Proceedings of the International Multiconference on Computer Science and Information Technology pp. 813–816
ISBN 978-83-60810-27-9
ISSN 1896-7094

Methods of Computer-Assisted Manual Control of Wheeled Robots

Viktor Michna, Petr Wagner, Jiri Kotzian
VŠB – Technical University of Ostrava
Department of Measurement and Control, Faculty of Electrical Engineering and Computer Science
17. listopadu 15, Ostrava-Poruba 708 33, Czech Republic
Email: {viktor.michna, petr.wagner, jiri.kotzian}@vsb.cz

Abstract*—This paper deals with possibilities of manual control of wheeled robots. The problem is tested on a robot-soccer application. The manual computer-assisted control module creates an interface between game controller and controlled robot. There are several ways of steering implementation. The simplest is a differential steering. The module currently uses semi-automatic steering which provides a capability to move with the robot as a point. This functionality requires interaction with vision system, usage of Kalman or similar filter and compensators. Next logical step is a fully-assisted steering. This steering system interacts with strategy and motion control module and provides additional functionality such as tracking the ball and shot. The objective is the control of many robots by only one player.

I. Introduction

THE robot-soccer is a popular application that has ability to attract young people and to popularize technical branches on the faculties of mechanical engineering, electrical engineering and computer science. It is also used as a test bed to evaluation and testing of new methods, algorithms and solutions in different technical branches.

Current solution of robot-soccer application consists of the Vision module, the Strategy module (divided into Higher strategy and Motion control), the Wireless transceiver and the Communication module. The Communication module provides interface for distributed data exchange among the other modules of the system [1].

This paper describes the development of computer-assisted Manual control module as the part of the 3rd generation of robot-soccer strategy system for VŠB-TUO robot-soccer team. The module has been developed for two reasons – strategy testing against human opponents and for attraction of young people to study at technical universities.

II. Manual control module interface

The manual control module allows a player to control the robots using connected game controllers or joysticks. It cooperates with other robot-soccer modules so it makes an interface between the game controller and controlled robot.

This work is being solved at the VSB-Technical University of Ostrava, Czech Republic. The project is supported by GAČR project 102/08/1429 and also by the internal grant SP/2010165 „Development and using new methods for design and realization of embedded systems".

A. Wheeled robot

The robot with basic parameters is in the Fig. 1. The relationship between velocities, angular velocity and radius of rotation is defined [2]

$$v=\frac{vl+vr}{2}\ ,\ \omega=\frac{v}{r}=\frac{vr-vl}{d} \qquad (1)$$

B. Game controller

After testing several types of controllers, the gamepad was chosen. It allows quick control of associated values.

The selected gamepad (Fig. 2) has

- 2 analog sticks
- 8-way POV (point of view) hat
- 12 buttons.

These controls can be utilized to comfortably support both left-hand and right-hand mode.

The position of main stick has coordinates $jx, jy \in \langle -1;1 \rangle$ in axis x, y respectively and button values $b_0, b_1, .. b_N \in \{0;1\}$.

III. Analysis of control methods

There are several ways of how manually control the robot – differential steering, semi-automatic steering, fully-assisted steering and group-assisted steering, from the simplest one to the most advanced.

A. Differential steering

The first generation of manual control used the simple differential steering. Its principle is based on direct

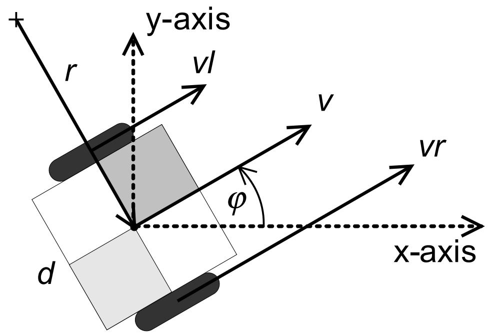

Fig 1. Basic motion parameters of the robot.

Fig 2. Gamepad controller.

conversion of desired velocity v_d and angular velocity ω_d of robot into the movement of stick

$$jx=\frac{\omega_d}{\omega_{max}}\ ,\ \ jy=\frac{v_d}{v_{max}} \tag{2}$$

Than, the velocities of robot's left and right wheel are defined

$$vl=jy\cdot v_{max}-jx\cdot(\omega_{max}\cdot d)/2$$
$$vl=jy\cdot v_{max}+jx\cdot(\omega_{max}\cdot d)/2 \tag{3}$$

This principle is shown in the Fig. 3. The computer-assisted control (CAC) block in this case is very simple and the control loop is opened. It leads to very difficult manual control of the robot to move it into desired position or orientation. Human is unable to both accurately and quickly control the rotation of the robot and to adjust its relative position. That is why this steering is unsatisfactory.

B. Semi-automatic steering

Current solution of the computer-assisted manual control is based on automatic control of the robot according to the desired direction and speed. This requires closer cooperation with the vision system. See the Fig. 4. The vision system supplies a position of the robot *rx*, *ry* in axis x, y of the playground and angular position $r\varphi$.

The objective of this control is to allow moving with the robot as it was a 2D point

$$jx=\frac{vx_d}{v_{max}}\ ,\ \ jy=\frac{vy_d}{v_{max}} \tag{4}$$

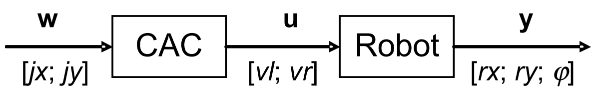

Fig 3. Schema of differential steering.

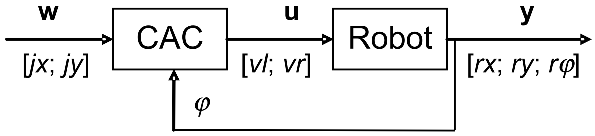

Fig 4. Schema of semi-automatic steering without filter.

where vx_d, vy_d are desired speeds along the axis x and y. There are two possible configurations, without a filter (Fig. 4) and with a filter (Fig. 5). The filter reconstructs the precise position and the speed of the robot. It also allows the CAC to compensate the error from the trajectory due to rotation of the robot. Currently, there is used an extented Kalman filter [1][3].

The schema of the CAC block is in the Fig. 6. Suppose that a player wants to move the robot to given direction with certain speed.

If the robot does not move it should first rotate around its vertical axis to a desired direction and then it should start moving with a desired speed. This will ensure following of a desired trajectory. This function is implemented in the Stationary rotation block. If actual velocity *v* of the robot is smaller than predefined minimum velocity v_0 and absolute value of angle error $\Delta\varphi$ is greater than a predefined maximum angle error $\Delta\varphi_0$ then desired speed v_d is zeroed. Simulated trajectories are shown in the Fig. 7.

If the robot is already moving, it is not possible to change the desired direction and to maintain the planned trajectory.

First, the robot has to rotate to the desired direction while it is still moving. Then it can gradually approach the trajectory that it would move as a mass point from the time of direction change in an ideal case. The tracking of ideal

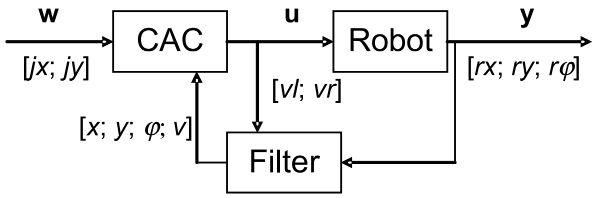

Fig 5. Schema of semi-automatic steering with filter.

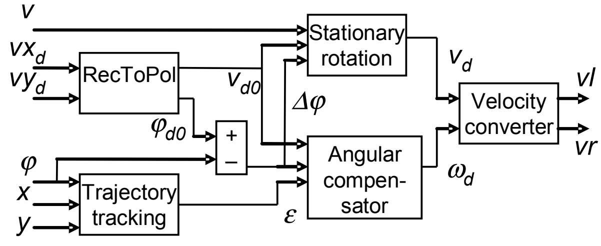

Fig 6. Schema of CAC for semi-automatic steering.

mass point is encapsulated in the Trajectory tracking block. The control requirements – direction (angle error $\Delta\varphi$) and position control (parameter ε) act against each other and this must be reflected by compensators design including system non-linearity and bounds of centripetal acceleration. The compensators are placed in the Angular compensator block. The Velocity converter block converts the pair of desired velocity v_d and angular velocity ω_d into the desired velocities *vl*, *vr* of left and right wheel. Simulated trajectories are shown in the Fig. 8.

If the robot is standing or moving, it is possible to invert the direction of robot. This reduces the required angle of rotation. This functionality is not shown in Fig. 6. If the robot is moving, the range of desired directions suitable for the inversion of robot direction is significantly lower. This is because gradual approach to the desired trajectory is more time efficient than stopping the robot and turning the direction of motion by 180 degrees.

C. *Fully-assisted steering*

This steering is next logical step in evolution of manual control module. See the Fig. 9. It requires cooperation with the Higher strategy and the Motion control module. The higher strategy controls all robots except the manually controlled ones and instructs them the desired positions and velocities [4]. The CAC block overrides the position and velocity for manually controlled robots. The motion control plans trajectory in real-time and controls position of the robots by changing the desired wheel velocity [5].

There are three modes of operation

- Free movement
- Tracking the ball
- Shot.

The free movement mode allows similar movement of the robot as semi-automatic steering. The difference is in this mode the robot will cooperate with other robots and avoid collision with them.

The tracking ball mode causes the robot to start catching the ball. It is activated by pressing the button b_0. If the stick

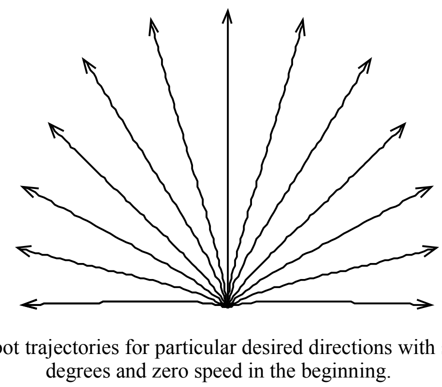

Fig 7. Robot trajectories for particular desired directions with step of 15 degrees and zero speed in the beginning.

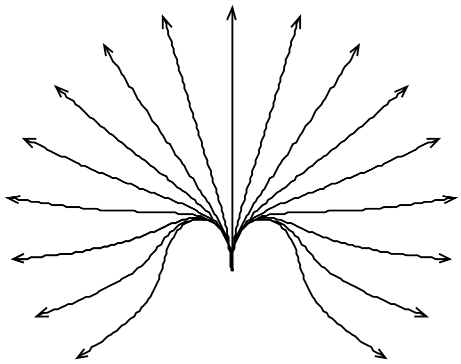

Fig 8. Robot trajectories for particular desired directions with step of 15 degrees and constant velocity.

is also applied, it controls the direction in which to meet the ball.

The shot mode makes the robot to shoot to the direction specified by stick.

D. *Group-assisted steering*

This steering will be the final extension of fully-assisted steering. The basic principle is same as previous except a fact that one player is controlling all of the robots.

Only one robot is controlled at the moment, so the capability of robot switching has to be developed. There are several possibilities of making such capability

- The robot nearest to the ball – the robot is selected automatically or by a button.
- Switching to a next robot – selection of the next robot in sequence.
- Select robot by POV-hat or stick – the robot is selected by analysis of current positions of robots on the playground.

It may be difficult for player to quickly recognize the selected robot on the playground. This can be solved by

- External lighting of the robot by LEDs.

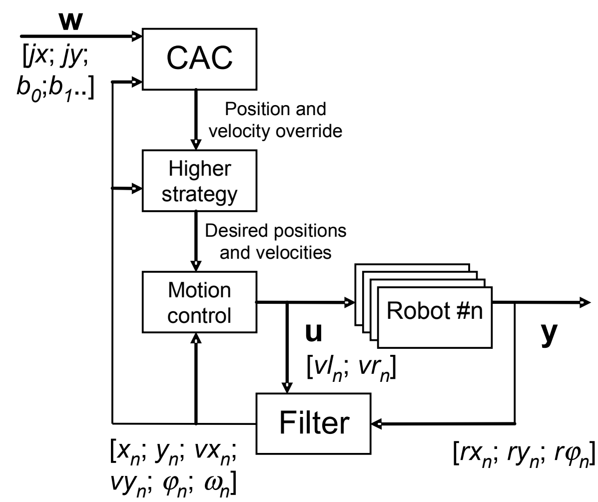

Fig 9. Schema of fully-assisted steering.

- Only virtual selection of the robot on soccer monitor.

IV. Additional functions

There are several additional requirements on the computer-assisted manual control. Some of them are general extensions, others are important for concrete type of steering.

A. *Collision Detection*

In the robot motion control, a collision detection is necessary. When a collision occurs, it leads not only to stagnation of the robot but it also causes that the underlying mathematical model of the robot gives false output. The model is part of the filter block in the motion control module and provides the predicted position and the speed of the robot based on control requirements. The information about collision can be used to the resetting of the model.

Another problem occurs if there is a permanent obstacle in the way. Then the described motion control is not able to react properly and rotate the robot. The solution is to stop the robot until it rotates and gets out of the collision.

Collision detection is performed in two ways

- A priori – detection is made on the basis of the position of the robot to known bounds (of the playground) and the direction of the velocity vector,
- A posteriori – detection is made on the basis of significant difference between the predicted robot position and its actual position.

B. *Game controls*

In addition to the control of the direction of the robot in two axes, there are more features important for the strategy testing. Some of these functions apply only on semi-automatic steering

- Acceleration – the possibility of a short-time increase of robot speed, this function is useful for shooting the ball.
- Brake – allows rotation of the robot in place.
- Tracking the ball – robot automatically targets towards the ball, allowing its easy hitting.

C. *Force feedback*

When the robot gets stuck, typically due to an impact of the robot or an attempt to pass through the barrier, the game controller vibrates strongly for a brief moment.

V. Conclusion

In this paper we described the Manual control module of the robot-soccer system. This module provides interface between the game controller and robot. Currently there is used the principle of semi-automatic steering, which is the step between the simple differential steering and the fully-assisted control method. The manual control module also uses additional functions as the collision detection and force feedback. Next research will focus on design of fully-assisted control. The requirements for the upcoming CAC methods were also described.

References

[1] P. Wagner, *et al.*, “Control of Group of Robots”, In *9th RoEduNet International Conference*, to be published, 2010.

[2] J. Kotzian, Z. Machacek, *et al.*, “Embedded control system for the mobile robot”, *WSEAS Transactions on Systems*, pp. 2261-2268, 2005.

[3] L. Jetto, S. Longhi, “Development and Experimental Validation of an Adaptive Extended Kalman Filter for the Localization of Mobile Robots”, *IEEE Transactions on Robotics and Automation*, Vol.15, pp. 219-229, 1999.

[4] V. Srovnal, B. Horak, *et al.*, “Strategy Description for Mobile Embedded Control Systems Exploiting the Multi-agent Technology”, *Lecture Notes in Computer Science*, pp. 936-943, 2007.

[5] G. Novak, M. Seyr, “Simple Path Planning Algorithm for Two-Wheeled Differentially Driven (2WDD) Soccer Robots”, *WISES 2004*, 2004.