Assignment Cover Sheet

Faculty of Science, Engineering and Built Environment

NAME: SATVIK SHARMA

STUDENT ID: 218595095

UNIT CODE: SIT122

ASSIGNMENT/PRAC No.: 2

ASSIGNMENT/PRAC NAME: Movement Control in Robotic Systems

DUE DATE: 12/04/2019

Plagiarism and collusion

Plagiarism occurs when a student passes off as the student's own work, or copies without acknowledgment as to its authorship, the work of any other person.

Collusion occurs when a student obtains the agreement of another person for a fraudulent purpose with the intent of obtaining an advantage in submitting an assignment or other work

Declaration

I certify that the attached work is entirely my own (or where submitted to meet the requirements of an approved group assignment is the work of the group), except where work quoted or paraphrased is acknowledged in the text. I also certify that it has not been submitted for assessment in any other unit or course.

I agree that Deakin University may make and retain copies of this work for the purposes of marking and review, and may submit this work to an external plagiarism-detection service who may retain a copy for future plagiarism detection but will not release it or use it for any other purpose.

DATE: 12/04/2019

An assignment will not be accepted for assessment if the declaration appearing above has not been duly completed by the author.

Report on Movement Control in Robotic Systems

Problem statement

The most basic feature every living being posses is the gift of movement. Whether it be a lion, a dog or a human being, they all move. Each individual can move from one point to the other by using different speeds, different directions et cetera. To move the NAO robot from one particular point to another can allow the humanoid to become more human-like. Since the robot is having limbs, this means that the robot can walk. Moreover, there are many robots on earth that can walk or even run. The best example will be of the robot Atlas developed by Boston Dynamics (https://www.youtube.com/watch?v=rVlhMGQgDkY&t=37s). if NAO is compared to Atlas, it does not walk as good.

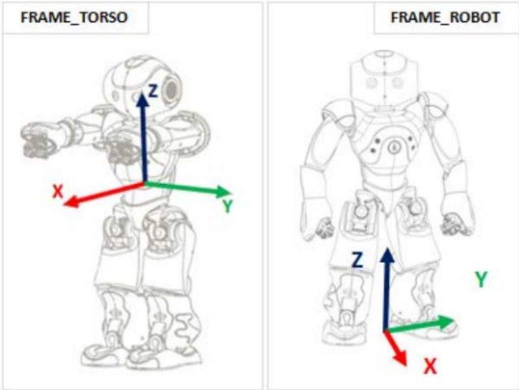
First problem is NAO does not move the precise distance given in the code. For example, if the robot is given to walk a distance of 1 metre, it does not walk the given exact distance. This might be due to different reasons such as friction in the carpet. This is the basic problem that needs to be corrected in the robot so as to make it market friendly.

Second problem that arises is that allow NAO to move in an arc such as a quarter circle with variation in radius. This is due to the reason that robot can move only in forward direction and change its direction according to the angle given in the code.

Background knowledge

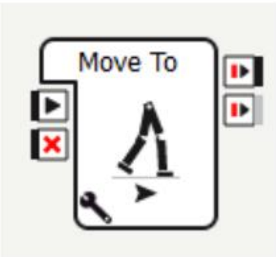
This task is done using Choregraphe 2.1.4 and not using the python. The robot uses a simple dynamic model called the “Linear Inverse Pendulum”, which means feedbacks are given from the joints, which leads to vigorous to small disturbances and upper- body’s oscillatory motion. This allow NAO, the mechanical person, to walk on different surfaces.

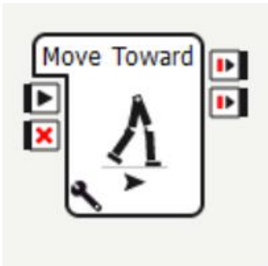
When giving a movement command, it is done in a frame of reference which means that the bot knows the initial and final position in a coordinate frame. The frame of reference used here is the world frame. There are two types of frames i.e. robot frame and the torso frame as depicted in the image below.



These two might look same, but this is only before the humanoid starts walking. The main difference between these two is that the robot frame moves along with the robot and the torso frame remains unchanged.

The next thing to know about is the commands that is to be given to the robot. The most basic command for NAO to walk is move to and move towards inbuilt feature.





In addition to above, one should know about event driven programmes and the one which is used over here is the python script box. If double-clicked on the box, it shows the python script which can be changed according to the need. For further reading, go to http://doc.aldebaran.com/2-5/naoqi/motion/almotion.html

Besides above, one should know about the gait parameters. http://doc.aldebaran.com/2-1/naoqi/motion/control-walk.html#control-walk-gait-parameters-table is the link for the gait parameters used in this report. Code for adding parameters and using them for NAO is given below.

```
mc = [
        ["MaxStepX", 0.01],
        ["MaxStepFrequency", 0.5]
     ]
self.motion.moveTo(0.5,0,0,mc)
```

The mc declares the value of each parameter while self.motion.moveTo allows the robot to move. Inside the bracket the first parameter is the distance x travelled in forward direction, second one is the y direction, third is the rotation along z axis and the fourth one call the parameters given above.

Investigations

To get the idea of movement in NAO, a number of investigations were conducted to get the robot move in a straight line up to a confined distance and moving it in a quarter circle with a variation in radius.

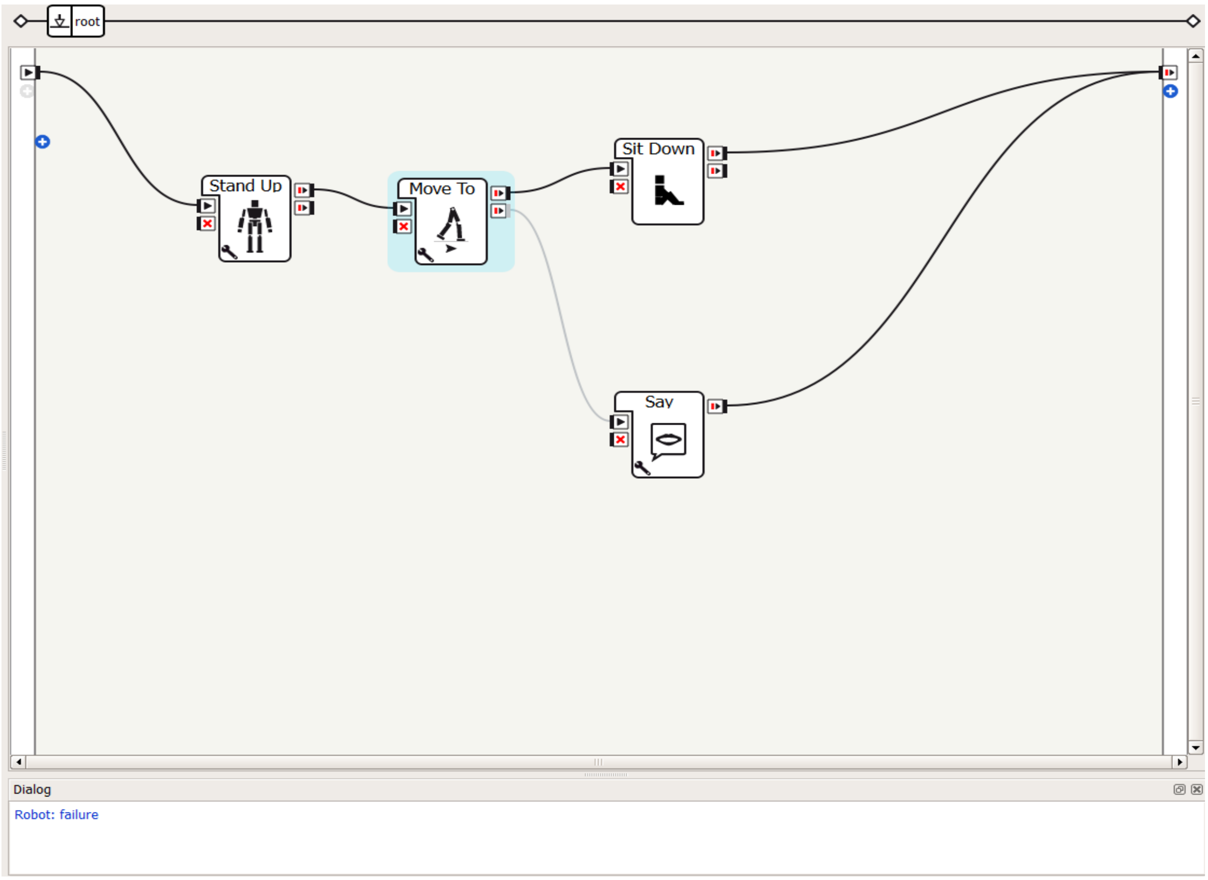
1. **Walking a confined distance**

Objective:

The objective here is to determine whether the robot walk to a stated distance that is 100 centimetres or 1 metre and if it does not walk that much, finding the average distance left for it to move and adding to move to box so that the robot moves 1 metre.

Solution Design:

First of all, place the robot in a proper place and connect it to choregraphe. Make a flow chart as given below.



In the move to box set the distance as (1.0,0.0,0.0)

```
def onUnload(self):
    self.motion.moveToward(1.0, 0.0, 0.0)
```

According to the above design, if the bot moves 1 metre then it will sit down or else it will say failure and keep on walking. But this does not actually happen and the android stops just before completing 1 metre because of the friction on the floor.

Risk assessment and precautions:

The main risk while performing this task is the falling of robot. Moreover, the robot requires a clear space around it, so that it does not tip over. It also requires full attention at each time.

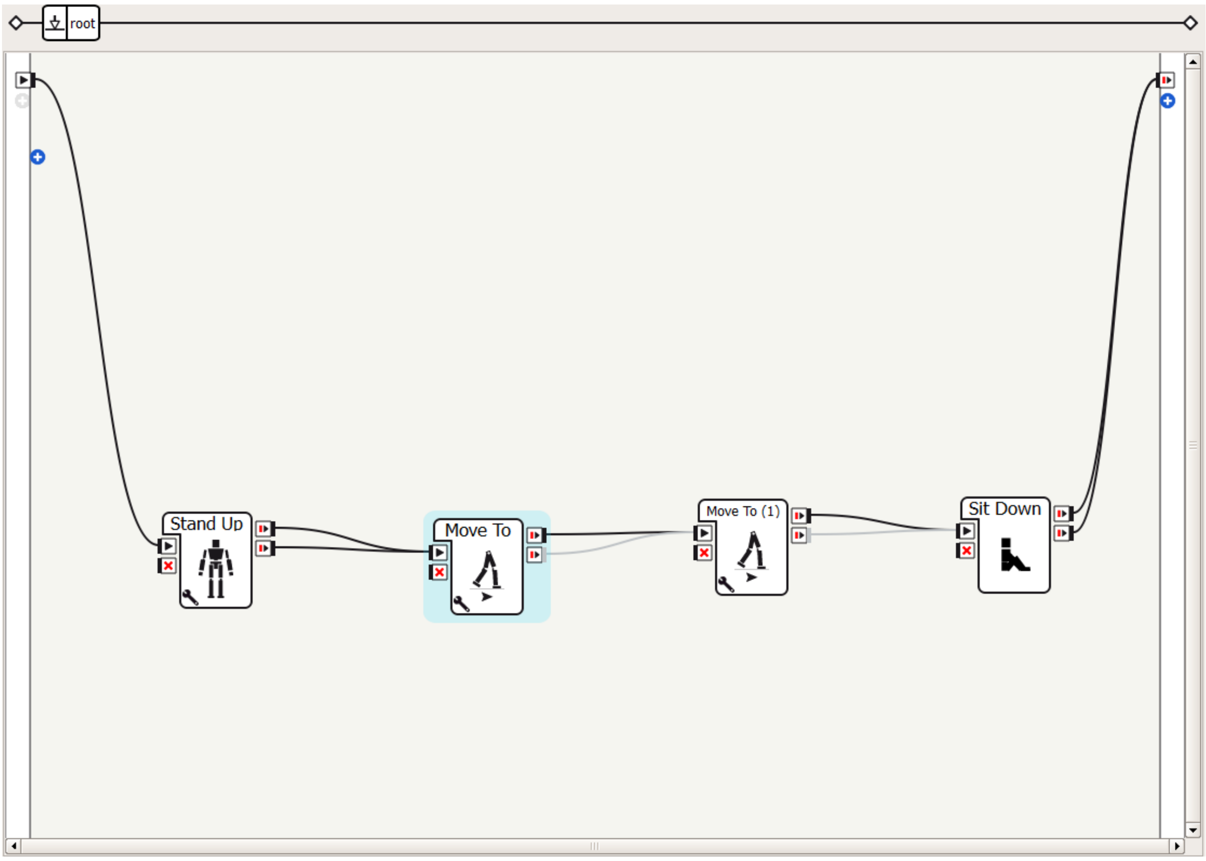
Testing procedure:

After designing the solution, repeat the solution for at least three times and use a measuring tape so as to find how much the robot covers distance. The three readings are given below

1. Reading 1 - 0.95metre
2. Reading 2 - 0.96metre
3. Reading 3 - 0.96metre

   Average distance covered- 0.95metre

The difference between the taken distance and the average distance i.e. 0.05metre should be added as a next 'move to' box.



Outcomes:

The robot moves the confined distance now. Initially there were some problems when the robot stood up, it used to go into different direction, but afterwards it was corrected after understanding NAO's cartesian coordinates.

Reflection:

The main thing to know about the NAO's world frame and robot frame which allow it to move. Furthermore, it has a lot of real-life applications which will allow the robot to mix up with the humans.

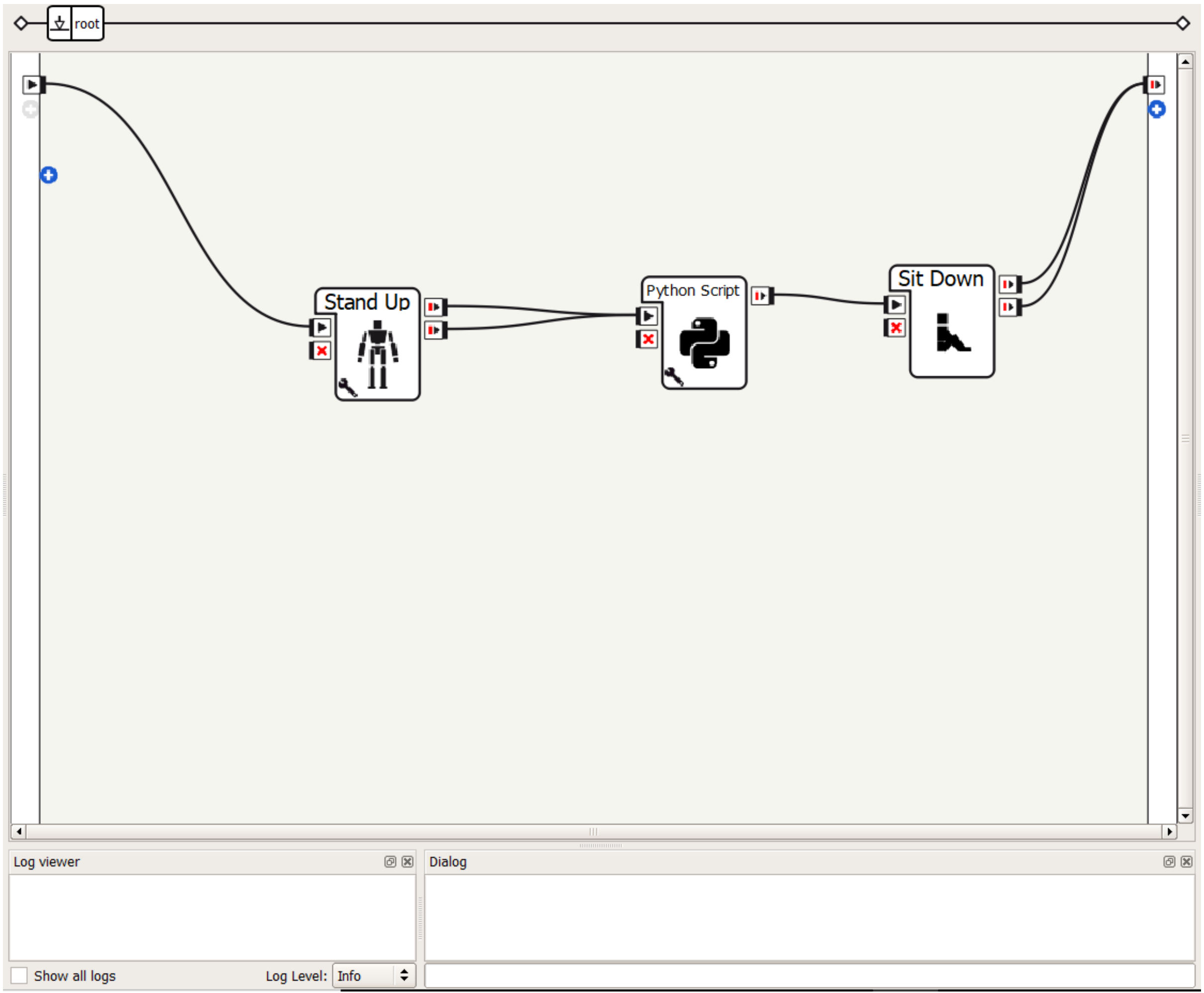
2.Moving NAO in a quarter circle with variation in radius

Objective:

The main purpose of this investigation is to find if NAO moves in a quarter circle properly with variation in radius and if it does not walk properly, adding parameters to it.

Solution design:

After connecting the robot to choregraphe, make a design as given below.



After making the design, double click the python script box and give a code as given below.

```python
class MyClass(GeneratedClass):
    def __init__(self):
        GeneratedClass.__init__(self)

    def onLoad(self):
        #put initialization code here
        pass

    def onUnload(self):
        #put clean-up code here
        pass

    def onInput_onStart(self):
        import almath
        import math
        pi=almath.PI
        self.motion = ALProxy("ALMotion")
        mc=[ ["MaxStepX", 0.04],
          ["MaxStepY", 0.14],
          ["MaxStepTheta", 0.349],
          ["MaxStepFrequency",1.0],
          ["StepHeight", 0.02],
          ["TorsoWx", 0.0],
          ["TorsoWy", 0.0] ]
        self.motion.moveTo(1,1,pi/2,mc)
        self.onStopped()
        pass

    def onInput_onStop(self):
        self.onUnload() #it is recommended to reuse the clean-up as the box is stopped
        self.onStopped() #activate the output of the box
```

The above code is given according to 1 metre.

Risk assessment and precautions:

Since the robot's movement is dealt with parameters, it can easily get a mixture of wrong parameters which can cause unstable walking in the robot and can topple it. In addition to above, the design should be first tested on the virtual robot or in webots.

Testing procedure:

```python
mc=[ ["MaxStepX", 0.04],
  ["MaxStepY", 0.14],
  ["MaxStepTheta", 0.349],
  ["MaxStepFrequency",1.0],
  ["StepHeight", 0.02],
  ["TorsoWx", 0.0],
  ["TorsoWy", 0.0] ]
self.motion.moveTo(1,1,pi/2,mc)
```

The investigation was done on the basis of decreasing radius from 1 metre to 0.7 metre and 0.5 metre. The above code works properly for 1 metre but when the radius is decreased, the chances of robot of falling down increases.

Using the values given in the gait parameter (doc.aldebaran.com/2-1/naoqi/motion/control-walk.html#control-walk-gait-parameters-table), change the values of the parameters accordingly. When given the same parameters as 1 metre, NAO starts wobbling and increases the chances of falling down. But when the parameters are decreased, the robot walks much better.

Outcomes:

After testing, it means that when the radius is small the speed should be less and the step distance should be small. Henceforth the parameters are changed according to the radius and needs to be changed every time in the python script.

Reflection:

The learning outcome in this problem is moving the robot in an arc with variations in speed, radius, step, height, et cetera. This will allow the robot to walk like humans do and can be used for heavy tasks which humans cannot easily do.

Conclusions

The solution to problem is the best solution as far as the documentation is provided. If in near future if the robots are developed as such that the actually detect the surface they are walking on and calculate the distance they have not actually travelled, will be a much better solution.

The second problem, which is NAO moving in a quarter circle. The solution provided is good enough for the robot, as it moves in a quarter circle with variation in radius. In contrast to the given solution, it can be improved if the robot takes the parameters, depending on the radius.

Further correction of these two problems will allow the robots to work domestically and in every day life as the robot's movement will allow it to do a number of tasks.

References

- Deakin university, SIT122 resources
- More documentations from http://doc.aldebaran.com