

ITAS and the Reverse RoboCup Challenge

Tarek Hassan and Babak Esfandiari

The Department of Systems and Computer Engineering,
Carleton University, Ottawa, Ontario, Canada, K1S 5B6
thassan@connect.carleton.ca,
babak@sce.carleton.ca

Abstract. ITAS is a tool that allows a human to play soccer in the RoboCup Soccer Simulator environment. This is essentially the reverse challenge to that of RoboCup. Instead of bringing the machine to the real world, ITAS strives to seamlessly interface man to the machine world. This presents a fundamental human-computer interaction design problem. This paper shows how the reverse RoboCup challenge can benefit the RoboCup community and what value it brings to robotics and AI research in large. An overview of the features of ITAS and its development using the Usability Engineering Lifecycle are then given, followed by a comparison with a related system, OZ-RP. ITAS is an open source project. The most recent releases are available at <http://itas.sourceforge.net>.

1 Introduction

ITAS, In The Agent's Shoes, is a RoboCup Simulation League soccer player that is controlled in real time by a human user. It provides the user with a representation of the sensory stimuli and the action commands that the RoboCup Soccer Simulator provides an agent. Originally developed as a tool with the purpose of logging a human player's interaction with the simulator as data for machine learning, it was quickly realized that developing a competitively performing human controlled player presented quite a challenge.

Since a human is as alien to the simulated environment as a robot is to the real world, this presents a fundamental human-computer interaction design problem. Development of ITAS has led to the recognition of the Reverse RoboCup challenge as an important undertaking unto itself and meeting this challenge became the project's main objective. The following is the statement of the Reverse RoboCup Challenge and an invitation to others to take on this challenge to develop the best human interface to the RoboCup Soccer Simulator (RCSS) environment.

"To develop a team of human controlled simulated soccer players that can win against the RoboCup Simulation League champion team."

2 Why the Reverse RoboCup Challenge?

2.1 Human Element in RoboCup Competition

Perhaps the most important benefit that the Reverse RoboCup Challenge brings to the RoboCup community is the introduction of a human element into the competition.

Even though RoboCup's mission statement is to defeat the human world soccer champions, so far the evolution of RoboCup players has been mostly based on competition amongst robots/agents. How can the progress of RoboCup towards its original goal be validated without the introduction of a human challenger in some form?

Until meticulous, full scale humanoid robotics are a reality, this problem cannot be addressed in the physical domain. However, human intelligence, strategy, decision-making and team work can be introduced to the competition by allowing real soccer players to play in the simulated soccer environment of the RoboCup Simulation League. As such, ITAS strives to be a tool that reflects the soccer play experience with respect to the kinds of decisions that are made in real time and the key environment and state information required to make those decisions.

2.2 Machine Learning

Generating data for machine learning was the original purpose of ITAS. To have a human player perform soccer tasks in RoboCup's simulated environment can be of great value. Logs of the player's actions will be based on the simulator parameters, facilitating interpretation and processing of the data for machine learning. However, the value of the data collected will depend on the quality of the human player's performance, which in turn depends on the quality of the interface to the simulated player.

2.3 Telerobotics

Interfaces developed for the reverse challenge can be adapted into interfaces for telerobotics control systems. The most obvious field being the RoboCup robotics leagues, as a start. ITAS's interface approach is particularly well suited for dealing with dynamic or unfamiliar environments since it is not dependant on relating incoming data to a predefined global model and placing the user in it, but rather presents the state of the environment as it is perceived in real time. The open source nature of the project makes it very accessible to robotics teams for such implementations and we encourage them to explore that option.

2.4 Other Areas

The reverse challenge makes for a great medium for research in collaborative and competitive interaction between artificial agents and human users in simulated or virtual environments. This can be of interest to researchers from many disciplines ranging from cognitive science and human factors to computer science and artificial intelligence.

Another exciting application for the reverse challenge is exploring the potential of the hybrid agent. This is an agent that combines the efficiency and accuracy of an artificial agent at performing tasks in the simulated environment, such as passing, dribbling, shooting and evasion, with the real-time strategic input of an actual soccer player. Such an agent may provide valuable competition to RoboCup champions with regards to facing a more human-like competitor.

Other areas of interest include user input devices. The complexity of the task at hand demands intuitive user control over the actions of an agent in a fast paced envi-

ronment, making it a good platform for testing new and different hardware. The OZ-RP group (another reverse RoboCup challenger) have implemented their interface on several platforms utilising a variety of control devices.

3 ITAS

3.1 Development Approach

ITAS development follows the processes defined by the Usability Engineering Life-cycle (UEL). The UEL consists of three phases, namely Requirement Analysis, Design/Verification/Development, and Installation, each composed of iterative sub phases. Perhaps the most valuable aspect of UEL to this project has been the focus on user profile and contextual task analysis that are part of the Requirement Analysis phase, as they have shaped the usability objectives of ITAS.

The contextual task analysis for this project considered two task domains which were important. The first is real life soccer. This is the native environment of the ideal user for ITAS, so understanding it goes a long way in understanding the user’s expectations and actions in the simulated environment. The physical characteristics and constraints of the real world are of little interest here, but rather the situational characteristics, as they provide the context in which the user will perceive the environment, set objectives and make decisions. The second domain is that of the simulated environment, ranging from the parameters, data objects and commands RCSS handles to the constraints and requirements of the ITAS interface. Understanding and then relating the two task domains to one another formed the base of ITAS’s interface design.

Soccer objectives change dynamically depending on game situations. We found the most significant state variable affecting a player’s task priorities to be ball possession (both relative to the individual and team-wide ball possession state). Team possession affects the whole team’s strategy whereas individual possession affects the particular player’s decisions. Based on observation of and discussions with our test group of experienced soccer players and the aforementioned states, a soccer task organization model was developed.

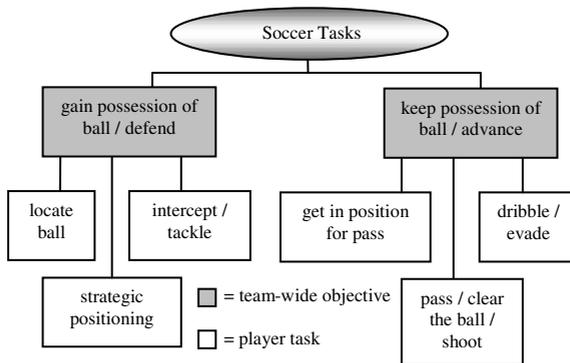


Fig. 1. Soccer task organization model

As shown in the Soccer task organization model, soccer play was broken down into two main team-wide objectives based on the team's state with regards to ball possession. Players work towards attaining these objectives or modes through performing the necessary key tasks. The key tasks named above define the decision-making domain in which we aim to place the ITAS user. So by providing the tools necessary to perform these tasks intuitively and efficiently, the user is left to make decisions as to when, where and how ('how' relating to the parameters of the task rather than it's mechanics) to perform these tasks.

With the exception of communication, which will be implemented in a later version of ITAS, each of the tasks defined in the task organization model requires the application of one or more of three basic skills, awareness of surroundings (objects and environment), maneuvering and ball handling.

These three skills relate almost directly to the three basic commands that RCSS provides; turn, dash and kick. The advanced tasks from the task organization model were designed by combining the use of these commands with interpretations of incoming sensory data to provide the user with higher level commands and a more informative and intuitive display. Conceptual mock-ups of the advanced controls and display features were discussed with our test group of soccer players and revised, before implementation. The tasks identified also helped design the drills that were used to test the different iterations of ITAS throughout development.

As well as following UEL, ITAS's development was split into two stages. An initial version, ITAS-basic, aimed to instantiate all the fundamental interactions between an agent and the RoboCup Soccer Simulator in the simplest form that is comprehensible to the user. This includes displaying real time visual and player state information sent by RCSS and providing the basic commands RCSS accepts such as dash, turn and kick. ITAS-basic is the platform on which more complex versions can be based. The second stage of ITAS is a version that provides the user with more advanced stimuli and control mechanisms. The advanced features in the most recent version of ITAS, as well as features currently in development, enrich the user experience by providing intuitive visual interpretations of the sensory information from the server as well as commands to perform more advanced tasks automatically.

3.2 Display

A birds-eye view was chosen to represent the player's visual field as it best represents the relative distances of all visible objects. The player is represented by a blue or red circle at the bottom of the screen. The limits of the player's field of view are indicated by shading the area outside of it in a dark green, helping the user focus on the relevant area of the display while maintaining situational awareness. Objects seen by the player are plotted in real time in the field of view.

A stamina-meter at the bottom of the screen indicates the stamina level of the player, changing color from green to orange then to red as the player's stamina level decreases to critical levels.

The basic visual information about the soccer field provided by RCSS is rather limited, as only 15 flags actually lie on field lines. The rest are on the outside of the field. Testers found the ITAS-basic (Fig. 2.b) display confusing since the familiar visual queues of a soccer field were missing. The latest ITAS (Fig 2.a), however,

calculates and draws missing field lines that aren't indicated as visible objects by inferring them from the visible flags. The inclusion of internal field lines, such as the kick-off circle and center line provides users with even more visual queues that aid in positional awareness.

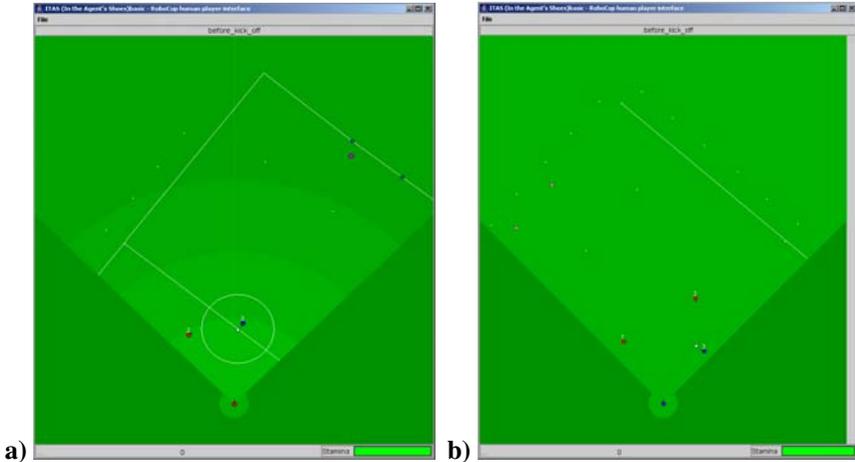


Fig. 2. a) ITAS screenshot b) ITAS-basic screenshot showing only the raw RCSS data

In RCSS, accuracy and consistency of visual information decrease to varying degrees over different ranges. These ranges are visually represented in ITAS using different shades of green in the visible area of the field of view, helping the user make better decisions and have more control over the reliability of what they see.

3.3 Controls

Movement – Instead of the basic dash command, ITAS has a run command which repeatedly sends dash commands as long as the left mouse button is held down. The speed of the running changes dynamically as the mouse pointer is moved closer or farther away from the player representation, allowing for smooth acceleration and deceleration.

Our testers had a hard time turning using the simple RCSS Turn command, as it involves an instant change of orientation in the field. This is analogous of closing one's eyes, turning and then opening them again. One needs to reestablish what it is they are looking at and where they are oriented, as the view of the environment is no longer persistent. This is even more confusing in the alien environment of the simulator. To fix this, we added a panning command. Panning is gradual turning involving many small turns, while the right mouse button is held down. The size of these turns (or the rate of panning) changes dynamically relative to the angle of the mouse pointer to the centre of the field of vision. This allows for a more persistent and less confusing world view.

ITAS also combines running and panning to allow for otherwise unachievable smooth movement. It took our testers a little time to get used to this feature, but was found to be incredibly useful once mastered. This feature allows players to easily move in circles, arcs and figure-8s, something that was very difficult to do in ITAS-basic.

Kick – Hitting the spacebar performs a kick in the direction of the mouse pointer and with a power relative to the distance of the mouse pointer from the on screen representation of the player.

Findball – This feature allows the player to locate and face the ball on the field with a single button press. This was implemented as a necessary component to other advanced features, rather than a stand-alone command.

Chaseball/Intercept – This command utilizes the Findball command to track the ball and the run command to reach it. This is a useful command for intercepting the ball or tackling. It's also a main component of the Dribble command, which is currently under development. It is initiated by holding the 'c' key and aborted upon its release.

Other ITAS commands include shoot, turn-neck, and view adjustment.

3.4 Performance and Testing

Using a set of drills which gauge performance of the basic skills required to perform the key tasks identified in **Fig. 1**, ITAS was continuously evaluated by a test group of soccer players that also have some computer experience. This verification process allows for iterative development of the different features and ensures the involvement of target users throughout development. These drills also serve as a great way to introduce new players to the ITAS interface. The set of drills includes simple drills that each focus on a specific action, such as movement or kicking. In the movement drills, for example, the player was timed while repeatedly traversing one of a set of predefined paths (circle, zigzag, figure-8). Other more complex drills required coordination of several different actions and communication with other players to perform a more complex task. These included passing drills, 'monkey-in-the-middle' and set plays such as cross-ins. Performance on these was gauged by the number of successful attempts at the task and the time taken to complete it. Results from the latest round of tests comparing drill performance between ITAS-basic and ITAS-advanced showed improved performance for part of the movement drills (particularly the figure-8s) and significant improvement for the complex task drills.

Although some games against agent teams have been played, ITAS isn't yet ready for full game testing against stronger RoboCup teams since two important features are still under development, namely Dribbling and Kick-to (which are discussed in the next section).

3.5 What's Next?

Completing ITAS's planned features is the next step. These planned features include Dribbling, Kick-to (which is a command for kicking the ball so as to end its movement within the vicinity of a target location – position of mouse click) and a symbolic

team communication tool. Also planned is an online platform for launching RCSS games with cooperating or competing ITAS players and the option of adding agent team players to play with and/or against.

Potential directions for ITAS once initially planned features have all been implemented include developing a hybrid agent (one that can act intelligently as well as accept real-time commands from a user), modifying ITAS to be used as a remote control interface for a RoboCup robot and experimenting with new user input hardware variants to the current mouse and keyboard setup.

4 Related Work

Another project that seems to be taking on the reverse RoboCup challenge is Nishino's OZ-RP (OZ-Real Players) from Japan. While both attempting to solve the same problem, ITAS and OZ-RP differ in their approach to the solution with regards to the display of the player's environment and the player control model.

Instead of limiting the user to the soccer player's perspective, OZ-RP fits incoming data into a global model and provides the user with a full field view. OZ-RP players also send head turn commands periodically so as to have 360° vision. These features have the great advantage of raising the user's positional awareness on the field. ITAS, on the other hand, aims to provide the user with a player oriented view which, while initially takes some getting used to, is a closer representation of a soccer player's experience. While OZ-RP's approach focuses on maximizing the player's performance through providing them with a complete world model, ITAS focuses more on trying to faithfully model the soccer playing experience relative to the player. Each approach has its merits. OZ-RP's approach allows the player to concentrate more on strategic decisions on a team level. ITAS's modeling of the soccer experience makes for a more up hill challenge, but aims to bring authentic human player-level involvement to the simulation league. The lack of a solid world model in ITAS also makes for a better candidate interface for adaptation for telerobotics and remote control in real world dynamic or unfamiliar environments.

The two systems also have different control models for the player's movement. OZ-RP adopts a rather intuitive 'run to click' approach, combining both the dash and turn commands into one atomic action which involves turning to where the mouse was clicked then running forwards until that point is reached. ITAS gives the user control over both running and turning separately. This makes for a more complex movement control model, but it allows for more fluent movement (e.g. arcs and figure-8s).

5 Summary

The primary purpose of the Reverse RoboCup challenge is to take steps towards introducing a human challenger into the RoboCup Simulation League. The Reverse Challenge also has the potential of opening several new areas of research to the RoboCup community. ITAS is an open source project and welcomes the RoboCup community to partake in and benefit from the RoboCup Reverse Challenge in the many creative ways that are possible.

References

1. D. J. Mayhew, *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design*, San Diego, CA: Academic Press, 1999.
2. J. Nielsen, *Usability Engineering*, San Diego, CA: Academic Press, 1999.
3. David Meister, Thomas P. Enderwick, *Human Factors in System Design, Development and Testing*, Lawrence Erlbaum Assoc; 2001
4. Junji Nishino, <http://www.fs.se.uec.ac.jp/~nishino/ozrp/logs/>, OZ-RP Website, 2002