Teleworkbench: Validating Robot Programs from Simulation to Prototyping with Minirobots (Demonstration)

A. Tanoto
System and Circuit Technology
Heinz Nixdorf Institute
University of Paderborn
33102 Paderborn, Germany
tanoto@hni.uni-paderborn.de

F. Werner, U. Rückert, and H. Li
Cognitronics and Sensor Systems Group, CITEC
Bielefeld University
33615 Bielefeld, Germany
{fwerner,rueckert,hli}@cit-ec.uni-bielefeld.de

ABSTRACT

This paper describes a Demo showing the role of the Teleworkbench in the validation process of a multi-agent system, e.g., a traffic management system. In the Demo, we show the capability of the Teleworkbench in seamlessly bridging the simulation and experimentation with real robots. During experiments, important information is logged for analysis purpose. Additionally, a graphical user interface enables geographically distributed users to perform some levels of interactivity, e.g., watch the video or command the robots.

Categories and Subject Descriptors

I.2.9 [**Computing Methodologies**]: Artificial Intelligence—*Robotics*

General Terms

Measurements, Experimentation

Keywords

multi-robot system, multi-agent system, robotics simulation, robotics experiments, telerobotics

1. INTRODUCTION

One of the challenging aspects in the development of multi-agent systems is their validation in real environment. For this purpose, robots are widely used as test platforms as they can interact with and change the environment. However, performing experiments with real robots is considerably tedious. It is a repetitive process consisting of several steps: *setup*, *execution*, *data logging*, *monitoring*, and *analysis*. Moreover, it also requires a lot of resources especially in the case of experiments involving many robots.

We have designed a system that can ease the tasks of conducting experiments with single or multi minirobots, called the Teleworkbench [4]. The aim of the system is to provide a standard environment in which users geographically distributed can test and validate their algorithms and programs

using real robots. As experiments run in a standardized environment, we can easily compare the results.

This paper describes a Demo showing the Teleworkbench as a platform that can assist users to conduct experiments for validating their multi-agent system using real robots. Six features that the Teleworkbench offers are: (i) *integration with robot simulator using a commonly known robot programming framework called Player/Stage* [2, 1], (ii) *support remote-download of user-defined robot programs*, (iii) *automatic environment building*, (iv) *data logging during experiment*, (v) *robot tracking upto sixty-four robots*, and (vi) *a visualization tool for experiment analysis*.

The scenario used in the Demo is a traffic management system involving many agents (see Figure 1): *Trafficlight Controller* (**TC**), *Blackboard* (**BB**), and *Robot Controller* (**RC**). The TC agent is responsible for controlling a set of traffic-lights at one location that requires traffic management, namely a *crossing*. In the current implementation, only one direction at a crossing can have a green light and there is no communication among TCs. TCs update the status of all traffic-lights via a *topic* at the BB in a *publish-subscribe* fashion. Any agent which needs the status of a specific traffic-light can subscribe to that particular topic. The RC agent is responsible for controlling a vehicle implemented on a minirobot Khepera III. Each vehicle has its specific route that may go through one or more crossings. The RC periodically updates the position of the vehicle and if the controlled vehicle is near to one crossing, it inquires BB for the status of the traffic-light. Accordingly, it will
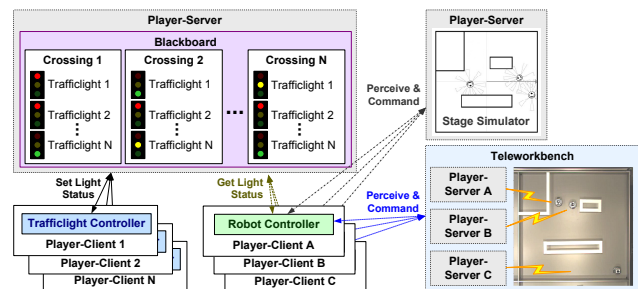

Figure 1: The block diagram of the traffic management system validated both on the simulator and the Teleworkbench.
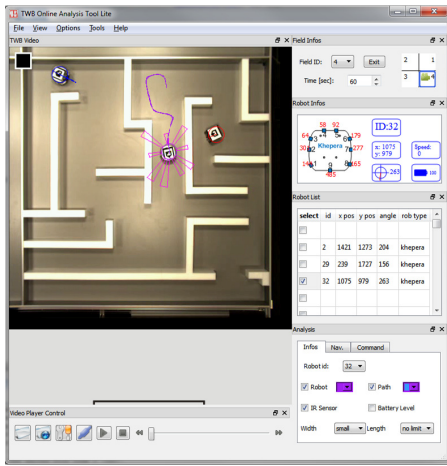
Figure 2: The GUI for online analysis tool.

Figure 3: The diagram showing the general system architecture of the Teleworkbench system.

command the vehicle to stop if the traffic-light is red or otherwise to continue following the route.

With the Teleworkbench, the validation process can be done seamlessly, from the simulator to the real environment. At first, a user can test the developed algorithm in a robot simulator. Afterwards, s/he can log in to the website and set up an experiment. During setup, some parameters of the experiment can be defined, e.g., the model of environment, the experiment duration, and the number of robots. When the experiment is set and ready, the Teleworkbench will first read the defined environment model and translate it to the real environment by using the gripper module. Afterwards, the uploaded programs are deployed and executed. There are two possible deployment platforms for the robot programs: PCs or robots. During experiments, the communicated messages among agents are logged and can be retrieved after the end of the experiment. At the same time, users can also observe the experiment using the developed graphical user interface (GUI) that can display the streamed video overlaid by some robot information such as robot symbol, robot path, and sensor information (see Figure 2).

2. SYSTEM DESCRIPTION

The modular and distributed system architecture of the Teleworkbench (**TWB**) is shown in Figure 3. Earlier papers [4, 5] describe the system in details. The following are short descriptions of some main components.

The TWB comprises a main experiment field of 3.6×3.6m that is partitionable into four sub-fields. Thus, up to four experiments can run in parallel. A gripper module with four degrees of freedom (3 translational and one rotational) allows automatic environment setup by placing plastic blocks or robots at predefined locations and orientations. Three different robotic platforms are currently used on the Teleworkbench: *Khepera II*, *Khepera III* from *K-Team Corporation* and the *BeBot* [3]. A 6-bit barcode-like marker is attached on top of each robot for position and orientation detection as well as for identification up to 64 robots. Five Prosilica GE1050 CCD cameras with a resolution of 1024 x 1024 pixels are mounted above the experiment field, four of which monitor the sub-fields. Each camera is connected to a video server that processes the video data to provide the
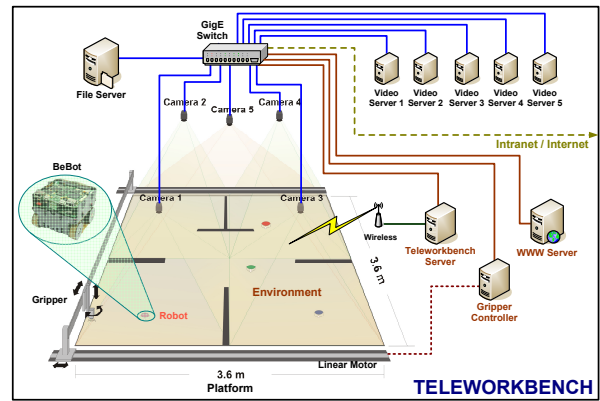
GPS-like position and orientation information of the robots as well as to record and stream the video. A server is responsible for the experiment scheduling and execution. Moreover, the server handles the message passing among robots via Bluetooth and WLAN. Another server is assigned as the intermediary between users and the TWB. A website is provided to enable users to perform different activities, e.g. set-up and execute experiments, retrieve experiment data, or watch live-video. A file server is deployed to store all data that accumulates during experiments that can be used for evaluation and analysis purpose. Additionally, an application programming interface (API) is provided to support users in developing a program that can interact with the robots or the TWB.

3. ACKNOWLEDGEMENTS

4. REFERENCES

[1] T. H. Collett, B. A. MacDonald, and B. P. Gerkey. Player 2.0: Toward a practical robot programming framework. In *Proc. of the Australasian Conf. on Robotics and Automation (ACRA)*, 2005.

[2] B. P. Gerkey, R. T. Vaughan, and A. Howard. The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. In *Proc. of the ICAR 2003*, pages 317–323, 2003.

[3] S. Herbrechtsmeier, U. Witkowski, and U. Rückert. Bebot: A modular mobile miniature robot platform supporting hardware reconfiguration and multi-standard communication. In *Proc. of the FIRA RoboWorld Congress 2009*, pages 346–356, 2009.

[4] A. Tanoto, U. Rückert, and U. Witkowski. Teleworkbench: A teleoperated platform for experiments in multi-robotics. In *Web-Based Control and Robotics Education*, volume 38, chapter 12, pages 287–316. Springer Verlag, 2009.

[5] F. Werner, U. Rückert, A. Tanoto, and J. Welzel. The Teleworkbench - a platform for performing and comparing experiments in robot navigation. In *Proc. of the Workshop on The Role of Experiments in Robotics Research at ICRA 2010*, May 2010.