A Mobile Robot for EUROBOT Mars Challenge

Tomáš Krajník, Jan Chudoba, and Ondřej Fišer

The Gerstner Laboratory for Intelligent Decision Making and Control
Department of Cybernetics, Faculty of Electrical Engineering
Czech Technical University in Prague
{tkrajnik,chudoba}@labe.felk.cvut.cz, fisero1@fel.cvut.cz

Abstract. The aim of this paper is to present an intelligent autonomous robot for competition EUROBOT 08. In this "Mission to Mars", two robots attempt to gather, sort and dump objects scattered on a planar rectangular play-field. This paper descripts robot hardware, i.e. electromechanics of drive, chassis and extraction mechanism, and software, i.e. localization, collision avoidance, motion control and planning algorithms. The experience gained by participating on both national and international round is evaluated.

1 Introduction

1.1 EUROBOT

The EUROBOT[1] association holds amateur robotics open contests, organized either in student projects, in independent clubs, or in educational projects. It started in 1998 in France as the national robotics cup. Nowadays, participants are not required to be Europeans, the only restriction is age: only one team member may be over 30 years old. This team member is allowed to advise and lead, but should not participate directly during implementation.

In a typical EUROBOT match, two autonomous mobile robots compete on a planar field with rectangular shape. Robots are limited in size - the maximum height is 0.35 m and their convex hull circumference must not exceed 1.2 m. After start, the robot can deploy its devices and extend its perimeter up to 1.4 m. Match duration is 90 seconds.

Competition rules change every year. This prevents from the situation in other leagues (e.g. FIRA[2]), where new teams have disadvantage compared to veteran participants and are selbom able to reach a good rank. The challenge of this year is called "Mission to Mars".

In this challenge, two opposing robots pick up floorball balls representing biological samples or frozen regolith and put them in three separate containers. Each successfully placed ball adds a certain score depending on its color, robot team color and container type. Special score bonus is added, if balls placed in a standard container form a certain pattern, representing biological samples encircled by ice. Detailed and precise rules description can be found at [3].

A. Gottscheber et al. (Eds.): EUROBOT 2008, CCIS 33, pp. 107–118, 2009.

1.2 Project Goal

The main objective of participation in this competition is educational. The competition proved to be an excellent opportunity for a group of students to design and manufacture a functional device able to solve the assigned task. This is a big difference compared to common educational project, when students solve only particular tasks and the ”right” solution is often known in advance. Moreover, students have to solve problems related to different areas of engineering: mechanics, electronics, computer science and artificial intelligence. The experience gained by working in a team under a time pressure is also invaluable.

1.3 Paper Structure

The paper is organized as follows: It begins with introduction. Next division is concerned with an overview of robot concept and desired activity. This will be succeeded by sections describing robot motion, extraction, localization, collision avoidance and planning subsystems. In the next section, we will evaluate the progress of the robot on both national and international round, compare its behaviour to other concepts. Subsequent chapter will conclude about robot ability to fulfill the desired task. Conclusion will be followed by acknowledgments and references.

2 Robot Overview

Our robot hardware consists of an alluminium chassis, two-wheel drive, ball extractor, power subsystem, two control units (drive and extractor), onboard PC, optical sensors, mechanical bumpers and laser rangefinder. We decided to use Ångström Linux operating system to be installed on the onboard computer. The software can be divided into localization, planning, motion control, collision avoidance and waste handling modules. Onboard PC software is programmed in C/C++, algorithms of the extractor and motion control boards have been realized in C.

Skeleton of the robot is formed by interlocked X-shaped alluminium beams. These proved to be firm and stable enough to support robot devices and protect them reliably during collisions. The remaining parts of the chassis is made of alluminium or cuprexit plates, which support sensors, electronics and extraction mechanism.

Drive subsystem has its own control unit capable of positioning both wheels independently. Motion control is based on a set of predefined standard movements, (e.g. from the start position to the vertical dispenser), reactive movement routines (e.g. moving along the standard container, docking) and on a position regulator. Three bumpers (two in the front and one in the back) serve as primary collision detection sensors. Aside from signals from these bumpers, motion control board monitors start, emergency stop and other buttons.

Localization module is based on odometry and URG-4LX laser rangefinder [4] located at the front of the robot. Two cooled containers at field corners serve as

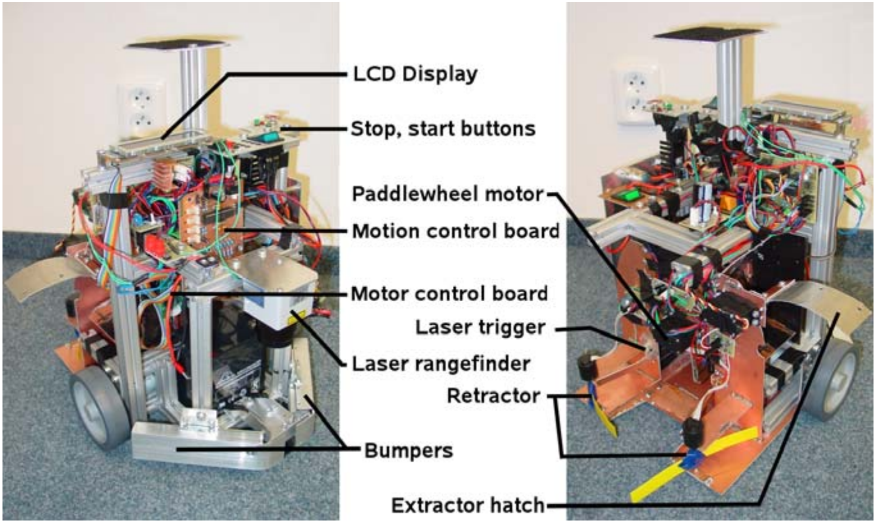

Fig. 1. Robot overview

easily detectable landmarks. Opponent robot position is calculated as well and taken into account in motion planning.

Extraction mechanism is a five vane paddlewheel with its own controller board, ball acquisition mechanism and hatch.

Planning subsystem implements a predefined Petri net [5], where places represent functions of individual software modules and transitions describe return values of these functions. In our approach, a place in a Petri net represents some functionality of robot subsystems (e.g. extractor: eject sample) mentioned in previous sections.

Collision avoidance uses A^*[10] algorithm operating on a grid map in combination with reactive avoidance.

At the beginning of the match, the robot attempts to pick up two colored balls from the vertical dispenser. During the extraction, its laser rangefinder provides data on opponent movement and reports it to the planning module. Planning module then decides, whether to pick up three white balls from ground plane or from the second vertical dispenser. After picking up three white balls, the robot manipulator is full and the robot heads towards the standard container while avoiding the opponent. During this movement, the robot calculates a sequence of manipulator actions in order to maximize the score while minimizing dumping time. Preparatory actions, i.e. preparing the first dumped sample at the hatch, are taken during docking. During the dumping procedure, laser rangefinder data are processed in order to detect adversary position. Opponent movement is analyzed to estimate the next course of action, i.e. picking from ground or extraction from dispensers.

During forward movement, laser rangefinder serves as a collision avoidance sensor. However it cannot provide opponent position during ball pickup, because

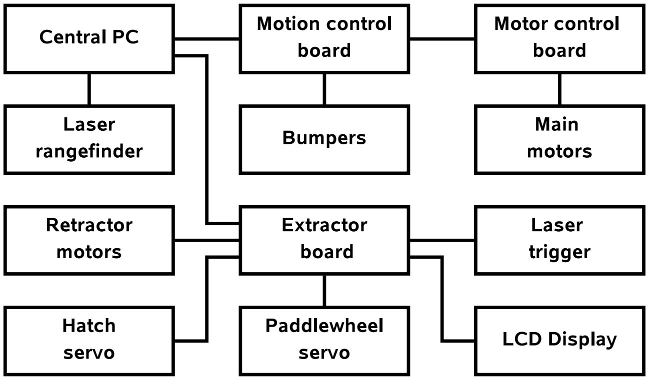

Fig. 2. System scheme

it is placed on the opposite side of the robot than the extraction mechanism. During ball pickup from the ground, the robot will have to rely on its bumpers to detect collisions. If the opponent is detected on collision course (or a collision is detected), our robot will wait if the opponent is going to clear the way. In case the path is not clear after a small moment, planning module will choose an alternative target. If no alternative target would exist, collision avoidance module tries to devise an alternative path to current destination.

3 Motion

The motion subsystem is in charge of moving the robot around the field while preventing damage done to our or opponent robot. It consists of two wheels with DC motors, motor control board, motion control board and mechanical bumpers. We have decided to use EGM30 12V/4.5W motors equipped with encoders and a 30:1 reduction gearboxes. The MD23 dual motor driver is able to control either speed or position of both motors independently. It can also report the current state of motor IRC counters. Acceleration and speed are monitored and limited by this controller to prevent wheel slipping and subsequent odometric imprecision. This motor controller is connected to the motion control board via an I^2C bus. The motion control board handles signals from bumpers and issues commands to the motor controller.

The motion control board is based on an Atmega microcontroller. It receives orders from onboard PC via an RS232 interface and translates them to commands for motor controller. In addition, it processes signals from three bumpers during the match and buttons used during robot starting procedure. These buttons are "system test", "change color", "change strategy" and "start". The "start" button is connected paralelly with a start cable connector.

Motion control unit stores latest information from bumpers and when requested, it transmits these data to the onboard PC by RS232 interface. Signal "bouncing" during the switch transitions is filtered by first order RC filters.

Collision detection, realized by three mechanical bumpers prevents damage by a robot attempting to move in blocked direction. In such situations, rotating wheels erode surface of the field - robot, which damages field gets disqualified. Detected collision also indicates, that odometric information is no longer reliable source of position information. Each frontal bumper is composed of two microswitches covered with a metal plate. Rear bumper is realized by a microswitch and serves as a indicator, that the robot has successfully docked to a dispenser.

Three basic operating modes are realized by the motion subsystem. The first one is a set of predefined sequences of commands for the drive controller. These are simple, short moves, during which we do not expect the opponent robot to interfere. A typical example is the first move of the match, i.e. moving from start area to first vertical dispenser. Second operating mode consists of a set of reactive behaviours. These translate sensory (bumper) inputs to simple commands. The third mode realizes "follow the carrot" [6] regulator. In this mode, onboard PC sends robot-relative coordinates of a target position and the robot moves towards it.

4 Extractor

Extractor is a part of robot which is responsible for sample pickup, transport and dump. It occupies almost half of the robot body. It is composed of a five vane paddlewheel, hatch, retractor, two laser triggers, control board and LCD display.

The most important part is the paddlewheel propelled by a DC motor with a IRC sensor providing 90 impulses per revolution. Paddle profile is optimized for efficient and reliable sample storage and dump. The wheel has five slots, each can store one ball. An infra red sensor provides contact-less detection, that the paddlewheel slots are in right position. The retractor is made of two floppy rotating lamellas. Its task is to push balls inside the paddlewheel lowest slot. The first laser trigger, placed behind the retractor, consists of a 5mW laser diode aimed at a photosensitive transistor. This light gate indicates that the sample was pushed inside by the retractor. Second laser trigger is placed inside the paddlewheel and indicates, that a ball was successfully extracted. One slot of the paddlewheel is fitted with a controllable hatch for sample ejection.

Extractor control board is based on an ATmega microcontroller. This microcontroller generates the PWM signal for the paddlewheel, retractor and hatch motors, controls laser trigger and handles LCD display. The controller is interconnected with the onboard PC via an RS232 interface. Apart from printing its own status on the LCD display, it acts as a bridge between the display and onboard PC. Microcontroller is using internal interrupts and timers for better communication and event handling.

The extractor is designed as a fairly independent function module capable of its own actions. Its main task is sample extraction from the field or vertical dispensers. During the sample pickup, retractor lamellas rotate ant push balls in front of the robot inside the paddlewheel slot. Once the ball acquisition is

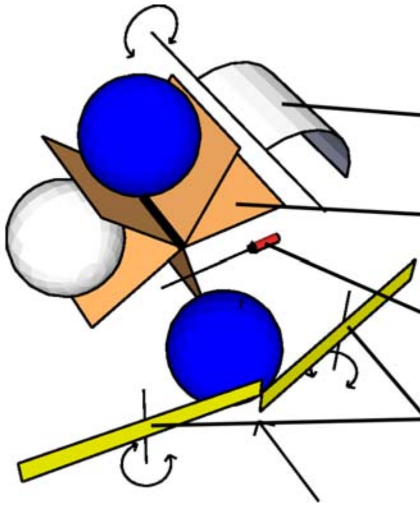

Fig. 3. Extractor scheme

detected by the first laser trigger, the paddlewheel rotates to prepare a free slot for next picked sample. The second laser trigger then checks, if the acquired ball is in correct position. Once the sample extraction is terminated by planning module or the extractor is full, retractor motion is stopped.

To initiate sample dump, the computer announces assumed paddlewheel status, i.e. which slots are occupied by which samples. After that, an optimal ball ejection sequence (e.g. so that the colored and white balls alternate) is planned. The sequence is then realized by rotating the paddlewheel and opening/closing the hatch. The last action status, i.e. success or reason of failure, is reported to onboard PC and printed on LCD display.

5 Localization

The task of the localization system is to estimate position of our and opponent robot on the playfield. To achieve this, we fuse data from odometry, bumpers and laser rangefinder.

Pulses of wheel IRC sensors are first counted by motor controller. These values are regularly read by the motion control board, which translates them to cartesian coordinates and heading. Onboard PC can obtain these data from motion control on request.

Odometry is insufficient, because when robot collides or slips, odometry data lose relevance. Furthermore, knowledge of the opponent robot position have proved to be useful - and this cannot be measured by odometry.

Because of that we use an independent localization sensor, which is based on an URG-04LX laser rangefinder mounted in front part of robot. Its scanning plane is parallel with field plane and is located at 0.16 m above the field level.

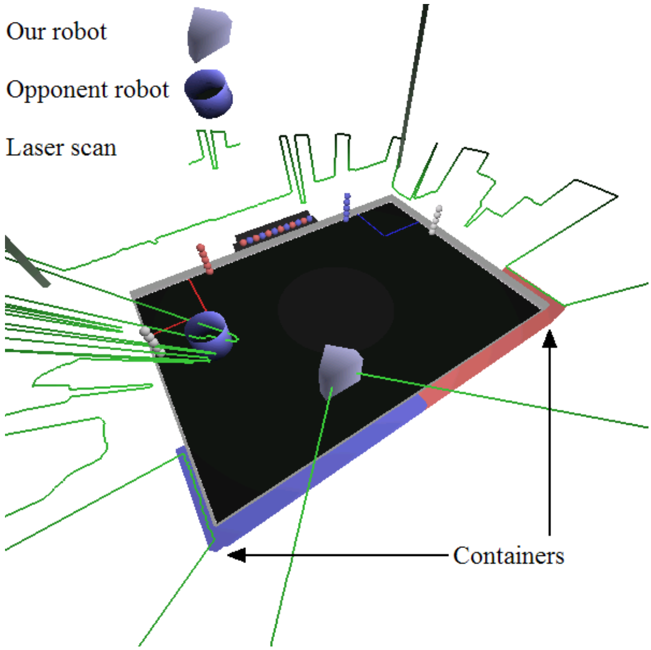

Fig. 4. Scanner data interpretation

Field of view is approximately 250 degrees, because it is slightly limited by the robot itself, sensor range is limited to 4 m, its resolution is up to 1 mm. One complete scan takes approximately 0.1 s and provides the robot with 720 distance measurements.

Localization algorithm works as follows: Laser rangefinder data are requested and motion control board is asked for odometric data. Lines of 0.6 m length, which correspond to cooled containers, are detected in the obtained scan. Cartesian positions of centers of detected lines are then computed.

From known and measured container positions, the pose of our robot is estimated. Most of the time, localization results are ambiguous, a result closest to odometry readings is chosen. This result is reported to motion control board to correct its position information. After computing our robot position, rangefinder measurements are transformed to the field absolute coordinates. Measurement laying inside of the field are segmented and centroid of the largest segment is calculated. This centroid represents opponent robot position.

6 Collision Avoidance

The purpose of collision avoidance module is to adapt robot movement to opponent robot position. In the first and second mode of motion (see section 3) collision avoidance module causes the robot to slow down or pause its movement.

During the "follow the carrot" mode, collision avoidance first checks, whether the desired trajectory intersects opponent position. If trajectory target actually lies inside of opponent position, collision avoidance reports failure and planning module has to devise a solution. If so, an occupancy grid model of the field with is created (cell size is 0.1 m). After that, A^*[10] algorithm is utilized to search the shortest collision free path. The resulting sequence of cells is first converted to a serie of points representing the planned trajectory. The trajectory is then smoothed out and redundant points are erased. The point closest to the robot then serves as an input for "follow the carrot" controller.

7 Planning

Planning is a process of creating a sequence of actions, which we have to apply to a system in order to transfer it from its start state to the goal states. Planning can be realized as search through the space of all possible plans, while looking for the optimal plan according to given optimality criterion. In our case, plan is a series of robot actions with optimality criterion given by game scoring rules. These are based on a number of collected and correctly placed balls during a ninety-second time interval. We do not utilize automatic methods for finding optimal plan, instead of this we try to propose suitable strategy leading to the maximum score. Our team has developed a fixed plan of actions, which should deal with every likely situation (e.g. crash with other robot, incorrectly identified sample). We decided to utilize Petri net [5] model, because it gives clear and effective representation of proposed plan. The Petri net is a bipartite graph consisting of places, transitions and oriented arcs. Every place in a Petri net has a positive integer number of tokens. The projection from a set of places to an integer set is called "marking" and represents state of modeled discreet system. A transition has input and output places. For input place, there exists an arc running from it to the transition and vice versa. If there are enough tokens in every input place of a transition, it can be "fired". During firing, tokens from input places are removed and appear in output places. The number of removed and added tokens is determined by arc weight. The major advantage of Petri nets is the ability to model parallelism and synchronization.

In our approach, a place in a Petri net represents some functionality of the robot subsystems (e.g. extractor: eject sample) mentioned in previous sections. These functions can be thought as simple robot behaviours [8]. If a token is present in a place, corresponding behaviour is activated. There can be more than one behaviour active at the same time, but within one subsystem, only one behaviour is in effect. Possible outcomes of particular behaviour are represented by output transitions (e.g. opponent far, opponent at dispenser). Because of Petri net formalism, we can precisely examine many important properties (deadlock, reversibility, liveness) of our plan. We also measure time needed for each action and use timed Petri nets extension to estimate plan duration. This can give not only quantitative evaluation of our robots ability to win, but also shows functions, which have major impact to plan length. During robot tuning in later stages of development, we can optimize these behaviours to obtain greater speed.

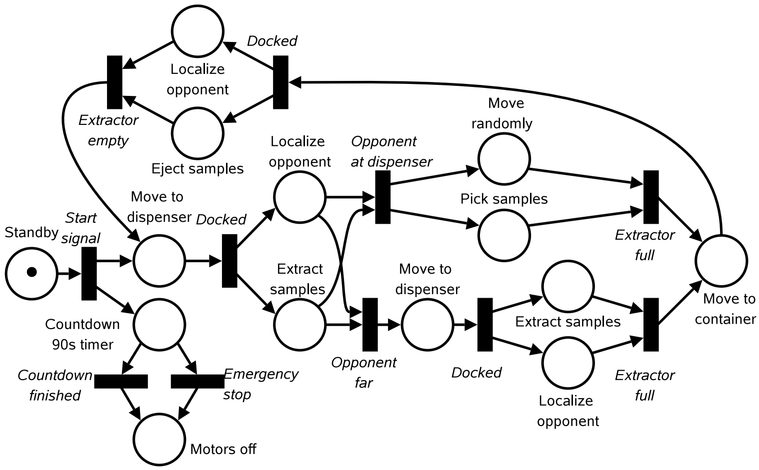

Fig. 5. Intended plan represented by a Petri Net

8 Testing and Competing

The robot was operational approximately seven weeks before the Czech national round. Thanks to other teams, which have build a testing field, we have been able to perform extensive testing. During the test trials, the overall concept has proven to be feasible. However, we have encountered several problems and had to deal with them by changing robot algorithms and hardware. These improvements were completed before the Czech national round and thus we were quite well prepared.

8.1 Testing

Wheel slipping was causing much trouble and we had to lower robot movement acceleration. The slippage problem was supressed, but not eliminated. Laser-based localization had significant problems if there were other planar objects around the field. These objects were often mistaken for containers and the robot lost track of its position. Since the rotation velocity of the URG04LX laser is quite low, laser data obtained during robot movement were deformed. Because of that, the robot had to stop before performing a localization procedure.

When dumping balls to the standard container, the robot position had to be 1-2 cm apart from the field edge. If the robot was closer or farther, balls bounced back to the field. Since the precision of the localization system is lower than 1 cm, the robot had to use its bumpers to detect the container. Balls ejected in the standard container collided with each other and sometimes switched their position. Therefore, we had to slow down the extractor in order to ensure the intended sequence of the balls in the standard container.

The most troublesome operation was extracting balls from vertical dispensers. When the robot did not dock precisely, acquisition of the samples took too much

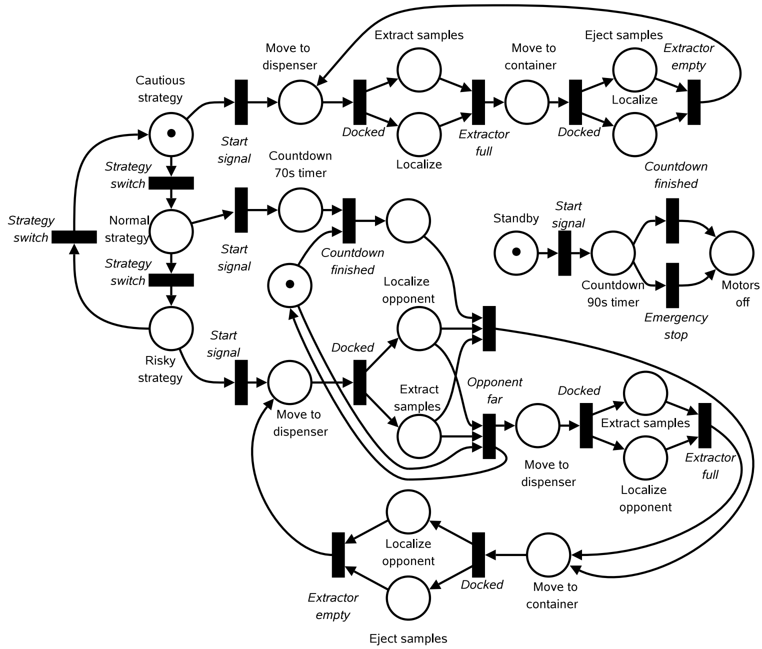

Fig. 6. Plan strategies represented by a Petri Net

time or the extractor failed completely. In order to improve precision, the laser scanner was used to detect dispenser position prior to docking. The rear bumper had to be adjusted to increase docking precision and reliability.

All these modifications improved reliability at the cost of speed. During trials, the duration and success rate of elementary actions were measured. We decided to acquire samples only from dispensers, since picking them up from the field has proven to be ineffective. For docking and movement actions, there was a tradeoff between speed and success rate. Whenever possible, the planning module was allowed to select current action speed and therefore its success rate. Now, the planning module implements three strategies, which are selected by a switch during the startup procedure:

– "Cautious" strategy maximizes the chance to score. All actions are performed slowly, the robot does not attempt to build a sample sequence. The robot extracts five colored balls and dumps them in the standard container. Then, the robot returns to the white sample dispenser, extracts all samples and dumps them as well. This strategy works best against weak teams, which are not likely to score more than 10 points. Maximum achievable score is 15 points. Success rate was approximately 80%.
– "Risky" performs all actions as quickly as possible. The robot attempts to create a full sequence of ten samples. If successful, the score reaches 27 points. Success rate was rather low - less than 20%, but worth of a try against strong teams.

– "Normal" maximizes average score. Action speed selection is based on remaining match time. The robot attempts to form the first sample sequence and then decides, whether to build another sequence or pick up and dump only colored samples. Maximum achievable score is 27 points. However, in 90% of the trials, the remaining time did not allow to form a second sequence. Thus, the most frequent score was 19 points. The approximate success rate is 50%.

8.2 Competition

During the Czech national round, we have tried "Normal" and "Cautious" strategies. The robot scored, but sometimes got lost when moving from the standard container to the dispenser. Later on we noticed, that the field has deteriorated (probably due to high humidity at storage location), which might be the cause of odometric localization error. Most robots suffered from the same problem even more, because their localization was based on odometry only. The robot was able to avoid moving opponents and switch to alternative plan is the opponent obstructed its path. We have maintained a fair score during qualifying rounds and reached the final. In the finals, we won by a fluke and ended up as Czech national champion.

At the international competition at Heidelberg, the robot kept stucking after it docked to the first dispenser. Out of five matches, the robot got stuck three times and did not score at all. However, we won one match with a score of 19 points.

9 Conclusion

Our robot has won the Czech national round and competed successfully at the international round at Heidelberg. However, extensive testing made it too much adjusted to the testing field. Therefore, the robot scored less than expected during competitions. It had problems to extract balls from different dispensers and its odometric localization was not working quite well due to wheel slippage.

Although the aim of this project was to create a robot capable of performing a task given by match rules of EUROBOT 08 competition, its main goal is educational. It should teach students to work as a team, evolve their knowledge in areas which are encompassed in robotics and apply theoretical principles they have learned previously. The robot itself is functional, and efforts, enthusiasm and increasing labor effectivity of team members indicate that educational goal has been achieved as well.

For the EUROBOT 2009, we would like to make modifications to the robot drive and odometric system in order to achieve greater speeds and more reliable odometric localization.

Acknowledgments

Construction of this robot would not be possible without support of Dr. Libor Přeučil, head of IMR, FEE CTU in Prague. We thank EUROBOT organizers, volunteers and team members for the oportunity to participate in the contest. The presented work has been supported by the Ministry of Education of the Czech Republic under program "National research program II" by the project 2C06005.

References

1. http://www.eurobot.org
2. http://www.fira.org
3. http://www.eurobot.org/eng/rules.php
4. http://www.hokuyo-aut.jp/
5. Petri, C.A.: Kommunikation mit Automaten. Ph. D. Thesis. University of Bonn (1962)
6. Barton, M.J.: Controller Development and Implementation for Path Planning and Following in an Autonomous Urban Vehicle. Undergraduate thesis, University of Sydney (November 2001)
7. Duda, R.O., Hart, P.E.: Use of the Hough transformation to detect lines and curves in pictures. Comm. ACM 15(1), 11–15 (1972)
8. Arkin, R.C.: Behavior Based Robotics. MIT Press, Cambridge (1998)
9. Kulich, M.: Lokalizace a tvorba modelu prostředí v inteligentní robotice (in czech). Ph.D. thesis. CTU-FEE, Prague, Dept. of cybernetics (2004)
10. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, pp. 97–104 (2003) ISBN 0-13-790395-2