TDP - Team Description Paper - RoboIME Humanoid (IEEE HRR) - IME

Luis D. P Farias[1], Fábio Suim Chagas[2] , Matheus Bozza[3], Marcos Paulo[4], Thainá L. H. Lima[5], Ana L. B. Silva[6], José L. O. Schramm[7], Mateus S. Carvalho[8], Mateus L. Silveira[9], Onias C. B. Silveira[10], Paulo F. F. Rosa[11]

Abstract—This document describes the project of RoboIME's Humanoid Team. It contains the methods, materials, and studies developed in order to participate in the IEEE HRR 2019. The overall concepts are in agreement with the rules of the Humanoid Robot Racing 2019. This is the first time that RoboIME joins in this category.

I. INTRODUCTION

The RoboIME team is a small group of the IME (Instituto Militar de Engenharia) who has participated in the LARC (Latin American Robotics Competition) since 2010 and has managed to achieve many good results in the categories in which it has joined. Mainly in the category SSL, where the team was consecrated champion in Larc 2017 and vice-champion in Larc 2018 and Robocup 2018 division B. Particularly, It is the first time the team is focused in IEEE Humanoid Robot Racing (HRR), which demanded much study, research and experiments. Although the group is new in this branch of competition, due to the exchange of experience with members of the other categories, the team is making fast progress to do our main objective for that first competition that is to have a robot with a stable walk.

II. HARDWARE

A. Mechanical Project

Initially was made a schematic of robot's parts, specifying which joints of the human the robot would also have. After studies, it was defined that our first robot would have 10 motors for the movement of the legs, 6 for the arms and 2 for the head. As you can see in figures 1 and 2.

The mechanical design was developed looking for similar proportions of a human being but limited by fixed components such as the battery, the motors and mainly by the rules of competition.

[1]University undergraduate student at Institute Military of Engineer, Rio de Janeiro, Rio de Janeiro, ˜ Brazil, {luisdavidxp, aninhabusesilva, mateusouza39, mateus.l.silveira, limarcospap, thainalucciola }@gmail.com and {m-bozza, lauroschramm}@hotmail.com

[2] Ph.D. student at Institute Military of Engineer, Rio de Janeiro, Rio de Janeiro, Brazil fsuim@yahoo.com.br

[3] Ph.D. and Professor at Institute Military of Engineer, Rio de Janeiro, Rio de Janeiro, Brazil rpauloime@gmail.com
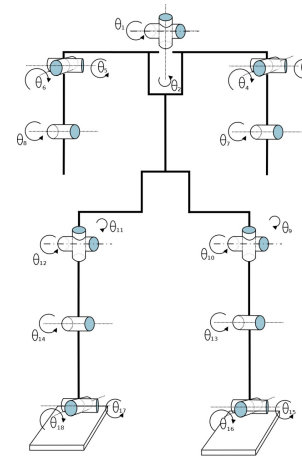
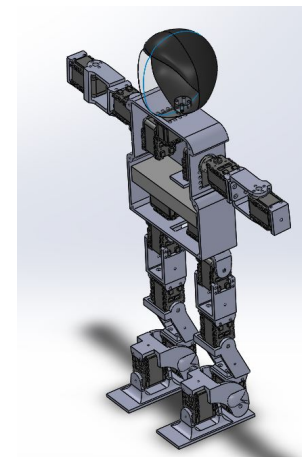Fig. 1.   Schematic of the robot showing its active joints.

Fig. 2.   Isometric robot's view.

B. Motor

The engine purchased for use in the robot is the DYNAMIXEL AX-12A. The motor that will be used has two degrees of freedom and the position control in the motor is carried out by a built-in micro-controller, so that its main controller is free to perform other functions. Thus, it was chosen because it allows feedback of some parameters, such as position, temperature, load, input voltage among others[1]. It is physically constituted of engineering plastic and its dimensions are found in figure 3:

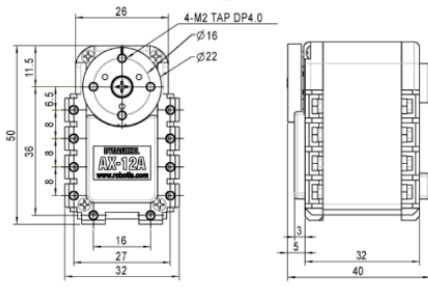The engine specifications are available in the table II-B:

Fig. 3. Technical drawing of the motor with dimensions

ITEM	SPECIFICATIONS
Weight	53.5g (AX-12/AX-12+), 54.6g (AX-12A)
Dimension	32mm * 50mm * 40mm
Resolution	0.29^o
Gear Reduction Ratio	254:1
Stall Torque	1.5N.m (at 12.0V, 1.5A)
No load speed	59rpm (at 12V)
Running Degree	$0^o \sim 300^o$ / Endless Turn
Running Temperature	$-5^oC \sim +70^oC$
Voltage	$9 \sim 12V$ (Recommended Voltage 11.1V)
Command Signal	Digital Packet
Link (Physical)	TTL Level Multi Drop
ID	254 ID ($0 \sim 253$)
Communication Speed	$7343bps \sim 1Mbps$
Feedback	Position, Temperature, Load, Input Voltage, etc.
Material	Engineering Plastic

TABLE I

ENGINE SPECIFICATION.

C. Central Process Unit

The board to be used is Raspberry Pi 3[2], which runs the Ubuntu, Raspbian and other distributions of Linux and allows programming in various languages such as Scratch and Python. Table II-C shows some specs.

ITEM	SPECIFICATIONS
SoC	Broadcom BCM2837
CPU	$4\times$ ARM Cortex-A53, 1.2GHz
GPU	Broadcom VideoCore IV
RAM	1GB LPDDR2 (900 MHz)
Networking	10/100 Ethernet, 2.4GHz 802.11n wireless
Bluetooth	Bluetooth 4.1 Classic, Bluetooth Low Energy
Storage	microSD
GPIO	40-pin header, populated
Ports	HDMI, 3.5mm analogue audio-video jack, $4\times$ USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)

TABLE II

RASPBERRY PI 3 SPECIFICATIONS.

D. Sensors

The following is a description of the sensors that will be used in the robot:

1) Force and pressure sensor: Force Sensitive Resistor (FSR) is a sensor that allows detecting the physical pressure, extrusion, and weight. They are simple to use and low cost. They are based on the change of resistance according to pressure: when no pressure sensor seems an infinite resistance, with increasing pressure the resistance has decreased. The graph shows the resistance versus pressure measurement by the sensor.
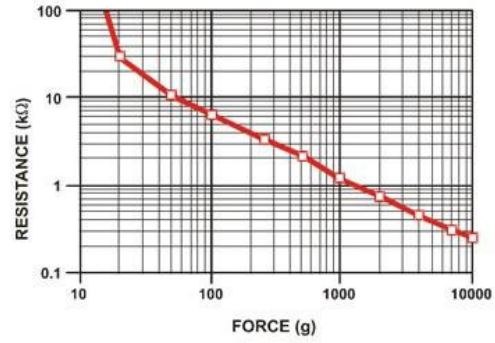

Fig. 4. Graph of resistance according to pressure

2) Accelerometer and Gyroscope: MPU9250 is a chip that has three modules, a 3-axis accelerometer, a 3-axis gyroscope and a 3-axis magnetometer (magnetic field sensor)[3]. Combined with the BMP180 pressure sensor, which is characterized by low power demand, reduced design cost and easy connection, the module uses the I2C interface to communicate with the micro-controller. Table II-D.2 shows some specifications.

ITEM	SPECIFICATIONS
communication	I2C interface
Power supply	3.3V
Dimensions of the module	15 x 15 x 3.5mm (without the pins)

TABLE III

MPU9250 SPECIFICATIONS.

3- Distance sensor:
The laser sensor can be used as a distance-measuring tool to detect the distance from it to obstacles. It can also be used as an eye of a robot, it provides distance information for robotic safe avoidance and route selection. It supports for system automation in a variety of machine control scenarios. It has a small size, lightweight and, low power consumption. TF Mini's maximum detection distance is 12 meters, see Figure 5. TF Mini also supports 100Hz sampling resolution, within 6 meters, its accuracy is within 4cm, 6×12 meters, accuracy within 6 cm. FOV of 2.3 degrees. Its anti-interference is strong, and can work in outdoor light; the overall weight is 4.7g.
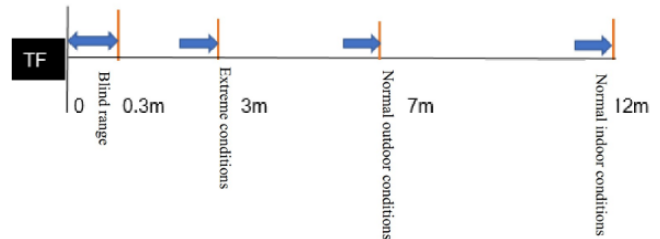

Fig. 5. Measurement range schematic diagram

E. Battery

The battery purchased for use in the robot is the TUR-NIGY 5000mAh 3S 20C Lipo Pack with XT-60. It is equipped with reinforced discharge conductors to minimize resistance and sustain high current loads, XT-60 connectors and JST-XH style balance connectors.

Fig. 6.   Battery

Battery specifications are available in the table II-E:

ITEM	SPECIFICATIONS
Recommended age	12 years and up
Weight	432 g
Constant discharge	20C
Peak discharge	30C
Capacity	5000mAh
Configuration	3S1P \ 11.1 V \ 3 Cell

TABLE IV

BATTERY SPECIFICATION.

F. Camera

In order to get the images from the environment, we plan to use a camera module for Raspberry Pi. It has 5mp and is capable to offer 2592 x 1944 resolution for static images and supports 1080p30, 720p60 and 640x480p60/90 for videos. It weighs nearly 3 grams and is able to manage 30 fps 1080p video with codec H.264 (AVC) and until 90 fps in the VGA mode. Its sensor has the dimensions 3,67 x 2,74 millimeters and a visual field of 2,0 x 1,33 m, at 2 m.

III. SIMULATOR

To elaborate the joints path planning we developed a simulator by using Matlab's Simulink module software, In order to test it quickly and safely.

For this we used the Simscape Multibody module that allows importing the robot from Solid Works and an additional library, Simscape Multibody Contact Forces Library, to simulate the contact between the robot's feet and the ground.

As it shows in figure 7. The simulator code is divided into 3 main parts:

- The robot assembly is responsible for the physical part of the robot.
- The movement part describes the movement calculated by the learning program that generates the robot's trajectory.

- And finally the part of the analysis of stability that will send information to this same program.
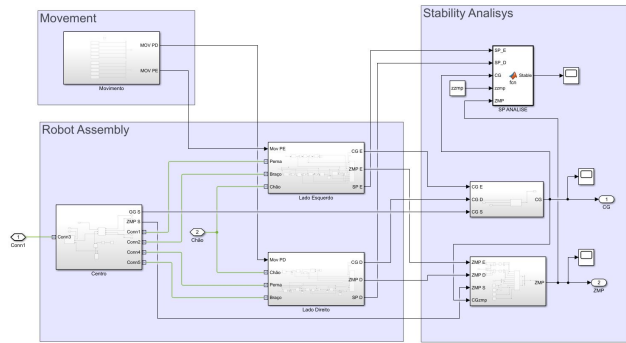

Fig. 7.   Simulator's Simulink code

While the robot is being modeled, a simplistic model was used to test the simulator and the learning code, but once it is finished it will be replaced by the actual model that was shown. A view of the simulator is shown in figure 8.
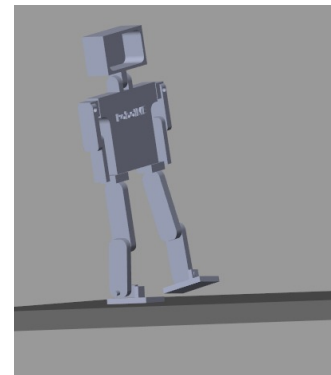

Fig. 8.   Simulator's Mechanic's Explorer view

IV. MOTION PLANNING

For the first model of walking, we adopted the model of statically stable movement, in this model the CoM(Center of Mass) always remains within the SP[1] (Support Polygon) during the entire motion. This implies that if the movement is stopped, the robot will remain in a stable position. This kind of stable movement is only valid for low walking velocities, which impose also low angular velocities in the joints[4].

For this, the full-body posture goal method was used. In this method, a set of angles of the robot's joints are defined, which determine its configuration so that each of these positions is stable and its sequence forms a walk. The algorithm consists in going from one configuration to another in a random way and verifies if this way was satisfactory, so far we only check if the robot falls or not. If the robot falls, another way it is chosen for him to go from one configuration to another. This repeats itself until it reaches the final configuration, thus completing its walk.

[1]Support Polygon is the convex polygon formed by the floor support points during robot walking.

Because this algorithm requires several tests, the simulator becomes essential. A schematic of the algorithm is shown in the figure 9.
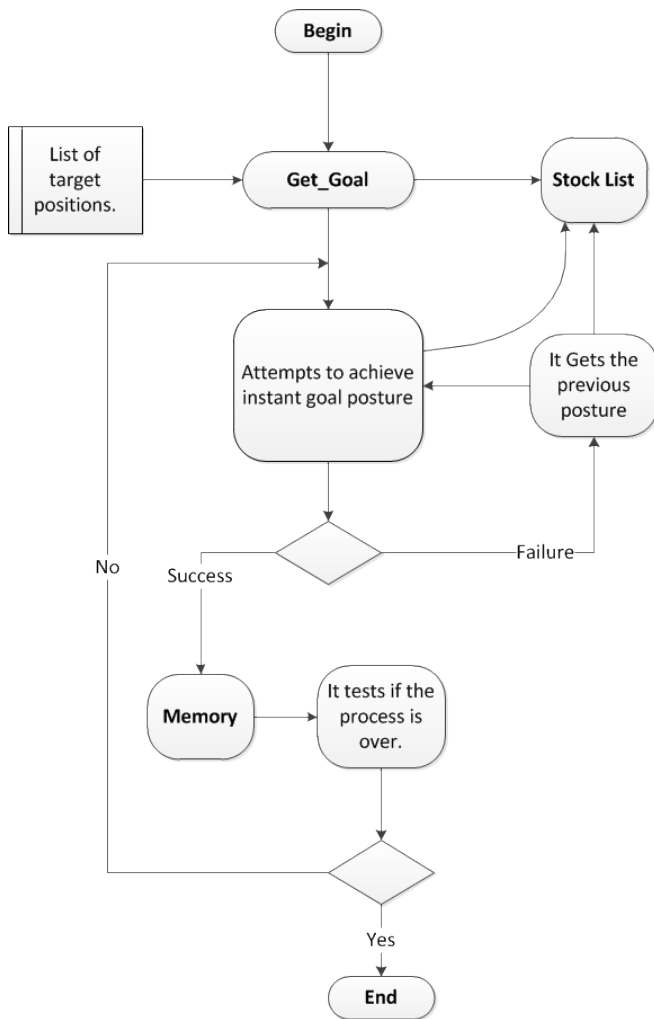

Fig. 9. Robot learning algorithm flowchart

V. COMPUTATIONAL VISION

The Computational Vision will be developed with the help of the OpenCV library, which is largely used due to its completeness. It enables us to identify the borders present into the image using for this purpose the FAST(Features from Accelerated Segment Test) algorithm which works, put simply, identifying if a pixel neighborhood (figure 10) is brighter, bigger intensity, or darker, lower intensity, than the chosen pixel and based in the result, deciding if it is part of a border. It will assist the robot deciding whether to change its path or not once that there will be obstacles in the way.

VI. CONCLUSIONS

This article shows RoboIME's work to participate in the IEEE Humanoid Racing category. We developed our own simulator, walk algorithm, and all robot design. We're starting to ride our robot and plumb the raspberry with the walk code. For the competition, we hope to have a complete
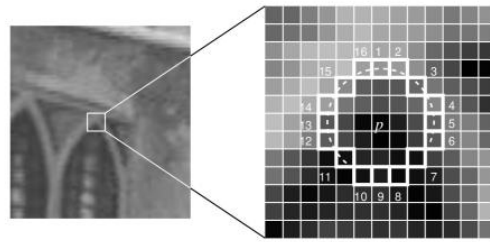

Fig. 10. Pixel analyzed neighborhood

robot, walking and, whether it's possible, recognizing the obstacles and overcoming them.

REFERENCES

REFERENCES

1 TROSSENROBOTICS. *Dynamixel AX-12 Datasheet*. 2019. ⟨https://www.trossenrobotics.com/images/productdownloads/AX-12(English).pdf\⟩.

2 LTD., R. P. T. *Raspberry Pi Compute Module 3+ and Raspberry Pi Compute Module 3+ Lite*. 2019. ⟨https://www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi_DATA_CM3plus_1p0.pdf⟩.

3 INVENSENSE. *MPU-9250 Product Specification Revision 1.1*. 2019. ⟨https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf⟩.

4 DEKKER, M. Zero-moment point method for stable biped walking. July 2009.

Robot Scheme		Dimensions
		Maximum height
		57,26 cm
		Maximum Width
		45,95 cm
		Leg Length
		24,23 cm
		Material
		PLA
		Board Specification
		Raspberry pi 3
Motors	Battery	Operational System
AX12A	TURNIGY 5000mAh	Raspbian
Internal Sensor		
IMU	Measures acelerations and rotations	
Encoder	Informs the velocity and position of the motors	
External Sensors		
TF Mini LiDARLaser Range Sensor	Measure Distances	
FSR 402 Force Sensor	Detect Mechanical Force and Pressure	
Camera	Detect Obstacles	