

IUIM3D RoboCup 2013 Team Description Paper

Hojjat Nikan Meghraze, Hedayat Vatankhah,
Karim Moradi, and HosseinAli Zare

Iran University of Industries & Mines (Tehran Center Branch),
Tehran, Iran

Abstract. In this paper motion control of an intelligent humanoid robot is considered. For each kind of movement initial trajectories are created by an offline method. After applying these trajectories to the robot and imposing the stability tests on them, the necessary modifications and corrections is done to these trajectories and it is guaranteed that the robot can do movement without falling down or tip over. The motion control of the soccer agents of IUIM3D team is also considered. We developed a system to represent action trajectories with two major component; trajectory generation system and trajectory control system which implemented by CPG and ZMP repeatedly. For other actions like kicking, a reinforcement learning methodology is being developed. In order to reduce noise and increase accuracy the Kalman Filter (KF) is used in sensing data. Also, we are developing a facilitating software called Simspark Toolkit, based on QT4 which provides some useful functionalities for researchers.

1 Introduction

Biped robots have better mobility than conventional wheeled robots, especially for moving on rough and uneven terrain. Study of these robots and their stability has been the main focus of too many researchers in the last decades.

Robocup competitions especially humanoid, standard and 3d soccer simulation leagues have provided researchers with a great platform to test the stability and advantages of their algorithms on humanoid locomotion and motion control. IUIM3D soccer simulation team presents our research results on locomotion and motion control of humanoid robots specially on Hoap and Nao.

Because of interdisciplinary and challenging characteristics of the humanoids, many controllers have been developed throughout three past decades.

Currently our agents use a walking algorithm based on off-line trajectory which is introduced in the next section. But also we are working on other approaches to generate more stable and controllable actions.

The approach which is used to generate actions should be able to produce stable and perturbation resistant actions. To achieve this goal, we developed a Central Pattern Generators (CPG) library and combined this by Zero Moment Point (ZMP), most famous mechanical method, to reach a technique to mimic human internal mechanisms of online trajectory production.

Tuning parameters of such a system is hard and mostly experimental. We solve this challenging problem by using Comprehensive Learning Particle Swarm Optimization (CLPSO).

We hadn't reached a complete stable walk via this approach yet but the current results provide us with promising feedbacks.

For the kick action that highly needs adaptation to the environment changes, like changes in the ball position, we are implementing a Reinforcement Learning (RL) methodology to learn the path between different robot states. A state is said to be the set of all agent joint angles.

Along this research scheme, we are working on a toolkit called Simspark Toolkit, to ease the process of research in a standard interface. Other researchers on the field, specially those who will use Simspark simulation environment can easily use and extend the toolkit as they need.

In the next section, we will present our work on robot's actions. In section 3, Simspark Toolkit is introduced. Finally, we will explain some ideas to work on, in a near future.

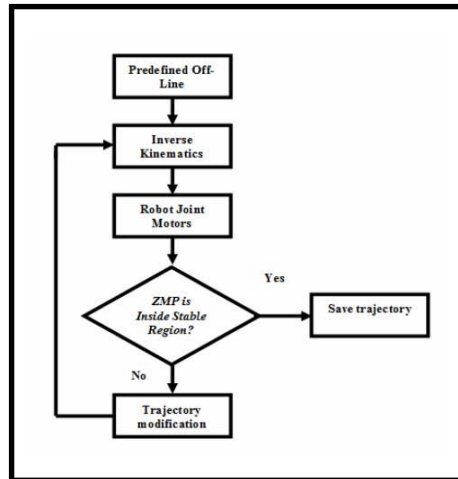


Fig. 1. Block diagram of the first phase

2 Generating Actions

2.1 Off-line trajectories

In humanoid, for each kind of movement a predefined trajectory is needed. If both foot and hip trajectories are known, all joint trajectories of humanoid robot will be determined by the kinematics constraint [1, 2]. To achieve the best stability, it's necessary to have smooth trajectories. Also it's required that at the start of

walking cycle (one of the legs leaves the ground) and end of the cycle, speed of the movement to be zero or near to zero, otherwise it severely causes the robot to tip over.

So considering a Spline function as it has been used in some other works sounds to be logical [1,2]. As a trade-off, a third order Spline function can be used. This function can be implemented by defining some break points in the trajectory and take into account that the first and second derivative of trajectory should be continuous at break points. Fig. 2 shows foot trajectory and its first and second order derivative in z-axis. The trajectory has been implemented using cubic periodic Spline function in MATLAB®. As it can be seen from the figure, the first and second order derivative are continuous.

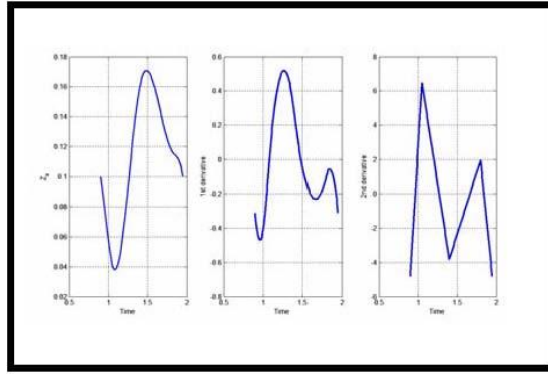


Fig. 2. Foot trajectory and its first and second order derivative in z-axis

2.2 Inverse Kinematics

Next step is applying the inverse kinematics. The attained trajectories should be converted from Cartesian space to joint space. By these equations the Cartesian values in the trajectories are changed to the motor angles and can be applied to the motors.

2.3 Stability Criteria

After issuing commands to the motors, now it's time to evaluate the predefined trajectory. For this means the ZMP point is calculated by the following equations.

$$x_{zmp} = \frac{\sum_{i=0}^n m_i (\ddot{z}_i + g) x_i - \sum_{i=0}^n m_i \ddot{x}_i z_i - \sum_{i=0}^n I_{iy} \ddot{\Omega}_{iy}}{\sum_{i=0}^n m_i (\ddot{z}_i + g)} \quad (1)$$

$$y_{zmp} = \frac{\sum_{i=0}^n m_i (\ddot{z}_i + g) y_i - \sum_{i=0}^n m_i \ddot{y}_i z_i - \sum_{i=0}^n I_{ix} \ddot{\Omega}_{ix}}{\sum_{i=0}^n m_i (\ddot{z}_i + g)} \quad (2)$$

In the above formula, m_i is the mass of link i , I_{ix} and I_{iy} are the inertia of component, Ω_{ix} and Ω_{iy} are the absolute angular components around x and y axis and n is the number of link. For stability this point should be inside the stable region. If this point located outside the stable region we should do some correction in the joint space (trajectories) to prevent this instability.

Because in our robot there is 21 DOF, it is a required to decide what kind of modification in which joint motor can result in best stability. For feasibility one of the hip parameters is considered as a variable. Here after we do all our trajectory modification just on this parameter to achieve the best stability.

2.4 CPG

Trying to implement human-like locomotion capabilities for robots, researchers have always been turn between a technological approach and a biological one. The former relies on concepts and techniques known from robotics. This approach can either be motivated by the attempt to find a better solution or lack of the comprehension of nature. The biological approach on the other hand starts by thoroughly analyzing the motion of animals or humans. These mechanisms are then adapted and translated into algorithms understandable to machines. Biological investigations suggest that locomotion in vertebrates and invertebrate, including human beings, is mainly generated by a rhythm generator which is called Central Pattern Generator (CPG) within the spinal cord. There are some typical model from engineering point of view such as neuron CPG model, Stein's model and leaky-integrated model. We use famous neuron oscillator model was proposed by Matsouka [6] which lay in leaky-integrated group. This oscillator has some good properties such as adaption item. This model composed of two mutually inhibiting neurons called flexor and extensor neurons (Fig. 3).

The implementation of the artificial CPG is usually achieved by explicit differential equations. Equations of Matsouka model is represented bellow.

$$T_{ri} \frac{dx_i}{dt} + x_i = -a_{ij} + s_i - b_i f_i \quad (3)$$

$$y_i = g(x_i) \quad (4)$$

$$T_{ai} \frac{df_i}{dt} + f_i = y_i^q \quad (5)$$

The use of CPGs offer certain advantages, like robust control of the joints and easy adjustment of walking speed and step length, also no dynamics model of the robot or the environment is needed. But also it has several drawbacks, such as a high number of parameters that have to be specified; and it is rather difficult to determine the appropriate parameter settings for the oscillator network in the CPG in order to generate a suitable pattern for control of robot's walking. In other words there is no methodology to tune the value of the parameters of CPG. In the previous work it was implemented in matlab and output of it sent to agent by a socket port [our TDP in Singapore competition]. In the current work it implemented in C++ languages and lay in team code. This approach

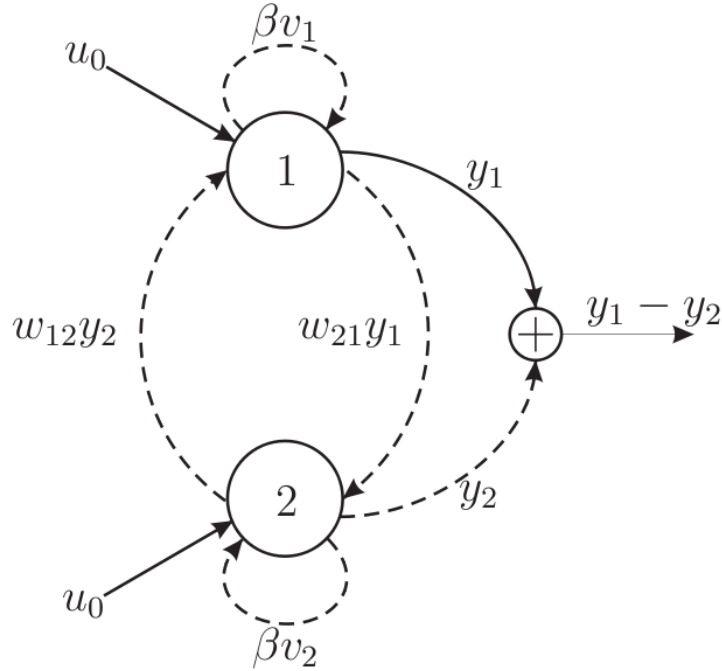


Fig. 3. A half center Central Pattern Generator

helps us to develop on-line learning instead of off-line learning in the previous experience, and also use feedback for adaptation.

For tuning the parameters of CPG, evolutionary computation methods such as Genetic Algorithm (GA), multi-objective Genetic Algorithm and so on are often used to optimize the parameters. Convergence is an important factor in the selection of a suitable method to prevent a robot from numerous iterations of the algorithm. Particle Swarm Optimization (PSO) [4] is a population based optimization method proposed by Kennedy and Eberhart in 1995 and is inspired by the social behavior of natural populations such as birds or fish. The main advantages of PSO are that it is simple to understand, easy to implement and quick in convergence compared to other global optimization algorithms such as Genetic Algorithms (GA) or Simulated Annealing (SA). In our first experience, we tried the canonical PSO, but this method could easily stick to a local minimum. We can get rid of the local minimum by using an enhanced method like Comprehensive Learning Particle Swarm Optimization (CLPSO) which we are still working on it.

Another successful humanoid walking implementation is based on mechanical approach, especially Zero Moment Point (ZMP). The ZMP is a point on the ground where the moments about two axes formed on the ground are zero. The basic concept of the ZMP criterion is that stability is guaranteed if the

ZMP is located within the support polygon formed by foot and ground. This method has been successfully implemented on a lot of humanoid robot. The most famous robots which have been developed by this method are QRIO and ASIMO. The most important disadvantage of all strategies which are based on the ZMP criterion is the need for an accurate model of the robot and its environment.

In the latest work we used hybrid controller based on both mentioned methods and combined advantages of them. The hybrid controller composed of two independent component; trajectory generator system and trajectory controller system. The trajectory generator system is composed of nonlinear oscillators based on Matsuoka model which generate desired trajectories of the joints. The trajectory control system is devised to keep the robot stable. Also, it is responsible to track the commanded motions which are generated by the trajectory generator system. The controller is fully described in [5] which was submitted to the Journal of Mechanical Science and Technology (JMST) and revised three times. We hope this article pressed as soon as possible. We also constructed a library and until now some actions such as straight walking have been successfully implemented by this and want to release this library during the competition.

3 Kalman Filter

As you know each real system includes of component of noise. If trigonometry is used to calculate the robot's position it can have a large error and can change significantly from frame to frame depending on the measurement at the time. This makes the robot appear as if it is "fast walking" around the field. The Kalman Filter [3] is a smarter way to integrate measurement data into an estimate by recognizing that measurements are noisy and that sometimes they should ignored or have only a small effect on the state estimate. It smooths out the effects of noise in the state variable being estimated by incorporating more information from reliable data than from unreliable data. The user can tell the Kalman Filter how much noise there is in the system (including the measurements, the inputs and the model) and it calculates an estimate of the position taking the noise into account. It is less likely to incorporate one bad measurement if the current position confidence is high.

The Kalman Filter algorithm also makes it very easy to combine measurements from different sources (such as vision measurements and locomotion data) and different times (updates as a robot is walking).

4 The Kick Action

Generating adaptive and online trajectories for special actions of a robot is an important and challenging issue in humanoid robots. We apply a novel method for online generation of an adaptive trajectory for the kick action of a humanoid robot using reinforcement learning. We obtained important joints for a kick action by visual inspection of human kick and statistical analysis of kick actions

of humanoid robot models in a simulated 3D soccer environment. We reduced the search space of the applied reinforcement learning algorithm by imposing some simplifications and restrictions. Finally we are employing a neural network to estimate the value function of the reinforcement learning algorithm.

5 Simspark Toolkit

Simspark Toolkit is an application software developed to extend the capabilities and to ease the use of Simspark. It has features that let the user to gain better results in less time. An interesting point about this software is its implementation and development under QT4 that permits interested developers to implement their extra features and changes easily to extend the software.

The Simspark toolkit has the ability of sending direct commands to the server. This feature enables the user to manually control the robot and change the angle of each joint separately (Fig. 4).

The Toolkit has an instance of a 2D monitor, a 3D monitor and their corresponding configuration can be changed easily. In the 2D monitor, one can drag and move the agent to set its position in the 3D environment. Also ball position and agents' formation can be easily changed via the 2D monitor.

Configuration of the settings can be done in manual or automatic mode. In the automatic mode, the server default values will be used; otherwise, the user can set configuration parameters manually.

Also, in 2D monitor, lines and polygons can be added to show the selected agent's frustum of view and corresponding distances.

Another interesting feature of the toolkit is its capability to show the requested data as plots (Fig. 5). Also, analytical information of such data will be shown. The data can be read from a local or can be read online from local or remote computer via TCP/IP connection

6 Future Work

The Kalman Filter equations above assume that the relationships between measurements, inputs and state variables are all linear. This is not always the case and a sometimes a different version of the Kalman Filter known as the Extended Kalman Filter (EKF) must be used. For the robots the sensor measurements are a non-linear combination of the state variable vector.

We are planning to use more joints of the robot for the CPG model of walking, to explore all of the search space of walking conditions. For this purpose, we need faster training algorithms to search the problem space, which is the most time consuming part of the system.

We are developing a decision making engine which is used to describe high level actions using low level actions as building block of the high level ones. This engine supports a hierarchical structure of actions and can break a high level action to a set of basic actions to achieve its goal. For example, going behind

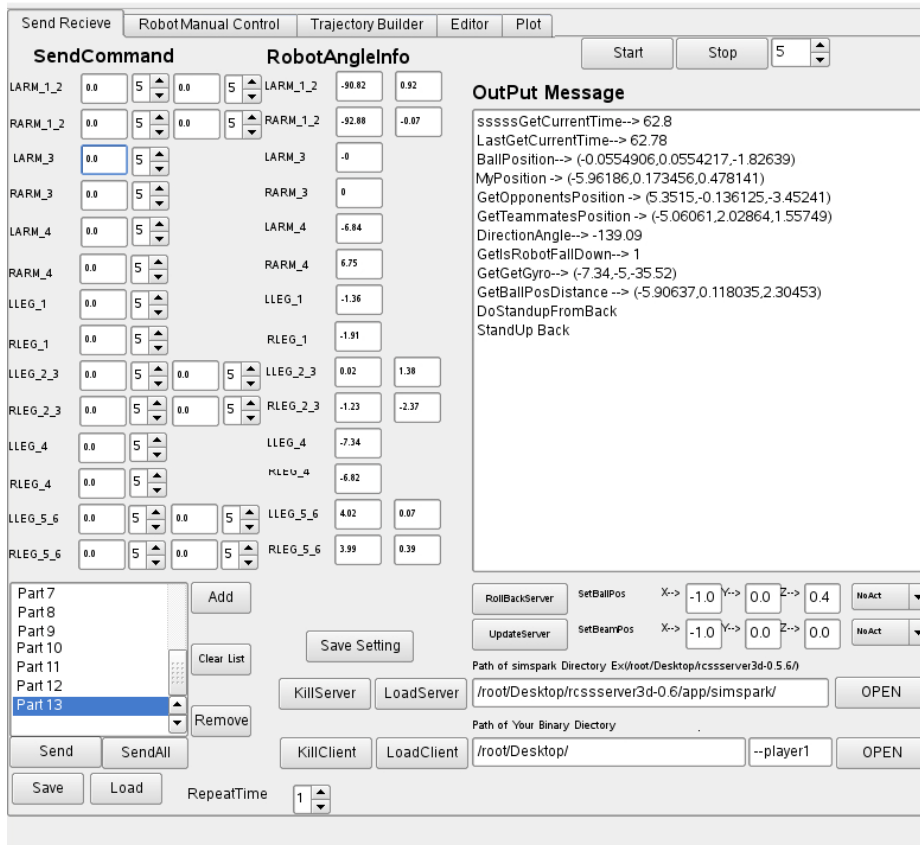


Fig. 4. Robot manual control

the ball will be substituted by low level actions like turning, walking, etc. High level actions can be dened using a GUI interface. For more exibility, a scripting facility will be implemented.

Also, for Simspark Toolkit we want to add joystick control support for easily tuning of the robot joints.

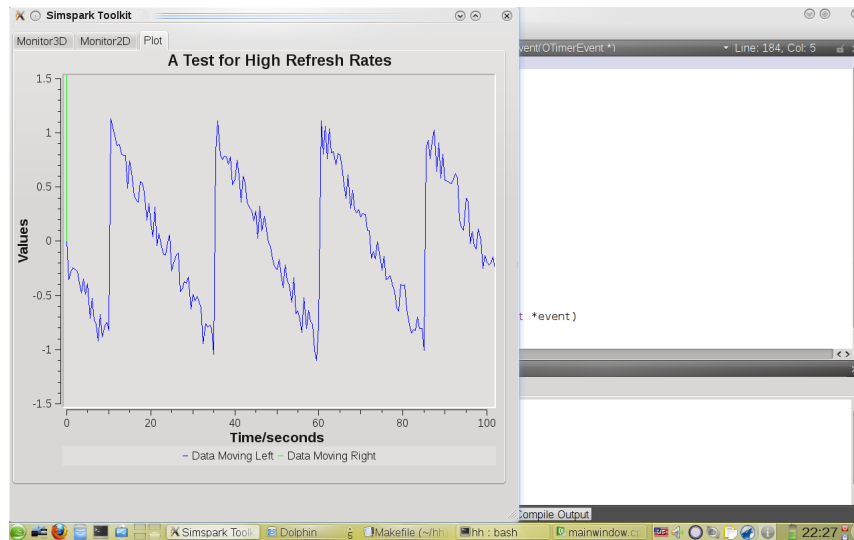


Fig. 5. Plotting in Simspark Toolkit

References

1. Q. Huang, S. Kajita, N. Koyachi, K. Kaneko, K. Yokoi, H. akira, K. Komoriya, K. Tanie. A High Stability, Smooth Walking Pattern for a Biped Robot, Proceeding of the 1999 IEEE, International Conference on Robotic & automation, Detroit, Michigan, May 1999
2. Z. Tang, Ch. Zhou, Z. Sun, Trajectory Planning for Smooth Transition of a Biped Robot, Proceeding of 2003 IEEE, International Conference on Robotic & Automation, Taiwan, September 14-19, 2003
3. P.S. Maybeck. Stochastic Models, Estimation and Control. New York:Academic, 1982, vols, 1 and 2.
4. Shi Y., Eberhart RC. Parameter selection in particle swarm optimization. In: Evolutionary programming VII proceedings of the seventh annual conference on evolutionary programming, pp. 591-600. New York (1998)
5. A. Massah and etc. A Hybrid Controller Based on CPG and ZMP for Biped Locomotion. Journal of Mechanical science and technology
6. K. Matsuoka. Mechanisms of frequency and pattern control in the neural rhythm generators. Biological Cybernetics, 56:345-353, 1987.