A thesis report on

REDUNDANCY CONTROL WITH POSITION PRIOR ORIENTATION OF SERIAL MANIPULATORS USING THE CONCEPT OF TASK PRIORITY

Submitted in the partial fulfillment of the requirement for
the award of the degree of

MASTER OF ENGINEERING

IN

CAD/CAM & ROBOTICS

Submitted by

Eram Neha

Roll No. 801181010

Under the guidance of

Dr. Ashish Singla

Assistant Professor

Mechanical Engineering Department

Thapar University, Patiala.

MECHANICAL ENGINEERING DEPARTMENT

Thapar University, Patiala

July 2013

CERTIFICATE

I hereby certify that the thesis entitled "**REDUNDANCY CONTROL WITH POSITION PRIOR ORIENTATION OF SERIAL MANIPULATORS USING THE CONCEPT OF TASK PRIORITY**" is an authentic record of my own work carried out in partial fulfillment of the requirements for the award of degree of **Master of Engineering** in **CAD/CAM & Robotics** during session Jan. 2013 - July 2013 at **Thapar University, Patiala** under the guidance of Dr. Ashish Singla (Assistant Professor, MED).

ERAM NEHA
(801181010)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Dr. Ashish Singla
Assistant Professor
Mechanical Engineering Department
Thapar University, Patiala

Countersigned By:-

Dr. S. K. Mohapatra
Senior Professor and
Dean of Academic Affairs
Thapar University, Patiala

Dr. Ajay Batish
Professor and Head
Mechanical Engineering Department
Thapar University, Patiala

ACKNOWLEDGEMENT

First and foremost, I would like to thank God, the Almighty, for His showers of blessings throughout my thesis to complete my work successfully.

I would like to express my deep and sincere gratitude to my thesis supervisor, **Dr. Ashish Singla**, Assistant Professor, Thapar University, Patiala for his continuous support and invaluable guidance throughout my work. His enthusiasm, sincerity, vision and motivation have deeply inspired me. He has taught me the methodology to carry out my thesis and to present my work as clearly as possible. It was a great privilege and honor to work and study under his guidance. I am extremely grateful for what he has offered me.

I would like to thank all the staff members of mechanical engineering department for their support and providing adequate lab facilities. Also, my thanks go to all my friends and classmates, who have supported me to complete my thesis work directly or indirectly.

Lastly, I am extremely grateful to my parents for their love, encouragement, prayers, care and sacrifices, they have offered for educating and preparing me for my future. Also, I express my thanks to my brother for his support and valuable prayers.

ERAM NEHA

ABSTRACT

In this thesis work, the redundancy resolution of serial robot manipulators is performed. Redundant manipulators have the characteristics that there exists infinite solutions to the inverse kinematic problem. Out of those infinite solutions, it is possible to choose certain specific solutions on merit, which can be selected on the basis of some performance criteria or priority of some tasks over the other. These priority subtasks can be position prior to orientation (PPO), torque minimization, obstacle avoidance, singularity avoidance etc. Out of these parameters, this thesis is focused on position prior to orientation. The concept of task priority is used, which is implemented in relation to the inverse kinematics problem of redundant manipulators. A required task is divided into a number of subtasks according to the order of priority. The redundancy resolution is performed to achieve the required orientation of the end effector while tracing a given trajectory.

The entire procedure is formulated using the pseudoinverse of the Jacobian matrix. A number of numerical simulations are performed for different trajectories which includes straight line, rectangular, triangular, circular trajectories to show the efficacy of the redundancy control scheme. Also to show the validity of the formulation, these cases are discussed further by tuning the parameters.

CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

$J(\theta)$	Jacobian matrix
l_1	Length of first link of the manipulator
l_2	Length of second link of the manipulator
c_1	Cosine of the joint angle θ_1
c_{12}	Cosine of the sum of joint angles θ_1 and θ_2
s_1	Sine of the joint angle θ_1
s_{12}	Sine of the sum of joint angles θ_1 and θ_2
W	Measure of manipulability
n	Degree of freedom
θ_i	Joint variables
m	Degree of task space
r	Manipulation vector
$\dot{r}$	Cartesian space velocity
$\dot{\theta}$	Joint space velocity
$\ddot{r}$	Cartesian space acceleration
$\ddot{\theta}$	Joint space acceleration
R^m	Range of task space
R^n	Range of Cartesian space
$R(J)$	Range space of jacobian
$R(J)^\perp$	Orthogonal complement of range space
$N(J)$	Null space of jacobian
$N(J)^\perp$	Orthogonal complement of null space
$J^\#(\theta)$	Pseudoinverse
I	Identity matrix
y, z	Arbitrary vectors

$\boldsymbol{\tau}$	Joint torque vector
$\boldsymbol{M}(\boldsymbol{\theta})$	Inertia matix
$\boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$	Torque vector
$\boldsymbol{r_1}$	Position of end effector
$\boldsymbol{r_2}$	Orientation of end effector
$\boldsymbol{e_i}$	Error vector
$\emptyset$	Orientation of the end effector with x-axis

CHAPTER 1 INTRODUCTION

1.1 REDUNDANT MANIPULATORS

Most of the conventional manipulators are usually designed to have not more than 6 degrees-of-freedom (DOF), which is the minimum one to perform any task in Cartesian space. A robot manipulator may require additional DOF, when working in a Cartesian space to follow a given trajectory so as to bypass hitting of the obstacles coming in its way.

When a robot has more DOF than that required to complete any given task, it is called as a redundant manipulator. A manipulator with seven DOF is said to be redundant for a task because six DOF are sufficient for complete position and orientation control of any end-effector. Redundant manipulators can also have six or less DOF because the particular task to be performed may require fewer DOF than the manipulator actually has.

A kinematically redundant manipulator has more number of joints (DOF) than required to completely specify the position and orientation of the end-effector. The linear equations relating unknown joint velocities to given end-effector velocity components in world coordinates have infinite solutions and the solutions are selected in such a way so as to get the best performance of the manipulator while tracking a desired trajectory. Performance of the manipulator depends upon the various criteria like torque requirements at joint motors, extent of obstacle avoidance, singularity avoidance etc [5].

Redundant manipulators are much more able than non redundant ones in many aspects, such as they can avoid collisions with obstacles in the working space, avoid singular states where the manipulators loses some degree of freedom, keep away from the motion limits of joints, reaching into holes etc. so that the robot manipulator becomes more flexible and adaptive like human arms [1].

1.2 TASK PRIORITY

There are certain tasks in which the position of the end effector is more important than the orientation, or vice versa. For example, both position and orientation must be controlled in arc welding and shape measurement, but the position of the end effector is more important than the orientation. On the other hand, with spray painting, or directing a camera to objects, orientation is more important. These tasks are considered to be composed of subtasks with different levels of significance and are known as tasks with the order of priority. The usual problems of redundancy utilization can be formulated in the framework of tasks with the order of priority [1].

When a redundant manipulator is required to trace a given trajectory of the end effector while avoiding obstacles in the workspace, trajectory tracing is given the first priority and obstacle avoidance is given the second priority. When a redundant manipulator is needed to trace a given trajectory of the end effector while avoiding singular points, trajectory tracing is given the first priority and singularity avoidance is given the second priority. For the tasks with the order of priority, if it is impossible to perform all of the subtasks completely because of the degeneracy or the shortage of degrees of freedom, it seems reasonable, then, to perform the most significant subtask preferentially and the less important subtasks (as well as possible) using the remaining degrees of freedom.

1.3 IMPORTANCE OF REDUNDANCY RESOLUTION

The need for using the redundant manipulators arises due to the following reasons:

1.3.1 Position Prior to Orientation (PPO)

Most of the complicated tasks given to the robots with many DOF can be decomposed into several subtasks with the order of priority. The task is divided into subtasks with different levels of significance according to their priority and known as task with the order of priority. Position prior to orientation means that the task is divided into two subtasks whose first priority is to follow the given trajectory that is the position and the second priority is given to the orientation of the end effector. For example, in case of arc welding process both position and orientation must be controlled, but the position of the end effector is more important than the orientation. Whereas in spray painting process the orientation is more importance. These tasks are considered to be composed of subtasks with different levels of significance and are known as tasks with the order of priority. The redundant manipulator is required to follow a trajectory of end effector while keeping the orientation of the end-effector fixed at certain angle. Each subtask is performed using the extra DOF that remain after all the subtasks with higher priority have been fulfilled.

1.3.2 Obstacle Avoidance

Obstacles are defined as any portion of an object with which contact is undesirable. Clearly, obstacle avoidance is essential for satisfactory task completion. Redundant manipulator involves the potential of obstacle avoidance from the manipulator's workspace and the use of fixed motion commands by choosing the best suitable path out of the infinite paths available. The satisfaction of the obstacle avoidance criteria would greatly increase the autonomy of the manipulators and results in a wider range of applications.

1.3.3 Singularity Avoidance

Singularity regions correspond to robot configurations that are close to a singular configuration, and in which the joint velocities required to achieve the end-effector motion in certain directions are extremely high. Therefore, it becomes important for a general purpose robotic manipulator to have more than six degrees of freedom. When the robot is in a singular configuration, there is at least one direction in which the end effector is unable to move. The joint velocities required to attain the end-effector velocity component in this direction are infinitely high. Mathematically, this occurs when the rank of the jacobian matrix is less than the number of its rows. In a configuration close to a singularity configuration, the joint velocities required to move in certain directions are extremely high. Robot's mobility, therefore, is very poor in these directions. Thus arbitrary directional changes of the end-effector become more difficult.

There are two types of singularities:-

- Kinematic singularities are those singular configurations at which the end effector loses all its mobility.
- Algorithmic singularities are those at which the end effector task and the constraint task conflict despite the redundant degrees of freedom.

1.3.4 Torque Minimization

Due to infinite solutions of inverse kinematics problem, the solutions needs to be chosen so that the torque required to move the motor is minimum. The next solution is chosen in reference to the previous one and thus aims at minimizing the torque of the motor.

1.3.5 Minimum Movement

The solutions are taken in such a way that there is minimum movement of the manipulator to reach the desired position. It also aims in choosing those solutions which require minor axis movements than major axis because the mobility of a robot depends on the direction of the end effector motion desired which is represented by the manipulator velocity ratio ellipsoid (MVRE). The minor axis of the MVRE represents the minimum value of MVR and the upper limit on the joint velocities required for the end effector to move in all directions. Thus a small minor axis implies low flexibility and proximity to singular configurations.

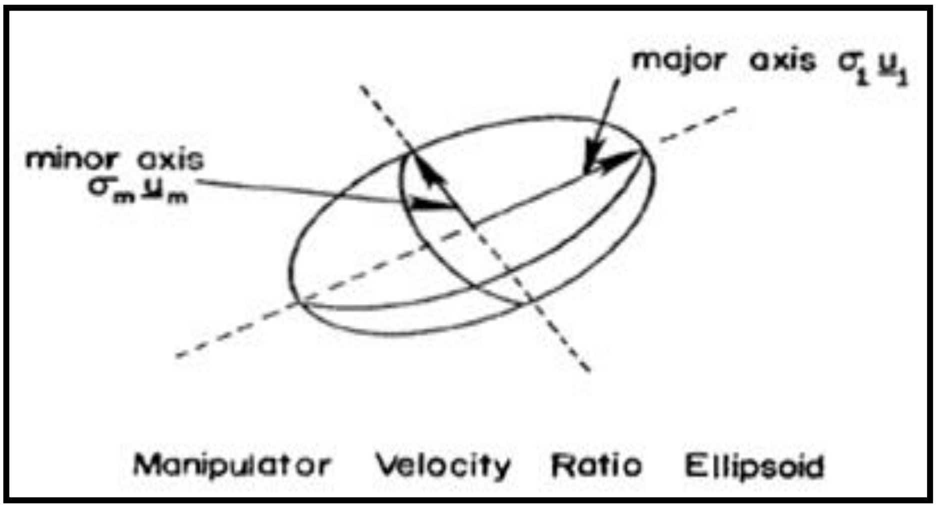

Fig. 1.1: Manipulator Velocity Ratio Ellipsoid [12].

1.3.6 Flexibility and Versatility

The redundancy of robot manipulators plays an important role in increasing the flexibility and versatility. Flexibility is important so that the robot can move in any direction freely. It is the robot's ability to change the direction of end-effector motion. Thus a robot is less flexible when it is near or in a singular configuration.

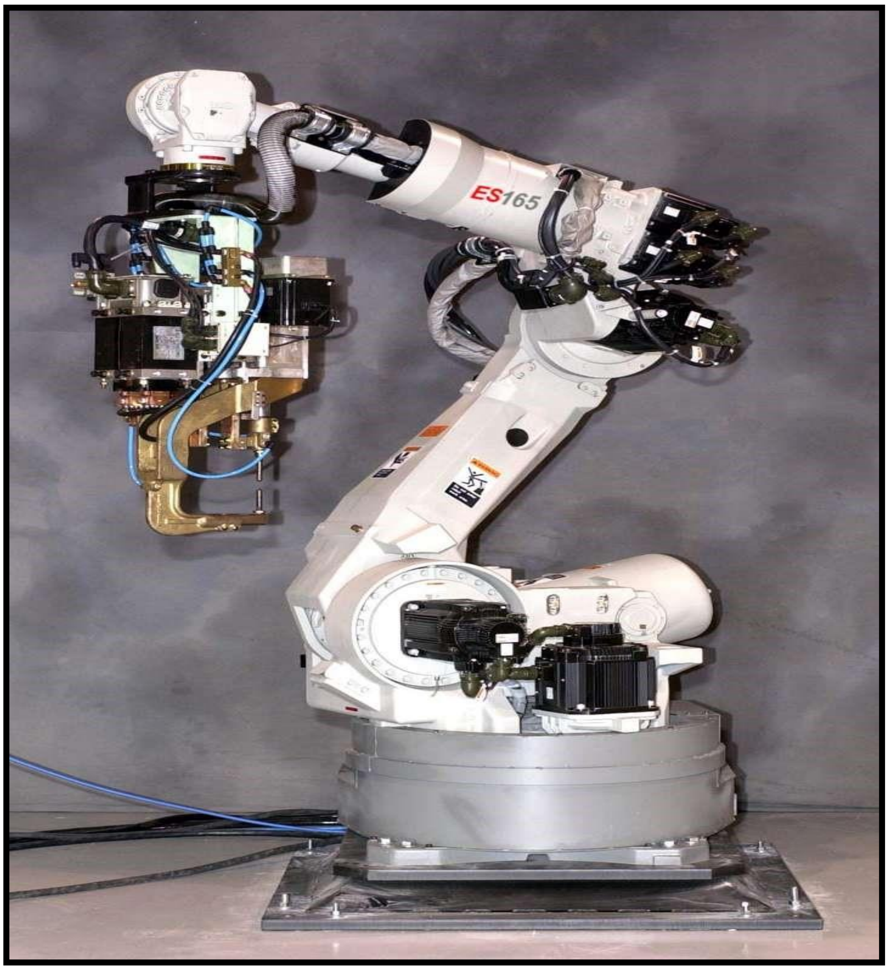

Fig. 1.2: A welding robot system with 7 DOF [23].

1.4 APPLICATIONS OF REDUNDANT MANIPULATORS

- Used for experimental purpose to avoid obstacles.
- In garments industries for the automation of sewing tasks.
- Redundant robots perform complicated welding operations, as shown in Fig. 1.2.

- Redundant manipulators are used for inspection inside nuclear plants, as shown in Fig. 1.3 and their redundancy is used to avoid collision with the environment.
- Used in sewerage blockage, spraying operations.
- Snake link manipulators shown in Fig. 1.4 are also redundant manipulators used in performing tasks in complicated areas.

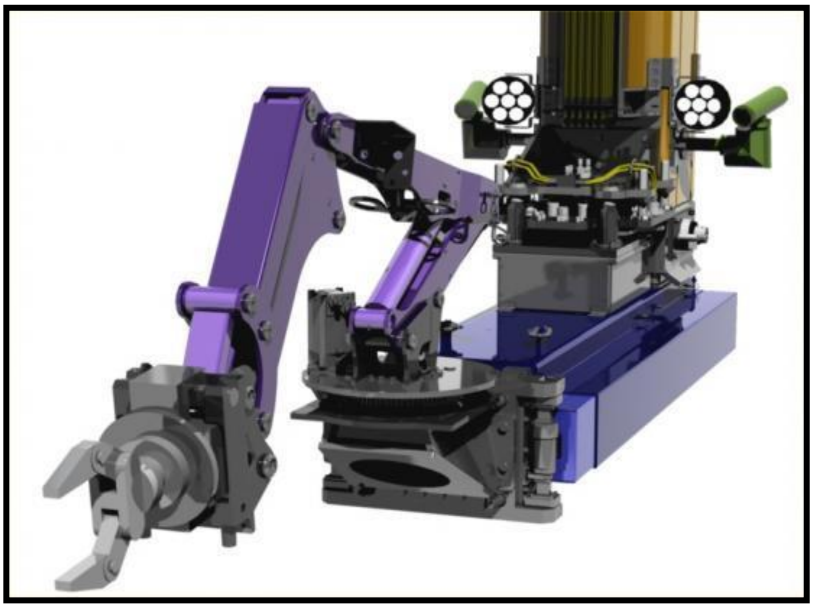

Fig. 1.3: Redundant Manipulators used in Nuclear Plants [24].

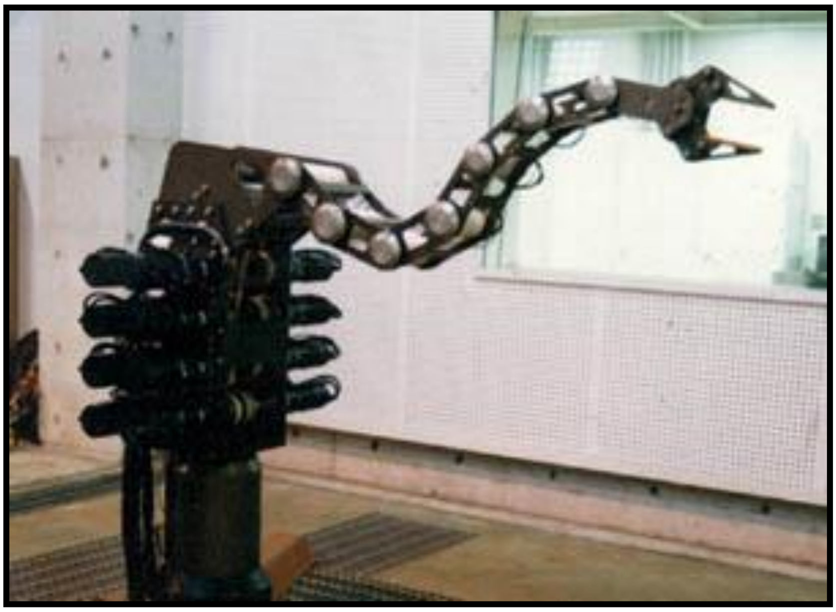

Fig. 1.4: A Snake-link Manipulator [25].

1.5 FEW EXAMPLES OF REDUNDANT MANIPULATORS

1. MELARM with two 7-DOF was developed in 1974 at a mechanical engineering laboratory, the ministry of international trade and industry (MITI), Japan.
2. UJIBOT a 7-DOF manipulator has a nonanthropomorphic structure. It was developed in 1979 at Kyoto University. This robot was used for experiments in obstacle avoidance.
3. Several multifingerded hand systems have been developed for research on multifinger coordination. Salisbury hand, a 3 DOF finger is one such kind, as shown in Fig. 1.5.

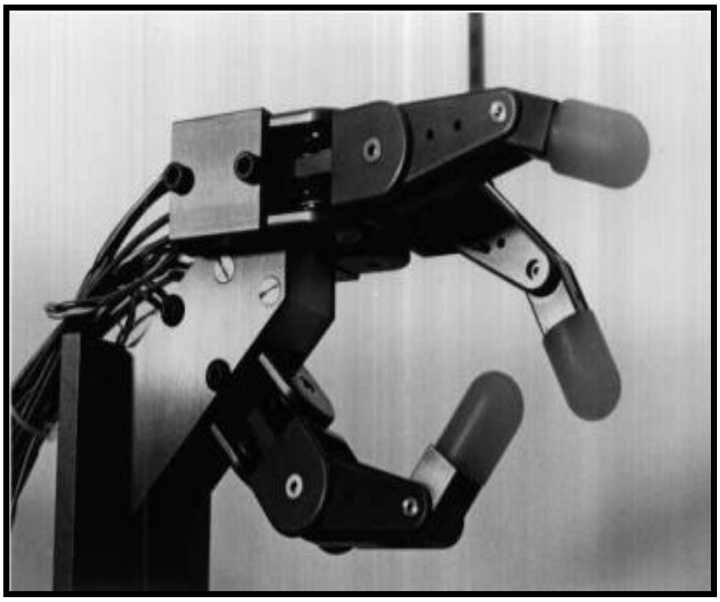

Fig. 1.5: Salisbury hand, 3-DOF fingers [1].

1.6 ORGANISATION OF THE THESIS

This thesis is organised as follows. In Chapter 2, the literature work that has been done so far in this area is discussed. Starting from the time when the idea came for active utilization of redundancy in which the author used the generalized inverse of the Jacobian matrix to get the general solution of joint velocity to the time when different algorithms are developed to utilize the redundancy and use it for avoiding obstacles, singularity avoidance, torque minimisation etc. Finally, a systematic framework to the problem of redundancy utilization using task priority is given.

In Chapter 3, the methodology related to redundancy resolution and task priority is given. The problem is formulated in the framework of resolved motion rate control. The basic concept of task priority including the use of redundant space and manipulable space is presented along with the generation of various equations associated with it.

Numerical simulations are carried out in Chapter 4 to show the effectiveness of the formulation. Results are obtained by using a MATLAB code and graphs are plotted for different trajectory motions with and without using task priority.

Finally results are summarised and possible future directions are mentioned in Chapter 5.

CHAPTER 2 LITRATURE REVIEW

2.1 INTRODUCTION

Most of the conventional robots came into existence in the early 60's. Thereafter, the robots got enormous popularity and find their place in lots of industrial applications. The trend from conventional manipulators to redundant manipulators changed in late 70's, when a lot of researchers worked in the domain of redundancy resolution. The most significant works in the domain of kinematic redundancy is summarized below.

2.2 LITRATURE SURVEY

In 1977, LIEGEOIS [3] discussed the active utilization of redundancy, in which he represented the general solution of joint velocity by means of the *generalized inverse* of the Jacobian matrix. He proposed that the gradient vector of a scalar function can be used as the arbitrary vector. He also showed the numerical simulation of utilizing redundancy for keeping the joint angles within their mechanical limitations.

In 1984, YOSHIKAWA T. [6] discussed the analysis and control of robot manipulators with redundancy. A *quantitative measure of manipulability,* is projected which is applicable to both redundant and non redundant manipulators. Based on this measure, a control algorithm for utilizing the redundancy for singularity avoidance is established. A control algorithm for avoiding obstacles is also discussed.

In 1985, YOSHIKAWA T. [7] discussed the manipulating ability of robotic mechanisms in positioning and orienting end-effectors and proposed a *measure of manipulability*. Some properties of this measure are derived, the best postures of various types of manipulators are given, and a 4 DOF finger is considered from the viewpoint of the measure. The postures somewhat resemble those of human arms and fingers.

Consider the two-joint link mechanism shown in figure, which is the simplest case of multi joint manipulators. When the hand position $[x, y]^T$ is taken as the manipulation vector $\boldsymbol{r}$, the jacobian matrix is given by

$$\boldsymbol{J(\theta)} = \begin{bmatrix} l_1 \cos\theta_1 + l_2 \cos(\theta_1 + \theta_2) & l_2 \cos(\theta_1 + \theta_2) \\ -l_1 \sin\theta_1 - l_2 \sin(\theta_1 + \theta_2) & -l_2 \sin(\theta_1 + \theta_2) \end{bmatrix} \quad (2.1)$$

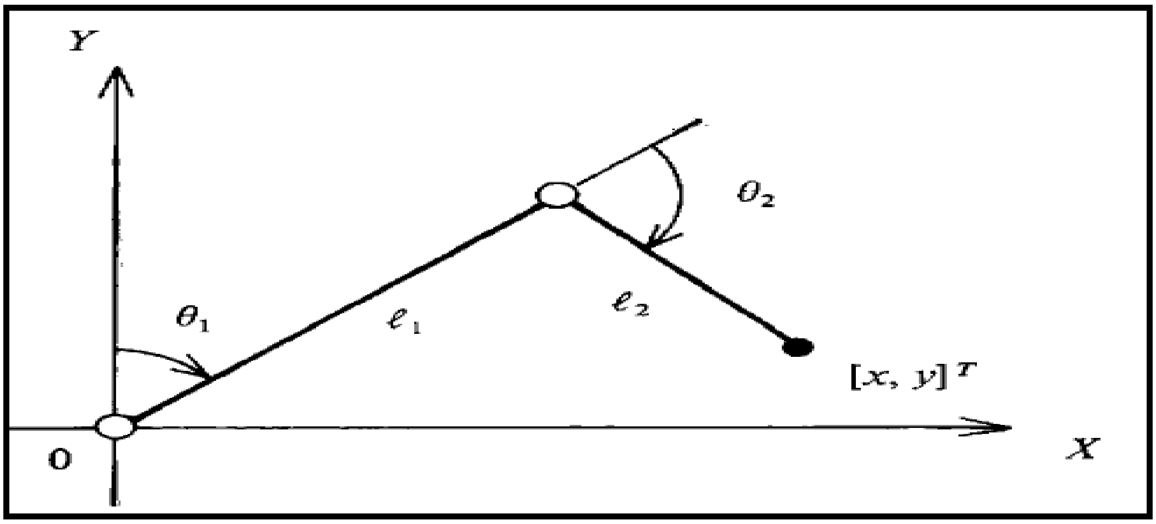

Fig. 2.1: Two joint link mechanism [7].

$$W = |\det \boldsymbol{J}(\theta)| = l_1 l_2 |\sin\theta_2| \quad (2.2)$$

Therefore, the manipulator takes its best orientation when $\theta_2 = \pm 90°$, for any given values of l_1, l_2 and θ_1.When the human arm is considered as a two-joint link mechanism by neglecting the degree of freedom of sideward direction at the shoulder and the degree of freedom of the wrist, it approximately satisfies the relation $l_1 = l_2$. Moreover, when we handle some object with our hands, the elbow angle is usually in the neighbourhood of 90°. Hence it could be said that people are unconsciously taking the best arm postures from the viewpoint of manipulability.

In 1985, YOSHIKAWA T. [8] proposed the concept of a *dynamic manipulability measure* of robot arms as a quantitative measure of their manipulating capability in positioning and orienting the end-effectors, which takes the arm dynamics into account. This measure is given on the basis of the relation between the joint driving force and the acceleration of the end-effector. Some properties of the measure are established. A two-joint link mechanism is analyzed to obtain its best posture under certain condition from the viewpoint of this measure. To illustrate the utilization of this concept for the design of robot manipulators a numerical example is also given.

In 1985, KLEIN et al. [9] presented an approach to determine the required joint angle rates for the manipulator under the constraints of multiple goals, the primary goal described by the specified end-effector trajectory and secondary goals describing the obstacle avoidance criteria. The decomposition of the solution into a particular and a homogeneous component effectively illustrates the priority of the multiple goals that is exact end-effector control with redundant degrees of freedom maximizing the distance to obstacles. An efficient numerical implementation of the technique permits sufficiently fast cycle times to deal with dynamic environments.

In 1987, NAKAMURA et al. [10] introduced the concept of *task priority* in relation to the inverse kinematic problem of redundant robot manipulators. The required task is divided into subtasks according to the order of priority. The joint motions of robot manipulators are calculated so that subtasks with lower priority can be performed utilizing the redundancy on subtasks with higher priority. This method is carried out using the pseudoinverse of the Jacobian matrices. Problems of redundancy utilization can mostly be built in the framework of tasks with the order of priority. The results obtained from the numerical simulations and experiments shows how effective this method is in controlling the redundancy.

In 1987, HOLLERBACH [11] used the method of *torque optimization* for redundancy resolution. The kinematic redundancies are resolved by the impact of joint torque. The effect of redundancy resolution on joint torque can be straightaway reflected when the generalized inverse is formulated in terms of accelerations and incorporated into the dynamics. Contrasting methods employing only the pseudoinverse with and without weighting by the inertia matrix are presented. The results show an unimagined stability problem during long trajectories for the null-space methods and for the inertia-weighted pseudoinverse method, but more infrequent for the unweighted pseudoinverse method.

In 1987, DUBEY R. [12] defined the manipulator velocity ratio to be a *quantitative measure of mobility* for robots with redundancy. It was shown that the mobility of a robot depends on the direction of the end-effector motion in demand, and it was represented by manipulator velocity ratio ellipsoid. This potential of the robot to change the direction of the end-effector motion was called flexibility. It was concluded that a small minor axis implies low flexibility and proximity to a singular configuration. So they presented a control scheme to increase the minor axis of MVRE or the minimum MVR while the robot is in motion. This scheme makes robots able to move in different directions more uniformly. Thus robot's flexibility increases and singularity regions are automatically avoided. The feasibility and effectiveness of the control scheme were demonstrated through simulation with graphics.

In 1987, BAILLIEUL [13] presented a coordinate-free version of the *manipulability index*. This permits the index to be evaluated in a meaningful way for a kinematically redundant mechanism operating under imposed functional constraints on the joint variables. Such an intrinsic quantity is useful for deciding which of several possible constraints yields the best performance of a given mechanism, and conversely, it may be employed as a constraint oriented figure-of-merit in the design of kinematically redundant manipulators. It also provides a new computational tool for determining the singularities inherent in most approaches to redundancy resolution by means of functional constraints.

In 1997, CHIAVERINI S. [15], developed a new task priority redundancy resolution technique by firstly studying the application of existing singularity robust methods applied on the redundant robotic arm to overcome the effects of algorithmic singularities. To demonstrate the effectiveness of the algorithm it is applied to a seven degree of freedom manipulator.

In 2000, LEE B.H. et al. [16], developed a new control method for kinematically redundant manipulators having the properties of torque-optimality and singularity-robustness. A dynamic control equation of joint torques, which should be satisfied to get the desired dynamic performance of the end-effector is formulated using the feedback linearization theory. The optimal control law is determined by locally optimizing an appropriate norm of joint torques using the weighted generalized inverses of the manipulator Jacobian-inertia product. This paper presents a new method for the robust and rough handling of robot kinematic singularities in the context of joint torque optimization. Control of the end-effector motions in the neighbourhood of a singular configuration is based on the use of the damped least-squares inverse of the Jacobian-inertia product.

In 2002, ZLAJPAH et al. [17] presented an approach, which deals with kinematic control algorithm for online obstacle avoidance which allows the kinematic redundant manipulator to move in an unstructured environment without colliding with obstacles. This approach is based on redundancy resolution at the velocity level. A velocity component in the direction away from obstacle is assigned at each point on the body of the manipulator, which is close to the obstacle. This avoiding velocity is defined in one dimensional space so as to avoid singularity problems when not enough redundancy is available locally. Hence the calculation of pseudoinverse also becomes simpler as it includes scalar division instead of matrix inversion. This algorithm has been implemented on four link manipulator and the fast cycle times of numerical implementation makes the algorithm suitable for real time control.

In 2003, QIAO et al. [18] presented a pseudo-distance function for a pair of convexpolyhedra, along with the algorithm for calculating its derivative. On this basis, a potential field-based approach for obstacle avoidance of kinematically redundant manipulators is developed, with the manipulator links and the environmental obstacles being geometrically modelled as a set of convexpolyhedra. The potential function is differentiable almost everywhere with respect to the joint configuration variables of the manipulator. It is incorporated in the 'null space projection scheme' in order to achieve obstacle avoidance.

Simulation examples are presented to show the effectiveness of the proposed method.

In 2006, KIM J. et al. [19] proposed a method of avoiding kinematic and algorithmic singularities by applying a task reconstruction approach while maximizing the task performance by calculating singularity measures. The proposed method is implemented by removing the component approaching the singularity calculated by using singularity measure in real time. The outstanding feature of the proposed task reconstruction method (TR-method) is that it is based on a local task reconstruction as opposed to the local joint reconstruction of many other approaches. The TR-method enables us to increase the task controller gain to improve the task performance whereas this increase can destabilize the system for the conventional algorithms in real experiments. In addition, the physical meaning of tuning parameters is very straightforward. Hence, task performance can be minimized even near the singular region while simultaneously obtaining the singularity-free motion. The advantage of the proposed method is experimentally tested by using the 7-dof spatial manipulator, and the result shows that the new method improves the performance several times over the existing algorithms.

In 2007, SINGLA A. et al. [20] discussed the concept of task priority and implemented on the inverse kinematic problem of redundant manipulators. A required task is divided into a number of subtasks depending on the order of priority. The redundancy is utilized to do various subtasks according to the order of priority like achieving the required orientation, obstacle avoidance etc. The entire procedure is formulated using the pseudo inverse of the Jacobian matrix. Numerical simulations were performed to verify the validity of the formulation and the solutions obtained for the redundancy control problems. The simulation results show that more degrees of redundancy implies fulfilment of more subtasks according to the given priority. But there is no single criterion which can work efficiently for any workspace and task. Therefore, according to the workspace and the order of priority, appropriate criteria must be selected to fulfil the lower priority sub-tasks.

In 2009, OETOMO D. et al. [21] worked on solving the problems related to singularity associated with robot manipulators. The main focus is on the discontinuity problem associated with singularity along with the removal of degenerate components. A complete treatment is presented so as to remove the singularities starting from the identification of singular configurations and then removing the degenerate components by different ways.

In 2012, YAHYA S. et al. [22] focuses on the advancement of singularity avoidance of three dimensional planar redundant manipulators by increasing its degrees of freedom without increasing the number of motors controlling the manipulator. The manipulability ellipsoids for the proposed manipulator have been obtained and compared with the ellipsoids of the PUMA arm. The manipulability measure values of both the proposed manipulator and PUMA arm have been calculated and analyzed, and the results of the illustrated examples show the ability of the proposed manipulator to be used specially for singularity avoidance.

2.3 OBSERVATIONS FROM LITRATURE SURVEY

1. The general solution of the joint velocity can be expressed by means of the generalized inverse of the jacobian matrix.
2. The measure of manipulability can be used to determine the best postures of the various types of manipulator.
3. Redundancy can be utilised for keeping the joint angles within their physical limitations.
4. Redundant manipulators are capable of avoiding obstacles, avoiding singularity etc. by making use of the pseudoinverse of the Jacobian matrix.
5. By dividing the task into subtasks with the order of priority, the degeneracy of degree of freedom can be overcome easily.

CHAPTER 3 METHODOLOGY

3.1 INTRODUCTION

In this chapter, the methodology and the basics behind the concept of task priority and redundancy resolution for position prior to orientation are discussed. The significance of Jacobian matrix and pseudoinverse in mapping the joint space to Cartesian is described. The problem is formulated using theory of task with the order of priority. Also, the idea behind trajectory planning is given which helps in planning different trajectories for the manipulator to follow in Cartesian space.

3.2 THE IMPORTANCE OF JACOBIAN MATRIX

The differentiation of a vector function of a vector variable is known as the jacobian and it is a matrix quantity. Jacobian maps joint space velocity to Cartesian space velocity. For any forward kinematics problem, consider a manipulator with n degrees of freedom, whose joint variables are denoted by $\boldsymbol{\theta_i}$, where i = 1, 2, ………, n. Assume that a class of tasks can be described by m variables $\boldsymbol{r_j}$, j =1, 2, ………, m $(m \leq n)$, and that the relation between $\boldsymbol{\theta_i}$ and $\boldsymbol{r_j}$ is given by

$$\boldsymbol{r} = \boldsymbol{f}\,(\boldsymbol{\theta}) \tag{3.1}$$

where, $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots\dots\dots, \theta_n]^T$ is the joint vectors, $\boldsymbol{r} = [r_1, r_2, \dots\dots.., r_m]^T$ is the manipulation vector, and the superscript T denotes the transpose. Differentiating Eqn. (3.1) gives,

$$\frac{\Delta r}{\Delta t} = \frac{\partial f}{\partial \theta} \cdot \frac{\Delta \theta}{\Delta t}$$

If, $\Delta \rightarrow 0$,

$$\frac{dr}{dt} = \frac{\partial f}{\partial \theta} \cdot \frac{d\theta}{dt}$$

So,
$$\dot{\boldsymbol{r}} = \boldsymbol{J}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} \tag{3.2}$$

Where,

$\dot{\boldsymbol{r}} \in \mathcal{R}^m$, $\dot{\boldsymbol{\theta}} \in \mathcal{R}^n$ and $\boldsymbol{J}(\boldsymbol{\theta}) = \frac{\partial f}{\partial \theta} \in \mathcal{R}^{m \times n}$. The matrix $\boldsymbol{J}(\boldsymbol{\theta})$ is the Jacobian [1].

Assume that the following condition is satisfied:

$$\text{Max rank } \boldsymbol{J(\theta)} = m \tag{3.3}$$

Failing to satisfy this condition usually means that the selection of manipulation variables is redundant and that the number of these variables m can be reduced. When condition (3.3) is satisfied, it is said that the degree of redundancy of this manipulator is $(n - m)$.

If, for some $\boldsymbol{\theta}$, Rank $\boldsymbol{J(\theta)} < m$, then it is said that the manipulator is in a singular state. This state is not desirable because the manipulation vector $\boldsymbol{r}$ cannot move in a certain direction, meaning that the manipulability is seriously deteriorated.
For redundant manipulators, as $n > m$, it has infinite solutions [6].

3.3 MANIPULABILITY AND MANIPULABLE SPACE

In Cartesian space, $\boldsymbol{r} \in \mathcal{R}^m$ is used to represent the kinematic output of a robot manipulator. Here $\boldsymbol{r}$ is called the manipulation variable and $\boldsymbol{J(\theta)} \in \mathcal{R}^{m \times n}$ is the corresponding jacobian matrix. As a generalized expression of Eqn. (3.2) the following equation is used:

$$\boldsymbol{\delta r} = \boldsymbol{J}(\boldsymbol{\theta})\boldsymbol{\delta\theta} \tag{3.4}$$

This equation shows that δr is a linear mapping of $\delta\theta$ by the jacobian matrix. The Fig. 3.1 explains this linear mapping. Here, $R(J)$ denotes the range space of $J(\theta)$, which is a subspace in the m-dimensional space of $\boldsymbol{\delta r}$. The dimension of $R(J)$ is the rank $\boldsymbol{J(\theta)} \leq \min(m, n)$. The range space is a space which the manipulator can reach by atleast one orientation.

The orthogonal complement of the range space $R(J)^{\perp}$ indicates the subspace made up of all of the kinematically unrealizable motions $\boldsymbol{\delta r}$. The range space of the jacobian matrix and its dimension are called **as manipulable space** and **degree of manipulability**, respectively.

The motion $\boldsymbol{\delta r}$ is kinematically realizable, iff it is a member of the manipulable space. The manipulable space changes as the configuration of the manipulator changes. Since, rank of jacobian matrix $\boldsymbol{J(\theta)}$ degenerates at singular points, so does the manipulable space. At singular points, a manipulator cannot move in degenerated directions [1].

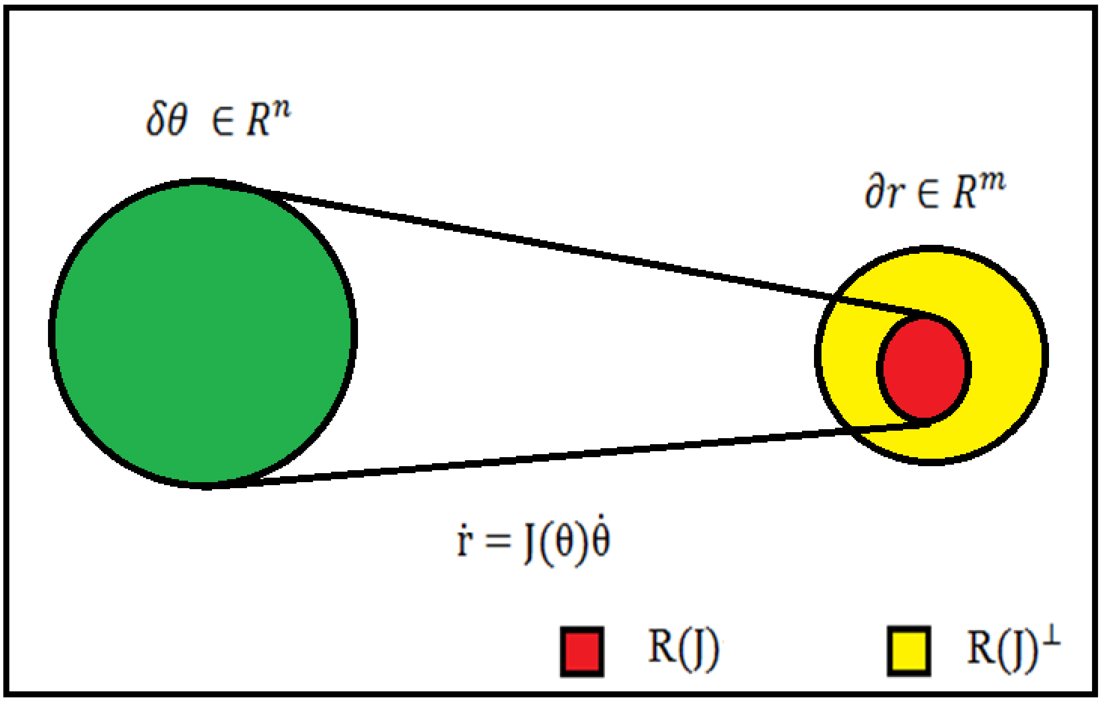

Fig. 3.1: Linear mapping of the manipulable space [1].

3.4 REDUNDANCY AND REDUNDANT SPACE

If the number of joints is greater than the dimension of the manipulation variable that is n > m, the manipulator is said to be redundant. Redundant manipulators have the characteristic that there exist infinite solutions of inverse kinematics problem.

Let $\partial\theta^1$ and $\partial\theta^2$ be two distinctive solutions of Eqn. (3.4). Then,

$$\boldsymbol{J(\theta)\ \partial\theta_e = J(\theta)\ [\partial\theta^1 - \partial\theta^2] = 0,} \tag{3.5}$$

where

$$\boldsymbol{\partial\theta_e = \partial\theta^1 - \partial\theta^2 \neq 0.} \tag{3.6}$$

These equations implies that the difference of two solutions is always mapped to the zero vector in $\boldsymbol{\delta r}$ space. The variety of solutions is represented by that of the vectors to be mapped to the zero vectors by the Jacobian matrix. It is known that such vectors span a linear subspace in n-dimensional space of $\boldsymbol{\delta\theta}$, which is called as *the null space* $\mathcal{N}(J)$.

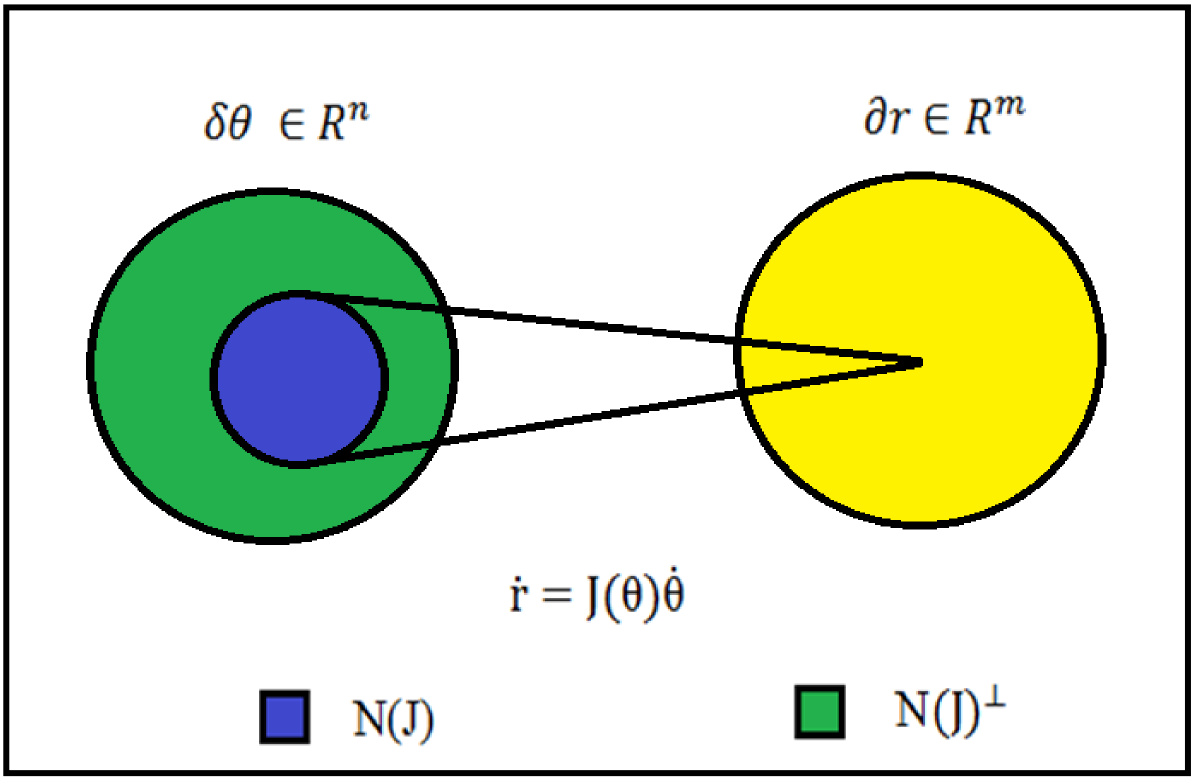

Fig. 3.2: Redundant space [1].

The null space of the Jacobian matrix, $\mathcal{N}(J)$, and its mapping is shown in Fig. 3.2. The null space of the Jacobian matrix and its dimension are called **as redundant space** and the **degree of redundancy** [1], respectively.

Note that, if $n \leq m$ and rank $\boldsymbol{J(\theta)} = n$, there is no non zero entry in the null space of $\boldsymbol{J(\theta)}$, since the column vectors of $\boldsymbol{J(\theta)}$ are all independent. Accordingly, degree of redundancy (DOR) becomes zero and the solution of Eqn. (3.4) becomes unique. A more general relationship between DOM and DOR is provided by using a well-known result from matrix theory that the range space and the null space of a matrix M $\in \mathcal{R}^{m\times n}$ should satisfy

$$\dim R(M) + \dim N(M) = n, \tag{3.7}$$

and

$$DOM + DOR = n. \tag{3.8}$$

This implies that degree of redundancy increases at singular points by the same number by which degree of manipulability decreases.

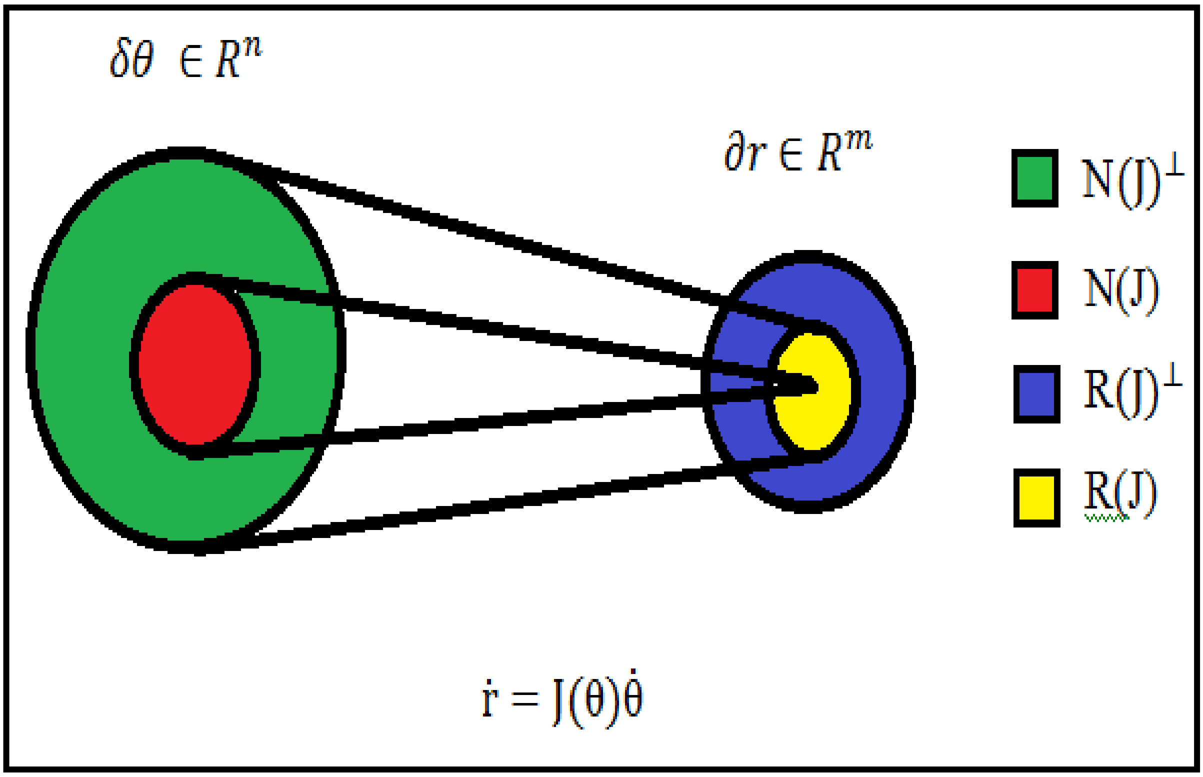

Fig. 3.3: Linear mapping of the Jacobian matrix [1].

3.5 PSEUDOINVERSE

The inverse of a non-square matrix is known as pseudoinverse. A pseudoinverse of a matrix $\boldsymbol{J(\theta)} \in \mathcal{R}^{m\times n}$ is a matrix $\boldsymbol{J}^{\#}(\boldsymbol{\theta}) \in \mathcal{R}^{n\times m}$.

Now,

$$\dot{\boldsymbol{r}} = \boldsymbol{J}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}. \tag{3.9}$$

Pre multiply Eqn. (3.9) by $\boldsymbol{J}^{\#}$ gives

$$\boldsymbol{J}^{\#}\,\dot{\boldsymbol{r}} = \boldsymbol{J}^{\#}\boldsymbol{J(\theta)}\,\dot{\boldsymbol{\theta}}. \tag{3.10}$$

It is known that $\boldsymbol{J}^{\#}\,\boldsymbol{J(\theta)}$ maps $\mathcal{N}(\boldsymbol{J}(\boldsymbol{\theta}))^{\perp}$. On the similar terms, $\boldsymbol{(I - J^{\#}J(\theta))y}$ maps $\mathcal{N}(\boldsymbol{J(\theta)})$

The linear mapping of pseudoinverse is shown in Fig. 3.4 where, the range space and null space of the pseudoinverse matrix along with their orthogonal compliments are shown.

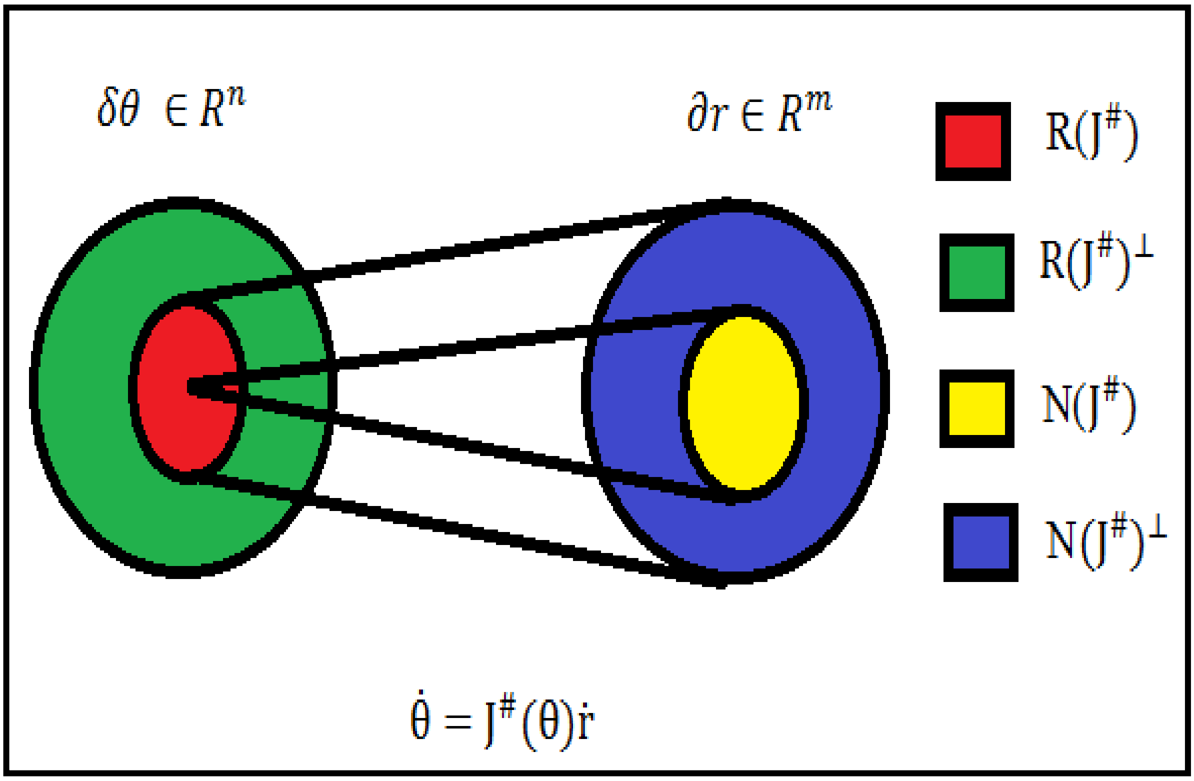

Fig. 3.4: Linear mapping of pseudoinverse $\mathbf{J}^{\#}(\boldsymbol{\theta})$ [1].

3.6 TASK WITH THE ORDER OF PRIORITY

The problem of redundancy utilization can generally be formulated in the framework of tasks with the order of priority. When a redundant manipulator is required to follow the trajectory of the end effector, while avoiding obstacles or kinematics singular points, trajectory following is given first priority, and obstacle or singularity avoidance is given second priority.

For tasks with the order of priority, if it is impossible to perform all the subtasks completely because of the degeneracy or the shortage of DOF, it would be reasonable to perform the most significant subtasks preferentially and the less important subtasks as much as possible using the remaining DOF. Even for a six DOF manipulator, the subtask decomposition between position and orientation is advantageous, because it will enlarge the reachable workspace of the first priority subtask by allowing incompleteness for the second priority subtask. In the following section, redundancy utilization is discussed based on the concept of tasks with the order of priority.

3.7 INVERSE KINEMATICS CONSIDERING THE ORDER OF PRIORITY

Assume that a task is composed of two subtasks, which will be performed according to the order of priority. The first priority subtask is specified using the first manipulation variable, $\boldsymbol{r_1} \in \mathcal{R}^{m_1}$ and the second priority subtask by the second manipulation variable, $\boldsymbol{r_2} \in \mathcal{R}^{m_2}$. The kinematic relationships between the joint variable $\boldsymbol{\theta} \in \mathcal{R}^n$ and the manipulation variable are expressed by

$$\boldsymbol{r}_i = \boldsymbol{f}_i(\boldsymbol{\theta}), \qquad \text{(i=1, 2)} \tag{3.11}$$

Their differential relationships are given as follows:

$$\dot{\boldsymbol{r}}_i = \boldsymbol{j}_i(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}, \qquad \text{(i=1, 2)} \tag{3.12}$$

For redundant manipulators, as $n > m$, the general solution to Eqn. (3.12) for $i = 1$ is obtained using the pseudoinverse of the jacobian matrix as follows :

$$\dot{\boldsymbol{\theta}} = \boldsymbol{J}_1^{\#}(\boldsymbol{\theta})\ \dot{\boldsymbol{r}}_1 + \{\boldsymbol{I} - \boldsymbol{J}_1^{\#}(\boldsymbol{\theta})\,\boldsymbol{J}_1(\boldsymbol{\theta})\}\boldsymbol{y} \tag{3.13}$$

Where $\boldsymbol{J}_1^{\#}(\boldsymbol{\theta}) \in \mathcal{R}^{n\times m_1}$ is the pseudoinverse of $\boldsymbol{J}_1(\boldsymbol{\theta})$, $\boldsymbol{y} \in \mathcal{R}^n$ is an arbitrary vector, and $\boldsymbol{I} \in \mathcal{R}^{n\times n}$ indicates an identity matrix. If the exact solution does not exist Eqn. (3.13) covers all the least squares solution [20] that minimize $||\dot{\mathbf{r}}_1 - \mathbf{J}_1(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}||$.

On substituting Eqn. (3.13) in Eqn. (3.12) for $i = 2$, to obtain,

$$J_2\{I - \hat{J}_1^{\#} J_1\}y = \dot{r}_2 - J_2\hat{J}_1^{\#}\dot{r}_1 \tag{3.14}$$

Obtain y that minimizes $||\dot{\mathbf{r}}_2 - \mathbf{J}_2(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}||$, same as Eqn. (3.13) as given by

$$y = \hat{J}_2^{\#}(\dot{r}_2 - J_2 J_1^{\#}\dot{r}_1) + \{I - J_2^{\#}\hat{J}_2\}z \tag{3.15}$$

where, $\hat{J}_2 = J_2\{I - J_1^{\#}J_1\}$ and $\boldsymbol{z} \in \mathcal{R}^n$ is an arbitrary vector.

The solution is obtained from Eqns. (3.13) and (3.15) as follows:

$$\dot{\boldsymbol{\theta}} = J_1^{\#}\dot{r}_1 + \{I - J_1^{\#}J_1\}\hat{J}_2^{\#}(\dot{r}_2 - J_2 J_1^{\#}\dot{r}_1) + \{I - J_1^{\#}J_1\}(I - \hat{J}_2^{\#}\hat{J}_2)z \tag{3.16}$$

The second term of the Eqn. (3.16) is reduced to $J_2^{\#}(\dot{r}_2 - J_2 J_1^{\#}\dot{r}_1)$.

Thus, $$\dot{\boldsymbol{\theta}} = J_1^{\#}\dot{r}_1 + \hat{J}_2^{\#}(\dot{r}_2 - J_2 J_1^{\#}\dot{r}_1) + \{I - J_1^{\#}J_1\}(I - \hat{J}_2^{\#}\hat{J}_2)z \tag{3.17}$$

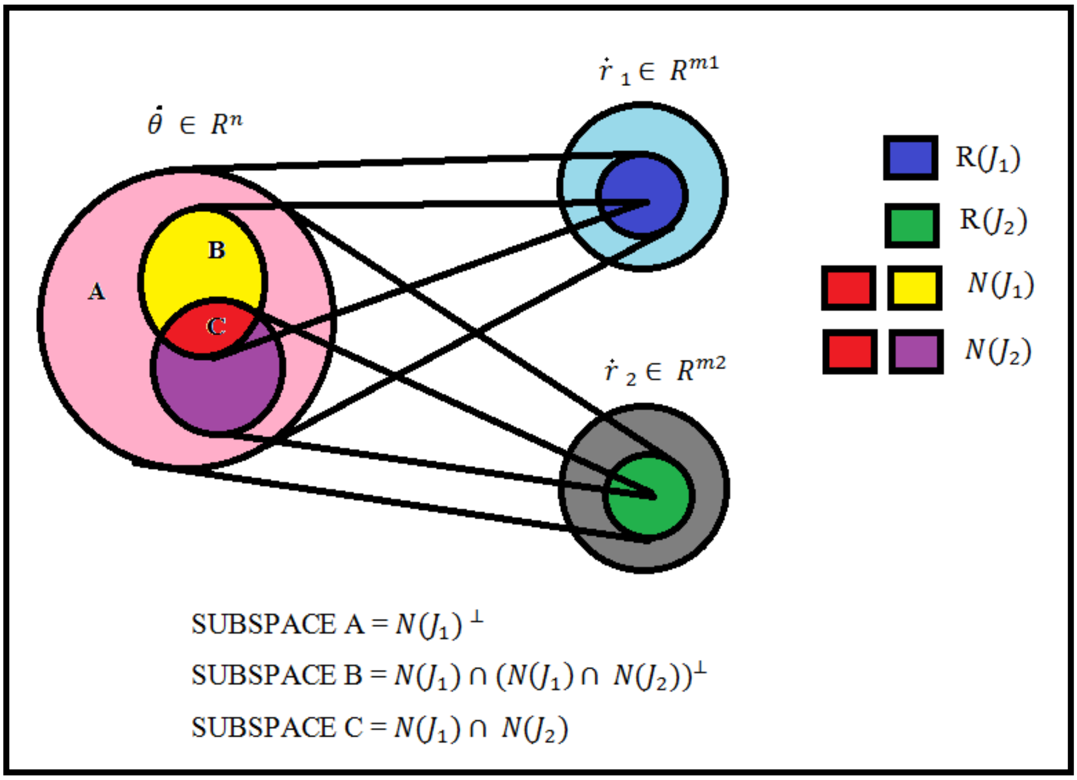

Fig. 3.5: Manipulable and Redundant spaces for the first and second manipulation variables [5].

The above equation represents the inverse kinematics solution taking account of the priority of the subtasks.

The manipulable space is the range space of the jacobian matrix which is denoted by R(J), and null space of the jacobian matrix *N(J)* is the the redundant space. Fig. 3.5 shows the general relationship between the manipulable and redundant spaces for the first and second manipulation variables. Subspace *A* shows all the possible contributions of $\dot{\boldsymbol{\theta}}$ to the first manipulation variable. Subspace *B* implies the contribution to the second manipulation variable without disturbing the first manipulation variable. Subspace *C* shows the remaining DOF that affect neither the first nor the second manipulation variables. Subspace *C* can be used for the third and higher manipulation variables, if necessary.

The first term in the right hand side of the Eqn. (3.17) is the least square mapping of $\dot{r}_1$ onto subspace *A*. The second term implies the least squares mapping of ($\dot{r}_2$ - $J_2 J_1^{\#} \dot{r}_1$) onto subspace *B*, where ($\dot{r}_2$ - $J_2 J_1^{\#} \dot{r}_1$) is the modified desired value of the second manipulation variable due to the effect of the first term on the second manipulation variable. The third term is the orthogonal projection of the arbitrary vector $\boldsymbol{z}$ onto subspace C. If there is a third manipulation variable, the arbitrary vector $\boldsymbol{z}$ is determined in the same manner as $\boldsymbol{y}$ [1].

In the case of $\boldsymbol{r}_2 = \boldsymbol{\theta}$, Eqn. (3.17) can be reduced to a simpler form using $\boldsymbol{J}_2 = \boldsymbol{I}$ as follows:

$$\dot{\boldsymbol{\theta}} = \boldsymbol{J}_1^{\#}\dot{\boldsymbol{r}}_1 + (\boldsymbol{I} - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1)\dot{\boldsymbol{r}}_2 \tag{3.18}$$

where $(\boldsymbol{I} - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1)^{\#} = (\boldsymbol{I} - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1)$ and the idempotency of $(\boldsymbol{I} - \boldsymbol{J}_1^{\#}\boldsymbol{J}_1)$ are used. In the above Eqn. (3.18), the term corresponding to the third term, in Eqn. (3.17) become intrinsically equal to zero, which means that zero degree of freedom remains for the higher manipulation variables, because the second manipulation variable $\boldsymbol{r}_2 = \boldsymbol{\theta}$ requires all the degree of freedom remaining after being used for $\boldsymbol{r}_1$.

3.8 PATH PLANNING

A serial robot manipulator may be regarded as a variable geometrical chain of links that associate the configuration of its end-effector to the Cartesian coordinate frame in which the base frame is fixed. Forward kinematics is mathematical geometrical relations that help in finding the end effector configuration by the making use of the joint coordinates. On the other hand, the inverse kinematics are mathematical geometrical relations that provide joint coordinates from a given end effector configuration.

The velocity and acceleration of each link of the manipulator can be found by expressing the Cartesian path of motion for the end-effector as a function of time. The first applied path function that can provide a rest-to-rest motion is a cubic path for the joint variable $q_i(t)$ between two points $q_i(t_0)$ and $q_i(t_f)$.

$$q_i(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \tag{3.19}$$

But to get zero acceleration or jerk at some point in our path, it is required to employ higher polynomials to satisfy the conditions. An n degree polynomial can satisfy $n + 1$ conditions.

$$\mathrm{q(t) = a_0 + a_1 t + a_2 t^2 + \cdots + a_n t^n} \tag{3.20}$$

It is also possible to split a multiple conditional path into some intervals with fewer conditions. The interval paths must then be connected to satisfy their boundary conditions.

A path of motion may also be defined based on different mathematical functions. Harmonic and cycloid functions are the most common paths. Non-polynomial equations introduce some advantages, due to simpler expression, and some disadvantages due to nonlinearities.

When a path of motion either in joint or Cartesian coordinates space is defined, forward and inverse kinematics must be utilized to find the path of motion in the other space.

3.8.1 Manipulator Motion by End-Effector Path

Cartesian path planning has more application in robotics, because it can control the level of force and jerk inserted by the hand of a robot to the carrying object. Path planning in Cartesian space also determines the geometric constraints of the external world. However, a Cartesian path needs inverse kinematics to determine the time history of the joint variables.

Here for a straight line trajectory, consider a rest to rest Cartesian path from point (1,0) to point (1,1). A cubic polynomial can satisfy the position and velocity constraints at initial and final points.

$$\left.\begin{aligned} Y(0) &= Y_0 = 0 \\ Y(1) &= Y_f = 1 \\ \dot{Y}(0) &= \dot{Y}_0 = 0 \\ \dot{Y}(1) &= \dot{Y}_f = 0 \end{aligned}\right\} \tag{3.21}$$

$$\boldsymbol{Y(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3} \tag{3.22}$$

On differentiating Eqn. (3.21),

$$\dot{Y}(t) = a_1 + 2a_2 t + 3a_3 t^2 \tag{3.23}$$

Solving the above equations to get the value of coefficients as

$$\left.\begin{aligned} a_0 &= 0 \\ a_1 &= 0 \\ a_2 &= 3 \\ a_3 &= -2 \end{aligned}\right\} \tag{3.24}$$

To get the Cartesian path as

$$\mathbf{X = 1} \tag{3.25}$$

$$\mathbf{Y = 3\,t^2 - 2\,t^3} \tag{3.26}$$

The manipulator moves from point (1,0) to (1,1) as the time changes from 0 to 1 minute and traces a straight line at *X=1* parallel to y axis. This way any desired trajectory can be traced by forming its equations.

3.9 SUMMARY

In this chapter, the concept of inverse kinematics considering the order of priority has been discussed, on the basis of which various simulations will be carried out in the next chapter. Main focussed is given on position prior to orientation, where the manipulator aims to follow a given trajectory along with maintaining a particular orientation of the end effector.

The main idea behind the trajectory planning has been discussed in this chapter, which is helpful in planning different trajectories for the serial robot manipulators to follow, in the next chapter.

CHAPTER 4 RESULTS AND DISCUSSIONS

4.1 INTRODUCTION

In this Chapter, different cases are presented and the redundancy of serial manipulators is resolved using the concept of task priority. Various simulations are conducted to show the redundancy resolution of the 3 link manipulator, when the manipulator aims to follow different trajectories like straight line motion, rectangular trajectory, triangular trajectory, circular and inclined line trajectory. All the simulations demonstrate the efficacy of redundancy resolution to perform multiple tasks with the concept of task priority. Moreover, the robot parameters are tuned, i.e. increasing the link length as well as DOF, to ensure the completeness of the subtasks.

4.2 POSITION PRIOR TO ORIENTATION

In this work, locally optimal control scheme is used, which resolves redundancy based on present information and this method is computationally economical. This control scheme is described and various simulations are performed for redundant manipulators using the task priority based redundancy control. The task is divided into subtasks with different levels of significance according to their priority and known as task with the order of priority. In this thesis, the redundant manipulator is required to follow a trajectory of end effector while keeping the orientation of the end-effector fixed at certain angle. So trajectory tracking may be given as the first priority and orientation has the second priority. Each subtask is performed using the extra DOF that remain after all the subtasks with higher priority have been fulfilled. Different simulations are performed by varying the parameters and shapes of trajectory [20].

4.3 NUMERICAL SIMULATIONS

Simulation results are presented and discussed in this section. The position of the end effector is given higher priority to its orientation for a three link manipulator. Results are obtained by choosing the most optimal solutions out of the infinite solutions available. Further it is shown that either by increasing the link lengths or by increasing the degree of freedom, it is possible to achieve both the desired position and orientation of the end effector. The simulation results show that more degrees of redundancy denote fulfilment of more subtasks according to the given priority.

This case is known as the position prior to orientation as the position of the end effector is given the first priority and the orientation has the second priority. The Fig. 4.1 shows the physical model of a planer 3-DOF manipulator having length of each link l_1, l_2 and l_3 respectively and the corresponding joint angles are θ_1, θ_2 and θ_3. It is assumed that the mass of each link is uniformly distributed, and that the manipulator is constrained in a horizontal plane that is it is a planer robot. The dynamics part is also required for making the code to carry out the required simulations.

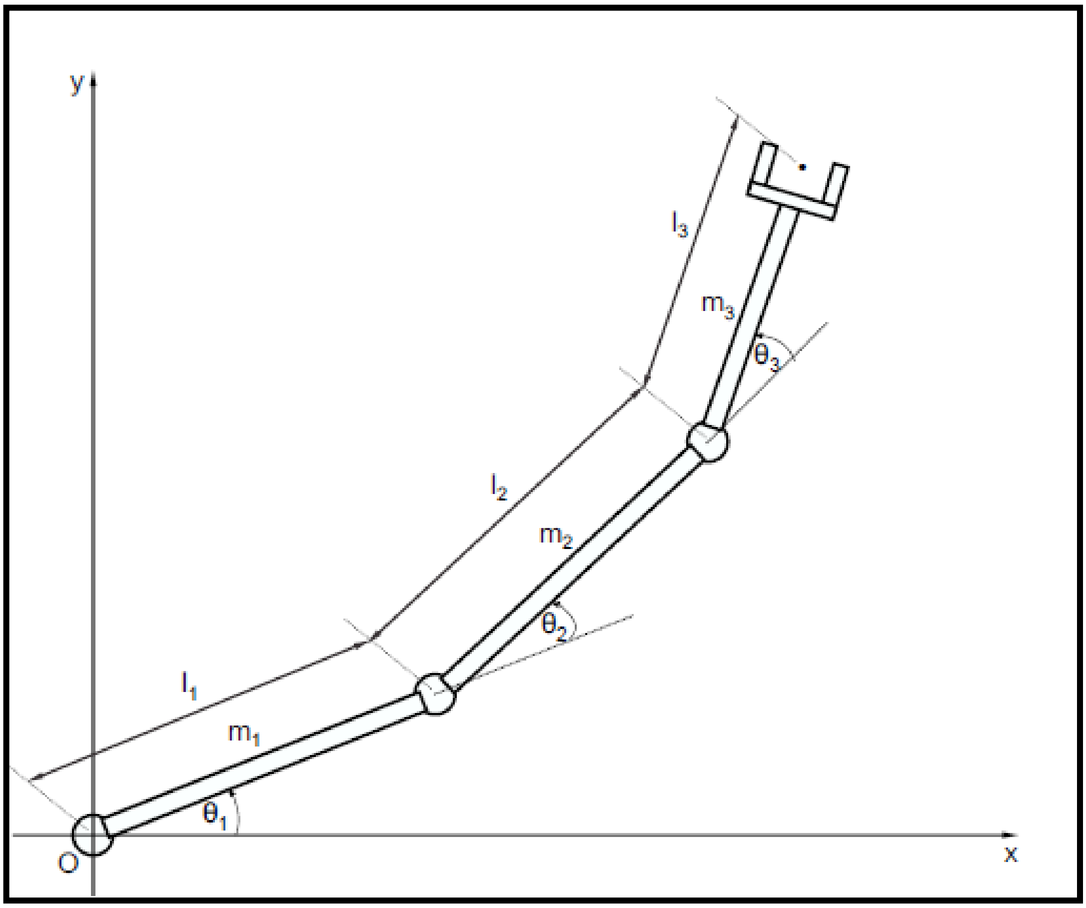

Fig. 4.1: Planar 3-DOF manipulator.

The dynamics of a general n degree of freedom manipulator is represented by

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \tag{4.1}$$

where $\boldsymbol{M}(\boldsymbol{\theta}) \in \mathcal{R}^{n\times n}$ is an inertia matrix, $\boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \in \mathcal{R}^{n}$ is a torque vector, which is produced by Centripetal and Coriolis forces and $\tau \epsilon \mathcal{R}^{n}$ is a joint torque vector. Let

$$\boldsymbol{r_1} = [x\ y]^T \tag{4.2}$$

$$r_2 = (\theta_1 + \theta_2 + \theta_3 \dots \dots \dots + \theta_n) \tag{4.3}$$

Where $\boldsymbol{r_1}$ represents the position of the end effector and r_2 is the orientation of the end effector of n-link manipulator. Here in this case, n = 3. Differentiate Eqn. (3.2) to obtain

$$\ddot{\boldsymbol{r_i}} = \boldsymbol{J_i}\ddot{\boldsymbol{\theta}} + \dot{\boldsymbol{J_i}}\dot{\boldsymbol{\theta}} \tag{4.4}$$

Now, a feedback control scheme is designed so that the following equation illustrates the closed loop characteristics

$$\ddot{\boldsymbol{e_i}} + c_{1i}\dot{\boldsymbol{e_i}} + c_{2i}\boldsymbol{e_i} = 0, \tag{4.5}$$

$$\boldsymbol{e_i} = \boldsymbol{r_i^d}(\boldsymbol{t}) - \boldsymbol{r_i}. \tag{4.6}$$

Where $\boldsymbol{r_i^d}(\boldsymbol{t})$ is the desired trajectory of the i^{th} manipulation variable, $\boldsymbol{e_i}$ is the error vector and c_{1i} and c_{2i} are the positive feedback coefficients. Now put the above feedback scheme in Eqn. (4.4) to get,

$$J_i\ddot{\boldsymbol{\theta}} = \ddot{\boldsymbol{r}}_i^{\boldsymbol{d}}(\boldsymbol{t}) - \dot{\boldsymbol{J}}_i\dot{\boldsymbol{\theta}} + c_{1i}\dot{e}_i + c_{2i}e_i = \boldsymbol{h}_i(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \boldsymbol{t}) \tag{4.7}$$

On comparing Eqn. (4.7) with Eqn. (3.2),it can be seen that both the equations are similar, so same approach is followed as before. Using the Eqn. (3.18), solve Eqn. (4.7) for i=2 and regarding the result $\ddot{\boldsymbol{\theta}} = \boldsymbol{J_2^{\#}h_2}$ as the desired acceleration of the second manipulation variable. To obtain,

$$\ddot{\boldsymbol{\theta}} = \boldsymbol{J_1^{\#}h_1} + (\boldsymbol{I_n} - \boldsymbol{J_1^{\#}J_1})\boldsymbol{J_2^{\#}h_2} \tag{4.8}$$

After all these calculations, lastly the necessary joint torque is calculated to generate the acceleration by putting Eqn. (4.8) into Eqn. (4.1).

Simulations are performed with different trajectories to be followed and graphs are plotted to show the best possible solution chosen to follow the given path. All this is achieved by making a general code for a robot manipulator using MATLAB and obtaining the desired plots.

4.3.1 Straight Line Trajectory

In the first simulation, which is shown in Fig. 4.2, a 3-DOF manipulator is used whose desired task which is given by $r_1^d(t)$ is to follow a straight line trajectory parallel to y-axis without any task priority that is the manipulator moves in any manner to trace the trajectory without concerning about any orientation of the end effector. On the other hand, when the concept of task priority is applied, the manipulator aims at following the straight line trajectory and to fulfil the desired orientation given by $r_2^d(t)$ that is the end effector should be perpendicular to the line which is obtained by maintaining $\emptyset = 0^{\circ}$. This simulation is shown in Fig. 4.3.

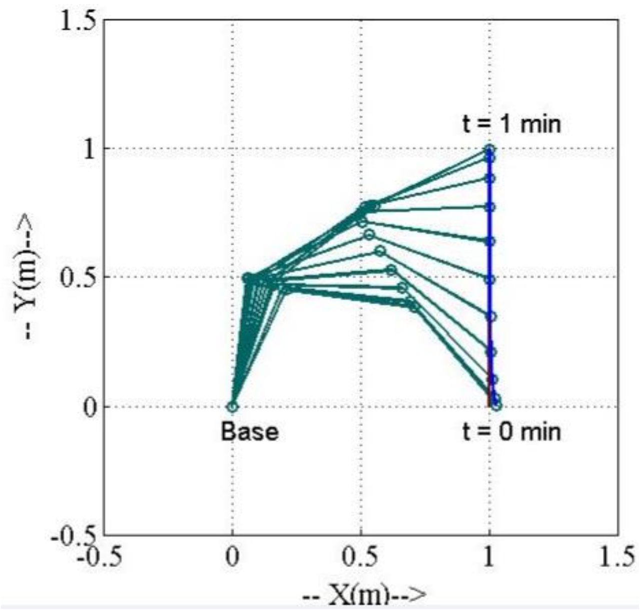

Fig. 4.2: 3-DOF manipulator motion without task priority following straight line trajectory.

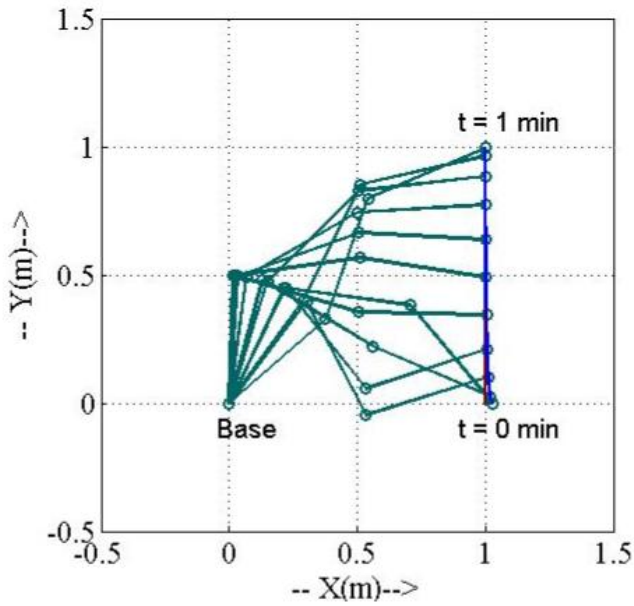

Fig. 4.3: 3-DOF manipulator motion with task priority at $\emptyset = 0^{\circ}$ following straight line trajectory.

Here, trajectory tracing by the three link manipulator is given the higher priority and the orientation of the end effector has the lower priority. That means the main preference will be given to follow the straight line path without any deviation from the desired path and thereafter aiming to keep the end effector perpendicular to the straight line.

The manipulator moves from point (1, 0) at time t = 0 min to point (1, 1) at time t = 1 min. So the time taken to perform the task is 1 min. The lengths of each link of the manipulator are taken to be equal and $l = [0.5 \quad 0.5 \quad 0.5]$ (m)and the initial configuration is taken as $\theta^0 = [65^\circ \quad -73^\circ \quad -42^\circ]$.

It can be clearly seen in Fig. 4.3, that both the subtasks i.e. tracking the desired trajectory and manipulating the desired orientation, are achieved in major part of the trajectory. However, towards the end of the trajectory, the manipulator is not able to maintain the required orientation. This is due to the reason that at the far most point of the trajectory, it is not possible to maintain the required orientation, due to link-length constraints. This problem can be rectified by increasing the link lengths of the manipulator and will be demonstrated in the next set of simulations.

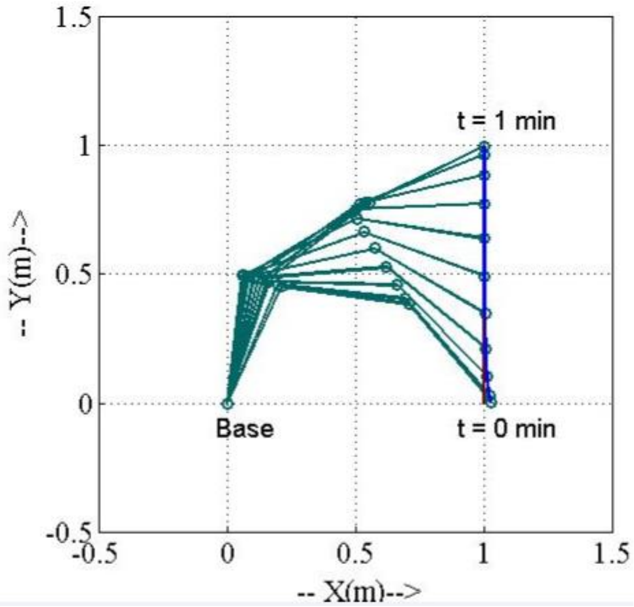

Fig. 4.4: 3-DOF manipulator motion without task priority following straight line trajectory.

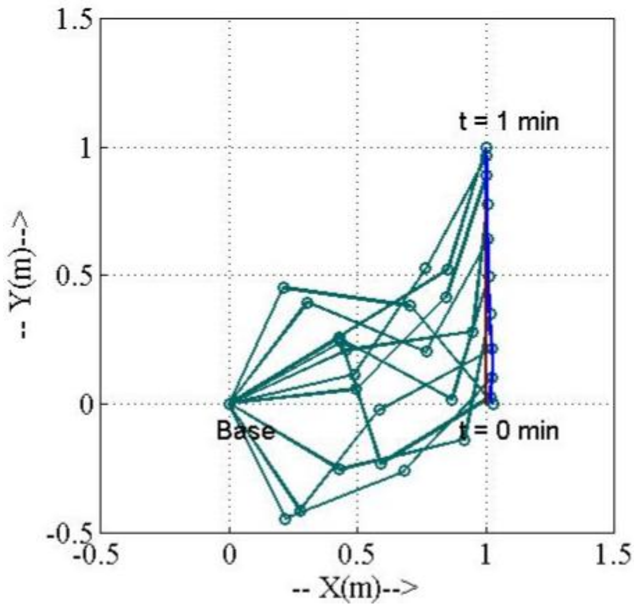

Fig. 4.5: 3-DOF manipulator motion with task priority at $\emptyset = 90^\circ$ following straight line trajectory.

In the next simulation run, as shown in Fig. 4.5, the desired trajectory is the same but the orientation of the end effector has been changed from 0° to 90° and now the end effector aims in moving parallel to the straight line that is perpendicular to *x*-axis. A comparison between the motion of the manipulator without task priority and with task priority is shown in the graphs below. It can be easily seen how the manipulator choose its own best orientations when its motion is without task priority, whereas in the case of task priority the solutions are difficult to obtain which completely satisfy both the subtasks.

From the above simulations, it can be seen that for a three link manipulator the second priority is not attained throughout the trajectory as both the subtasks cannot be satisfied simultaneously and the priority is given to the first manipulation variable.

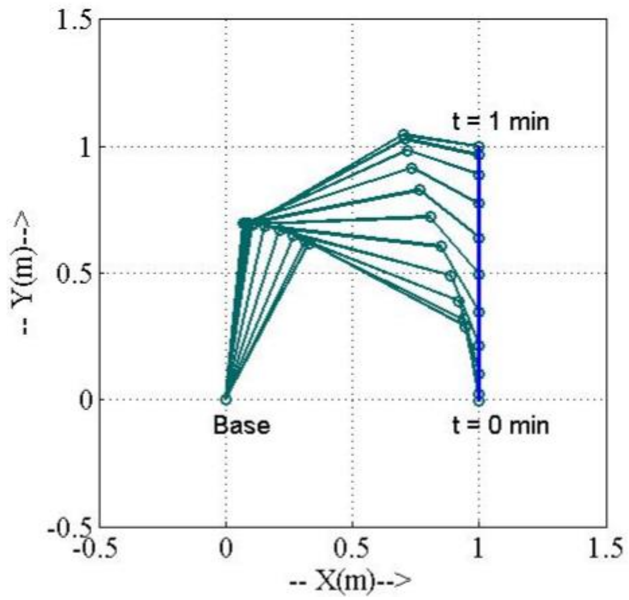

Fig. 4.6: 3-DOF manipulator motion for increased link length without task priority following straight line trajectory.

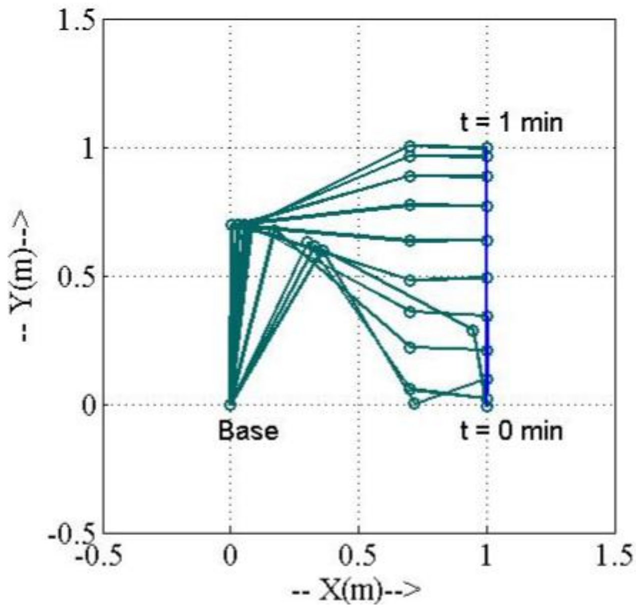

Fig. 4.7: 3-DOF manipulator for increased length with task priority at $\emptyset = 0^{\circ}$ following straight line trajectory.

In the next two simulation runs, as shown in Figs. 4.7 and 4.9, are used to see the validity of the formulation by tuning the parameters. The overall length of the robot manipulator is increased and the new length $l = [0.7 \;\; 0.7 \;\; 0.3]$ (m). Also, the new initial configuration is taken to be $\theta^0 = [62^{\circ} \;\; -90^{\circ} \;\; -52^{\circ}]$. Figs. 4.6 and 4.8 shows the motion of the manipulator with increased link length without following task priority in comparison with those which follow the task priority. It can be clearly seen that on increasing the overall link length of the manipulator both the subtasks are performed with maximum accuracy. The solutions obtained after increasing the link lengths are more precise than the previous results for both the cases of $\emptyset = 90^{\circ}$ and 0°.

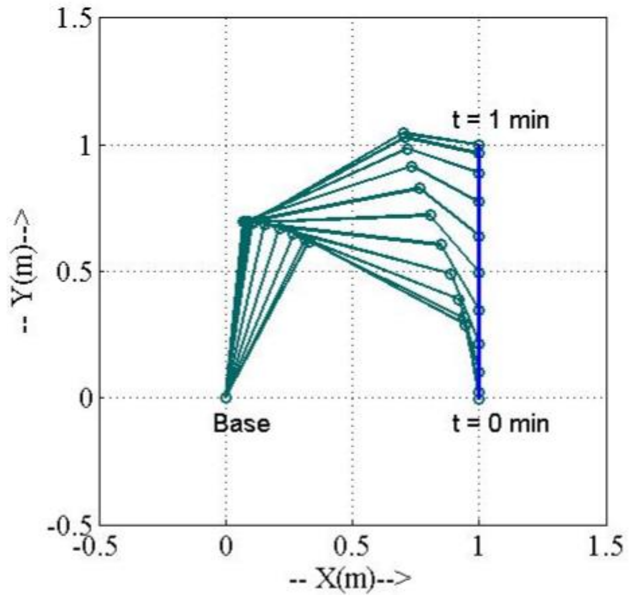

Fig. 4.8: 3-DOF manipulator motion for increased link length without task priority following straight line trajectory.

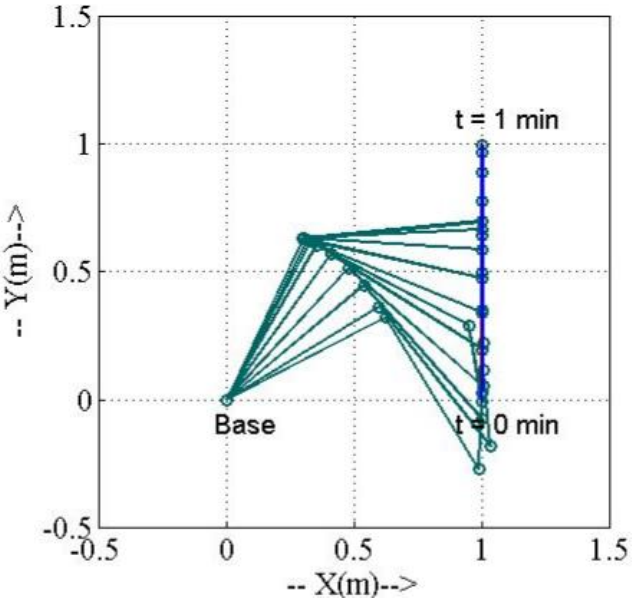

Fig. 4.9: 3-DOF manipulator for increased length with task priority at $\emptyset = 90^{\circ}$ following straight line trajectory.

This shows the validation of how the extra redundancy can be utilised to fulfil the desired tasks.

The feedback coefficients used in Eqn. (22) are chosen as $c_{11} = 20\ (1/\mathrm{min})$,

$c_{21} = 100\ (1/\mathrm{min}^2)$, $c_{12} = 100\ (1/\mathrm{min})$ and $c_{22} = 2500\ (1/\mathrm{min}^2)$.

4.3.2 Rectangular Trajectory

The concept of task priority is applied on a three link manipulator whose aim is to follow the two given subtasks. The primary subtask is to follow the rectangular trajectory given by $r_1^d(t)$ and the secondary subtask is to fulfil the desired orientation of the end effector given by $r_2^d(t)$. Also the simulations are compared for motion of the manipulator without task priority and with task priority along with their orientations.

Trajectory tracing by the three link manipulator is given the higher priority and the orientation of the end effector has the lower priority. That means the main preference will be given to follow the rectangular path without any deviation from the desired path and thereafter aiming to keep the end effector parallel or perpendicular to x-axis.

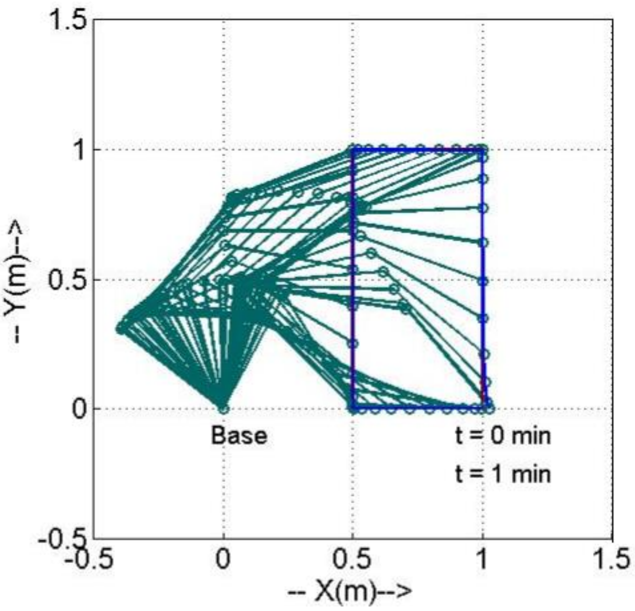

Fig. 4.10: 3-DOF manipulator motion without task priority following rectangular trajectory.

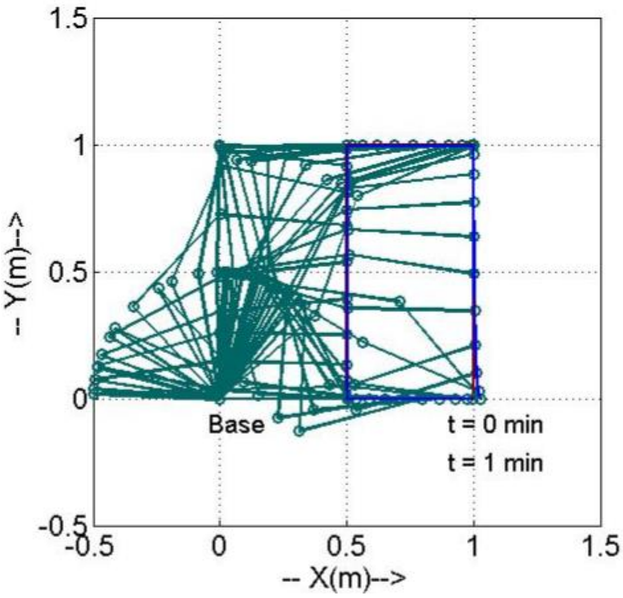

Fig. 4.11: 3-DOF manipulator motion with task priority at $\emptyset = 0^{\circ}$ following rectangular trajectory.

In the simulation run as shown in Fig. 4.10 the manipulator moves along a rectangular trajectory of length 1 meter and width 0.5 meter without any task priority. The end effector aims to follow the rectangular trajectory with its best suitable orientation.

On the other hand Fig. 4.11 shows the simulation when the concept of task priority is applied to the manipulator which works to follow rectangular trajectory along with the fulfilment of the desired orientation given by $r_2^d(t)$ that is the end effector should be parallel to x-axis which is obtained by maintaining $\emptyset = 0^{\circ}$. The manipulator starts and ends its motion from point (1, 0). The lengths of each link of the manipulator are taken to be equal and $l = [0.5 \;\; 0.5 \;\; 0.5]$ (m)and the initial configuration is taken as $\theta^0 = [65^\circ \;\; -73^\circ \;\; -42^\circ]$.

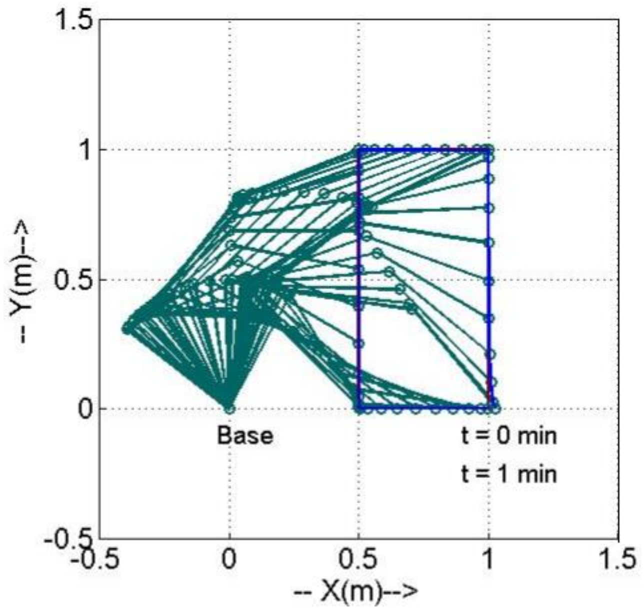

Fig. 4.12: 3-DOF manipulator motion without task priority following rectangular trajectory.

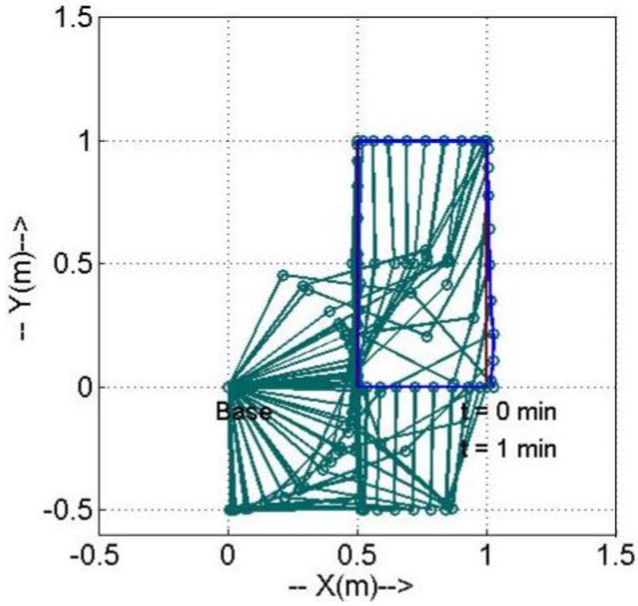

Fig. 4.13: 3-DOF manipulator motion with task priority at $\emptyset = 90^\circ$following rectangular trajectory.

In the next simulation as shown in Fig. 4.13, the desired trajectory is the same but the orientation of the end effector has been changed from 0° to 90° and now the end effectors aim is to move perpendicular to x-axis. A comparison between the motions of the manipulator without task priority and with task priority can be seen by comparing Figs. 4.12 and 4.13.

It can be clearly seen how the manipulator choose its own best orientations when its motion is without task priority, whereas in the case of task priority the solutions are difficult to obtain which completely satisfy both the subtasks and thus the second priority is not attained throughout the trajectory as both the subtasks cannot be satisfied simultaneously and the priority is given to the first manipulation variable.

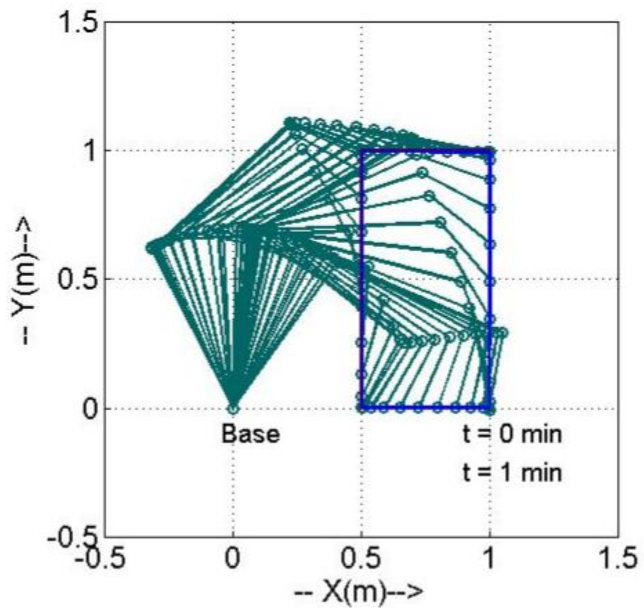

Fig. 4.14: 3-DOF manipulator motion for increased link length without task priority following rectangular trajectory.

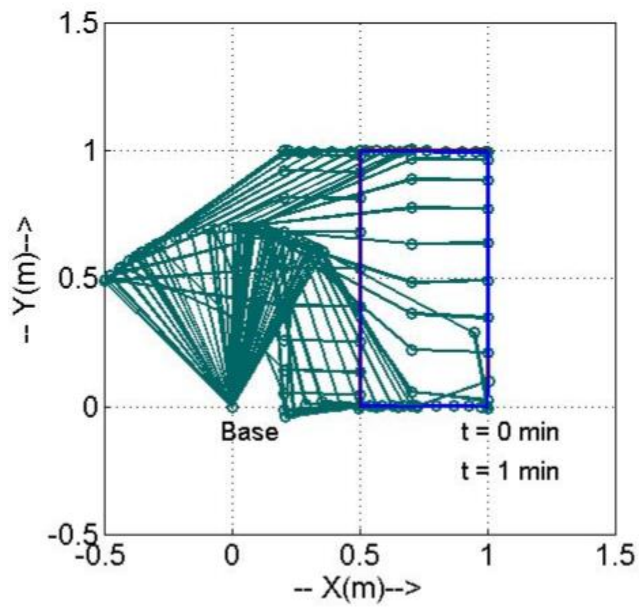

Fig. 4.15: 3-DOF manipulator for increased length with task priority at $\emptyset = 0^{\circ}$ following rectangular trajectory.

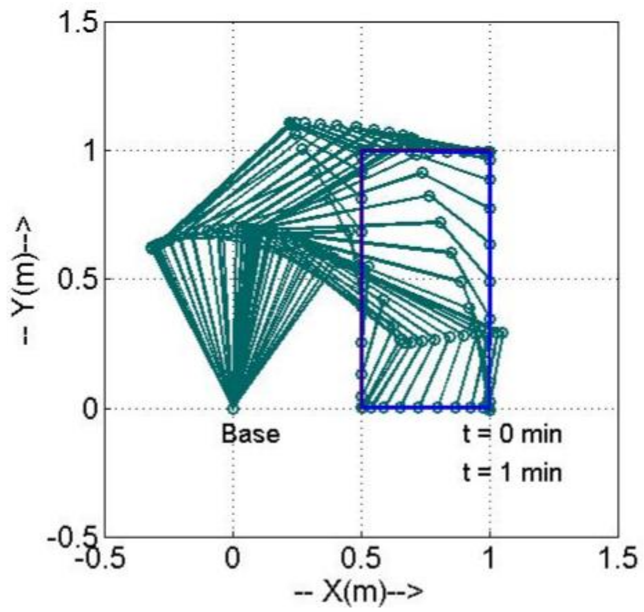

Fig. 4.16: 3-DOF manipulator motion for increased link length without task priority following rectangular trajectory.

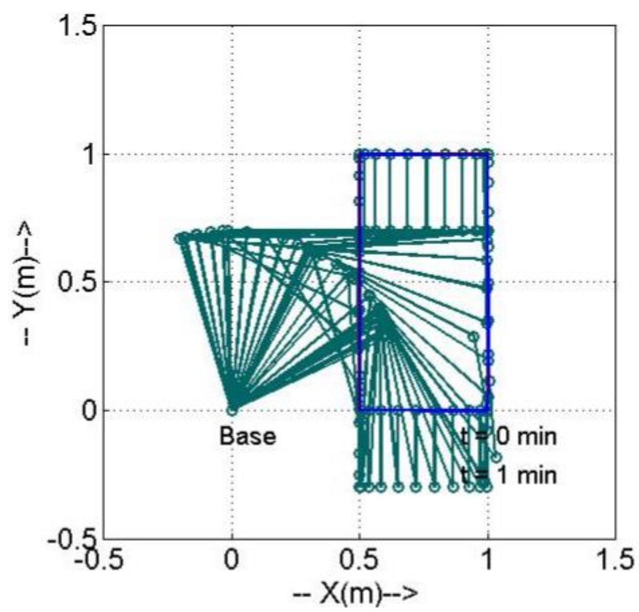

Fig. 4.17: 3-DOF manipulator for increased length with task priority at $\emptyset = 90^{\circ}$ following rectangular trajectory.

The next two cases as shown in Figs. 4.15 and 4.17 to see the validity of the formulation by tuning the parameters. This is achieved by increasing the overall length of the robot manipulator and the new length is given as $l = [0.7 \;\; 0.7 \;\; 0.3]$ (m). Also the new initial configuration is taken to be $\theta^0 = [62^\circ \;\; -90^\circ \;\; -52^\circ]$.

Figs. 4.14 and 4.16 shows the motion of the manipulator with increased link length without following task priority in comparison with those which follow the task priority.

It can be clearly seen that on increasing the overall link length of the manipulator both the subtasks are performed with maximum accuracy. The solutions obtained after increasing the link lengths are more precise than the previous results for both the cases of $\emptyset = 90^\circ$ and 0°. This shows the validation of how the extra redundancy can be utilised to fulfil the desired tasks.

4.3.3 Triangular Trajectory

Similar is the case of triangular trajectory and here also both the subtasks of tracking the desired trajectory and manipulating the desired orientation, are achieved in major part of the trajectory. But, the manipulator is not able to maintain the required orientation. This is done to the reason that at the far most point of the trajectory, it is not possible to maintain the required orientation, due to link-length constraints.

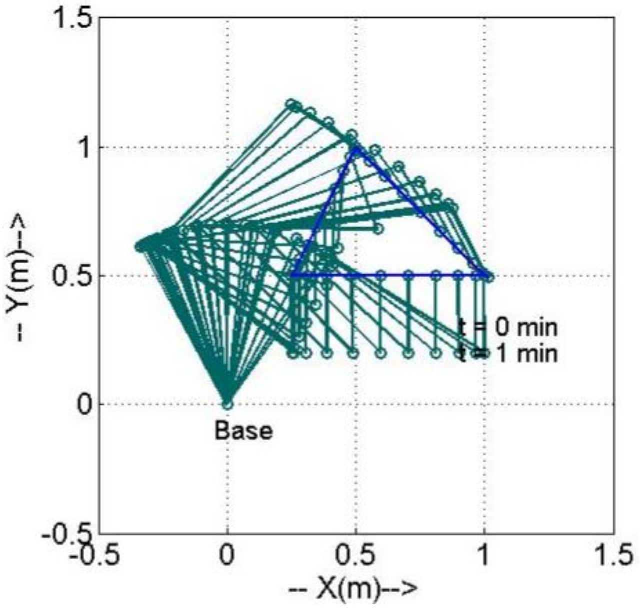

Fig. 4.18: 3-DOF manipulator motion for increased link length without task priority following tiangular trajectory.

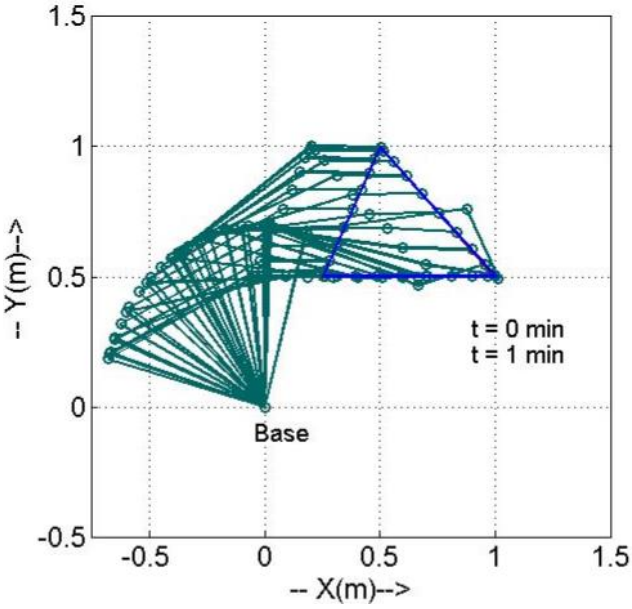

Fig. 4.19: 3-DOF manipulator for increased length with task priority at $\emptyset = 0^\circ$ following tiangular trajectory.

This problem can be rectified by increasing the link length of the manipulator and will be demonstrated in the next set of simulations. In the simulation shown in Fig. 4.18 the manipulator moves along a triangular trajectory without any task priority with increased length of the manipulator, whose three vertices are at (1, 0.5), (0.5, 1) and (0.25, 0.5). The end effector aims to follow the triangular trajectory with its best suitable orientation.

On the other hand Fig. 4.19 shows the simulation when the concept of task priority is applied to the manipulator which works to follow triangular trajectory along with the fulfilment of the desired orientation in such a way that the end effector should be parallel to x-axis which is obtained by maintaining $\emptyset = 0^{\circ}$. The manipulator starts and ends its motion from point (1, 0.5). The increased lengths of each link of the manipulator are taken to be equal and $l = [0.7 \;\; 0.7 \;\; 0.3]$ (m) and the initial configuration is taken as $\theta^0 = [75^{\circ} \; -68^{\circ} \; -70^{\circ}]$.

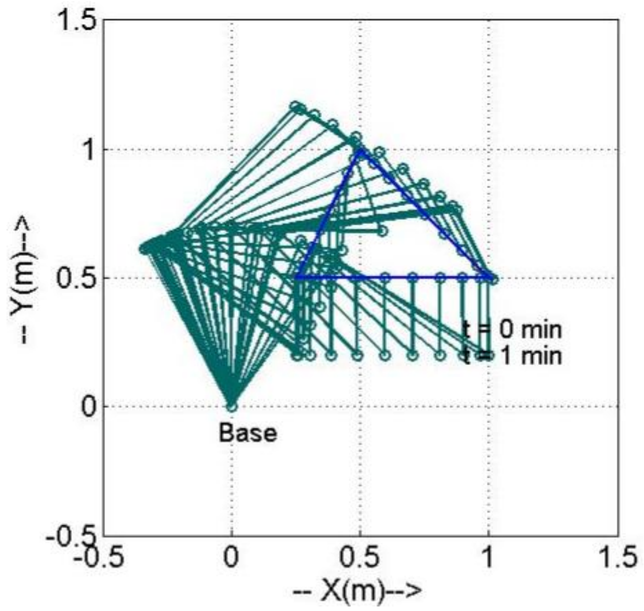

Fig. 4.20: 3-DOF manipulator motion for increased link length without task priority following tiangular trajectory.

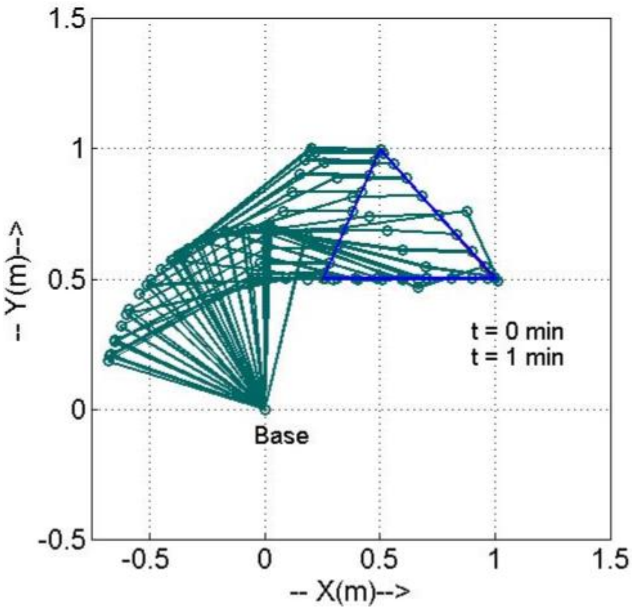

Fig. 4.21: 3-DOF manipulator for increased length with task priority at $\emptyset = 90^{\circ}$ following tiangular trajectory.

Similarly from the simulation shown in Fig. 4.21, the desired trajectory is the same but the orientation of the end effector has been changed from 0° to 90° and now the end effectors aim is to move perpendicular to x-axis. A comparison between the motions of the manipulator without task priority and with task priority can be seen by comparing Fig. 4.20 and Fig. 4.21. and it can be clearly observed how the manipulator chooses its own best orientations when its motion is without task priority whereas in the case of task priority, it can be clearly seen that on increasing the overall link length of the manipulator both the subtasks are performed with maximum accuracy. The solutions obtained after increasing the link lengths are more precise than the previous results for both the cases of $\emptyset = 90^{\circ}$ and 0°. This shows the validation of how the extra redundancy can be utilised to fulfil the desired tasks.

4.3.4 CIRCULAR TRAJECTORY

Similar results are obtained when the manipulator moves along a circular trajectory without any task priority with increased link length of the manipulator as shown in Fig. 4.22. The circular trajectory is having its centre at origin with the radius of 0.6 meter. The end effector aims to follow the circular trajectory with its best suitable orientation.

On the other hand Fig. 4.23 shows the simulation when the concept of task priority is applied to the manipulator which works to follow the circular trajectory along with the fulfilment of the desired orientation in such a way that the end effector should be parallel to x-axis which is obtained by maintaining $\emptyset = 0^{\circ}$.

The new increased lengths of each link of the manipulator are taken to be equal and $\mathrm{l} = [0.7 \quad 0.7 \quad 0.3]$ (m) and the initial configuration is taken as $\theta^0 = [80^{\circ} \ -125^{\circ} \ -50^{\circ}]$. Similarly from the simulation shown in Fig. 4.25, the desired trajectory is the same but the orientation of the end effector has been changed from 0° to 90° and now the end effectors aim is to move perpendicular to x-axis.

A comparison between the motions of the manipulator without task priority and with task priority can be seen by comparing Fig. 4.24 and Fig. 4.25. and it can be clearly observed how the manipulator chooses its own best orientations when its motion is without task priority whereas in the case of task priority, it can be clearly seen that on increasing the overall link length of the manipulator both the subtasks are performed with maximum accuracy. The solutions obtained after increasing the link lengths are more precise than the previous results for both the cases of $\emptyset = 90^{\circ}$ and 0°. This shows the validation of how the extra redundancy can be utilised to fulfil the desired tasks.

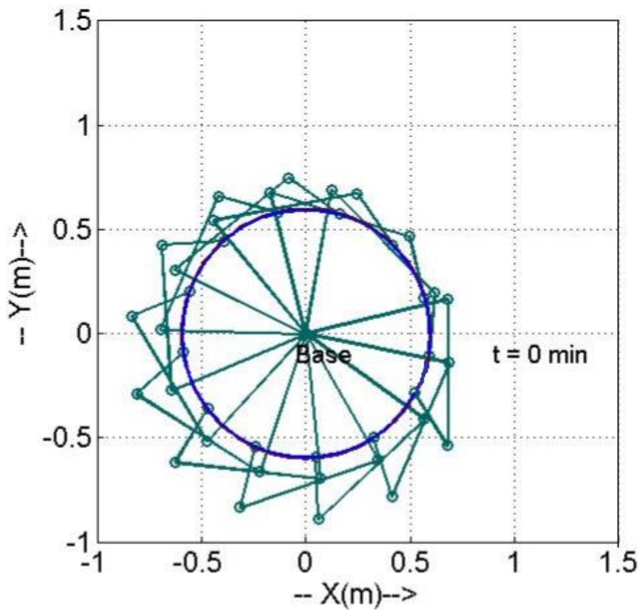

Fig. 4.22: 3-DOF manipulator motion for increased link length without task priority following circular trajectory.

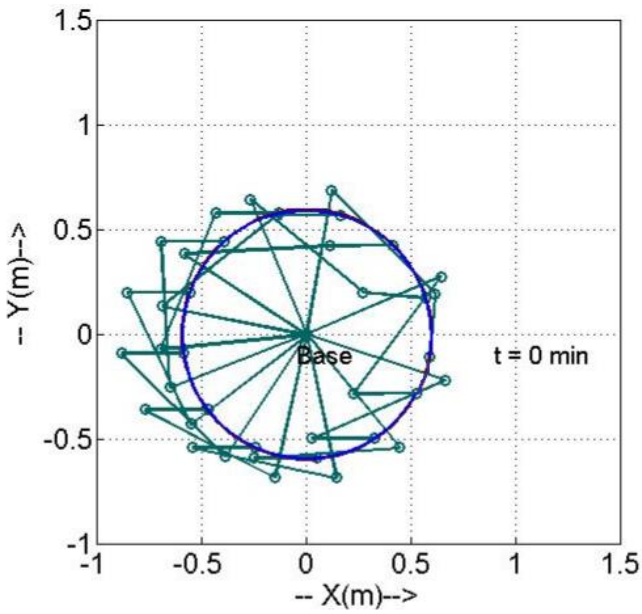

Fig. 4.23: 3-DOF manipulator for increased length with task priority at $\emptyset = 0^{\circ}$ following circular trajectory.

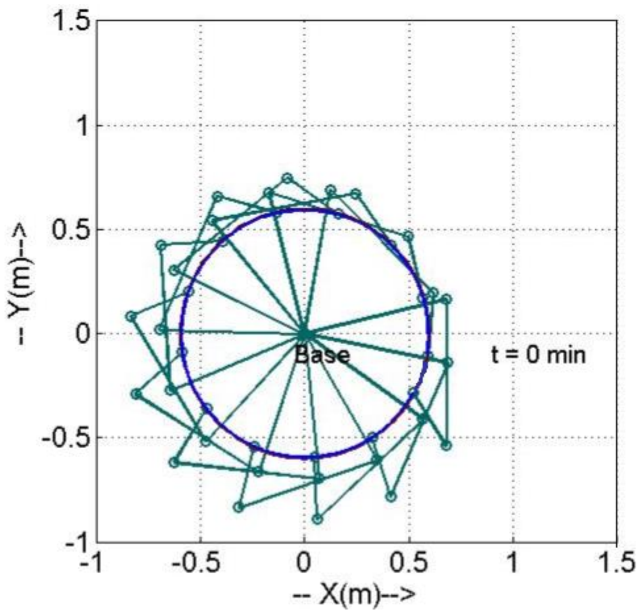

Fig. 4.24: 3-DOF manipulator motion for increased link length without task priority following circular trajectory.

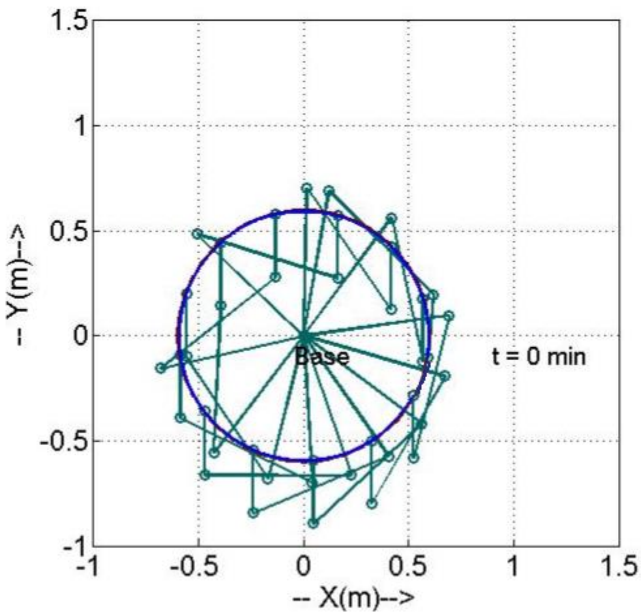

Fig. 4.25: 3-DOF manipulator for increased length with task priority at $\emptyset = 90^{\circ}$ following circular trajectory.

4.3.5 Inclined Line Trajectory

This is the different case in which, the DOF of the manipulator is increased keeping its total length constant to see the validity of the formulation. Now, the concept of task priority is applied on a three link manipulator whose aim is to follow the two given subtasks. The primary subtask is to follow the inclined line trajectory given by $r_1^d(t)$ and the secondary subtask is to fulfil the desired orientation of the end effector given by $r_2^d(t)$. Also the simulations are compared for motion of the manipulator without task priority and with task priority along with their orientations.

Trajectory tracking by the three link manipulator is given the higher priority and the orientation of the end effector has the lower priority. That means the main preference is given to follow the inclined path without any deviation from the desired path and thereafter aiming to keep the end effector along the inclined line, parallel to x-axis and perpendicular to x-axis.

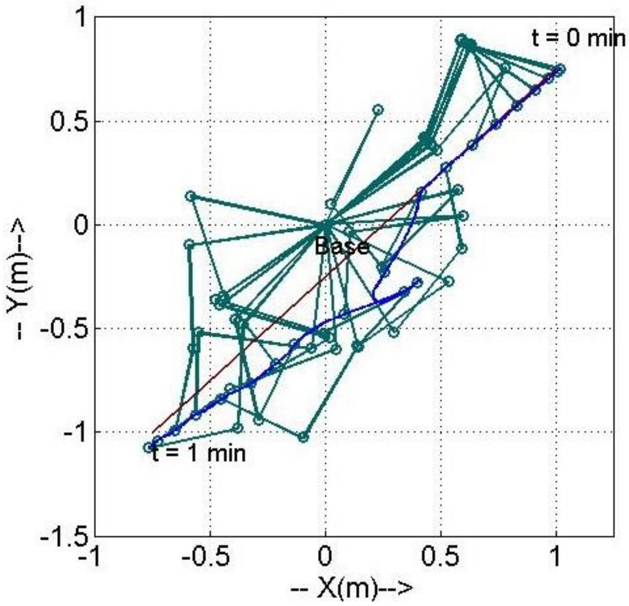

Fig. 4.26: 3-DOF manipulator motion without task priority following inclined line trajectory.

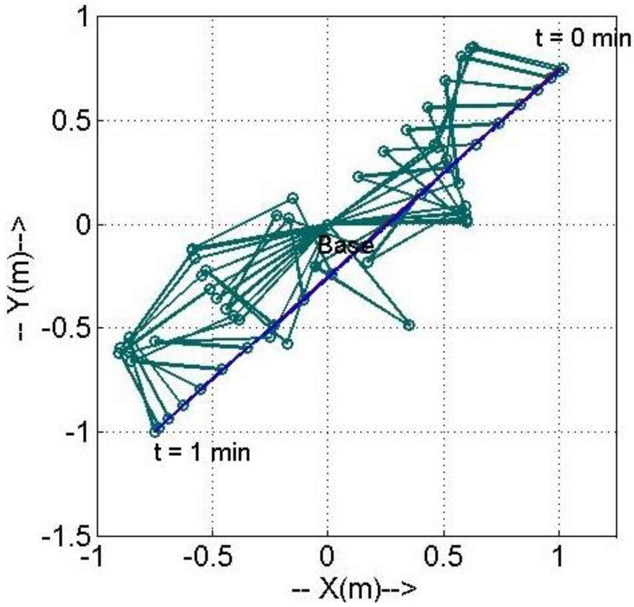

Fig. 4.27: 3-DOF manipulator motion with task priority at $\emptyset = 0^{\circ}$ following inclined line trajectory.

The simulation for manipulator motion without task priority is shown in Fig. 4.26 where the manipulator aims to move along the inclined line trajectory which is shown by the red line and the actual path traced by the manipulator is given by the blue path. Here the manipulator is unable to follow the given path because it is reaching at singular points and after that its joint velocities becomes extremely high in certain directions. This is the case of inverse kinematics when manipulator moves without any task priority. The end effector aims to follow the inclined line trajectory with its best suitable orientation.

On the other hand Fig. 4.27 shows the simulation when the concept of task priority is applied to the manipulator which works to follow the given trajectory along with the fulfilment of the desired orientation given by $r_2^d(t)$ that is the end effector should be parallel to x-axis which is obtained by maintaining $\emptyset = 0^{\circ}$. The lengths of each link of the manipulator are taken as $\mathrm{l} = [0.6 \quad 0.5 \quad 0.4]$ (m) and the initial configuration is taken as $\theta^0 = [40^{\circ} \quad 30^{\circ} \quad -85^{\circ}]$. It can be clearly seen that by using the concept of task priority the manipulator is able to trace the given trajectory thus overcoming the problem of singularity.

In the simulation shown in Fig. 4.29, the desired trajectory is the same but the orientation of the end effector has been changed from 0° to 90° and now the end effectors aim is to move perpendicular to x-axis. A comparison between the motions of the manipulator without task priority and with task priority can be seen by comparing Figs. 4.28 and 4.29.

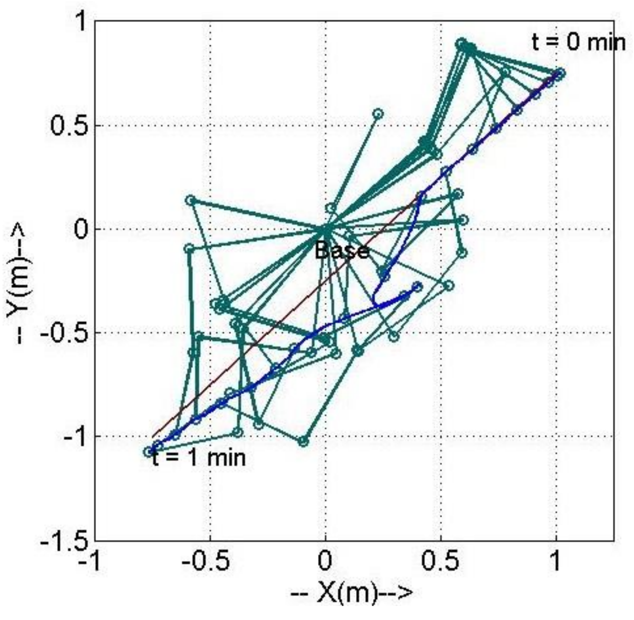

Fig. 4.28: 3-DOF manipulator motion without task priority following inclined line trajectory.

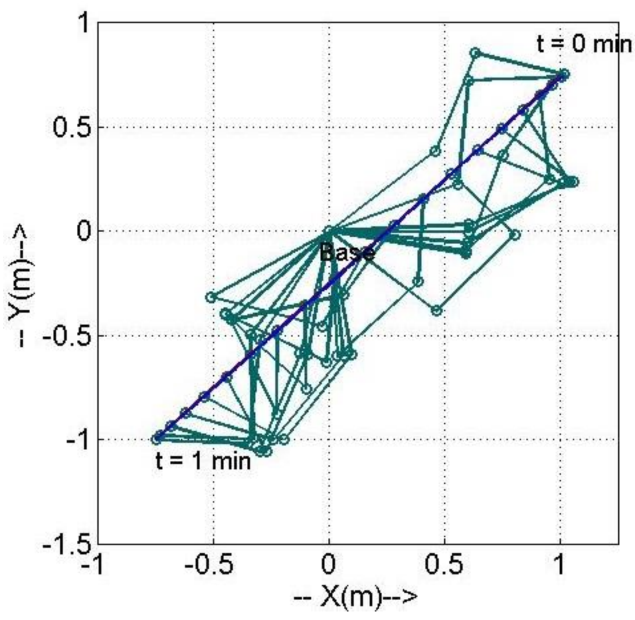

Fig. 4.29: 3-DOF manipulator motion with task priority at $\emptyset = 90^{\circ}$ following inclined line trajectory.

It can be seen how the manipulator choose its own best orientations when its motion is without task priority whereas in the case of task priority the solutions are difficult to obtain which completely satisfy both the subtasks and thus the second priority is not attained throughout the trajectory as both the subtasks cannot be satisfied simultaneously and the priority is given to the first manipulation variable.

Similarly, same conclusions are drawn from the simulations shown in Fig. 4.31, where the end effector of the manipulator moves along the inclined line which is at 45° to the x-axis.

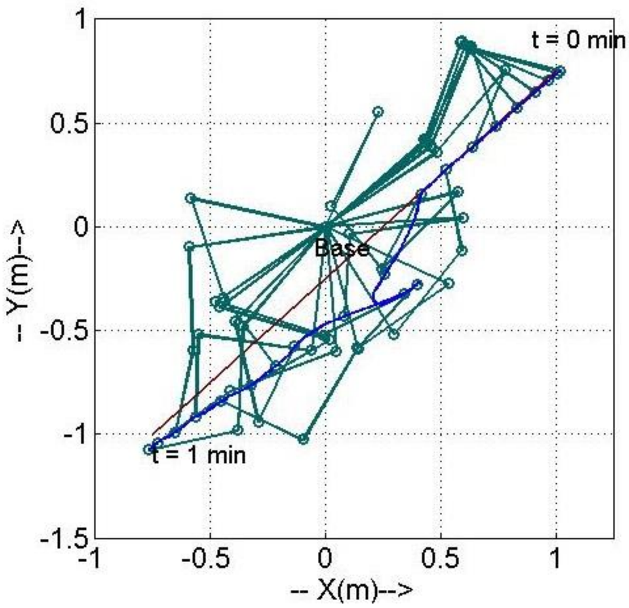

Fig. 4.30: 3-DOF manipulator motion without task priority following inclined line trajectory.

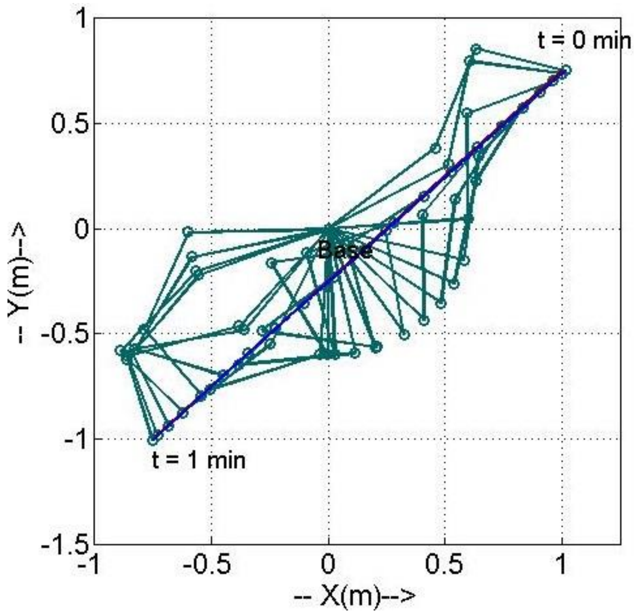

Fig. 4.31: 3-DOF manipulator motion with task priority at $\emptyset = 45^{\circ}$ following inclined line trajectory.

The above simulations shows that by using the concept of task priority singularity can be avoided and the manipulator can easily trace the desired trajectory which is the primary subtask. However, the secondary subtask cannot be fulfilled completely and the manipulator could not maintain the exact orientation of the end- effector.

It can be clearly seen in Figs. 4.27, 4.29 and 4.31, that both the subtasks are achieved in major parts of the trajectory. But, towards the end of the trajectory, the manipulator is not able to maintain the required orientation. This problem can be rectified by increasing the DOF of the manipulator, keeping the total length of the manipulator constant. This case is demonstrated in the next set of simulations.

Now the cases are discussed shown in Figs. 4.33, 4.35 and 4.37 to see the validity of the formulation by tuning the parameters for the motion of the manipulator, whose end effector is required to be oriented at $0^{\circ}, 90^{\circ}$ and 45°, respectively. This is achieved by increasing the DOF of the robot manipulator to four i.e. $n = 4$, without changing the overall length of the manipulator. So the new length, $l = [0.4 \quad 0.4 \quad 0.4 \; 0.3]$ (m). Also the new initial configuration is taken to be $\theta^0 = [45^{\circ} \; 35^{\circ} \; -70^{\circ} \; -30^{\circ}]$.

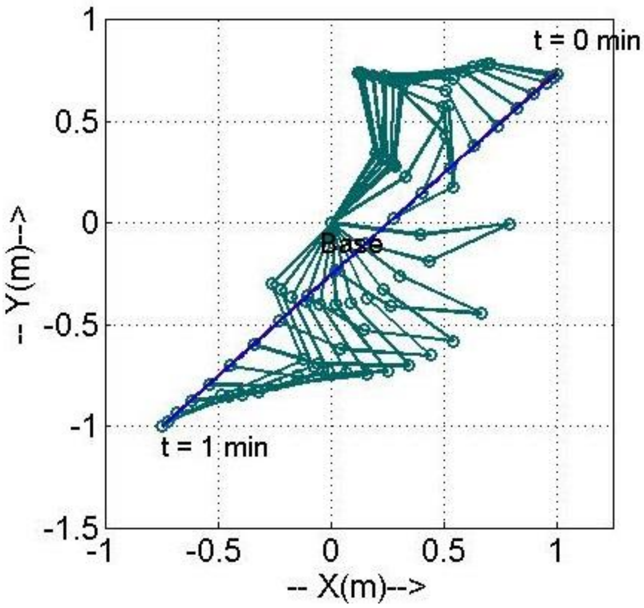

Fig. 4.32: 4-DOF manipulator motion for without task priority following inclined line trajectory.

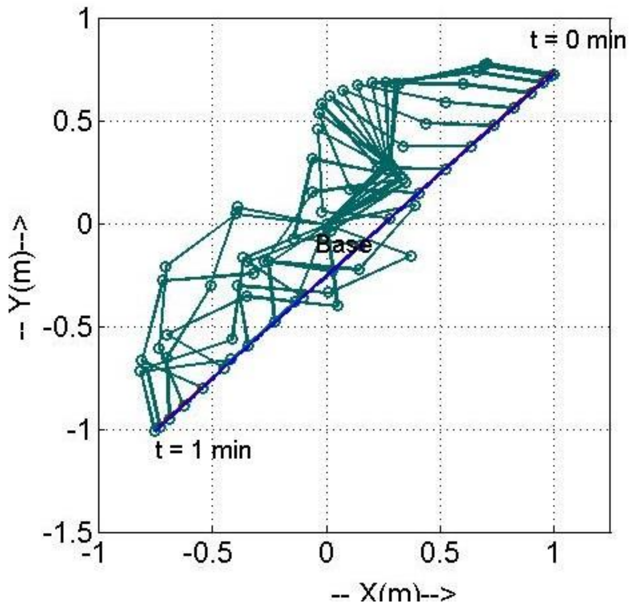

Fig. 4.33: 4-DOF manipulator with task priority at $\emptyset = 0^{\circ}$ following inclined line trajectory.

Figs. 4.32 and 4.34 shows the motion of the manipulator with increased DOF keeping the overall length of manipulator constant as 15 meter without following task priority in comparison with those which follow the task priority. It is shown that on increasing the DOF to 4, the singular configurations can be avoided even in the case, when the manipulator is not following task priority.

Thus, both the subtasks are successfully completed by increasing the DOF of the manipulator, while maintaining the overall length as constant.

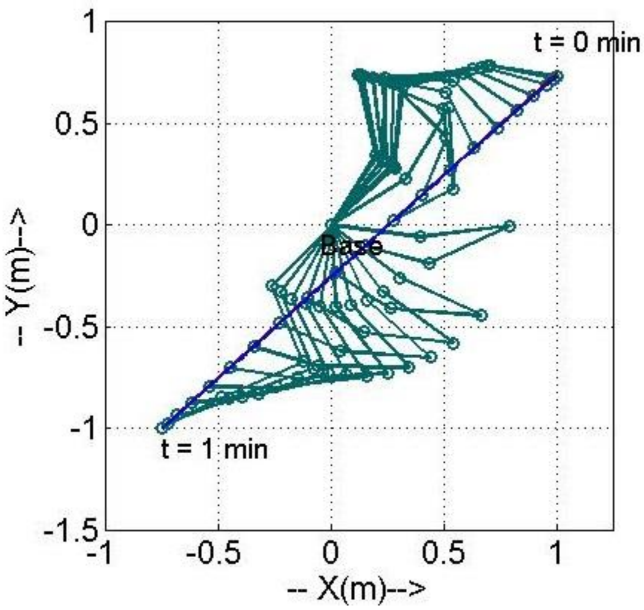

Fig. 4.34: 4-DOF manipulator motion for without task priority following inclined line trajectory.

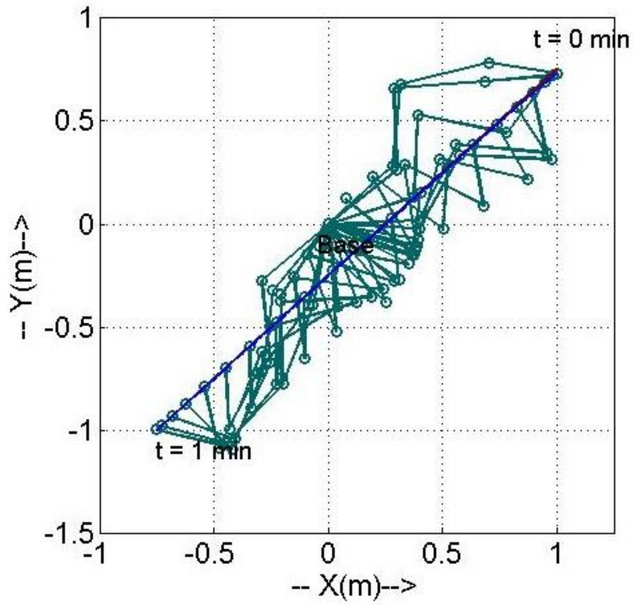

Fig. 4.35: 4-DOF manipulator with task priority at $\emptyset = 90^{\circ}$ following inclined line trajectory.

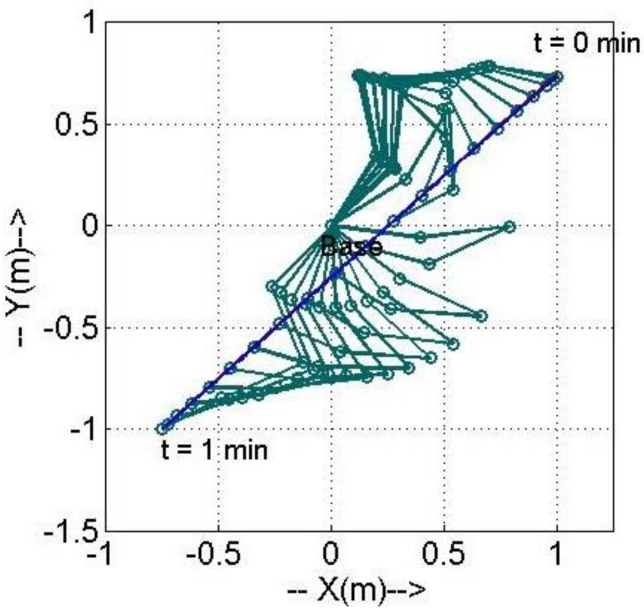

Fig. 4.36: 4-DOF manipulator motion for without task priority following inclined line trajectory.

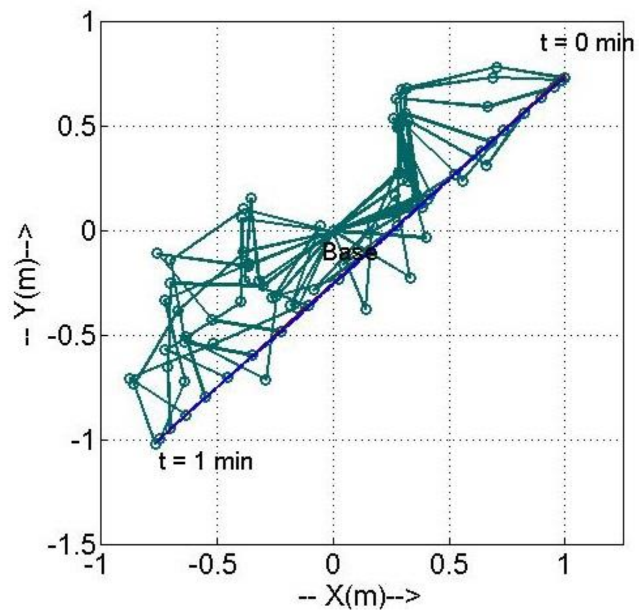

Fig. 4.37: 4-DOF manipulator with task priority at $\emptyset = 45^{\circ}$ following inclined line trajectory.

4.4 SUMMARY

This chapter has presented different case studies to show the efficacy of the concept of task priority in resolving the redundancy of serial manipulators. The redundancy resolution of the 3-link robot manipulator is applied while the manipulator is following straight line motion, rectangular trajectory, triangular trajectory, circular trajectory and inclined line trajectory. It is demonstrated that both the subtasks can be successfully completed either by increasing the overall length of the manipulator or by increasing the DOF while keeping the total length constant. In all the simulations, it is well demonstrated that it is possible to perform multiple tasks with the concept of task priority.

CHAPTER 5 CONCLUSIONS AND FUTURE DIRECTIONS

5.1 CONCLUSIONS

Various conclusions are drawn from the results that are obtained after simulating for trajectory planning along with the desired orientation of the end effector by utilizing the concept of task priority.

1. The purpose of quantifying the ability of redundant manipulators in manipulating their end effector can be solved by studying the redundancy of robot manipulators.

2. The purpose of task priority was discussed and an inverse kinematics solution was derived taking into account the order of priority, which can be regarded as an optimal solution suitable for real time redundancy control.

3. This concept of task priority was implemented for redundancy utilization of robot manipulators by dividing a task into a number of subtasks.

4. Numerical simulations are performed in order to verify the effectiveness of the solution for redundancy control problems. Results shows that more degree of redundancy implies fulfilment of more subtasks according to the given priority.

5. The appropriate criteria to fulfil the lower priority subtasks is selected according to the workspace and the order of priority.

6. It is confirmed that dividing a task into subtasks with the order of priority is a remedy for overcoming the degeneracy of degree of freedom. Hence, a number of trajectories are planned and the manipulators are aimed to trace the given trajectory along with maintaining its end effector at the desired orientations.

7. The first priority is given to follow the desired trajectory and the second priority is to maintain its end effector at a particular orientation while following the trajectory.

8. Results shows that for a 3-DOF case the second priority is not attended throughout the trajectory as both the subtasks cannot be satisfied simultaneously and the priority is given to the first manipulation variable.

9. To overcome this problem tuning of the parameters is done and it is observed that both the subtasks can be completed successfully either by increasing the overall length of the manipulators or by increasing its degree of freedom, keeping the length of robot constant.

5.2 FUTURE DIRECTIONS

1. In this thesis, the concept of task priority has been implemented on planar manipulators only. The work can be extended effectively to spatial manipulators also.

2. The work can be extended to demonstrate the effectiveness of Task Priority on a physical prototype.

3. The work can be extended to handle other subtasks like obstacle avoidance, singularity avoidance, torque optimization, minimum movement etc.

REFERENCES

[1] Y. Nakamura, Advanced Robotics: Redundancy and Optimization, Addison-Wesley Publishing company, 1991.

[2] J. J. Craig, Introduction to Robotics Mechanics and Control, 2nd ed., Pearson Education, 2003.

[3] A. Ligeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-7, no. 12, pp. 868−871, 1977.

[4] C. A. Klein and C. H. Huang, "Review of pseudoinverse control for use with kinematically redundant manipulators," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 13, no. 2, pp. 245−250, 1983.

[5] Y. Nakamura and H. Hanafusa, "Task priority based redundancy control of robot manipulators," *Proc. of International Symposium on Robotic Research*, 1984.

[6] T. Yoshikawa, "Analysis and control of robot manipulators with redundancy," *The First International Symposium on Robotic Research*, pp. 735−747, 1984.

[7] T. Yoshikawa, "Manipulability of robotic mechanisms," *International Journal of Robotics Research*, vol. 4, no. 2, pp. 3−9, 1985.

[8] T. Yoshikawa, "Dynamics manipulability of robot manipulators," *Journal of Robotic Systems*, vol. 2, no. 1, pp. 113−124, 1985.

[9] A. Maciejewski and C. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *International Journal of Robotics Research*, vol. 4, no. 3, pp. 109−117, 1985.

[10] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task priority based redundancy control of robot manipulators," *International Journal of Robotics Research*, vol. 6, no. 2, pp. 3−15, 1987.

[11] J. M. Hollerbach and K. C. Suh, "Redundancy resolution of manipulators through torque optimization," *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 308−316, 1987.

[12] R. Dubey and J. Y. S. Luh, "Redundant robot control for higher flexibility," *Proc. IEEE International Conference on Robotics and Automation*, pp. 1066−1072, 1987.

[13] J. Baillieul, "A constraint oriented approach to inverse problems for kinematically redundant manipulators," *Proc. IEEE International Conference on Robotics and Automation*, pp. 1827−1833, 1987.

[14] T. Yoshikawa, "Basic optimization methods of redundant manipulators," *Laboratory Robotics and Automation*, vol. 8, no. 1, pp. 49−60, 1996.

[15] S. Chiaverini , "Singularity-Robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Transactions on Robotics and Automation,*vol. 13,no. 3,pp. 398−410, 1997.

[16] C. Chung, B. Lee, M. Kim, and C. Lee, "Torque Optimizing Control with Singularity-Robustness for Kinematically Redundant Robots," *Journal of Intelligent and Robotic Systems*, vol. 28, no. 3, pp. 231−258, 2000.

[17] L. Zlapah and B. Nemec, "Kinematic control algorithms for on-line obstacle avoidance for redundant manipulators," *IEEE/RSJ Intl. Conference on Intelligent Robots and Systems,* vol. 17, no.3, pp. 1898−1903, 2002.

[18] X. Zhu and H. Qiao, "Obstacle avoidance for kinematically redundant manipulators using polyhedral approximations," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 217, no. 5, pp. 533−542, 2003.

[19] J. Kim, G. Marani, W. Chung, and J. Yuh, "Task reconstruction method for real-time singularity avoidance for robotic manipulators," *Advanced Robotics*, vol. 20, no. 4, pp. 453−481, 2006.

[20] A. Singla, P. Kulkarni, S. Kumar, and B. Dasgupta,"Redundancy control of robot manipulators using task Priority," *International conference on advances in control and optimization of Dynamical systems,* pp. 1−7, 2007.

[21] D. Oetomo and M.H. Ang Jr, "Singularity robust algorithm in serial manipulators," *Robotics and Computer-Integrated Manufacturing*, vol. 25, pp.122−134, 2009.

[22] S. Yahya, M. Moghavvemi and H. A. F. Mohamed, "Singularity avoidance of a six degree of freedom three dimensional redundant planar manipulator," *Computers and Mathematics with Applications,* vol. 64, pp.856−868, 2012.

[23] http://www.motoman.com/feature/7axis-bots.php#sthash.o6y9cdlw.dpbs

[24] http://www.caithness-business.co.uk/article.php?id=1258

[25] http://www.robot.mes.titech.ac.jp/hirose/robot/snake/ct_arm/photo1.left.jpg