MRL Team Description Paper for Humanoid KidSize League of RoboCup 2013

Mostafa E. Salehi[1], Reza Safdari, M. Reza Najafipour, Amir Salimi, Mohammad Aghaabbasloo, Erfan Abedi, Roham Shakiba, Meisam Teimouri, Bahareh Foroughi, Ehsan Nazari, Mojtaba Ghanbari, and Mostafa Azarkaman,

Mechatronics Research Lab, Dept. of Computer and Electrical Engineering,
Qazvin Islamic Azad University, Qazvin, Iran
Email: [1]m.e.salehi@qiau.ac.ir
Web: http://mrl.ir

Abstract. This team description paper presents the specifications of the MRL kidsize humanoid robot system which contains different parts including robot vision, motion control, world modeling, self-localization, and behavior. MRL humanoid team is developed under the RoboCup 2013 rules to participate in the kidsize humanoid soccer league competition in Eindhoven, the Netherlands and like the last year we will introduce a referee with sufficient knowledge of the rules available during the competitions. We use DARwIn-OP as our base platform and we have modified this platform in vision, motion control, world modeling, self-localization, behavior, embedded control board, and also the robot embedded operating system as will be discussed in the related sections.
Keywords: RoboCup, Kidsize Humanoid League, Bipedal Locomotion, Artificial Intelligence, Embedded System Design

1 Introduction

RoboCup uses soccer as a research area to develop a team of humanoid robots that can win the human world champion soccer team in 2050. In the Humanoid league, human-like fully autonomous robots play soccer against each other and meanwhile handle stable walking, modeling and kicking the ball, visual perception of the ball, players, and the field, and self-localization. The RoboCup soccer playing robots introduce challenges in design, control, stability, and behavior of autonomous humanoid robots.

The MRL project was started in 2003 in the Mechatronics Research Laboratory in Islamic Azad University, Qazvin branch looking onward to enhance the knowledge of robotics and the MRL humanoid kidsize soccer league is aimed to develop a humanoid platform for research and education. Our research center has the honor to hold the RoboCup IranOpen from 2003 to 2013. MRL has nine qualified teams and has had a successful history in RoboCup for many years. Our humanoid soccer playing team is one of the developing soccer-playing humanoid robots in the RoboCup Humanoid League and has participated in RoboCup and IranOpen Humanoid League in 2011 and 2012. In 2012 we had the honor to be in the top 8 teams among 22 participating teams. This year we are planning to participate in the kidsize humanoid competition for the third time in RoboCup 2013 in Eindhoven, the Netherlands. Our mission is to fulfill our study in motion control, vision, world modeling and localization, artificial intelligence, and embedded system design.

MRL Humanoid Kid Size team consists of one Ph.D., eight graduate, and eight undergraduate students from software, hardware, electronics, and mechatronics. The other team members are: Moien Shirkhorshidi, Pooya Pishbin, Alireza Tabaie, Human Heidari, Khashayar Ghamati, and Elahe Mansuri.

2 Overview of the System

We have used DARwIn-OP (Dynamic Anthropomorphic Robot with Intelligence Open Platform) [1] in our soccer playing team for RoboCup2013. The kinematic structure with 20 DoF can be seen in Fig.1. The actuators used in our robots are the MX28 servo motors. The motion mechanism consists of 20 degrees of freedom distributed in six per leg, three per arm and two degree of freedom moving the neck horizontal and vertical. Physical specification of the robot is illustrated in Table 1. Our developments for the kidsize humanoid robot include the design and construction of modular software architecture based on the Upenn RoboCup released code [2]. The software contains robot applications including autonomous motion and walking controller, self-localization base on vision, planning, and communication.

Considering the processing power of humanoid soccer playing robots, we need to use a customized operating system for special purposes. We have customized the Linux kernel for our robots in order to have a proper scheduling getting the best result. In fact we build up a light specific distribution of Linux for DARwIn-OP. The DARwIn-OP uses Ubuntu distribution as default. Since it is a general purpose distribution, it imposes processing load to the robot. We have improved some quality factors such as performance and running time by deploying our novel light distribution and using specific CPU scheduler, file system and customized kernel which only installs the minimum required libraries and applications. Because we have built up our distribution for humanoid robots and there is no need to have a GUI (Graphical User Interface), we have only provided BASH as a CLI (Command Line Interface) user interface. After comparing a few different CPU schedulers, finally we have chosen the RSDL for scheduling processes. RSDL maintains the priority array. Different processes want to run according to the priority. The processes at the highest priority are allowed to execute and will be given the time slices. It uses Round-Robin algorithm to rotate through time slices [3]. For our file system we have deployed BRTFS. Considering that our distribution is Debian-based, we use APT as package manager.

The robot hardware consists of the mechanical structure and the driver circuit board which we are modifying and developing as a customized embedded system for humanoid soccer playing robot. Each robot is able to detect the ball and goal by scanning the field, walk towards the ball, and kick when it catches the ball. The project is still in progress and some developed methods are described in the current report.

Fig.1. kinematic structure of DARwIn robot.

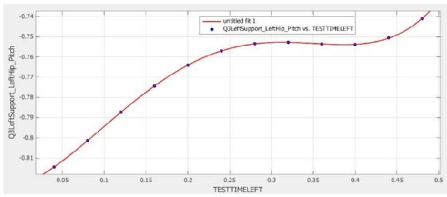
Table 1. Physical measurements of the DARwIn.

Feature	DARwIn
Height:	45.5 cm
Weight:	2.8 Kg
Walking Speed:	24 cm/s
Degrees of freedom:	20 in total
Servo motors:	20 MX-28
Sensors:	Touch sensor and IMU
Embedded PC board:	Fit-PC2i

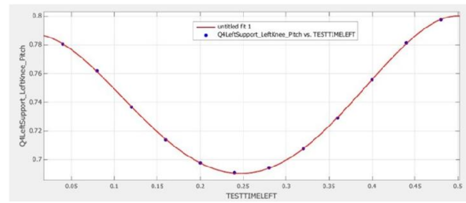
DARwIn-OP consists of a USB camera, two embedded processing systems, gyro and acceleration sensors, servo motors, batteries and some user interfaces such as switch and LED. Images are captured by the USB camera, the camera sends image signal to the main CPU board. The CPU processes the image data to detect positions of ball, goals, and other robots by color-based image processing. A particle filter is employed to localize the robot in the soccer field. We also have used wireless communication between the robots. Exploiting the vision and network data we select the next behavior of the robot according to the robot role and the priority of the behaviors. The defined behaviors are composed of simple motions to support more complex tasks.

3 Motion Control

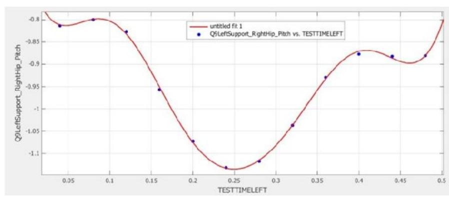
One of the challenging research areas in humanoid robots is the walking and stability. In this section we introduce our methodology and the proposed evolutionary algorithm that is used to modify the DARwIn robot motion. Motion of joints in biped robots can be studied in two categories: a) movement position, b) angular position [4]. Mechanical methods are based on the dynamics of the robot and information from the environment. In our methodology we focus on angular positions of the joints and we use sinusoid equations to generate motion gait patterns for swing leg and we use polynomial equations for support leg which leads to higher performance and stability during the walk. We have also used an evolutionary algorithm in tuning the parameters of robot motion. These parameters are the coefficients of equations that are set for each joint and create the sequence of joint angles for robot walking. The evolutionary algorithm that we use in this implementation is the PSO algorithm. We have produced single step cycles in 0.48s and record the angular position value of each joint in each 0.04s (angular position for both swing leg and support leg). Fig.2 represents some samples for angular positions for hip, knee, and ankle joints which are fitted to sinusoid and polynomial mathematical equations.



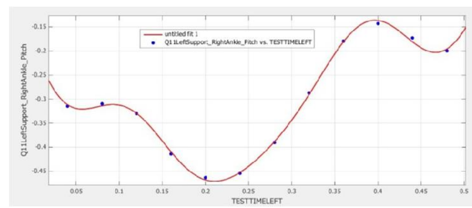
a) **Left Hip Pitch (Left leg Support**



b) Left Knee Pitch (Left leg support)



C) **Right Hip Pitch (Left leg support)**



d) Right Ankle pitch (Left leg Support)

Fig. 2- Samples of angular positions for (a, c) Hip joint, (b) Knee joint, (d) Ankle joint.

The PSO algorithm is an evolutionary algorithm which is based on plural intelligent of particles. This bio- inspired algorithm is based on iterations and probability. In our methodology we present each particle with a structure of four attributes that is shown in Fig. 3.

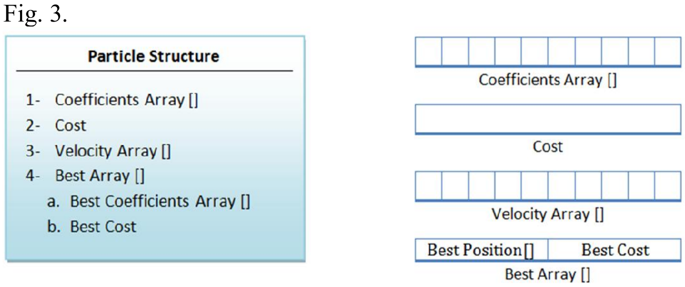


Fig. 3- Particle structure in our model.

The coefficients array is the parameter aimed to be optimized. This array is updated with two equations that are shown in equation (1) and (2). These equations are position and velocity updates in the PSO algorithm. The fitness function for this algorithm is produced by maximum robot movement in X direction and minimum robot movement in Y direction. The fitness function is calculated with equation (3).

$$v_i(t+1) = \beta\left(v_i(t) + c_1 r_{1i}\left(pBest_i - x_i(t)\right) + c_2 r_{2i}\left(gBest_i - x_i(t)\right)\right) \tag{1}$$

$$\beta = \frac{2}{\left|2 - \omega - \sqrt{\omega^2 - 4\omega}\right|}, \qquad \omega = c_1 + c_2\,, \qquad \omega > 4$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{2}$$

$$\sum_{i=1}^{n} \frac{\sqrt{(x-x_0)^2 + (y-y_0)^2} - |(y-y_0)|}{x_{destination} - x_{offset}} \tag{3}$$

The maximum speed of DARwIn is 24 cm/s. Stability control is based on the robot's gyroscope and the controller receives data from this sensor via A/D converter. According to these data, the robot detects a fall and prevents fall. When the robot falls it detects the fall and stands up smoothly. The robot can stand up from lying on its back and its front side as well.

4 Robot Vision

Vision is one of the most important interfaces for robot perception [5]. The main vision sensor is a camera that is located in the robot's head. This camera model of DARwIn-OP is Logitech c905 that uses USB2 connection with 2 Megapixel 640×480 resolutions (up to 1600×1200, 10fps or 1280×720, 30fps) in YUYV color space capturing 30 frames per second. At the first step, we used V4L2 module to grab the raw output of the camera, then the grabbed image was converted to HSI color space and is mapped to the field's colors, using a color look-up table to segment the image according to the color. For robot's color learning phase we used color look-up table for segmentation -the same as what we had done last year. One of the leading problems of this approach is its dependency to the light intensity and the other problem is that it takes a pretty long time to set the color look-up table manually. The light intensity is an uncontrolled factor in humanoid robot operational environment. According to our previous research [6], the HSI color space is less affected to variations in light intensity comparing to other color spaces. To solve the first problem we used HSI color space and for the second one, we deployed autonomous color look-up table which the TT-UT Austin villa team has already implemented [7].

Edge detection in real time is one of the challenges of humanoid robots. There are several solutions for solving this problem. In embedded system we must find the best solution that has low processing overhead. One way for reducing the processing load overhead is using evolutionary algorithm for this purpose. We are trying to use PSO algorithm in edge detection in Darwin robot.

5 World Modeling

World model is a key component in intelligent and autonomous robots. Modeling the system consist of a model for each static and dynamic object in the field of play. These models are formed by the incoming data from the sensors of the robot. Due to the noise and uncertainty of observations and limitations in humanoid sensors, tracking the surrounding environment of the robot is an important challenge. This year we have implemented models for self-localization and ball tracking and we are working on modeling obstacles.

Self-localization

An essential capability of a soccer playing robot is to robustly and accurately estimate its pose on the field. With respect to the limited field of view and limitation in robot sensors, tracking the pose is a complex problem. Last year we utilized SDMCL [8] to estimate the robot pose. Although this model has a good accuracy and low cost, but suffer from low stability. In [9], a hybrid approach based on fuzzy grid and EKF is presented that has good stability and high accuracy as well. In the same manner we combined MCL and EKF methods. Firstly, both models are initialized with maximum uncertainty: MCL particles are distributed uniformly with equal probability; EKF position is set at center of the field, with an uncertainty that covers the entire field. Then, these two models are updated with odometry data and vision measurements. Whenever the differences between MCL and EKF poses exceed a predefined threshold value, and the MCL particles have been converged, the EKF estimation is supposed to be erroneous and EKF is reinitialized with MCL. With this approach we have achieved both accuracy and stability.

Ball tracking

Ball is one of the most important dynamic objects in the field of play that should be tracked by every player. For this purpose we utilize kalman filter. Issues that affect this model are robot odometry, ball moving, kicking the ball and vision measurements. Moreover, uncertainty of the model is increased when the robot does not detect the ball. In situations where the uncertainty is high, we refine the model with teammate cooperations.

6 Behavior Control

Due to the essence of AI as our super-field there is a spectrum of problems and their relevant spectrum of solutions. For handling this spectrum and with respect to the hierarchical structure of these problems (and solutions) as are described in [10] we have defined a simple hierarchical structure constituted from three layers. The topmost layer namely *Game* is a simple equivalent of higher level mind of player (maybe coach) and its main duty is the management of the game state during the match. The second layer which is labeled as *Player* is a simple model of player thought and is responsible for selecting player proper action during the match such as path planning and positioning. The bottom-most layer namely *Body* which is responsible for execution of the selected actions consists of two state machines. Body state machine for player's body actions and Head state machine for player's head actions.

The UPennalizer structure is formed based on state machines. The three state machines which are called Game, Body and Head are implemented. The Game state machine is responsible for receiving RoboCup game controller commands. The Body state machine is for player body action and main decision making during the match. The Head state machine handles head's actions during the match.

As our contributions in the behavior, firstly we have applied most of the improvements in the bottom layer as player's action parameter optimization. In second layer we have implemented new path planning algorithm which uses Ferguson splines and particle swarm optimization (PSO) for planning optimum path through obstacles [11]. The selected algorithm reaches the best existing path during at most 60 iterations and a swarm size of 20. Also we have proposed an algorithm for increasing the safety attention which is submitted to an international conference for review. A sample image of our developed algorithm output is shown in Fig. 4. Experimental results indicate that in 92% of the configurations our new proposed method plans a path considering both length and safety. This means we select the shortest path if it is safe enough or the safest path with a maximum of 14% overhead in the path length. And in the remaining 8% of cases it is possible to plan safer or the safest path with the average length overhead of 11% or 25% respectively. Our new algorithm is also is submitted to an international conference for review.

In the most top layer we are developing a higher semantically hierarchy layer which satisfies our system view point. The concentration of our future work is in the top layer which can provide a basis of game strategy development. In Player layer we will implement other player-level algorithms for problems like positioning. Also optimization of implemented path planning algorithm is in progress as explained.

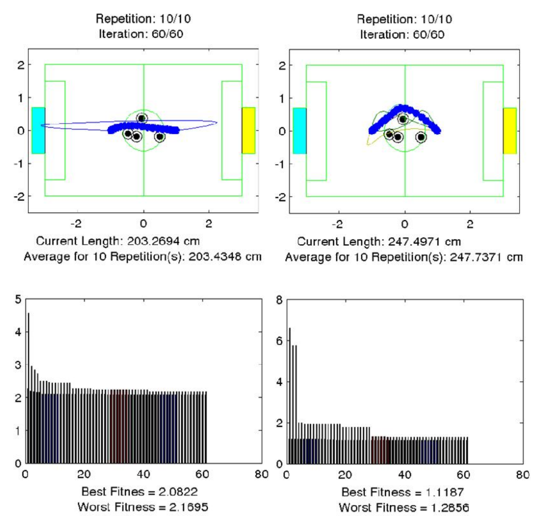


Fig. 4. This is a sample of our new algorithm output. The left column shows previous method and right column the new developed method. The figure in top shows space configuration and the bottom one shows convergence of fitness function diagram. While previous algorithm planes a path through obstacles our algorithm planes a safer path which is about 20% longer.

7 Conclusion

In this paper we have presented the specifications of the hardware and software of MRL kidsize humanoid robot system developed under the RoboCup 2013 rules. MRL commits to participate in RoboCup 2013 in in Eindhoven, the Netherlands with further enhanced hardware and software based on the achievements of previous year and also commits to introduce a referee familiar with the rules of the Humanoid League.

We use DARwIn-OP as our base platform and we are working on this platform with about totally 20 graduate and undergraduate students modifying and optimizing the platform in vision, motion control, world modeling, self-localization, behavior, embedded control board, and also the robot embedded operating system as is discussed in the related sections. Up to now we have 3 published and 2 submitted papers in the related research fields.

References

[1] DARwIN OP, http://www.romela.org/main/DARwIn_OP:_Open_Platform_ Humanoid_Robot_for_Research_and_Education, retrieved Jan 2013.

[2] Upenn RoboCup: Code, available online: https://fling.seas.upenn.edu/~robocup/wiki/index.php?n=Main.Code, retrieved Jan 2012.

[3] Shen Wang, Yu Chen, Wei Jiang, Peng Li, Ting Dai, Yan Cui "Fairness and Interactivity of Three CPU Schedulers in Linux" *In 15th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, RTCSA,* pp. 172-177, 2009

[4] N. Shafii, S. Aslani, O. M. Nezami and S. Shiri, "Evolution of biped walking using truncated fourier series and particle swarm optimization," *Springer, Robocup*, pp. 344-354, 2010.

[5] Rafael C. Gonzales and Richard E. Woods, Digital Image Processing, Prentice Hall, 2001.

[6] Mojtaba Ghanbari, Reza Safdari, Ehsan Nazari, and Mostafa E. Salehi "Evaluating the Effect of Intensity Variations on Object Detection in Humanoid Soccer Playing Robots," *in Proceedings the 2013 RSI International Conference on Robotics and Mechatronics (ICROM 2013), in press,* 13 -15 Feb 2013.

[7] M. Sridharan and P. Stone, "Autonomous Planned Color Learning on a Mobile Robot Without Labeled Data," *in Proc. of ICARCV,* pp.1-6, 2006.

[8] Wei Hong, Changjiu Zhou and Yantao Tian, Robust Monte Carlo Localization For Humanoid Soccer Robot, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics,* 2009.

[9] Francisco Martın, Vicente Matellan, Pablo Barrera, and Jose M. Canas, "Localization of legged robots combining a fuzzy-Markov method and a population of extended Kalman filters," *Robotics and Autonomous Systems*, vol. 55, pp. 870-880, 2007.

[10] Sven Behnke and Jorg Stuckler, "Hierarchical reactive control for a team of humanoid soccer robots," *IEEE-RAS International Conference on Humanoid Robots,* pp. 622-629, 2007.

[11] Wu Xianxiang, Ming Yan and Wang Juan, "An improved Path Planning Approach Based on Particle Swarm Optimization," *Hybrid Intelligent Systems (HIS),* pp. 157-161, 2011.