Multi Robot Systems: The EagleKnights/RoboBulls Small-Size League RoboCup Architecture

Alfredo Weitzenfeld
University of South Florida
Computer Science and Engineering Department
Tampa, FL 33620-5399
(813) 974-3652
aweitzen@cse.usf.edu

ABSTRACT
In this paper we present the system architecture of the Eagle Knights/RoboBulls Small Size League RoboCup Team. In this league two teams composed of five autonomous robots each compete against each other in a medium size field. This league is one of the fastest and most thrilling in RoboCup permitting teams to develop complex coordination strategies. We explain the three main components of the architecture: Vision System, AI System and Robots.

Keywords
robotics, robocup, autonomous, vision, architecture.

1. INTRODUCTION
RoboCup [1] is an international joint project to promote AI, robotics and related field. In the Small Size League, two teams of five robots up to 18 cm in diameter play soccer on a 4 by 5.5 m carpeted soccer field. Figure 1 shows a schematic diagram of the playing field and computer setup. Figure 2 shows a picture of one of our robots.

Aerial cameras send video signals to a vision system computer that computes robots and ball positioning on the field. This information is then passed to an AI system that produces control commands sent to the robots via wireless communication. Additional information is provided by a referee box indicating the sate of the game.

The robot architecture of the EagleKnights/RoboBulls consists of four main components as shown in further detail in Figure 3: (1) vision system, (2) AI system, (3) robots and (4) referee:

1. The **vision system** digitally processes two video signals from the cameras mounted on top of the field. It computes the position of the ball and robots on the field, including orientation of robots in our team. Resulting information is transmitted to the AI system.
2. The **AI system** receives the information from the vision system and makes strategic decisions. The actions of the team are based in a set of roles (goalkeeper, defense, forward) that exhibit behaviors according to the current state of the game. This module includes functionality to avoid collisions with robots of the opposite team. The decisions are converted to commands that are sent to the five robots via a wireless link.
3. The **robots**, five in total, execute commands sent from the AI system by generating mechanical actions. This cycle is repeated from 30 to 60 times per second.
4. The **referee** communicates external game decisions (penalties, goal scored, start of the game, etc.) to each team sending a set of predefined commands to the AI system through a serial link.

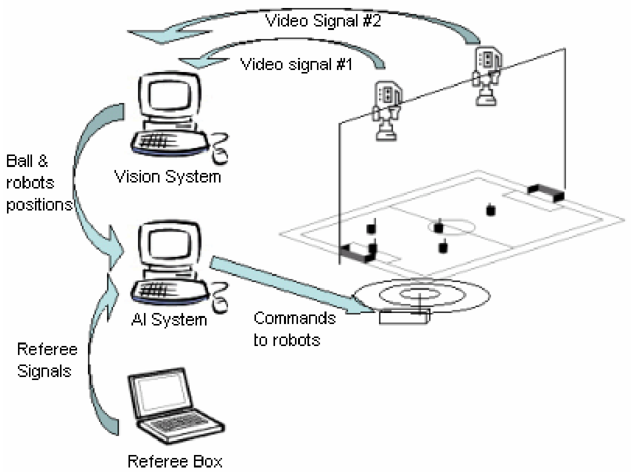

Figure 1. System configuration of the Small-Size League. Aerial cameras send video signals to a computer processing the vision system that in turns sends its output to an AI system responsible for remote robot control.

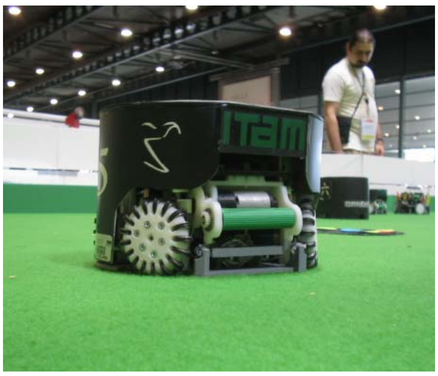

Figure 2. Small Size League RoboCup robot.

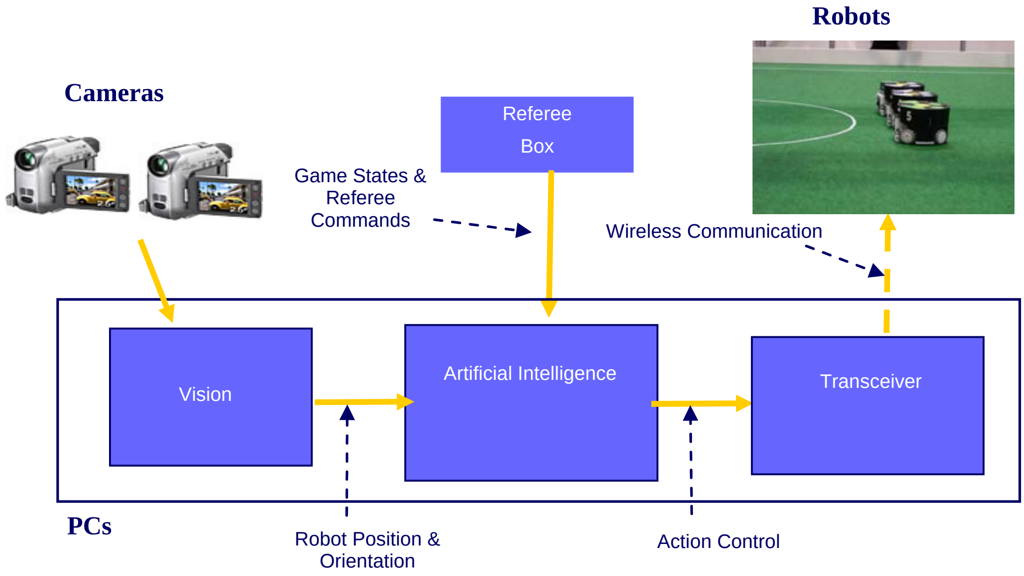

Figure 3. RoboCup Small-Size league block diagram. Visual input from cameras mounted on top of the field are processed by the Vision module to provide the AI module with robot positions and orientations. The AI module sends action command to the robots via a transceiver.

In the next sections we describe in more detail each component of the architecture.

2. VISION SYSTEM

The vision system is the only source of sensory feedback in the system. If data returned by the vision system is inaccurate or incorrect the overall performance of the team will be severely affected. That is why the vision system should be robust enough to compensate for possible errors.

The main object characteristics used by the vision system are the colors defined in the rules of the SSL [2]. The ball is a standard orange golf ball. The robots of one team must have on top of them a 50 mm blue colored circle while the other team must have a yellow patch. The main tasks of the vision system are:

- Capture video in real time from cameras mounted on top of the field.
- Recognize the set of colors specified by the rules in correspondence to objects of interest in the field (robots and ball).
- Identify and compute the orientation and position of robots in the field.
- Compute the position of robots of the opposite team.
- Track objects in the field and get their moving vector.
- Transmit information to the AI system.
- Adapt to different light conditions (color calibration procedure).

The vision system [3] consists of several modules where each module is a functional block with a specific task as shown in Figure 4:

- CAPTURE MODULE. We use two AVT Guppy 1/3" progressive scan CCD cameras with an IEEE 1394 link. The frame capture is the AVT software that allows us to configure the resolution of the image, space color and frame rate. By default we capture RGB images with a 720x480 resolution at 60 fps.
- PREPROCESSING MODULE. The preprocessing module is used to improve the quality of the image, such as brightness, contrast, gamma, exposure, white balance, etc.
- OBJECT CALIBRATION MODULE. This module is a tool to establish the thresholds of each color component according to the space color defined for every object of interest (robots and ball). The calibration is done in HSV color space where selected thresholds are more robust to changes in field lighting and color changes than in previous versions. The HSV thresholds are transformed to RGB values to improve segmentation speed and avoid costly color space transformations.
- SEGMENTATION MODULE. This module assigns each image pixel into object classes. The module consists of two segmenters, each one using separate thresholds values assigned to each camera for every object of interest. The HSV thresholds are mapped to a complete RGB color space cube in such a way that a 32 color segmentation can be done with just one access to memory.
- BLOB BUILDER MODULE. This module links segmented pixels with blobs. Before reaching this module the image is composed of separate pixels. Once a blob is constructed relevant information can be easily computed, such as color areas, centroids, bounding boxes, etc. A Run Length Encoding and four-neighbor search are computed. A joint list of blobs for the two cameras is generated for each color.
- ACTIVATION/DEACTIVATION MODULE. This module enables or disables the use of a particular robot. Sometimes a team can play with a smaller number of robots.

- RECOGNITION MODULE. This module identifies each object in the field selecting the color regions that better adjust to objects searched. Each object in the field is searched within a small square area. This square moves with each frame according to the prediction returned from an Extended Kalman Filter (EKF) in the case of the ball and opponent robots. Robots are tracked using their moving vector provided by the AI system. With this method the processing prediction time and noise are reduced. The system includes specific selection criteria for every kind of object in the field. For the ball the biggest orange blob is selected (close to 85 pixels with an image resolution of 720x480). For the robots of the opposite team the selection criteria looks for blobs with corresponding central patch color with an area closest to 115 pixels (the area of the patch is bigger than the ball). For same team robots the procedure is similar to the one used for robots of the opposite team, although in addition to the central patch, a search for extra patches is necessary. The extra patches are employed for identification and orientation computation. In the case where one object is not found inside its square area a sub sample segmentation is made in the entire field in order to relocate all objects and resume the tracking step.
- GEOMETRIC CALIBRATION MODULE. This module computes the internal and external parameters of the cameras using the Tsai method [4]. These parameters are used to correct the distortion produced by the camera lenses and to convert camera coordinates to world coordinates and vice versa.
- LOCALIZATION MODULE. This module computes the position and orientation of objects in the field. It uses camera parameters obtained in the Geometric Calibration module to correct distortions in the image. Computation uses a set of pre-defined points, each one representing a well known landmark in the field (corners, midline point, etc). When a point appears in both camera images, corresponding coordinates are used to match and discard duplicate pixel data.
- KALMAN FILTER MODULE. This module consists of an Extended Kalman Filter used to reduce noise generated during blob centroid computation. The module predicts the ball and opposite robots position in the next frame. An adjustable box is used to limit the search of their position in the next frame. In addition, moving vector for each object are also computed.
- GRAPHIC DISPLAY MODULE. This module is responsible for displaying video images in the screen and for generating basic drawing functions such as lines, circles, etc.
- TRANSMISSION MODULE. This module consists of UDP network link used for communication between the vision system and the AI system that process on separate machines. The module builds a structure appropriate for data transmission. The information sent to the AI system includes same team robot positions and orientations, together with ball and opposite team robot positions and the moving vector for all objects.

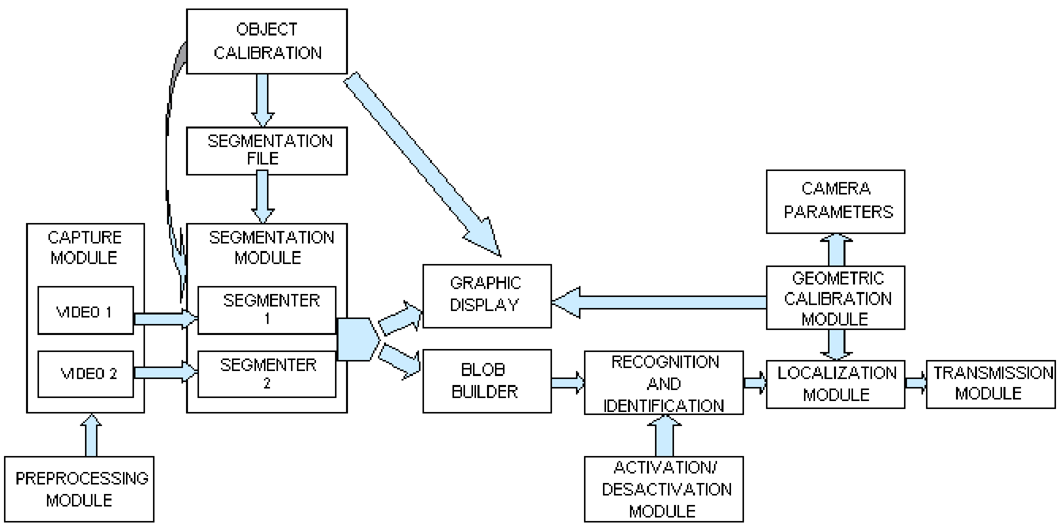

Figure 4. Vision System Architecture consisting primarily of the following modules: Capture, Segmentation, Calibration, Recognition and Identification, Activation and Deactivation, Localization and Transmission.

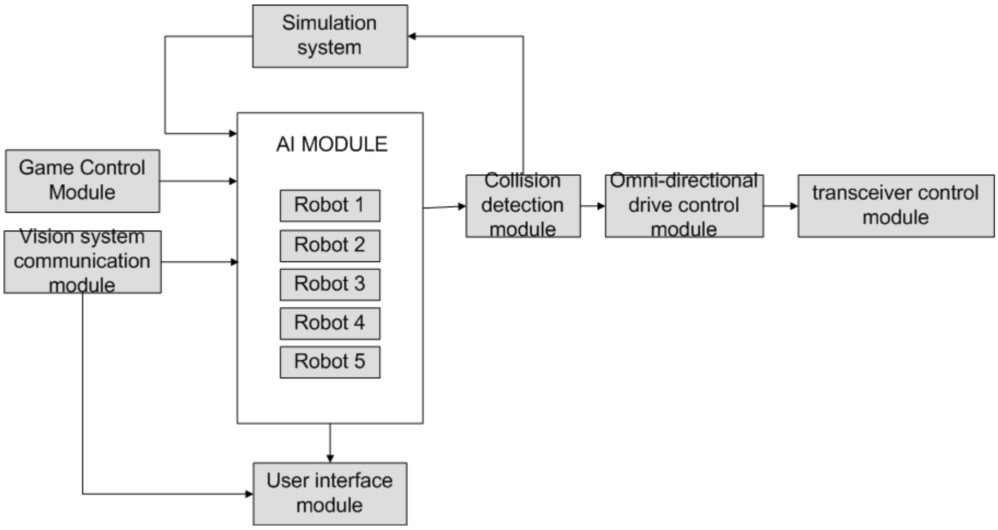

Figure 5. Artificial Intelligence System Architecture consisting primarily of the following modules: Game Control, Vision Communication, User Interface, Simulation, Collision Detection, Omni-directional drive control and Transceiver Control.

3. AI SYSTEM

The AI System consists of eight modules: Artificial Intelligence, Simulation System, Collision Detection, Transceiver Communication, Drive Control, User Interface, Vision System Communication and Game Control. This system is designed in a way that the user can test each module separately and independently from the robots. The system includes a dynamic simulator in two and three dimensions to test system functions, including collision detection, AI and robot control.

The artificial intelligence system includes a main thread that loops and calls each of the different modules as shown in Figure 5. The detailed description for each of the modules is as follows:

- VISION SYSTEM COMMUNICATION MODULE. This module provides via packets the vision system commands representing the game scenario corresponding to robots and ball coordinates, angles and moving vector.
- GAME CONTROL MODULE. This module receives referee commands through a serial interface and returns the game state of the game.
- AI MODULE. This module receives the robots and ball positions, robots orientations, game state, robots roles, shooting direction and field configuration. With this information the system calculates future position and actions to be taken by each robot. The strategy used depends on the configuration of a tree that analyzes all possible actions. Actions are classified according to their importance. For each node of the tree one or more evaluations are used. Each evaluation has a group of possible results associated with a particular score. During the main program loop the tree is evaluated. The trajectory to take from root to leaf (final action) depends on the highest score of the evaluation result on each level using a Best First Search method. Once the system has reached a final action like passing, shooting, or blocking, the robot moving vector, its linear and angular velocity and the activation of the kicker and dribbler devices is defined. Robots are coordinated through different roles: Goalkeeper, Defense, First, Second and Third Forward. The final action for each role is defined using the position of the robots and the ball. With the Extended Kalman Filter used in the vision system it is possible to know where the ball is moving in order for defenders and goalkeeper to intercept it. The module is also used to send or receive passes and to shoot to goal.
- USER INTERFACE MODULE. This module constantly displays current game state. The information includes robot positions, orientations, speed, game state, control commands to the robots and the configuration of the AI system.
- SIMULATION SYSTEM. This module tests the operation of the artificial intelligence system without using the real vision system or the actual robots. It is useful to debug and test actions in the artificial intelligence module. The field is visualized with a two or a three dimension graphics interface.
- COLLISION DETECTION MODULE. This module receives current and final position of the robots and generates a new path generated by a GET (geometrical exploring tree) [5]. With this method it is possible to easily avoid the opposite's robots and goals. The method works in real time (five robots in more than 60 fps). GET constructs a tree during every control iteration. It can combine different types of obstacles with geometrical figures. Robots are represented by circles and goals by rectangles. To generate the planner the tree starts with a root in the initial point being classified as an exploring node. The final point is already been defined by the AI module. The steps of the algorithm are as follows: First select an exploring node of the tree and try to reach the goal advancing a small predefined distance. If there are no obstacles interfering with the new point then the tree is extended and the new point replaces the last exploring node. In order to know if an obstacle interferes between the initial point and the goal the geometry must be considered. A vector "A" is defined from the initial point to the goal. In the case of a circular obstacle a possible intersection between the circle and the vector "A" is

calculated as shown for example in Figure 6. In case of an intersection the distance between the obstacle and the new extension is validated to be smaller than a radio "R".
In the example shown in Figure 5 there are two intersections: "P1" and "P2". When the tree reaches radio "R", it will generate two possible routes one to each side of the obstacle. These two points are now considered as exploring nodes. While the exploring node can not freely reach the goal then it continues rounding the obstacle until there are no further intersections or obstacles. In such case a new obstacle is defined as the new exploring node obstacle and it continues surrounding the new obstacle. Figures 7 and 8 show the generation of the GET with one and several obstacles.
In the case of the goals the algorithm works similarly but surrounding them with a rectangle. Once the tree has reached a point close enough to the goal the nearest path is chosen and the robot can be sent directly to the first point before intersection or in the direction of the next node towards the goal. This algorithm is repeated every cycle until the robot reaches its goal.

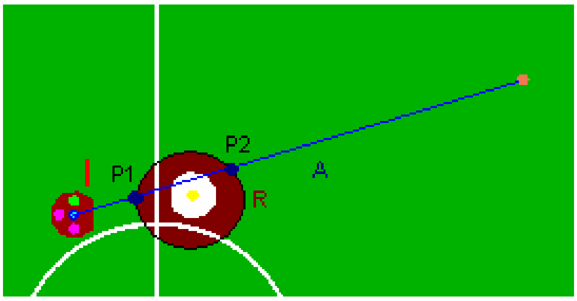

Figure 6. Collision detection of a circular obstacle with robot 1 trying to reach the ball.

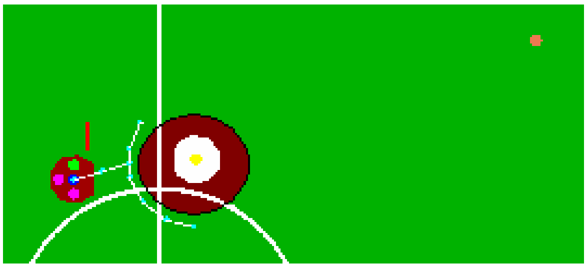

Figure 7. The tree is generated until an obstacle is found.

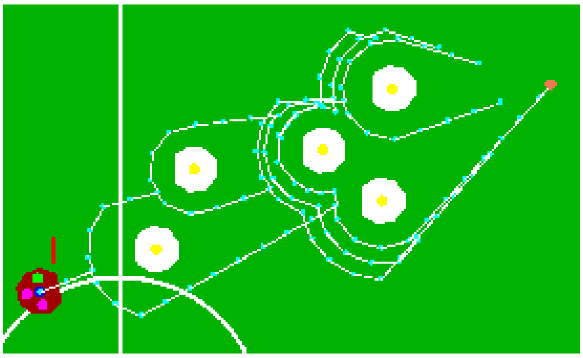

Figure 8. The tree finds the goal avoiding multiple obstacles.

- TRANSCEIVER COMMUNICATION MODULE. This module builds the packets sent to the robots using a transceiver. The information sent to each robot is the moving vector and the angular speed of each robot.

4. ROBOT

We designed and built five omnidirectional robots. Each robot has five Faulhaber 2224P0212 motors with gearheads 14:1 (four motors for the wheels and one for the dribbler), a low resistance solenoid, a DSP - Digital Signal Processor, a transceiver, a single printed circuit board and two Lithium Polymer batteries. The height of the robot is 140 mm, the maximum diameter of its projection to the ground is 178 mm, and the maximum percentage of ball coverage is 19%. The robots were manufactured using a CNC ABC plastic machine.

The robot receives commands from the AI system in the PC. It includes the following functional elements:

- ROBOT ID. Each robot incorporates an identification circuit manually setup with a dipswitch making it easy to modify the robot ID if necessary.
- DSP. The robot micro controller is a Texas Instruments TMS320LF2812 fixed-point single chip DSP. This device offers low power and high-performance processing capabilities, optimized for digital motor and motion control. The DSP consists of six major blocks of logic: (1) External program and data memory, (2) I/O Interface, (3) Standard I/O, in addition to other modules not currently used in our design. The modules used are:
  1. External program and data memory. The RAM module is used in debugging the software with the Parallel Port JTAG Controller Interface.
  2. I/O Interface. It contains different kinds of pins: (i) Capture units used for capturing rising pulses generated by the motor encoders which can be used to measure speed and direction of the moving motor. (ii) PWM outputs having an associated compare unit. A periodic value is established to determine the size of the PWM, and the compare value is used to change the duty cycle.
  3. Standard I/O: used to read and write values for transceiver communication, motor, kicker and dribbler control.
- MOTOR CONTROL. The motor encoders generate a number of square pulses for each completed turn as shown in Figure 9. Each pulse is captured using the DSP and the feedback speed is computed into the omni-directional module. To control the motors speed a PWM signal sent back to the motor. This information is obtained by the omni-directional module.
- WIRELESS COMMUNICATION. Wireless communication is controlled by two Radiometrix RPC-914/869-64 transceivers with radio frequency at either 914MHz or 869MHz. The transceiver module is a self-contained plug-in radio incorporating a 64kbit/s packet controller with a parallel port interface. Data is transferred between the RPC and the host (either DSP or PC) four bits at a time using a fully asynchronous protocol. The nibbles are always sent in pairs to form a byte, having the Least Significant Nibble (bits 0 to 3) transferred first, followed by the Most Significant

Nibble (bits 4 to 7). Two pairs of handshake lines REQUEST & ACCEPT, control the flow of data in each direction.

- OMNI-DIRECTIONAL DRIVE CONTROL MODULE. This module receives the movement vector including linear and angular velocities from the transceiver. To control the motor speeds to steps are completed. First the capture from the motor enconders is used to get the motors speed. The speed of each motor generates the actual linear and angular velocities of the robot. The second step is to use these velocities along with transceiver velocities as inputs to the PID algorithm. There are three independent PID algorithms in the process: the linear speed projection in the x and y coordinates of the robot and the angular velocity. The output of the PID is turned into speeds for each motor (using the motors geometry in the robot) and finally they are controlled to the correct speed with a PWM signal.
- KICKER CONTROL SYSTEM. Small Size soccer robots use different kicking designs to push the ball. We use a push type solenoid that kicks the ball. Solenoid kicker system needs a high power supply. For size restrictions robots have six 7.4V/700mA batteries, equivalent to 31 Watts of power. With this amount of power we obtain less than the solenoid requires for a minimum performance. The main idea in power elevation is to store energy, then discharge it when solenoid is activated. To solve this power problem we implement a four-layer system as follows:

1. Voltage transformation. The 14.8 dc voltage obtained from the batteries is increased using a (PicoElectronics IRF100S) dc-dc converter to reach 110 volts. The output is used to charge up two 2200mF capacitors. The converter is controlled using a control pin of the DSP with a relay and a transistor. The robot can kick approximately every 10 seconds.
2. Discharge and solenoid activation. An infrared sensor system in the bottom of the robot senses if the robot has the ball. The DSP sends a high-level output bit when the robot is in score position. To discharge the capacitors into the solenoid, the Discharge layer uses both the DSP kick bit and the infrared ball detector output bit to discharge the capacitors. Because the capacitors charge level is very high, the robot discharges it using a power MOSFET. A PWM signal is sent to the MOSFET to control the flow of current through it and thus controlling the intensity of the kick.

5. CONCLUSIONS

We presented an overview of the SSL EagleKnights/RoboBulls team. The collision detection, kicker device and omni-directional robot control have been described in detail. Our team has been consistently competing in regional and world RoboCup tournaments obtaining, 2nd and 3rd place in US Open and 1st place in Latin American Open. We have participated in the last RoboCup competitions: Osaka, Japan 2005, Bremen, Germany 2006 and in Atlanta USA 2007. We have released to the public the Vision System and documentation of our electronics and DSP software to promote the participation of others teams in this initiative. More information can be found in http://robotica.itam.mx/ and http://www.usfrobobulls.org. Videos of previous participations can be found in http://robotica.itam.mx/ingles/small_size/fotos.phtml.

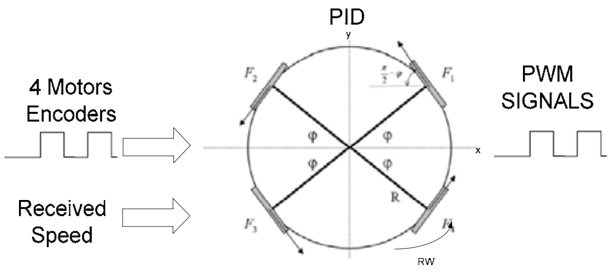

Figure 9. Motor control using Pulse Width Modulation (PWM) and Proportional-Integral-Derivative controller (PID).

6. ACKNOWLEDGMENTS

This work is supported by French-Mexican LAFMI, UC MEXUS CONACYT, CONACYT grant #42440, and “Asociación Mexicana de Cultura, A.C.” in Mexico and by ARO, SPAWAR and by the USF College of Engineering in the US.

7. REFERENCES

[1] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa. Robocup: The robot world cup initiative. In Proceedings of the IJCAI-95 Workshop on Entertainment and AI/ALife, 1995. http://www.robocup.org.

[2] Verret, Ball, Kiat Ng. Laws of the F180 League - Release 3.00a. http://www.itee.uq.edu.au/~wyeth/F180%20Rules/index.htm

[3] Martínez-Gómez, L.A., and Weitzenfeld, A., 2004, Real Time Vision System for a Small Size League Team, Proc. 1st IEEE-RAS Latin American Robotics Symposium, ITAM, Mexico City, October 28-29.

[4] Tsai, R.Y. A versatile camera calibration technique for high accuracy 3D machine vision using off-the-shell TV cameras and lenses. IEEE Journal of robotics and Automation, 1987.

[5] Elnagar, A., Basu, A. Local path planning in dynamic environments with uncertainty. Multisensor Fusion and Integration for Intelligent Systems, 1994.