BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI
HYDERABAD CAMPUS
SECOND SEMESTER 2022-2023
ME F366 LABORATORY PROJECT
PROJECT TITLE & PLAN OF WORK

Date: 07/02/2023

1. Title of the project

   Humanoid Robot Control

2. Need for the study

   To simulate the RMRC algorithm used to numerically solve the inverse kinematics problem for a tree-type robot so that it can be extended to humanoid robots and their control. Also, to evaluate the performance of the algorithm in generating different curves in the end effector space of such robots.

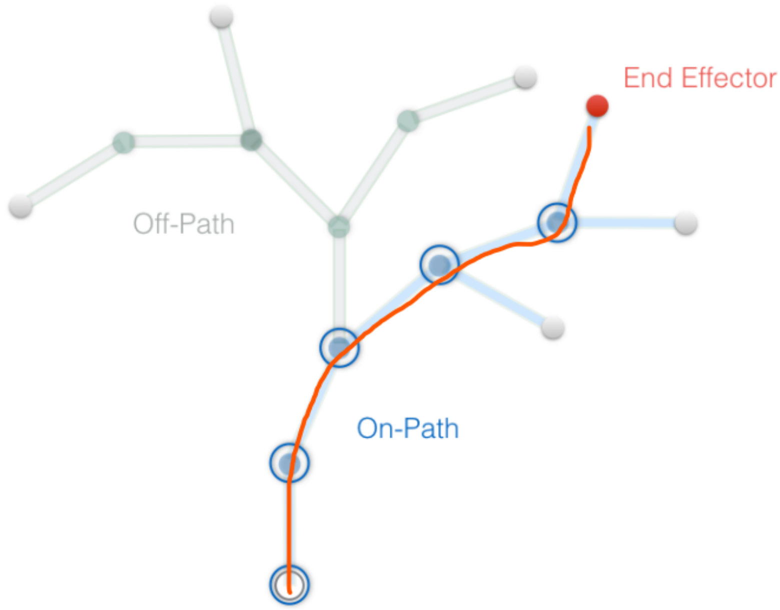
3. Objectives (Bullet points)

   - Model the tree-type robot in MATLAB using structures and classes.
   - Compute the Jacobian of a general tree-type robot for a given end-effector.
   - Develop the RMRC (Resolved Motion Rate Control) of the tree-type robot considering one or more than one end-effectors.
   - Plan the trajectory of the robot using the given control scheme.

4. Literature Review

   Tree-type manipulator:

   A tree-type manipulator is essentially a robotic manipulator with each link connected to one, two, or more than two links and forming a branched/tree-like structure. For such a manipulator, we can equivalently reduce the problem of controlling a given end effector as follows:



Resolved Motion Rate Control (RMRC):

It is a step-by-step method for solving the inverse kinematics problem of a given manipulator. In this method, we locally evaluate the jacobian (relationship between joint space velocity and end effector's velocity) and update it after each step. This instantaneous jacobian is inverted and used to calculate the step in joint space for a small step in the cartesian space.

$$J(\theta(t)) \cdot \dot{\theta}(t) = \dot{x}(t).$$

$$\dot{\theta}(t) = J^{+} \cdot \dot{x}(t) + (I - J^{+}J) \cdot z(t).$$

The type of Jacobian inverse evaluated above is called the Moore-Penrose inverse, and it is a pseudoinverse.

5. Work Plan (Include Detailed Methodology with Time Schedule)

The work plan is divided into the following parts:

a. Understanding the math behind spatial joints (more than 1 dof: screw theory)

b. Modeling tree-type robot:

    1. Create a treeRobot-class capable of handling different types of links (binary, ternary, etc.), different methods for dealing with transformations and kinematics, etc.
    2. Inside the class, create a default link-structure that is modifiable depending on our needs, with different parameters like physical dimensions of the link, type of link, etc.
    3. Using the class methods and the link objects, build a custom tree-robot ready for analysis.

c. Write the RMRC code for controlling the end effectors' positions of the given tree-type robot:

    1. For a given end effector, evaluate what is the effective serial manipulator that needs to be considered for inverse kinematics using dfs (depth-first-search).
    2. Considering a given set of initial conditions, write the algorithm to iteratively evaluate the inverse kinematics of this end effector using the jacobian.

d. Using the RMRC algorithm written above, try to test different trajectories for the tree-type robot:

    1. First, try to make a single end effector trace sample trajectories like a straight line, circle, square, etc.
    2. Then increase a level further and try to make two end effectors trace the same and different trajectories simultaneously.

3. Check the performance of different solutions of inverse kinematics available in RMRC against how each algorithm holds up against singularities.

The Detailed Schedule is as follows:

S. No.	Task to be completed	Deadline
1	Understanding forward, inverse, and differential kinematics of manipulators in terms of twist theory.	14th Feb 2023
2	Create a class capable of creating a general tree-type manipulator with methods like jacobian, transformation, etc.	18th Feb 2023
3	Create a link structure that can be used by the class for creating the robot object.	20th Feb 2023
4	Instantiate the robot object and build a custom tree	20th Feb 2023
5	Write the code for DFS to get the effective serial manipulator to be considered given an end effector.	22nd Feb 2023
8	Understand the RMRC algorithm that is used in iterative solution of inverse kinematics as relevant to screw theory.	28th Feb 2023
7	Implement RMRC first for a single end effector and try simulating for a given set of initial and final conditions	10th Mar/25th Mar 2023
8	Add a second effector and try simulating for a different set of initial and final conditions for the two end effectors	29th Mar 2023
9	Try generating standard trajectories like a straight line, square, circle, etc.	8th April/15th April 2023

10	Vary the type of inverse jacobian computation in the RMRC algorithm and check each methods' performance in generating the desired trajectory	15th April 2023

6. References

(n.d.).

Julien, A., & Nagase, K. (2018). Dynamics and Control of Tree-Type Manipulator Systems using Exponential Coordinates. *Conference on Automation Science and Engineering (CASE).* Munich: IEEE.

Lynch, K. M., & Park, F. C. (2017). *Modern Robotics Mechanics, Planning, And Control.* Cambridge University Press.

Murray, R. M., Li, Z., & Sastry, S. S. (1994). *A Mathematical Introduction to Robotic Manipulation.* CRC Press.

O'Neil, K. A., Chen, Y.-C., & Seng, J. (1997). Removing Singularities of Resolved Motion Rate Control of Mechanisms, Including Self-Motion. *IEEE Transactions on Robotics and Automation*, 741-751.

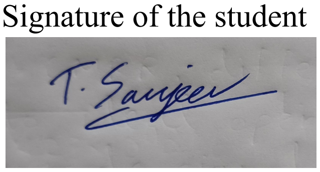
Rabbani, S. A. (2019, November). *A Recipe To Cook Jacobian*. Retrieved from Shahin Rabbani: https://www.shahinrabbani.ca/jacobian

Wang, H., Li, R., Ge, L., & Cao, C. (2014). Comparative Study of Redundancy Based on Resolved Motion Rate Control Algorithm. *International Conference on Information and Automation.* Hailar: IEEE.

7. Expected Knowledge to be gained after completion of the project (Bullet points)

- Tree-type manipulator theory
- Computation of inverse kinematics using a numerical approach like RMRC.
- Given an inverse kinematics solver, different ways of generating a desired trajectory of the end effector.

Signature of the student



Name: Thathapudi Sanjeev Paul Joel
ID No: 2020AAPS0120H