Sensor based one-way communication in multiple mobile robot systems: an experiment

Ida Hönigmann*†, Manuel Eiwen*, Matthias Guzmits*, Cornelius Kahofer*, Peter Kain* and Christoph Schnabl*
*Technical Secondary College
Department of Computer Science
2700 Wiener Neustadt, Austria
†Corresponding author Email: i15013@student.htlwrn.ac.at

***Abstract*—Sensors provide an easy and readily available way of communication in multiple mobile robot systems usually used in Botball® competition. In this work we compare four different ways of communication, including touch sensor, distance sensor, and light sensor. We present an experimental setup and discuss its results. It turns out that using a light sensor for communication outperforms all other sensor based communication methods, both in setup time and in terms of robustness and reliability. As a further advantage, a light sensor is a standard component in Botball® robots and hence does not require additional hardware.**

I. Introduction

Some tasks are too complex or even impossible to accomplish by a single robot. Consider, e.g., two sub-tasks to be carried out at the same time at different locations far apart. In this case using multiple robots can be a solution.

When more than one robot is used to complete a given task, they have to be coordinated. A key aspect of coordinating robots is to enable a communication between them. While most of the time wireless communication is the most accurate and cheapest way, it sometimes can not be used. In this case, sensor based communication is a good alternative. In the present work, we evaluate its effectiveness in an experimental setup.

This paper is structured as follows: Section II summarizes existing literature on sensor based communication between multiple mobile robot systems. Section III lists various ways of communication using sensors. In section IV-A we describe the experimental setup. In section IV-B we list the robots used in the experiment. A description of the experiment can be found in section IV-C. We continue with a summary and a discussion of the results in section V. We conclude this paper with section VI repeating the most important results.

II. Study of Literature

Yan et al. discuss different problems which can be solved with multiple mobile robot systems (MMRSs)[5]. They define communication as a mode of interaction between robots. This interaction enables the robots to share information about position, sensor data, intentions and actions with others in the system. Communication can be classified into three types including: interaction via the environment, interaction via sensing and interaction via explicit communications. Another classification differentiates indirect and direct communication.

For direct communication some sort of communication module has to be mounted to the robot. For various examples see [5]. In indirect communication robots get information from other robots in the system through the environment. In order for robots to notice changes in the environment they rely on sensor data. For example, robots can communicate by detecting each other or collecting items dropped by other robots in the environment. This kind of communication is imitated by the collective behavior of bees and ants. E.g., Yamada et al. describe an adaptive action selection method without explicit communication for dynamic multi-robot box-pushing [4]. Another example is described by Kube and Bonabeau in [1]: their robots mimic ants, which have to cooperate to move prey too large to be transported by a single individual.

According to Yan et al. the use of direct communication can ensure the accuracy of the information to be exchanged between robots. However, this kind of communication is not expandable to a vast amount of robots, since it may cause a decrease in system performance. This has been studied by Rybski et al. in [3]. A solution to this problem has been proposed by Rekleitis et al. in [2]. Their robots communicate only when they are within line of sight of each other.

We, however, investigate on different indirect communication approaches in this paper, which can be used for cases that allow no direct communication.

III. Ways of Indirect Communication

Let us consider indirect communication between two robots of an MMRS. We can distinguish between synchronous and asynchronous indirect communication. Let us start with synchronous indirect communication, also referred to as timing.

A. Timing

An often used method of coordinating multiple robots is to time their actions. In this method, all robots have synchronized timers. The actions of each robot happen according to a fixed temporal schedule. The given task is accomplished without the robots knowing of each other.

This method is inexpensive as there is no need to buy additional sensors and is relatively easy to implement. However, there are some drawbacks: There must be an exact temporal model of both the robots and the environment to ensure successful performance of the MMRS. In highly dynamic real world scenarios this precondition is hardly ever met. Even if so, a lot of testing is required. On the other hand, this mode of indirect communication can easily be extended to more than two robots, if all robots have synchronized timers.

B. Touch

Indirect communication based on physical contact between robots requires a touch sensor, mounted to at least one of the robots. Other robots can trigger actions or pass on information when bumping onto the sensor.

Using a touch sensor as means of communication will make the MMRS more flexible with respect to a changing environment. The cost of the sensor is comparably low and implementation relatively straight forward. However, mounting and triggering the sensor can turn out as a challenge. The bigger the sensor, the easier triggering, but the harder mounting it to the robot, and the other way around.

C. Distance

Using distance sensors as a way of indirect communication requires dedicated hardware triggering a signal as soon as a fellow robot approaches.

Different from using a touch sensor, communication can happen at a distance between the communication partners. However, one can usually not differentiate between friend and foe: in an MMRS not every robot within reach of the distance sensor is a desired communication partner. Even if the robot within the range of the sensor is the right partner, communication might be unreliable. Furthermore, information about the direction of movement is missing or hard to deduce.

Distance sensors are comparatively expensive and require calibration before use.

D. Light

Light sensor based communication requires a light source on one robot and a light sensor on an other. Prerequisite to this mode of communication is alignment of source and sensor, otherwise communication will fail. In theory, the distance between the robots can be large, however, with increasing distance the alignment problem becomes more acute. In a complex, dynamic environment exact positioning of robots is a challenge and additional light sources pose additional problems. Yet, light sources and sensors are very cheap and often readily available.

IV. Experiment

A. Task and Environment

First of all, we describe the task and the environment. The environment is a flat square table ($8' \times 8'$) whose surface is a pebble grain white fiberglass reinforced plastic panel. The table is lit by fluorescent lamps located on the ceiling of the room. On the table robots A and B are placed at a distance of $20''$. Robot B's task is to move $2'' \times 2'' \times 2''$ colored foam blocks from its starting position to an end position, where robot A takes over the block. Robot A's task is to use its robotic arm to move the foam block to the target area, see Fig. 1 and Fig. 2. The crucial part of the experiment is the communication which ensures correct take over of the foam cube. Successful completion of the task is achieved as soon as the cube touches the target area. For a description of the realignment block see Section IV-C.

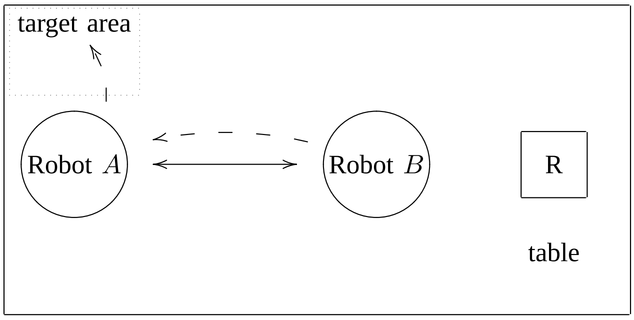

Fig. 1. Schematic figure of the environment. Solid arrows indicate movement of robots. Dashed arrows indicate movement of foam blocks. R is a realignment block. For a description of the realignment block see Section IV-C.

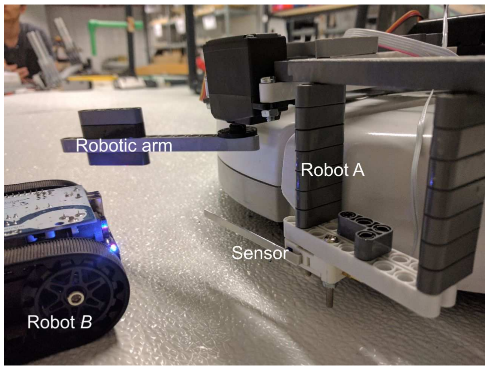

Fig. 2. Robot A in touch sensor configuration after successfully handing over the foam cube from robot B.

B. Materials

Robot B is an Arduino®-controllable Zumo robot sized $3.86'' \times 3.86'' \times 1.54''$. It moves with two gearmotors coupled to a pair of silicone tracks and is equipped with a stainless steel bulldozer-style blade. The Arduino® controller has a ATmega328P 16MHz microprocessor and 2kB RAM. It is programmable using C.

Robot A is an iRobot® Create with a diameter of $13.9''$and a height of $3.6''$. A robotic arm is attached to a servo to take over the foam cube. Robot A is controlled by a KIPR Link robot controller, which is connected to one of several sensors at a time. The controller has a ARMv5te 800MHz microprocessor and 18 MB RAM. It is programmable in C.

C. Implementation

The overall experiment comprises four sub-experiments, differing in the mode of communication between the robots.

In all sub-experiments, ten foam cubes have to be transported from robot B's starting position, placed by a member of the team, to the target area. The entire sub-experiment runs fully automatically, with placement of foam blocks being the only human intervention. As we focus on communication between robots A and B we want to get rid of any effects due to track instabilities. Therefore, we realign robot B upon each return to its starting position by bumping against a realignment block with its blade. Furthermore, we avoid any inconsistencies caused by turning. Hence, robot B just moves back and forth always facing the same direction.

We now describe each sub-experiment in detail.

1) Sub-experiment Timing: When timing is used for coordinated cooperation of robots, they have to be synchronized at some point t_0 in time. In this sub-experiment we manually start the robots synchronously after a count-down. Robot B is equipped with a foam cube and starts moving immediately. Robot A stays idle. After six seconds robot B has arrived and waits for robot A to move its robotic arm to take over the foam cube and move it to the target area. Four seconds after arrival the robotic arm is reset to its initial position. Robot B moves back to its starting position where another foam block is placed on it. For statistic purposes, these actions are repeated ten times.

2) Sub-experiment Touch: Here, we use a KIPR Long Lever sensor as a touch sensor on robot A to trigger the servo, which moves the robotic arm.

Robot B moves until it triggers the touch sensor, then moves back to an optimal position for take over and waits there for two seconds. After the touch sensor was triggered, robot A waits for one second, then the robotic arm moves the foam block to the target area. The robotic arm is reset to its initial position after five seconds. Robot B moves all the way back to its starting position where another foam block is placed on it. These actions are repeated ten times.

3) Sub-experiment Distance: In this sub-experiment, a KIPR Large IR (Top Hat) is used as a distance sensor mounted on robot A.

Robot B starts moving until it triggers the distance sensor and waits for robot A to register robot Bs arrival and take over the foam cube by moving the robotic arm. After four seconds the servo moves back to its initial position and robot B moves back to its starting position to get another foam block. These actions are repeated ten times.

4) Sub-experiment Light: For this sub-experiment robot A was equipped with a light sensor pointing down onto the table from a position above the servo. A Xiaomi 4S, with the flash light turned on, was mounted upside down on robot B as a light source.

Upon starting Robot B it moves until triggering the light sensor. Robot A takes over the foam cube by moving its robotic arm. The servo moves back to its starting position after four seconds. Robot B moves back to its starting position and gets another foam block. These actions are repeated ten times.

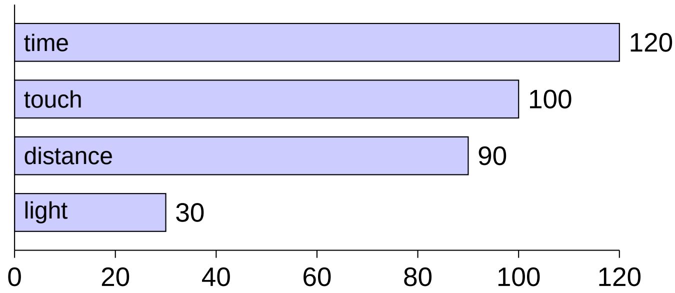

Fig. 3. Setup times (in minutes).

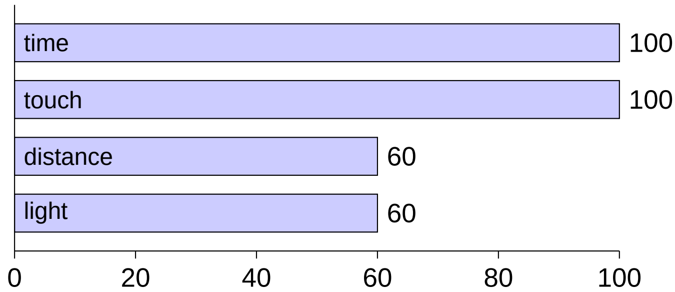

Fig. 4. Timing of the sub-experiment (in seconds).

TABLE I
FAILED TAKE OVERS OF FOAM BLOCKS.

	time	touch	distance	light
first try	0%	50%	60%	0%
final try	0%	0%	30%	0%

V. RESULTS AND DISCUSSION

Each sub-experiment required a different amount of time for setup, i.e., for fine-tuning parameters. The majority of the time was used for modifying parameters of the temporal domain, followed by sensitivity parameters. Sub-experiment *Timing* took longest, which may also arise from the fact, that this was our first sub-experiment. Sub-experiment Light took the least amount of time. For details see Fig. 3.

We measured the time from starting the robots until robot Bs final return to its starting position, i.e., all ten foam blocks have been taken over by robot A. For details of the results, see Fig. 4.

We considered a run of a sub-experiment as failed, if the foam block was not properly taken over by robot A. For details of the failed attempts of take over, see Table I.

A. Setup

It turned out that the setup of the sub-experiment *Timing* took longest. Here we had to adjust idle times between individual actions of robots A and B based on a large number of trials. We extended the time interval between the arrival of robot B and it moving back to its starting position, in order to make sure robot A has enough time to take over the foam cube. Additionally, robot B moves back longer than it moves toward robot A to ensure it realigns at the dedicated block.

The most difficult part of the setup for sub-experiment *Touch* was mounting the sensor and making sure robot B triggered it correctly without losing its alignment. For more details on the position of the sensor see Fig. 2. Too short movement of robot B will not trigger the sensor. On the other hand, if robot B bumps into the sensor with high velocity, it turns up to 45 degrees and thus does not hit the realignment block when returning to its starting position. Once off track, all further attempts to deliver the foam block fail, which was the reason for five fails in the first attempt (see Table I). To make the track of robot B more stable, we adjusted its movement. Additionally robot B moves back slightly in order to ensure robot A is able to take over the foam block correctly.

Unfortunately the sub-experiment *Distance* turned out to have the most difficult setup. In the first try, six foam block deliveries failed. We tried to modify the distance threshold of the sensor and the movement of robot B, but could not achieve a result better than three fails, see Table I.

Unexpectedly, using the light sensor for communication is surprisingly easy in setup. Although using an ad-hoc approach for the light source, calibration of the sensor was simple. We did not have any fails in the first trial.

In conclusion, light turned out as easiest concerning setup. However, all sub-experiments after the sub-experiment *Timing* profited from the experience and data we collected about movement of robot B.

B. Timing

When using time for synchronization of robot A and B we had to introduce idle times to ensure proper take over of the foam cubes. This makes the sub-experiment *Timing* slow in comparison to sub-experiments *Distance* and *Light*. The same is true for sub-experiment *Touch*. Here extra time is required for the extra movement described in Section V-A.

C. Fails

The main source of failure was robot B not being in a position where it triggers the sensor, or robot A not being able to take over the foam block correctly. The more foam blocks have to be delivered, the more crucial is the presence of means to ensure alignment of the track of robot B, in our case the realignment block. Without it, no sub-experiment would have succeeded.

VI. Conclusion

In conclusion using a light sensor as a mean of communication between robots turned out to be not only easy in setup, but also effective. Moreover, light sensors and light sources are cheap and readily available in many robotic tool kits. In Botball®, a light sensor is used to start the robots, therefore, the only component additionally required for light based communication is a light source on one or more of the communication partners.

In our experiments we used the simplest way of communication, i.e., just signaling “I am here!”. A more complex communication pattern can be implemented in a straight forward way, e.g. using some code, which we will examine in future work.

Acknowledgments

The authors would like to thank Dr. Michael Stifter for his support throughout the work on this paper.

References

[1] C.R. Kube and E. Bonabeau. Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, 30(1):85–101, 2000.

[2] I. Rekleitis, V. Lee-Shue, A. Peng, and H. Choset. Limited communication, multi-robot team based coverage. *Proceedings - IEEE International Conference on Robotics and Automation*, 2004(4):3462–3468, 2004.

[3] P.E. Rybski, S.A. Stoeter, M. Gini, D.F. Hougen, and N.P. Papanikolopoulos. Performance of a distributed robotic system using shared communications channels. *IEEE Transactions on Robotics and Automation*, 18(5):713–727, 2002.

[4] S. Yamada and J. Saito. Adaptive action selection without explicit communication for multirobot box-pushing. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 31(3):398–404, Aug 2001.

[5] Z. Yan, N. Jouandeau, and A.A. Cherif. A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems*, 10, 2013.