Carleton University
The Department of Systems and Computer Engineering
1125 Colonel By Drive
Carleton University
Ottawa, Ontario
K1S 5B6
Canada
+1 (613) 520 5740
+1 (613) 520 5727

© 2002 Carleton University

1 Introduction.....	4
1.1 ITAS	4
1.2 RoboCup	4
1.3 RoboCup Soccer Simulator Server	4
2 Getting Started.....	5
2.1 What you need	5
2.1 Joining a game	5
3 The Display	6
3.1 The Field	7
3.2 The View Cone	7
3.3 The Players & the Ball.....	7
4 The Controls.....	8
5 For Developers.....	Error! Bookmark not defined.
6 Contacts & More Information	9
6.3 ITAS Development Team	9
6.4 Useful Links	9
References	11

1 Introduction

1.1 ITAS

ITAS = In The Agent's Shoes

This tool was developed to put a human user in the place of a RoboCup agent for the purpose of logging their actions and their reactions to different game situations. This log could then be used for machine learning for development of software agents. ITAS could also be used to train, test or challenge an AI team by allowing the human players to create desired game situations through their actions (e.g. try different attack approaches to challenge a goalie agent's adaptability).

1.2 RoboCup

RoboCup is an international research initiative driven by competition. Its ultimate goal statement is, "By 2050, develop a team of fully autonomous humanoid robots that can win against the human world champion team in soccer." [1] By providing a standard and well-known challenge, RoboCup promotes AI and robotics research and innovation worldwide. An annual world cup competition is held with different leagues that focus on different aspects of the RoboCup challenge. There are four different robotic leagues and one RoboCup Simulation league. More information about RoboCup is available at www.robocup.org.

1.3 RoboCup Soccer Simulator Server

The RoboCup Simulation league focuses on the artificial intelligence aspect of the RoboCup project by abstracting away from the problems of robotics. The physical field environment is replaced with a simulated field, provided by The RoboCup Soccer Simulator Server (RCSSS hereon) [2]. RCSSS is a server that simulates the soccer field environment and the objects on the field. Players (the software agents) are clients that log onto this server and interact with it by receiving world information and sending commands. Essentially, each agent is the brain of a player, receiving sensory stimuli and issuing commands based on these stimuli. The basic commands available to

the agent for controlling the players actions are 'dash', which makes the player run forward with a given power, 'turn', which makes the player turn with a given moment and 'kick' which kicks the ball with a given power. Additional information about RCSSS is available at sserver.sourceforge.net.

2 Getting Started

2.1 What you need

First, to run ITAS, you will need a Java Runtime Environment (J2RE 1.4 or higher is recommended) and the ITAS software. If a game server is already running, all one needs is the ITAS software, and the server's address and port number. If you wish to start your own server, you will also need the RCSSServer software and a monitor program to setup and control the server environment.

What you need	Where to get it
ITAS (latest version)	itas.sourceforge.net
J2RE 1.4 or higher	sun.java.com/j2se
RCSSServer 8.0 or higher (if running a server)	sserver.sourceforge.net
RCSS monitor (if running a server)	sserver.sourceforge.net

2.1 Joining a game

Make sure that your JRE is properly installed and that your CLASSPATH is set to the right location.

Unzip the ITAS files into a directory of your choice.

From a terminal or prompt, go to your ITAS directory and run ITAS using the following command line:

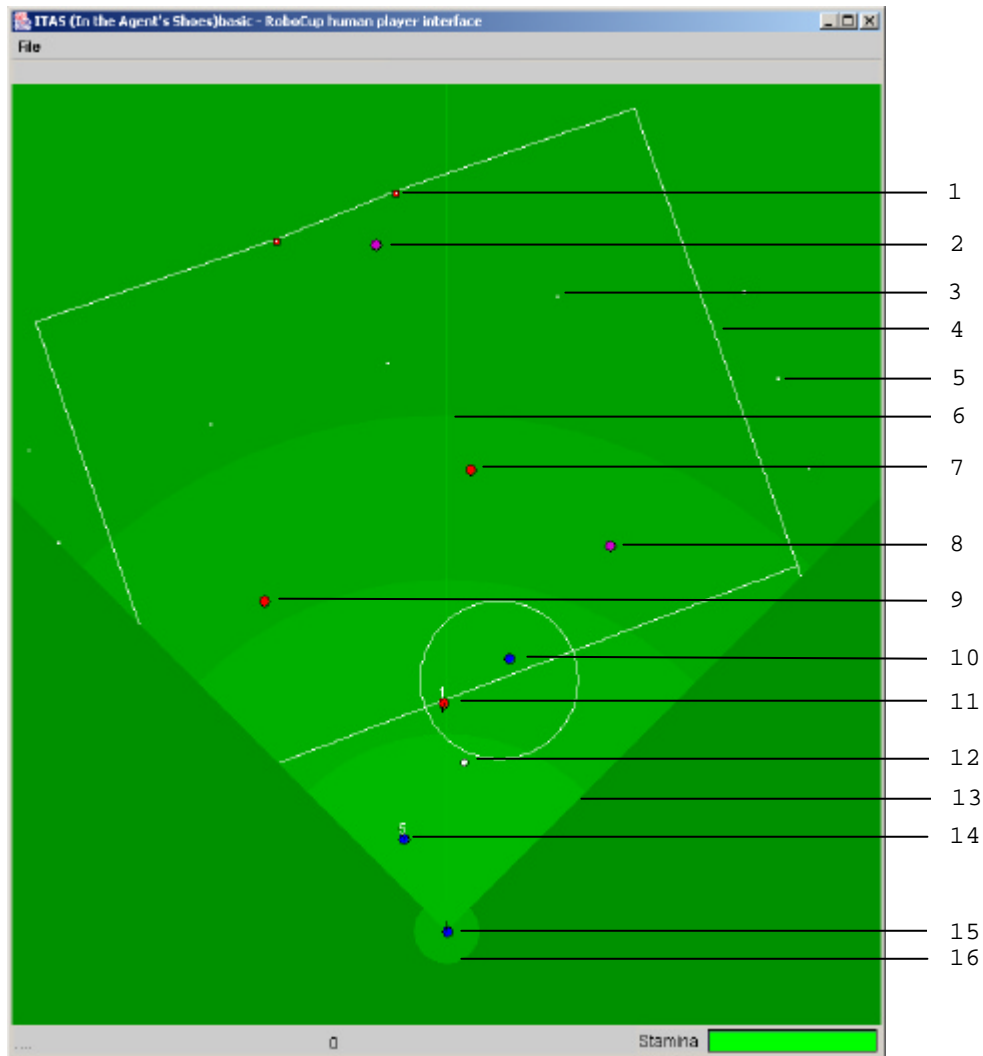
```
java ITAS -host <hostname> -port <port#> -team <teamname>
```

(defaults: hostname="", port="6000", team="GreyMatterUtd")

3 The Display

Your player's view is displayed in a bird's-eye perspective (top-down). Everything in the display is relative to your on-screen player(14). i.e. the player and their view cone are static and everything else moves relative to them.

This is a screenshot of the ITAS interface.



- | | |
|--------------------------|-----------------------------|
| 1. goal post (red side) | 9. opponent |
| 2. unidentified player | 10. teammate |
| 3. inner flag | 11. opponent |
| 4. field line | 12. the ball |
| 5. outer flag | 13. view-cone |
| 6. view-cone center line | 14. teammate |
| 7. opponent | 15. user's player |
| 8. unidentified player | 16. peripheral vision range |

3.1 The Field

The field is defined by flags and lines. The white lines(3) are the field lines and define the perimeter of the field, the center line and kick-off circle. Flags are markers located on and outside of the field. 36 flags(1) are located around the outside of the field, 6 flags mark the actual perimeter of the field, 4 flags indicate the goalposts(2) for each side (the colour of the goalpost depends on the side) and 7 flags(4) mark key points on the field (center field and penalty boxes).

3.2 The View Cone

The view cone(13) indicates the limits of the player's normal vision. The shape of the view-cone depends on the player's view width (see controls). The screen shot shows the default setting of 90°.

The view cone is further broken down into different ranges of vision quality. These can be identified by their different shades of green.

Beyond the farthest range(6), nothing is known about the player except for that they are a player. In the next farthest range(8), the player's team can be identified, but not consistently. In the second closest range, the player's team can be identified consistently and the player's number and direction can be identified inconsistently. Within the closest range(11), a player's number, direction and team are always identifiable.

The circle around the user's player indicates their peripheral vision(15). If an object is outside of the view cone and in this circle, it is visible, but no details about the object are available.

3.3 The Players & the Ball

The ball(10) is a small white circle with a black outline.

A player is drawn as a circle with a black outline. Depending on the position and distance of the player, their

player number (white numbers), the direction they are facing (marked by a small black dash) and their team colour (left=blue, right=red & unknown=purple) are visible. Different player representations are seen in the screenshot above (12, 9, 7 & 5).

4 The Controls

Running:

Clicking the left mouse button will make your player dash forwards with a power relative to the distance of the mouse click from your player on screen. If the mouse button is held down, however, this causes the player to run (dash repeatedly) until the mouse is released. The player's running speed is relative the distance of the mouse pointer from the player and will change dynamically as this distance changes.

Turning:

Clicking the right mouse button will make the player try to turn to face the point where the mouse was clicked. The player will turn with a moment equal to the angle of the mouse click from the view-cone center line. How far the player turns is affected by their speed.

Panning:

Holding down the right mouse button will cause the player to first turn once towards the mouse pointer, then start panning (turning gradually) towards the pointer at a rate relative to the angle of the pointer from the view-cone center line. The panning rate changes dynamically as this angle changes.

Note: You can run and pan at the same time. Experiment with this and you'll find your manoeuvrability will improve drastically.

Kicking:

Pressing the space-bar will make your player kick the ball towards the point where the mouse was clicked with a power relative to the distance of the mouse pointer from your player (like dash) and an angle equal to the angle of the mouse pointer (like turn).

Shooting:

Pressing 'alt' will make your player kick the ball in the same manner as pressing the space-bar will, except the power of the kick will always be the maximum power.

View width:

Pressing 'w' will switch your view-cone angle to one of three settings, 45°, 90° and 180°. The larger the angle, the slower the refresh rate of data coming from the server.

Zoom in/out:

The + & - keys will zoom in and out respectively.

5 Contacts & More Information

5.1 ITAS Development Team

ITAS was developed in the [Department of Systems and Computer Engineering](#) at [Carleton University](#) in Ottawa, Ontario Canada by Tarek Hassan under the supervision of Dr. Babak Esfandiari and Paul Marlow.

For support or for any inquiries regarding ITAS, please contact:

thassan@connectmail.carleton.ca

5.2 Useful Links

For more ITAS information, the latest news and releases of ITAS:

<http://itas.sourceforge.net>
(official ITAS website)

<http://www.sourceforge.net/projects/itas/>
(ITAS SourceForge page)

For more information on RoboCup in general:

<http://www.robocup.org>
(official RoboCup site)

<http://www.sce.carleton.ca/netmanage/robocup/>

(Carleton's RoboCup site - a little old, but has useful information and downloads)

For more RCSSS information, news and releases of RoboCup simulation:

<http://sserver.sourceforge.net>
(official RCSSS website)

References

- [1] <http://www.robocup.org>, RoboCup website
- [2] <http://sserver.sourceforge.net>, RCSS website