B. TECH. PROJECT REPORT

On

Smart Mobile Manipulator using Intelligent Intervention Method

BY

Manish S.Raut

Uma Bharti Meena

Vivek Khokhar

DISCIPLINE OF MECHANICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

November 2016

Smart Mobile Manipulator Using Intelligent Intervention Method

A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degrees

of
BACHELOR OF TECHNOLOGY
in

MECHANICAL ENGINEERING

Submitted by:
Manish S.Raut(130003022)
Uma Bharti Meena(130003036)
Vivek Khokhar(130003039)

Guided by:
Dr. M. Santhakumar and Dr.D.L.Deshmukh

INDIAN INSTITUTE OF TECHNOLOGY INDORE

November 2016

CANDIDATE'S DECLARATION

We hereby declare that the project entitled **"Smart Mobile Manipulator using Intelligent Intervention Method"** submitted in partial fulfillment for the award of the degree of Bachelor of Technology in 'Mechanical Engineering' completed under the supervision of **Dr. M. Santhakumar (Assistant Professor, IIT Indore) And Dr.D.L.Deshmukh (Assistant Professor,IIT Indore)** is an authentic work.

Further, we declare that we have not submitted this work for the award of any other degree elsewhere.

Signature and name of the student(s) with date

Manish S. Raut **Uma Bharti Meena** **Vivek Khokhar**

CERTIFICATE by BTP Guide(s)

It is certified that the above statement made by the students is correct to the best of my/our knowledge.

Signature of BTP Guide(s) with dates and their designation

Dr.M.Santhakumar
Asst. Prof. Mechanical Engg.
IIT Indore

Dr.D.L. Deshmukh
Asst. Prof. Mechanical Engg.
IIT Indore

Preface

This report on "Smart Mobile Manipulator using Intelligent Intervention Method" is prepared under the guidance of Dr. M.Santhakumar and Dr.D.L.Deshmukh

Through this report we have tried to give a detailed design of a smart mobile manipulator suitable for its use in industry for cleanliness purposes.We have covered every aspect of the proposed design and its mathematical modelling.We fabricated the prototype and performed a task successfully which shows that the design is technically sound and feasible, and can have numerous applications in industry.

We have tried to the best of our abilities and knowledge to explain the content in a pellucid manner. We have also added 3-D CAD models and figures to make it more illustrative.Equations of motion and Simulation results are also given in the respective section.

Manish S.Raut
(130003022)
B.Tech. IV Year
Mechanical Engineering

Uma Bharti Meena
(130003036)
B.Tech. IV Year
Mechanical Engineering

Vivek Khokhar
(130003039)
B.Tech. IV Year
Mechanical Engineering

Acknowledgements

We wish to thank our guide,**Dr. M. Santhakumar and** co-guide, **Dr.Devendra Deshmukh** for their kind support and valuable guidance and giving us the opportunity to work for the B.Tech project under their supervision.We owe profound gratitude to them who took keen interest in our project work and we are extremely fortunate to have their guidance. It is their vital encouragement and support, due to which we became able to complete the design and technical report

We would like to thank Discipline of Mechanical Engineering and in total IIT Indore for providing us the necessary equipment and funds for the fabrication process.We express thanks to **Mr A.Petare, Mr.Satish** and entire **Workshop staff** for helping us in prototyping. We would like to extend our sincere regards to **Mr. Jayant Mohanta, Mr. Yogesh Singh, Mr.M.Muralidharan, Mrs.Swati Mishra** (CRC PhD Scholars) and **Mr.Hemant** for helping in the lab activities. We are thankful to get constant encouragement, support and guidance from B.Tech Project evaluation committee. It is their help and support, due to which we are able to complete the project.

Manish S.Raut
Uma Bharti Meena
Vivek Khokhar
B.Tech. IV Year
Discipline of Mechanical Engineering
IIT Indore

Abstract

This Report dealt with design and development of a 10 DoF smart mobile manipulator mainly designed for monitoring and cleaning purposes in structured as well as unstructured environment . Major focus had been laid on design of an efficient manipulator capable of doing the desired task intelligently in coordination with the mobile base which is well equipped with sensors in order to do a task smartly and autonomously without need of any external intervention. Firstly,a suitable configuration has been chosen based on the kinematics, dynamic analysis,workspace analysis of two possible configurations.All the analysis plainly justify the selection of the manipulator configuration.A conceptual design of the proposed mobile manipulator featuring a camera for visual input and a vacuum cleaner has been given.The Components such as joint actuators,mobile base motors, chosen based on the dynamic analysis results,are also described herein.The camera is being used via Image Processing in order to detect waste material to be disposed off. The limited sequence control algorithm of the manipulator validates the accurateness of modelling and simulation as well as the physical prototyping and its controlling.The system can be customized easily to be used in other areas like ware house automation, mining, services, rescue, etc.The paper ends with conclusion and discussion on the future scopes of development of the smart mobile manipulator.

Table of Contents

1.INTRODUCTION

Mobile manipulator is a robotic system which has manipulator (or arm) mounted on a mobile base. There are multifarious manipulators available in market. For instance, Tiago by Pal Robotics, MM-500 by Neobotix, MMO-500 by Neobotix, RB-1 by Robotnik, etc.Their colossal use is found in warehouse automation, mining, construction, forestry, military, etc. and can be put into use in areas perilous for humans as in atomic power plants, fertilizer plants, space exploration, health-care, and home-care.The mobile manipulator can be used in structured as well as unstructured environment.

1.1 Mobile Base

The mobile base provides flexibility to the system and an unlimited workspace on a surface. Out of the multitudinous mobile base available in the market, few are listed below:

1. Baxter Mobile Base

   A mobile base with the use of omnidirectional wheels for agility, advanced interface to control the base and to integrate it with the system, and an expandable modular design (Fig.1)

2. Ridgeback

   An omnidirectional development platform with precision control for forward, lateral or twisting movements, and easy to integrate with a variety of manipulators and sensors (Fig.2)

3. Otto

   A self-driving warehouse robot that can transport 3300 lb loads at speeds up to 4.5 mph ,while tracking along optimal path and avoiding any collision.(Fig.3)

4. PMB-2

   A mobile base with compact design for navigation ,and fully compatible with ROS(Fig.4)

Fig.1

Fig.2

Fig.3

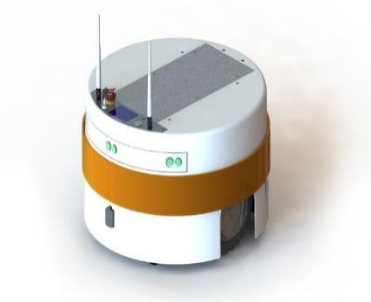


Fig.4

1.2 Manipulator

A manipulator is typically composed of links, joints, and end-effector. The links are the rigid bodies connected by joints and thus, forming the mechanism. At the free end of the chain of the links that make up the manipulator is the end-effector that interacts with its environment to perform tasks. The manipulator provides dexterity to the robotic system. It has relatively simple design and it can be readily adapted to given purposes. It has been built for various applications, such as remote handling, machine tools, medical robots, simulators, micro-robots, and humanoid robots. It can be classified on according to various criteria such as degrees of freedom, workspace geometry, motion characteristics, and control.

1.2.1 Degrees of Freedom

Ideally, a manipulator should possess at least 6 Degrees of freedom in order to reach a particular point in space. Based on this, there are three types of manipulators:

1 General-Purpose Manipulator

A manipulator having six degrees of freedom (Fig.5)

1. Redundant Manipulator

A manipulator having more than six degrees of freedom (Fig.6)

2. Deficient Manipulator

A manipulator having less than six degrees of freedom (Fig.7)

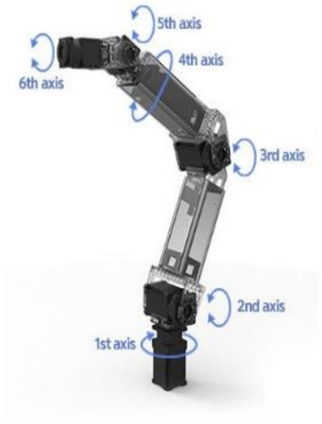


Fig.5

Fig.6

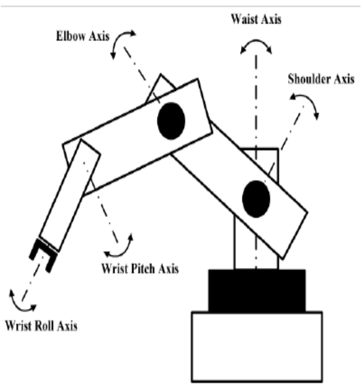


Fig.7

1.2.2 Kinematic Structure

According to its structural topologies, a kinematic structure can be classified into three categories :

1. Serial Manipulator: A manipulator that has only one kinematic chain connects the fixed base to the end-effector (Fig.8)
2. Parallel Manipulator: A manipulator that has more than one kinematic chain connects the fixed base to the end-effector (Fig.9)
3. Hybrid Manipulator

   A manipulator which consists of both open and closed-loop chains. (Fig.10)

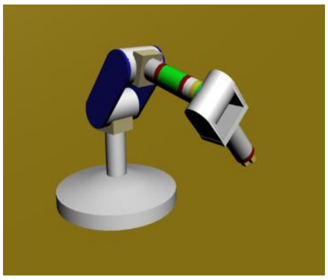


Fig.8

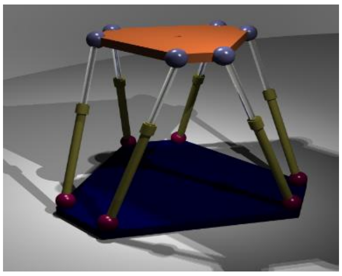


Fig.9

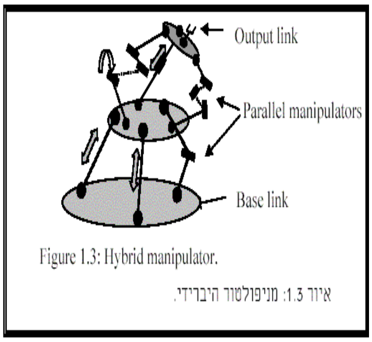


Fig.10

1.2.3 Workspace Geometry

Workspace is that volume of space that the end-effector of manipulator can reach. For a solution to exist, the specified goal point must lie within the workspace. Sometimes, it is useful to consider two definitions of workspace: Reachable workspace is that volume of space that the robot can reach in atleast one orientation. Dexterous workspace is that volume of space that the robot end-effector can reach with all orientations. Clearly, the Dexterous workspace is a subset of reachable workspace.The classification based on the Workspace geometry is shown in the below table.

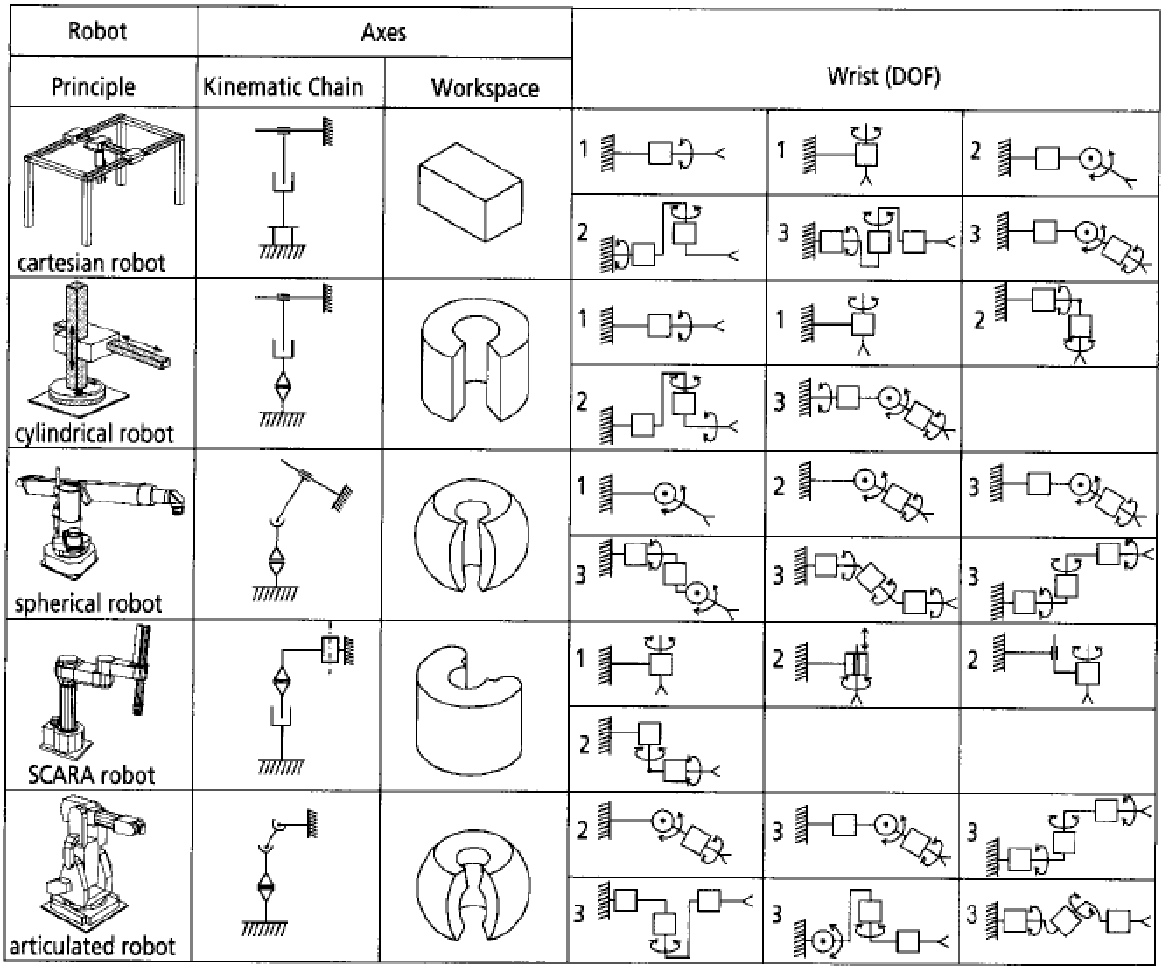


Table 1: Manipulator Clsssification based on Workspace Geometry

The manipulators are classified kinematically on the basis of first three joints of arm as the remaining joints used for the wrist are for controlling the orientation. Therefore, the manipulators can be of following five types:

1. Cartesian (PPP or three prismatic joints)
2. Cylindrical (RPP or one rotary and two prismatic joints)
3. Spherical (RRP or two rotary and one prismatic joint)
4. SCARA (Selective compliance assembly robot arm or RRP)
5. Articulated (RRR)

1.2.4 Motion Characteristics

Based on their nature of motion, mobile manipulators can be classified into three types:

1. Planar Manipulator

   A manipulator in which all the moving links move in planes parallel to one another

2. Spherical Manipulator

   A manipulator in which all the links perform spherical motions about a common stationary point.

3. Spatial Manipulator

   A manipulator in which atleast one of the moving links possesses a general spatial motion.

1.3 Different types of Industrial Manipulators

In industries many types of industrial manipulators are used according to their requirements. Some of them are listed below.

1.3.1 Cartesian coordinate robot:

In this industrial robot, its 3 principle axis have prismatic joints or they move linear thorough each other. Cartesian robots are best suited for dispensing adhesive like in automotive industries. The primary advantage of Cartesians is that they are capable of moving in multiple linear directions. And also they are able to do straight-line insertions and are easy to program. The disadvantages of Cartesian robot are that it takes too much space as most of the space in this robot is unused.

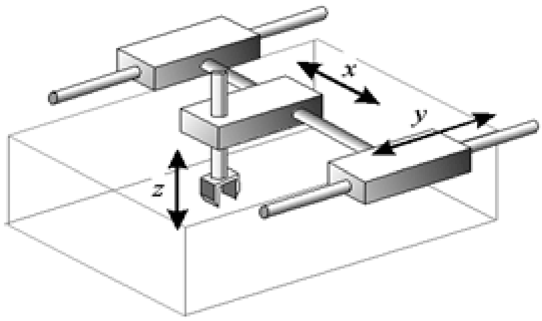


Fig.11

1.3.2 SCARA Robot:

The SCARA acronym stands for Selective Compliance Assembly Robot Arm or Selective Compliance Articulated Robot Arm. SCARA robots have motions similar to that of a human arm. These machines comprise both a 'shoulder' and 'elbow' joint along with a 'wrist' axis and vertical motion. SCARA robots have 2 revolute joints and 1 prismatic joint. SCARA robots have limited movements but it is also its advantage as it can move faster than other 6 axis robots. It is also very rigid and durable. They are mostly used in purpose application which require fast, repeatable and articulate point to point movements such as palletizing, DE palletizing, machine loading/unloading and assembly. Its disadvantages are that it has limited movements and it is not very flexible.

Fig.12

1.3.3 Cylindrical robot:

It is basically a robot arm that moves around a cylinder shaped pole. A cylindrical robotic system has three axes of motion – the circular motion axis and the two linear axes in the horizontal and vertical movement of the arm. So it has 1 revolute joint, 1 cylindrical and 1 prismatic joint. Today Cylindrical Robot are less used and are replaced by more flexible and fast robots but it has a very important place in history as it was used for grappling and holding tasks much before six axis robots were developed. Its advantage is that it can move much faster than Cartesian robot if two points have same radius. Its disadvantage is that it requires effort to transform from Cartesian coordinate system to cylindrical coordinate system.

Fig.13

1.3.4 PUMA Robot:

The PUMA (Programmable Universal Machine for Assembly, or Programmable Universal Manipulation Arm) is the most commonly used industrial robot in assembly, welding operations and university laboratories. It is more similar to human arm than SCARA robot. It has great flexibility more than SCARA but it also reduces its precision. So they are used in less precision work like assembling, welding and object handling. It has 3 revolute joints but not all the joints are parallel, second joint from the base is orthogonal to the other joints. This makes PUMA to compliant in all three axis X, Y and Z. Its disadvantage is its less precision so it can't be used in critical and high precision needed applications.

Fig.14

1.3.5 Polar Robots:

It is sometimes regarded as Spherical robots. These are stationary robot arms with spherical or near-spherical work envelopes that can be positioned in a polar coordinate system. They are more sophisticated than Cartesian and SCARA robots but its control solution are much less complicated. It has 2 revolute joints and 1 prismatic joint to make near spherical workspace. Its main uses are in handling operations in production line and pick and place robot.

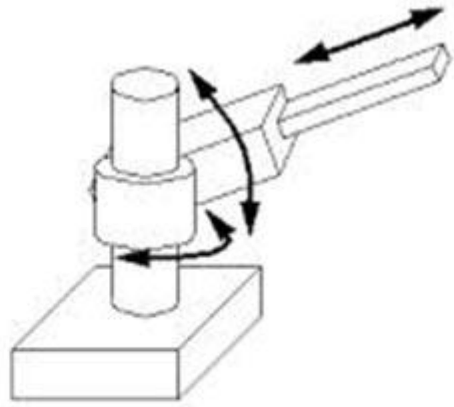


Fig. 15

1.4 Proposed Mobile Manipulator System

Till now, we have seen that manipulators as well as mobile bases are already available in the market which are able to perform certain kind of tasks. Using the manipulators for the purpose of area monitoring and cleaning it off the solid waste objects as well as small dust particles,is a now-a-days a hot topic of designing.But the combination of a manipulator and a mobile base is a very rarely explored area in robotics as it becomes quite complicated to control and model mathematically when both are combined than when done individually and in combining both it becomes expensive for an application purpose such as cleaning operations.But when a manipulator is attached to a mobile base,it greatly enhances the capability and applicability of the manipulator.Hence,we have attempted here to design a mobile manipulator with optimal configuration and optimal cost ,which will have artificial intelligence so that it can be used for cleaning operations indoor as well as outdoor and work in collaboration with similar devices using IoT.

The propounded design for 10 DoF mobile manipulator involves a mobile Omni-base providing 3DoF, a 4DoF manipulator and a 3DoF end-effector joint. The mobile base consists of 4 Omni-wheels each attached to the motor shaft by couplings. The Omni-wheels are wheels with small discs around circumference. These discs are perpendicular to the turning direction, hence, allows the motion in lateral as well as transverse direction accounting for 2DoF of the base and rotational motion accounts for the 3^{rd}DoF of the base.

The possible configurations for the manipulator are RPRR and RRRR since revolute joint(R) and the prismatic joint(P) are the single degree of freedom joints which are easiest to control.More number of revolute joints than that of the prismatic joint makes the system more simple and stable.So,in order to choose between the two we need to do some comparison based on the mathematical modeling and its simulations.Hence Based on the dynamic force analysis in ADAMS, static force analysis and workspace analysis in MATLAB, the best configuration out of the two is chosen.The spherical joint at the end of third link provides 3DoF.

1.4.1 Novelties

The customized design of 10 DoF mobile manipulator is suitable for indoor as well as outdoor cleaning. The robotic system has an additional feature of vacuum cleaning along with pick and place feature. The vacuum cleaning can be used to clean the objects which aren’t of comparable size and cannot be picked up by manipulator. As of now, no robotic systems are there which provides this feature along with manipulation. Hence, the proposed design is innovative in terms of functionalities.

- It is a multipurpose robot which has applications in future research, specially, in IoT. Using IoT, various such robots can collaborate among themselves to decide that robot in close proximity to the desired location will perform the intended task.
- Visual sensing is another feature that will enable operations in an unstructured environment.
- It can also be used for wet detritus.

1.4.2 Applications

- The 10DoF system can be used for uninterrupted round the clock cleanliness monitoring. Employment of human labour in such areas will be uneconomical.
- It can be solution for areas where humans are incapable of rendering the desired services, for example ,sensitive and important parts of institutes , space exploration, military operations, home-care and health-care,etc.

2. SELECTION OF MANIPULATOR CONFIGURATION

Manipulator is an important part in task handling of robotics system. Here, for the purpose of picking and placing two configurations have been analyzed. Selection of proper configuration gives suitable workspace, efficient energy consumption and higher dexterity. In the present case, analysis of RRRR and RPRR configuration has been done.

The robot consists of mobile base 3DOF and a manipulator with 4 DOF. Besides this, at the end-effector, there is a 3DOF arm. Overall the robot is a 10DOF system. Therefore among the possible configurations for this particular application, two 4 DOF manipulators (RPRR and RRRR) have been chosen for analysis and one out of the two is selected on the basis of dynamic analysis done in ADAMS, workspace and static force analysis done in MATLAB.

2.1 Comparison Based On Dynamic Analysis:

1. RRRR Configuration and Its Dynamic Analysis

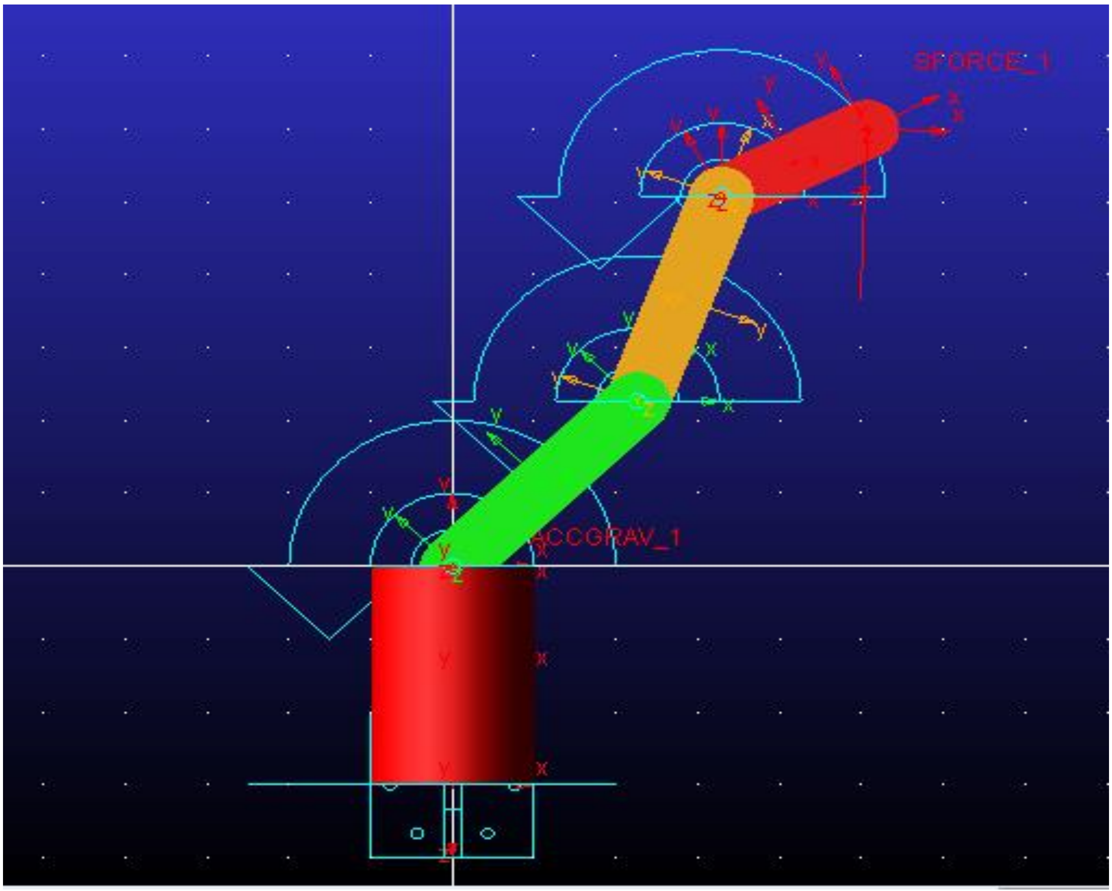


Fig. 16 RRRR manipulator in ADAMS environment

Dynamic Analysis involves the calculation of the forces and torques of joint space when the system is in motion. Since the robot will be in motion for the most of the time so the dynamic analysis becomes indispensable. Also, for finding the actuator torque this analysis is imperative.

There are two problems related to the dynamics of a manipulator that we wish to solve. In the first problem, we are given a trajectory point, e, and ë, and we wish to find the required vector of joint torques, r. This formulation of dynamics is useful for the problem of controlling the manipulator. The second problem is to calculate how the mechanism will move under application of a set of joint torques. That is, given a torque vector, r, calculate the resulting motion of the manipulator, g, é, and 0. This is useful for simulating the manipulator.Since we are focusing on controlling the manipulator for time being,we'll do only the first type of dynamic analysis.

It has 4 rotary joints and nowadays it is widely being used in industry. It also provide large workspace to size ratio .For the purpose of simulation, lengths chosen for various links are

link 1 = 160mm, link 2 = 150mm and link 3 = 130mm. Payload is same for both configuration, i.e., 1Kg. Since it is the proposed maximum load that our end-effector is intended to pick.These are the condition during simulation, joints velocity is 10 RPM and payload is 1Kg.The model is simulated with above parameters for 8 seconds and the graphical results are obtained

<u>Graphical Output Using ADAMS</u>

Fig.17 Shoulder Motor Torque

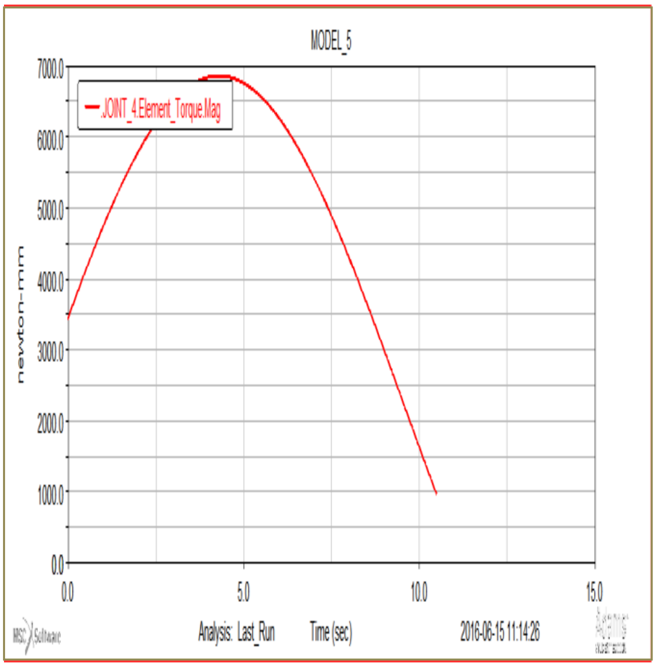


Fig.18 Link3 Motor Torque

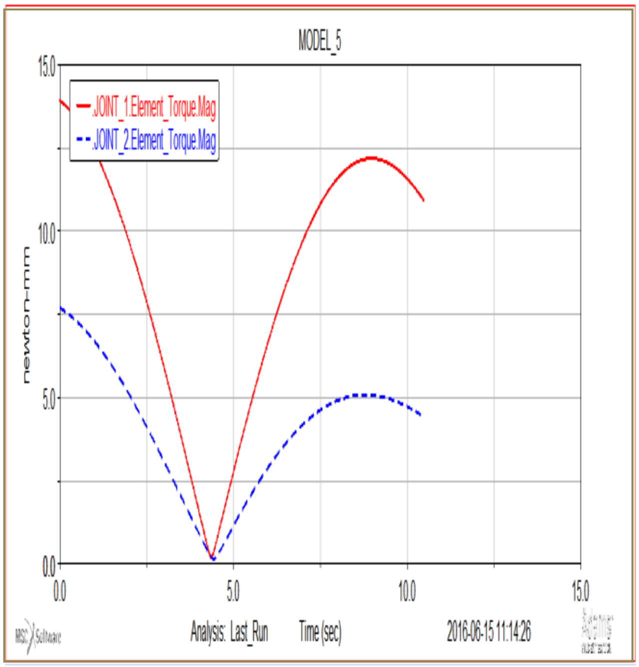


Fig.19 Link 2 Motor Torque

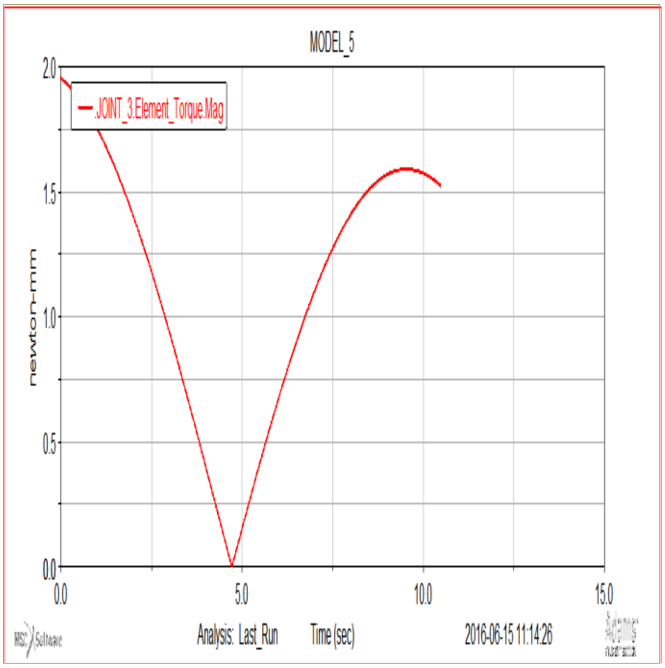
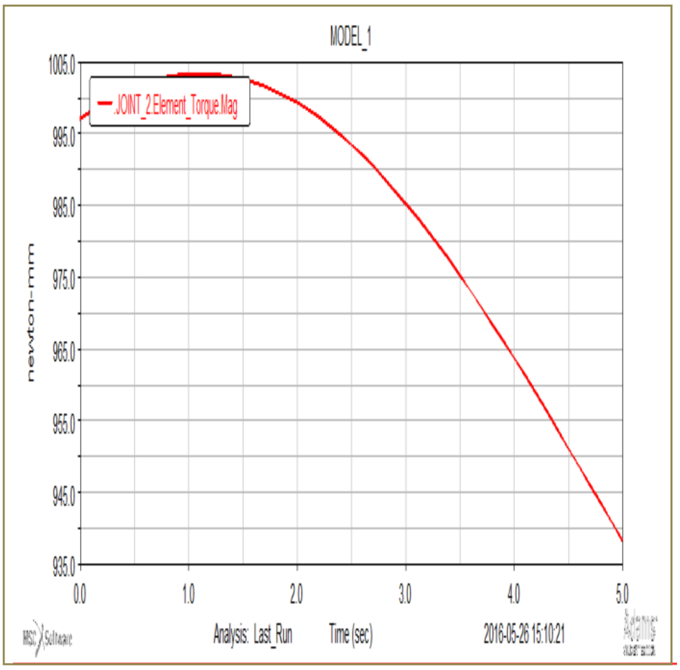


Fig.20 Link1 Motor Torque



2. RPRR Configuration and Its Dynamic Analysis

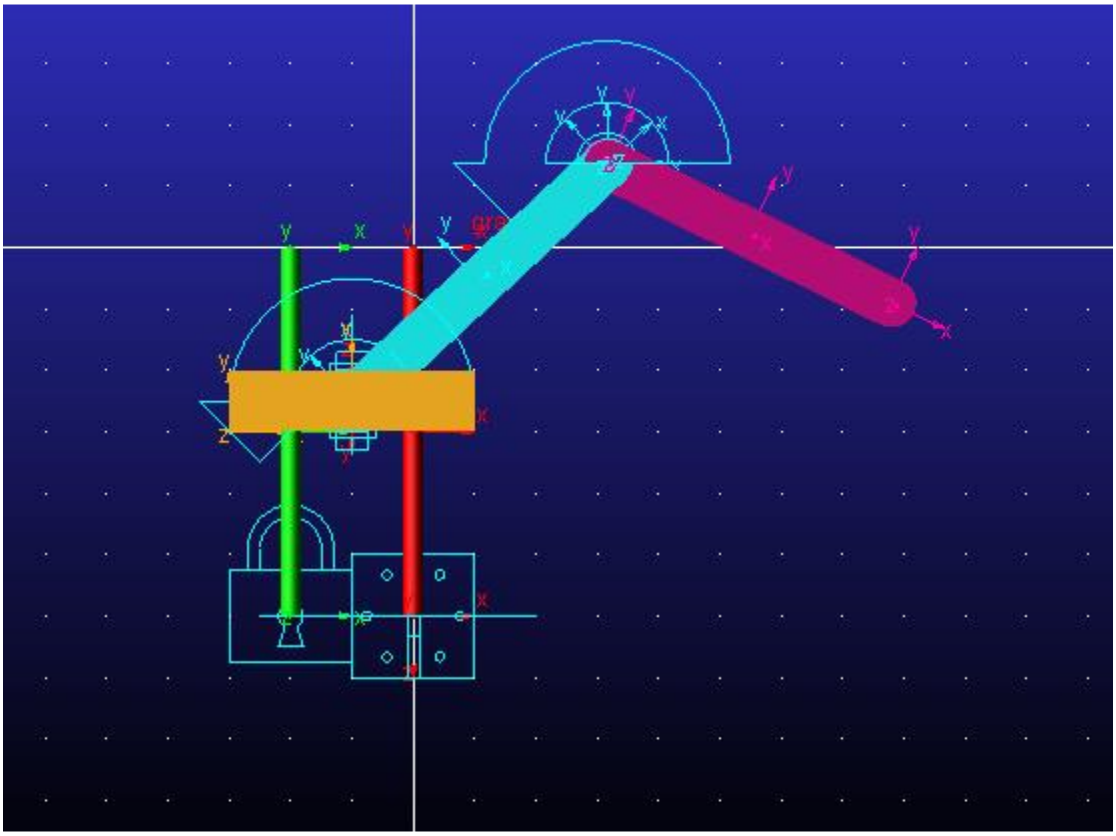


Fig. 21 RPRR manipulator in ADAMS environment

It has 3 rotary joint and one prismatic joint .it Provides better platform for camera and other instrumentations here we can put camera over slider. Manipulator has easy manoeuvre for up and down motion by just moving slider and without changing the configuration. We know the orientation of all links and from that we can find the position at which the end effector will reach so we used Direct kinematics .it provide large singularity free workspace. There are some point at which the end-effector cannot reach those points will be include in singularity.For the purpose of simulation, various link lengths chosen are

link 1 = 150mm, link 2 = 130mm, lead screw =160mm and lead screw diameter=8mm.payload=1Kg.

These are the condition during simulation, joints velocity is 10 RPM and payload is 1Kg.The model with above parameters is simulated for 9 seconds and the graphical results are obtained.Linear velocity of prismatic joint is taken as 10 mm/sec

Graphical Output Using ADAMS

Fig.22 Shouder Motor Torque

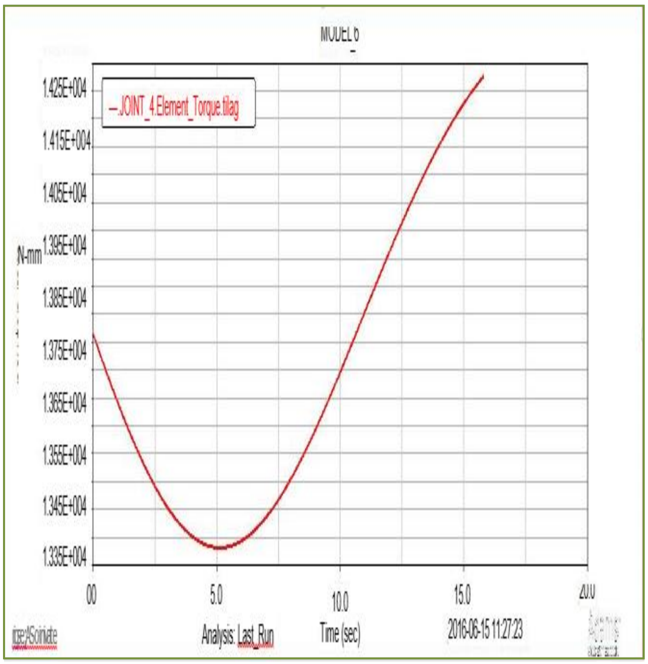


Fig.23 Link1 Motor Torque

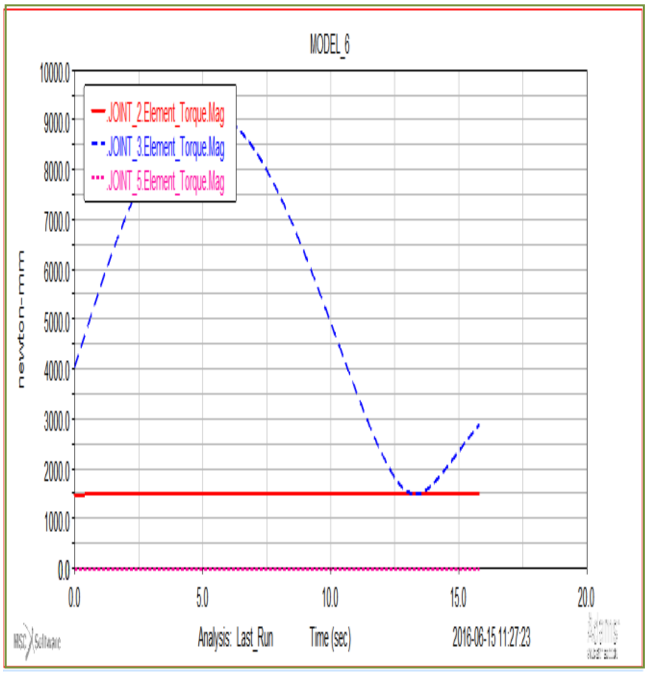


Fig.24 Link3 Motor Torque

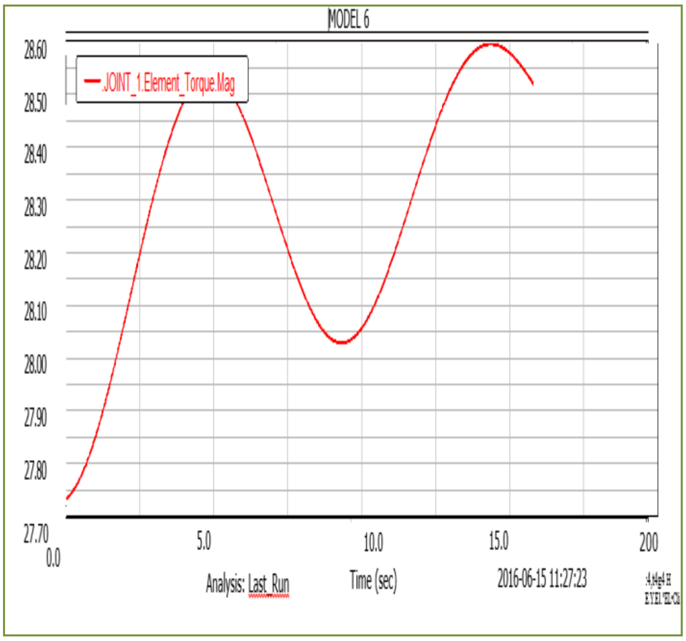
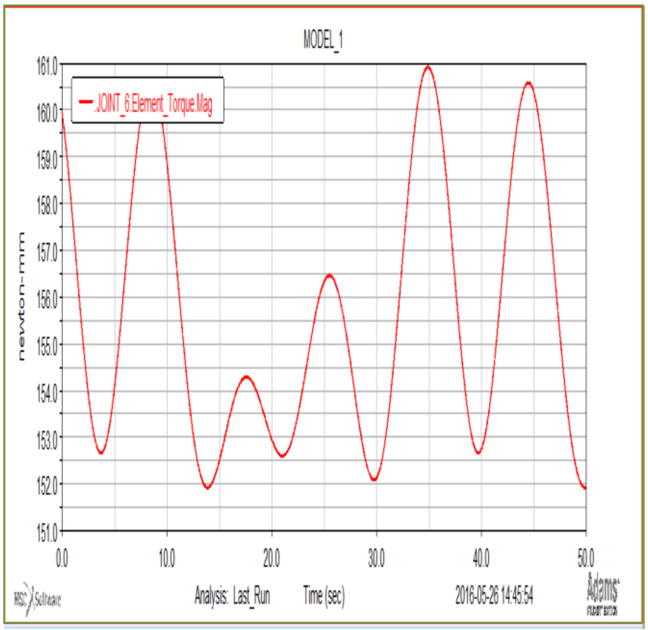


Fig.25 Link2 Motor Torque



Based on the analysis,the Maximum torque for shoulder motor of RPRR configuration is almost double that of RRRR configuration and overall,the joint torques in RPRR configuration are more than the torques in RRRR configuration. Hence, the RRRR turns out to be a better configuration over RPRR based on the dynamic analysis.

2.2 Comparison based on Static Force Analysis in MATLAB

Static force analysis is basically finding out the joint torques at the respective joints by applying some desired load at the end effector such that the whole system is in static equilibrium. This is done to find out the maximum load the manipulator can handle where all the actuators will be locked as if it's in static equilibrium. So we'll be able to find the stall torques of the motor.In our case we have the maximum payload of 1 Kg and the respective stall torques in N-m required for the two configuration are given below. The force vector applied at the end-effector is

F5= [-7.0711 ; 0 ; 7.0711].

	RPRR	RRRR
T4	1.6971	1.3
T3	4.5971	2.36
T2	7.07	3.96
T1	0	0

Table 2: Comparison based on Static Force Analysis

It's clearly visible that the stall torque requirements at the joints for the RPRR configuration are higher than that of RRRR configuration. So RRRR configuration comes out to be most efficient based on Static force analysis.

2.3 Comparison based on Workspace Analysis

Workspace is that volume of space that the end-effector of the manipulator can reach. For the end effector to be able to span the desired volume space, the specified goal points must lie within the workspace.So workspace is one of the most important parameter to compare.The more workspace associated is the better configuration it is.The position of the end-effector is a function of joint angles of the manipulator which can be represented using kinematic equations. Finding out the Task space using the joint space is called as forward kinematics.There are various methods to solve the forward kinematics, for our configurations DH parameter method is convenient.

The ***Denavit–Hartenberg parameters*** (also called ***DH parameters***) are the four parameters namely $a_{i-1}, \alpha_{i-1}, d_{i,} \theta_i$ associated with a particular convention for attaching reference frames to the links of a spatial kinematic chain.Jacques Denavit and Richard Hartenberg introduced this convention in 1955 in order to standardize the coordinate frames for spatial linkages.

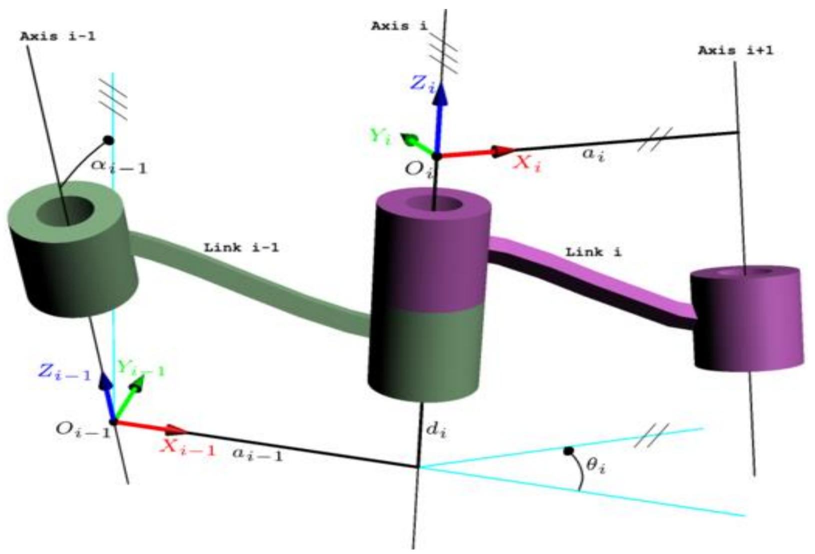


Fig26.

Assigning the frames to the links and the four link parameters in DH parameterization method

Table 3

i	α_{i-1}	a_{i-1}	θ_i	d_i
1	0	0	θ_1	0
2	90	0	θ_2	0
3	0	l_1	θ_3	0
4	0	l_2	θ_4	0
5	90	0	0	l_3

DH parameters for the RRRR configuration

Table 4

i	α_{i-1}	a_{i-1}	θ_i	d_i
1	0	0	θ_1	0
2	0	0	0	d_2
3	90	0	θ_3	0
4	0	l_1	θ_4	0
5	90	0	0	l_2

DH parameter for the RPRR configuration

Above are the DH parameters found for the two configurations. The offsets are assumed to be null otherwise.Now, we derive the general form of the transformation that relates the frames attached to neighboring links. These individual transformations are then concatenated to solve for the position and orientation of link n relative to link 0.If we wish to write the transformation that transforms vectors defined in {i} to their description in {i-1}, we may write

$$^{i-1}P = {}^{i-1}_{i}T\,{}^{i}P$$

And $${}^{i-1}_{i}T = Screw_x(a_{i-1}, \alpha_{i-1})Screw_z(d_i, \theta_i)$$

Where $Screw_Q(r, \phi)$ stands for the combination of a translation along an axis $\hat{Q}$ by a distance r and a rotation about the same axis by an angle ϕ, then the general form of ${}^{i-1}_{i}T$ is obtained:

$${}^{i-1}_{i}T = \begin{pmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i\cos\alpha_{i-1} & \cos\theta_i\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1}d_i \\ \sin\theta_i\sin\alpha_{i-1} & \cos\theta_i\sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \_\_\_\_\_(1)$$

Transformation matrices and forward kinematic equation for RRRR configuration:

$${}^{0}_{1}T = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$ {}^{1}_{2}T = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{2}_{3}T = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & l_1 \\ \sin\theta_3 & \cos\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{3}_{4}T = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & l_2 \\ -\sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{4}_{5}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -l_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Now, in order to find the position and the orientation of the last link we have to relate frame {5} to frame{0} ,which can be found by the transformation ${}^{0}_{5}T$

$$ {}^{0}_{5}T = {}^{0}_{1}T \; {}^{1}_{2}T \; {}^{2}_{3}T \; {}^{3}_{4}T \; {}^{4}_{5}T $$

Hence,

$$ {}^{0}_{5}T = \begin{bmatrix} c(\theta_2+\theta_3+\theta_4)c\theta_1 & s\theta_1 & s(\theta_2+\theta_3+\theta_4)c\theta_1 & c\theta_1\left(l_2c(\theta_2+\theta_3)+l_1c\theta_2+l_3s(\theta_2+\theta_3+\theta_4)\right) \\ c(\theta_2+\theta_3+\theta_4)s\theta_1 & -c\theta_1 & s(\theta_2+\theta_3+\theta_4)s\theta_1 & s\theta_1\left(l_2c(\theta_2+\theta_3)+l_1c\theta_2+l_3s(\theta_2+\theta_3+\theta_4)\right) \\ s(\theta_2+\theta_3+\theta_4) & 0 & -c(\theta_2+\theta_3+\theta_4) & l_2s(\theta_2+\theta_3)+l_1s\theta_2-l_3c(\theta_2+\theta_3+\theta_4) \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Hence from the above transformation matrix we get the end effector position of the manipulator .i.e.

$$ \begin{aligned} P_x &= c\theta_1\left(l_2c(\theta_2+\theta_3)+l_1c\theta_2+l_3s(\theta_2+\theta_3+\theta_4)\right) \\ P_y &= s\theta_1\left(l_2c(\theta_2+\theta_3)+l_1c\theta_2+l_3s(\theta_2+\theta_3+\theta_4)\right) \\ P_z &= l_2s(\theta_2+\theta_3)+l_1s\theta_2-l_3c(\theta_2+\theta_3+\theta_4) \end{aligned} $$

Forward kinematic equation of RRRR

where $\cos\theta_i, \sin\theta_i$ *is replaced with* $c\theta_i,\ s\theta_i$ *respectively*

Similarly, using the DH table of the RPRR configuration we can find its forward kinematic equations. we obtained.

$$
\begin{aligned}
P_x &= c\theta_1\left(l_2 s(\theta_3+\theta_4)+l_1 c\theta_3\right)\\
P_y &= s\theta_1\left(l_2 s(\theta_3+\theta_4)+l_1 c\theta_3\right)\\
P_z &= d_2 - l_2 c(\theta_3+\theta_4)+l_1 s\theta_3
\end{aligned}
$$

Forward kinematic equation of RPRR

The above kinematic equation gives the end-effector position in terms of the joint angles.

Once we found the forward kinematics, we can easily plot the set of data points for the above equations using MATLAB which will give us the Workspaces for respective configurations. The workspace comparison of both configurations is given below.

For RRRR configuration,

$l_1 = .16m,\ l_2 = .15m,\ l_3 = .13m,\ \theta_1$ varies from 0° to 360°, θ_2 varies from 0° to 160°, θ_3 varies from 0° to 200°, θ_4 varies from 0° to 200°.

For RPRR configuration,

$l_1 = .15m,\ l_2 = .13m,\ \theta_1$ varies from 0° to 360°, d_2 varies from 0 to $0.16m$, θ_3 varies from 0° to 200°, θ_4 varies from 0° to 200°.

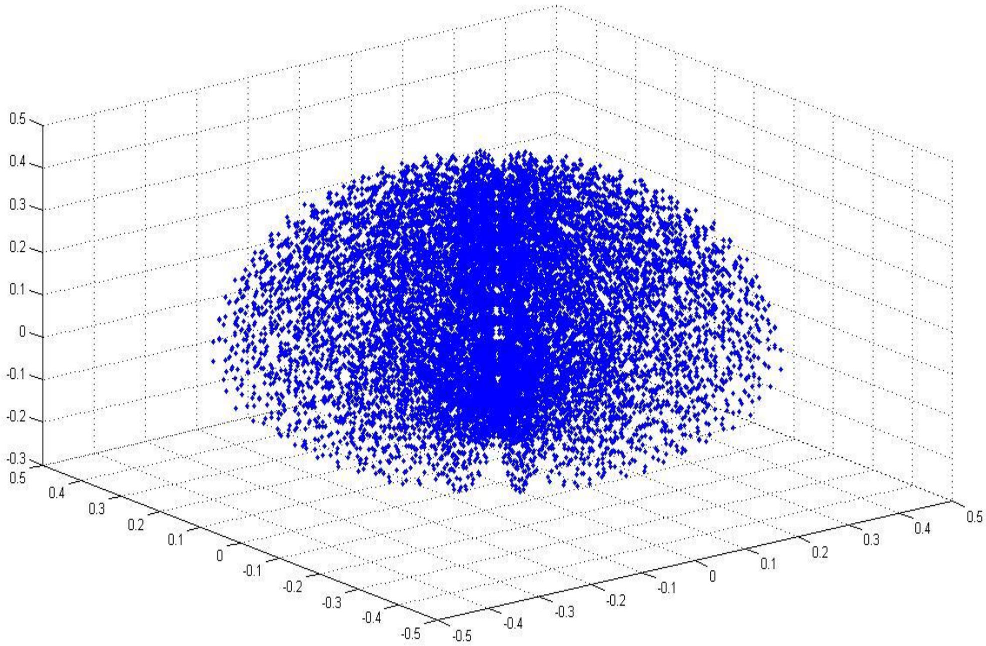


Fig.27 3D view of RRRR Configuration Workspace Workspace

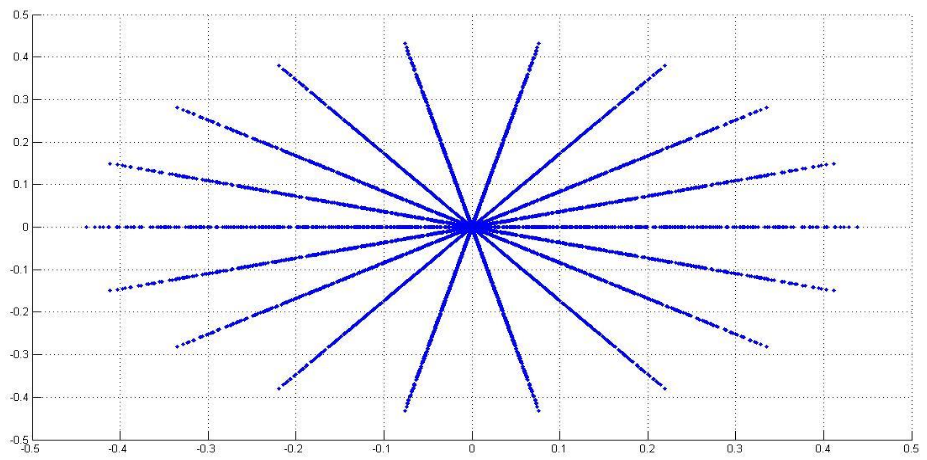


Fig.28 Top view of RRRR Configuration Workspace

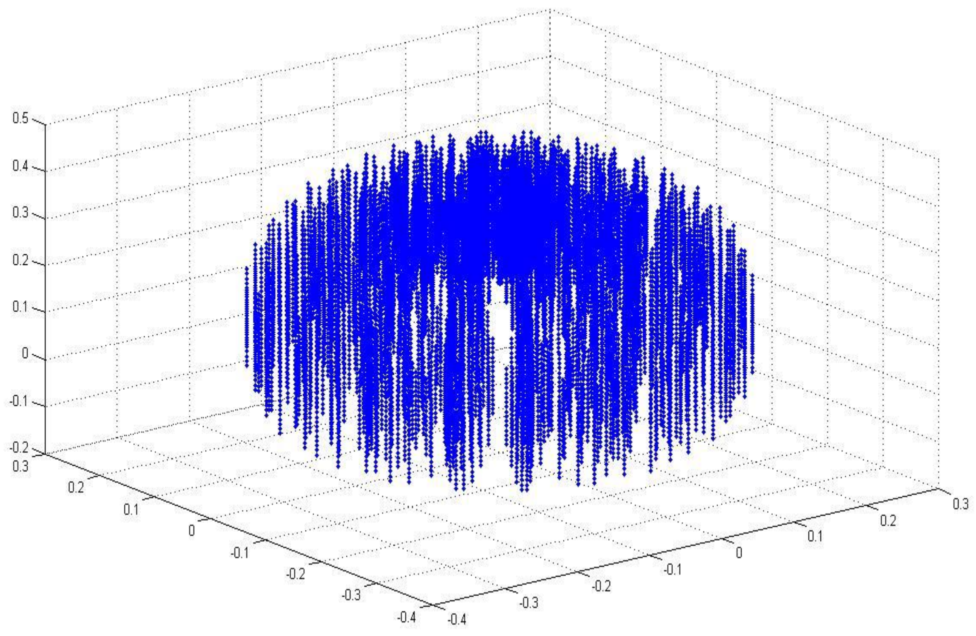


Fig.29 3D view of RPRR Configuration Workspace

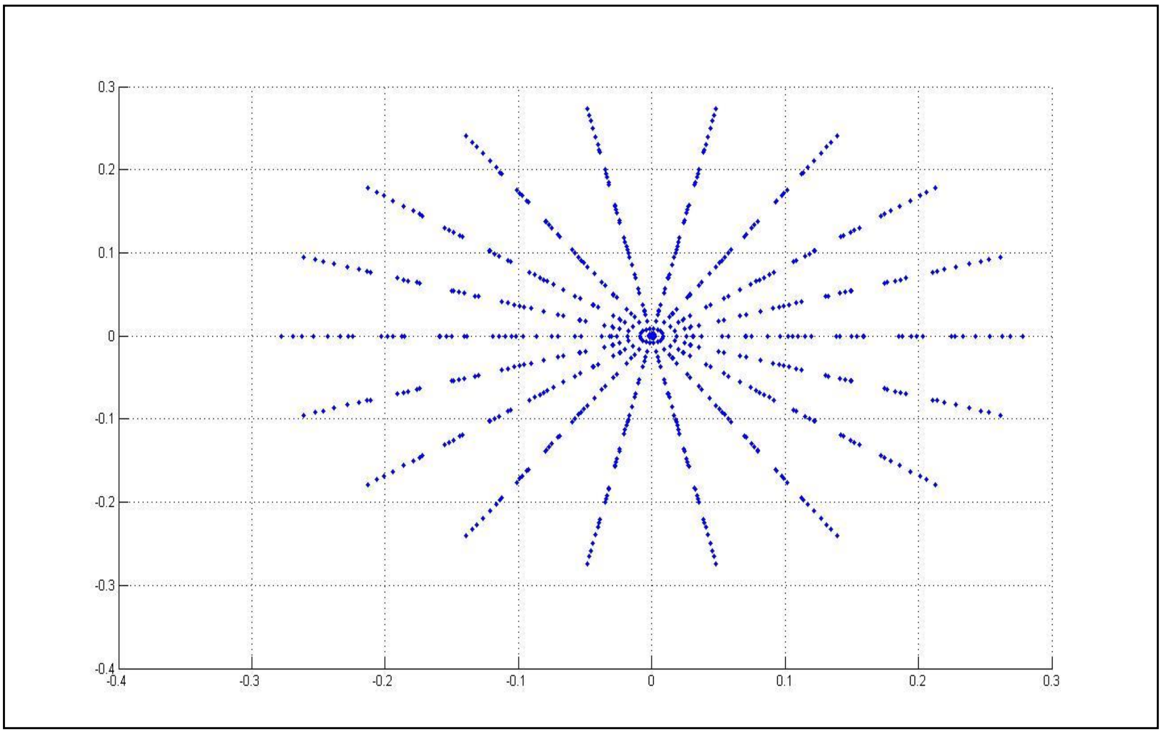


Fig.30 Top view of RPRR Configuration Workspace

The total volume swept by the end-effector of :

RRRR configuration is 0.3567 m^3

RPRR configuration is 0.1313 m^3

Since the RRRR Configuration has larger workspace, it can be inferred from above three analysis that the **RRRR configuration** is best suitable for our manipulator.

3. SYSTEM DESCRIPTION

3.1 Conceptual Design

Concept design is needed to make informed decisions. The greatest obstacle in pursuing this goal is that the conceptual design phase rarely has enough information available to conduct a rigorous analysis. However, there are a number of techniques available that are commonly used in industry to assist in design decision-making. Many of these techniques are also useful during the conceptual design phase.

The first task during conceptual design is to gain a better understanding of the problem. Create a list of actions the robot takes, with a corresponding amount of time each of those actions take. Include details like interacting with human beings and obstacles, and how long those interactions may take.

The values for a representative, well-estimated size and weight of the robot will be required for the conceptual design. This means that before knowing exactly what components have to used, make well-informed guesses as to which components might used and how much they weigh, use those guesses to drive design choices, and then eventually compare the final weight to the guessed weight to see if there is a need to re-select any components.

There will also be need to visualize how large the robot can be when it comes to selecting the components. Create a 3D representation of the estimated design of the robot using softwares like Adams, Solidworks, Catia, etc .

In the present case, a conceptual design was made in Solidworks. Various components like base, omniwheel, links, vacuum cleaner, joints, etc. are made and integrated to make the robot assembly. The various features and commands available in Solidworks for designing are:

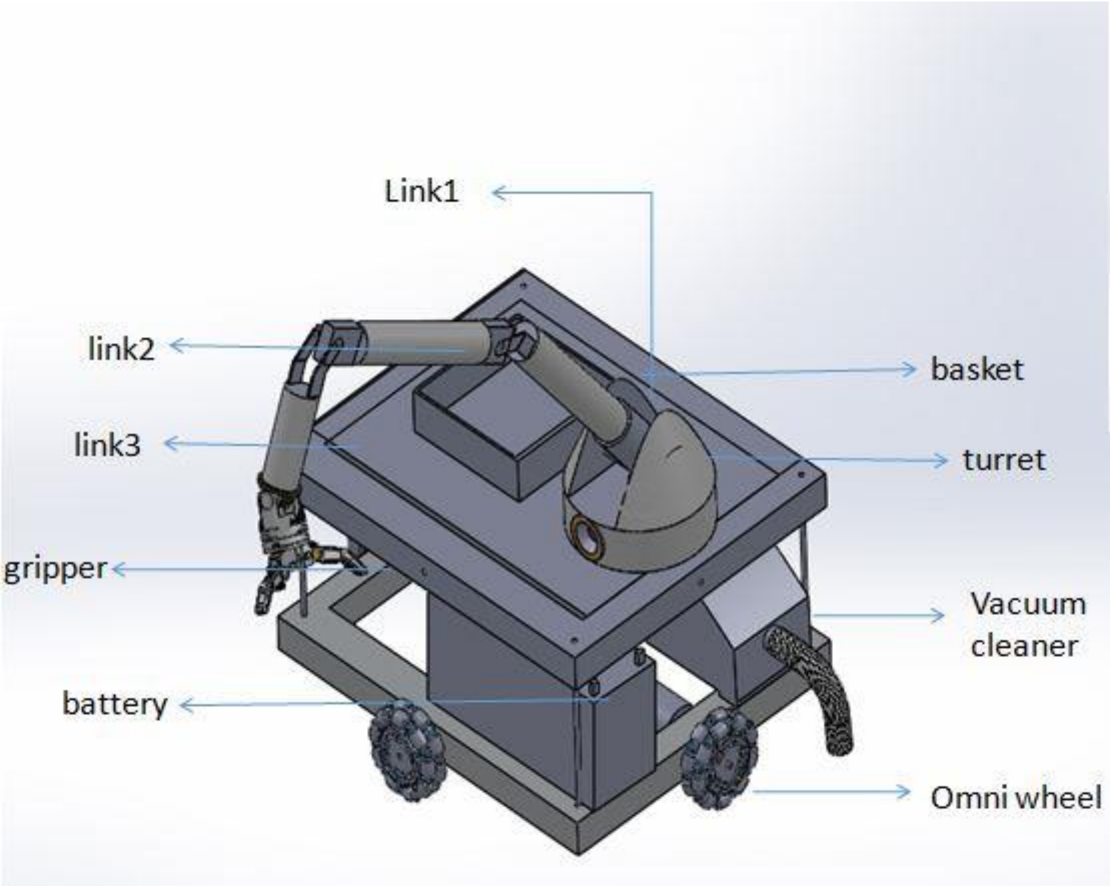


Fig.31 Isometric View of CAD model

Fig.32 Front Veiw of CAD

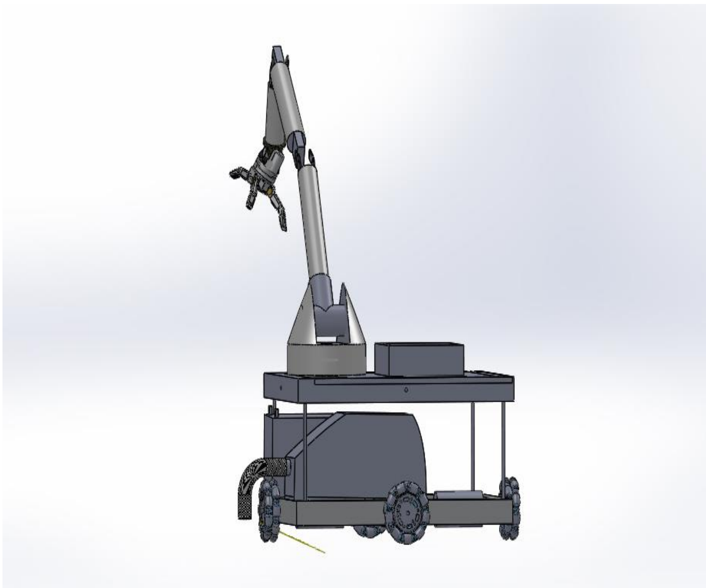


Fig.33 Side Veiw of CAD

3.2 Specifications of Components used in prototype

1. High Precision Quad Encoder Geared DC Motor

Specifications:

- Operating Voltages: 24VDC
- Load current: 3A max
- No load Speed: 12 RPM
- 88560 Counts per revolution
- Rated Torque: 40 Kg-cm

Fig.34 High Precision Quad Encoder Geared DC Motor: Actuator for manipulator joints

2. High Torque Encoder DC Servo Motor 10RPM with UART/I2C/PPM Drive

Specifications:

- 10RPM 12V DC motors with Metal Gearbox and Metal Gears
- 350gm weight
- 120kgcm torque
- No-load current = 800 mA, Load current = upto 7.5 A(Max)
- 0.2deg resolution optical encoder integrated on motor output shaft

Fig.35 High Torque DC Geared Motor:
Actuator for base motors and the shoulder

3. Camera

S. no.	**Features**	
1.	Sensor	• CMOS Sensor Technology
2.	Resolution	• Sensor Resolution:1920 x 1080 • 5 Mega pixel(2560 x 2048 pixel,interpolated)
3.	Imaging features	• Automatic face tracking • Digital pan,digital tilt,and 3x digital zoom • Auto focus from 0.1m to ≥ 10m • Automatic image adjustment with manual override • Up to 30 frames per second
4.	Frequency Response	• 100 Hz – 18 kHz

Table 5: Camera specifications

Fig.36 Microsoft HD camera(Life Cam Studio)

4.Vacuum Cleaner

Its'a 12 volts 100 Watt low power vacuum cleaner just suitable for our application.

A high power vacuum cleaner can be used for more efficient cleaning.

Fig. 37 Bergmann Hurricane Hi-Power Vacuum Cleaner

5. Omniwheel

Specifications:

- Diameter: 100mm
- Body material: Nylon
- Net weight: 290g
- Load capacity: 20kg

Fig.38   Dual row Omniwheel

3.3 Prototype Design

The components were chosen based on the results of conceptual design and results of the analysis.After all the components are received,we need to do the assembly very intelligently and properly.The main consideration is that of the couplings to be used for the base motors and link motors.You need to very careful while designing those couplings.Some of the images of the prototype are shown below.

Fig.39 Side View of Mobile Manipulator

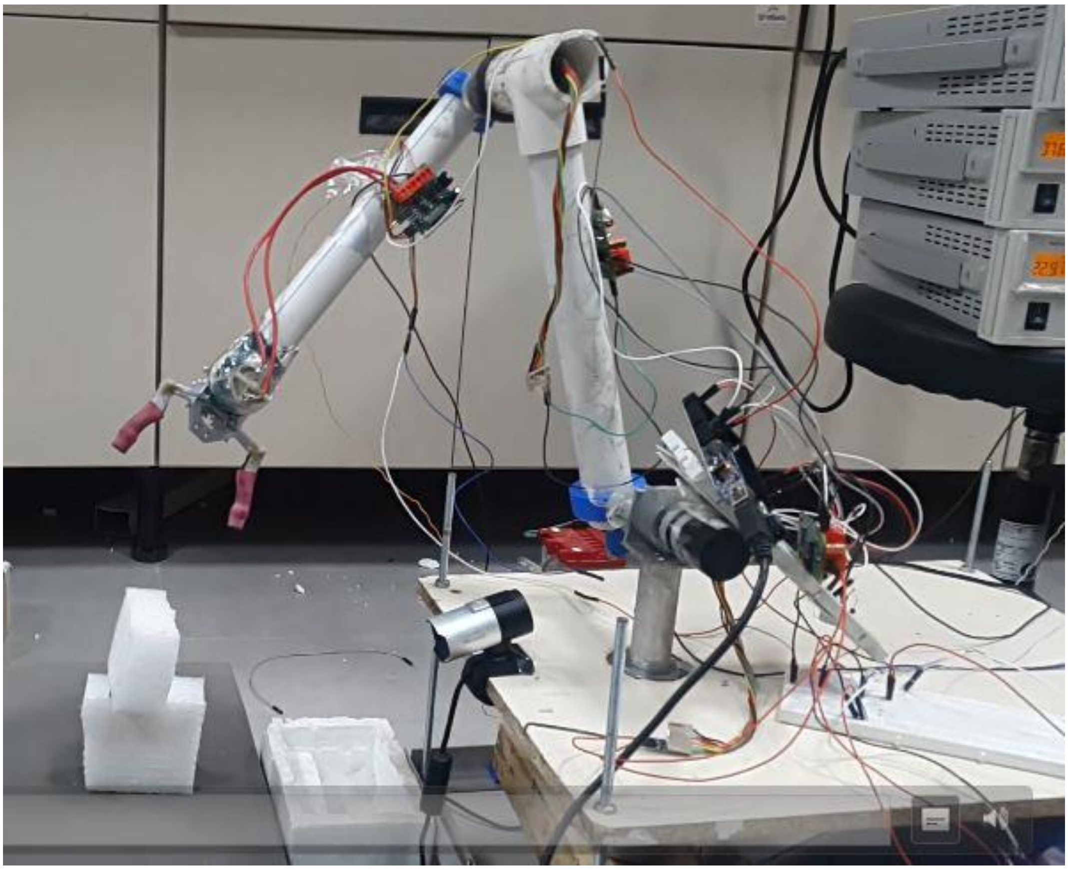


Fig. 40 Manipulator is trying to pick up an object

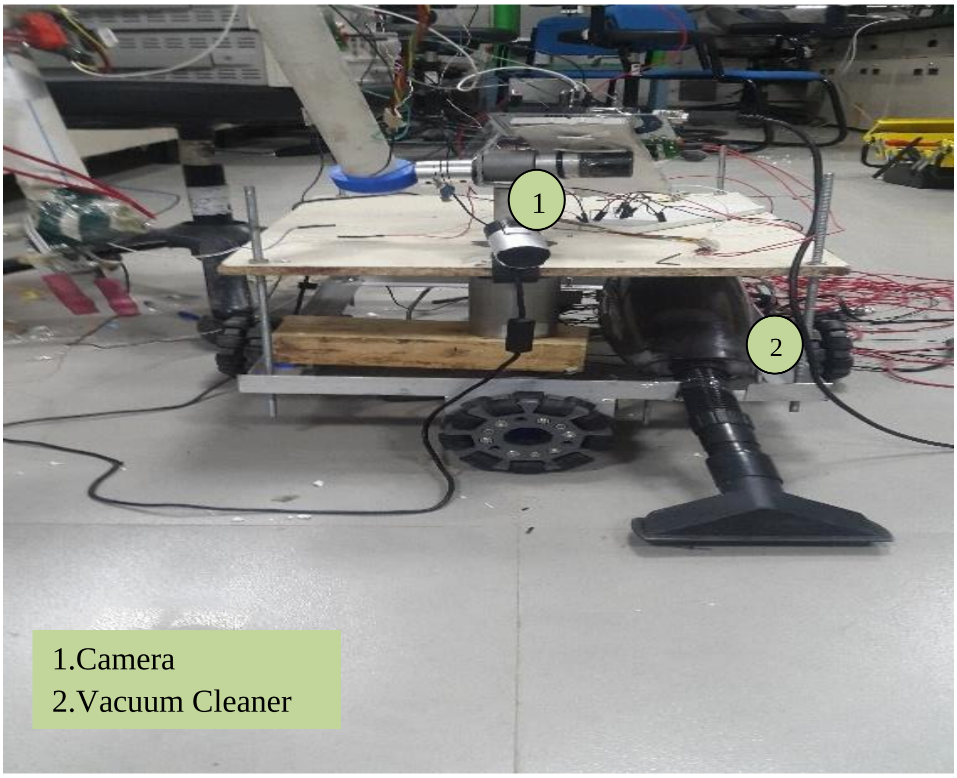


Fig. 41 Front View of the Mobile Manipulator:

4. DYNAMICS OF THE MANIPULATOR

Although We already calculated the maximum joint torques while the manipulator is in motion in ADAMS,but we haven't got the torque values in equation form in terms of the known parameters which is necessary in order to control the manipulator.So we did it in MATLAB using Newton-Euler Dynamics Algorithm.

4.1 The iterative Newton—Euler dynamics algorithm

The complete algorithm for computing joint torques from the motion of the joints is composed of two parts. First, link velocities and accelerations are iteratively computed from link 1 out to link n and the Newton—Euler equations are applied to each link in order to find the inertial force and the torque acting at the centre of mass of each link. Second, forces and torques of interaction and joint actuator torques are computed recursively from link n back to link 1.This is done by writing a force balance and moment balance equation based on a free body diagram of a typical link(see Fig.30). The equations are summarized next for the case of all joints rotational

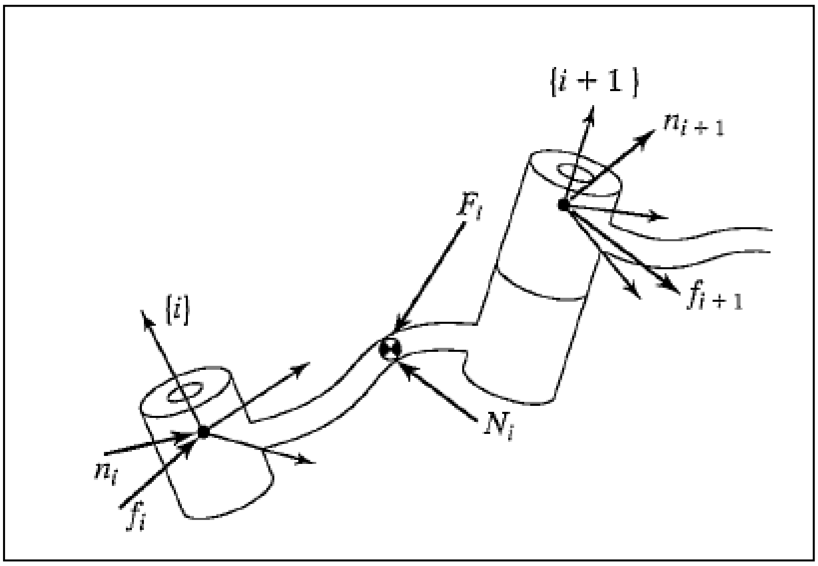


Fig.42

${}^{i}f_i$ = *force exerted on link i by link* $i-1$

${}^{i}n_i$ = *torque exerted on link i by link* $i-1$

F_i = *force acting on link i*

N_i = *Moment acting on link i*

${}^{C}I$ = *inertia tensor of link i in frame C located at centre of mass(CoM)*

${}^{i+1}_{i}R$ = *Rotation matrix describing B w.r.t. A*

${}^{i}P_{C_i}$ = *Position vector of CoM in link frame i*

$Outward\ iteration: i : 0 \rightarrow 5$

$${}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R\,{}^{i}\omega_i + \dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1},$$

$${}^{i+1}\dot{\omega}_{i+1} = {}^{i+1}_{i}R\,{}^{i}\dot{\omega}_i + {}^{i+1}_{i}R\,{}^{i}\omega_i \times \dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} + \ddot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1}$$

$${}^{i+1}\dot{v}_{i+1} = {}^{i+1}_{i}R\left({}^{i}\dot{\omega}_i \times {}^{i}P_{i+1} + {}^{i}\omega_i \times \left({}^{i}\omega_i \times {}^{i}P_{i+1}\right) + {}^{i}\dot{v}_i\right)$$

$${}^{i+1}\dot{v}_{C_{i+1}} = {}^{i+1}\dot{\omega}_{i+1} \times {}^{i+1}P_{C_{i+1}} + {}^{i+1}\omega_{i+1} \times \left({}^{i+1}\omega_{i+1} \times {}^{i+1}P_{C_{i+1}}\right) + {}^{i+1}\dot{v}_{i+1}$$

$${}^{i+1}F_{i+1} = m_{i+1}\,{}^{i+1}\dot{v}_{C_{i+1}}$$

$${}^{i+1}N_{i+1} = {}^{C_{i+1}}I_{i+1}\,{}^{i+1}\dot{\omega}_{i+1} + {}^{i+1}\omega_{i+1} \times {}^{C_{i+1}}I_{i+1}\,{}^{i+1}\omega_{i+1}$$

$Inverd\ Iterations : i : 6 \rightarrow 1$

$${}^{i}f_i = {}^{i}_{i+1}R\,{}^{i+1}f_{i+1} + {}^{i}F_i$$

$${}^{i}n_i = {}^{i}N_i + {}^{i}_{i+1}R\,{}^{i+1}n_{i+1} + {}^{i}P_{C_i} \times {}^{i}F_i + {}^{i}P_{i+1} \times {}^{i}_{i+1}R\,{}^{i+1}f_{i+1}$$

$$\tau_i = {}^{i}n_i^{T}\,{}^{i}\hat{Z}_i$$

Using the above algorithm,knowing the values of θ_i, $\dot{\theta}_i$, $\ddot{\theta}_i$ and solving in MATLAB will give us the torques at the joints. The effect of gravity loading on the links can be included quite simpiy by setting

${}^{0}\dot{v}_0$ = G,

where G has the magnitude of the gravity vector but points in the opposite direction. This is equivalent to saying that the base of the robot is accelerating upward with 1 g acceleration. This fictitious upward acceleration causes exactly the same effect on the links as gravity would. So, with no extra computational expense, the gravity effect is calculated.So using this algorithm,we got the following joint torque expressions.Here we have used gravity as 'G'

T1 $= (l_2^2 m_2 \ddot{\theta}_1)/6 + (l_2^2 m_3 \ddot{\theta}_1)/2 + (l_2^2 m_4 \ddot{\theta}_1)/2 + (l_3^2 m_3 \ddot{\theta}_1)/6 + (l_3^2 m_4 \ddot{\theta}_1)/2 + (l_4^2 m_4 \ddot{\theta}_1)/6 + (m_1 r^2 \ddot{\theta}_1)/2 + (l_2^2 m_2 \ddot{\theta}_1 C(2\theta_2))/6 + (l_2^2 m_3 \ddot{\theta}_1 C(2\theta_2))/2 + (l_2^2 m_4 \ddot{\theta}_1 C(2\theta_2))/2 + (l_4^2 m_4 \ddot{\theta}_1 C(2\theta_2+2\theta_3+2\theta_4))/6 + (l_3^2 m_3 \ddot{\theta}_1 C(2\theta_2+2\theta_3))/6 + (l_3^2 m_4 \ddot{\theta}_1 C(2\theta_2+2\theta_3))/2 + (l_2 l_3 m_3 \ddot{\theta}_1 C(2\theta_2+2\theta_3))/2 + l_2 l_3 m_4 \ddot{\theta}_1 C(2\theta_2+2\theta_3) + (l_2 l_4 m_4 \ddot{\theta}_1 C(2\theta_2+\theta_3+\theta_4))/2 + l_2 l_4 m_4$

$\ddot{\theta}_1 C(\theta_3+\theta_4))/2 - (l_2^2 m_2 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2))/3 - l_2^2 m_3 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2) - l_2^2 m_4 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2) - (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2+2\theta_3+2\theta_4))/3 - (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_2+2\theta_3+2\theta_4))/3 - (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_4 S(2\theta_2+2\theta_3+2\theta_4))/3 + (l_2 l_3 m_3 \ddot{\theta}_1 C(\theta_3))/2 + l_2 l_3 m_4 \ddot{\theta}_1 C(\theta_3) + (l_3 l_4 m_4 \ddot{\theta}_1 C(\theta_4))/2 + l_3 l_4 m_4 \ddot{\theta}_1 C(2\theta_2+2\theta_3+\theta_4))/2 - (l_3^2 m_3 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2+2\theta_3))/3 - (l_3^2 m_3 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_2+2\theta_3))/3 - l_3^2 m_4 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2+2\theta_3) - l_3^2 m_4 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_2+2\theta_3) + (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_4-2\theta_3))/48 + (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_3-2\theta_4))/48 + (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_4-2\theta_3))/48 + (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_3-2\theta_4))/48 + (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_4 S(2\theta_4-2\theta_3))/48 + (l_4^2 m_4 \dot{\theta}_1 \dot{\theta}_4 S(2\theta_3-2\theta_4))/48 - l_2 l_4 m_4 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2+\theta_3+\theta_4) - (l_2 l_4 m_4 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_2+\theta_3+\theta_4))/2 - (l_2 l_4 m_4 \dot{\theta}_1 \dot{\theta}_4 S(2\theta_2+\theta_3+\theta_4))/2 - (l_2 l_4 m_4 \dot{\theta}_1 \dot{\theta}_3 S(\theta_3+\theta_4))/2 - (l_2 l_4 m_4 \dot{\theta}_1 \dot{\theta}_4 S(\theta_3+\theta_4))/2 - (l_2 l_3 m_3 \dot{\theta}_1 \dot{\theta}_3 S(\theta_3))/2 - l_2 l_3 m_4 \dot{\theta}_1 \dot{\theta}_3 S(\theta_3) - (l_3 l_4 m_4 \dot{\theta}_1 \dot{\theta}_4 S(\theta_4))/2 - l_3 l_4 m_4 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2+2\theta_3+\theta_4) - l_3 l_4 m_4 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_2+2\theta_3+\theta_4) - (l_3 l_4 m_4 \dot{\theta}_1 \dot{\theta}_4 S(2\theta_2+2\theta_3+\theta_4))/2 - l_2 l_3 m_3 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2+\theta_3) - (l_2 l_3 m_3 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_2+\theta_3))/2 - 2 l_2 l_3 m_4 \dot{\theta}_1 \dot{\theta}_2 S(2\theta_2+\theta_3) - l_2 l_3 m_4 \dot{\theta}_1 \dot{\theta}_3 S(2\theta_2+\theta_3)$

T2 $=(l_2^2 m_2 \ddot{\theta}_2)/3 - 5\sqrt{2}\, l_3 C(\theta_4) - 5\sqrt{2}\, l_4 + l_2^2 m_3 \ddot{\theta}_2 + l_2^2 m_4 \ddot{\theta}_2 + (l_3^2 m_3 \ddot{\theta}_2)/3 + (l_3^2 m_3 \ddot{\theta}_3)/3 + l_3^2 m_4 \ddot{\theta}_2 + l_3^2 m_4 \ddot{\theta}_3 + (l_4^2 m_4 \ddot{\theta}_2)/3 + (l_4^2 m_4 \ddot{\theta}_3)/3 + (l_4^2 m_4 \ddot{\theta}_4)/3 - 5\sqrt{2}\, l_3 S(\theta_4) - 5\sqrt{2}\, l_2 C(\theta_3+\theta_4) - 5\sqrt{2}\, l_2 S(\theta_3+\theta_4) + G l_3 m_3 C(\theta_2+\theta_3)/2 + G l_3 m_4 C(\theta_2+\theta_3) + + G l_2 m_2 C(\theta_2)/2 + + G l_2 m_3 C(\theta_2) + + G l_2 m_4 C(\theta_2) + (G l_4 m_4 C(\theta_2+\theta_3+\theta_4))/2 + (l_2^2 m_2 \dot{\theta}_1^2 S(2\theta_2))/6 + (l_2^2 m_3 \dot{\theta}_1^2 S(2\theta_2))/2 + (l_2^2 m_4 \dot{\theta}_1^2 S(2\theta_2))/2 + (l_4^2 m_4 \dot{\theta}_1^2 S(2\theta_2+2\theta_3+2\theta_4))/6 + (l_3^2 m_3 \dot{\theta}_1^2 S(2\theta_2+2\theta_3))/6 + (l_3^2 m_4 \dot{\theta}_1^2 S(2\theta_2+2\theta_3))/2 - (l_4^2 m_4 \dot{\theta}_1^2 S(2\theta_4-2\theta_3))/96 - (l_4^2 m_4 \dot{\theta}_1^2 S(2\theta_3-2\theta_4))/96 + (l_4 l_2 m_4 \dot{\theta}_1^2 S(2\theta_2+\theta_3+\theta_4))/2 - (7 l_4 l_2 m_4 \dot{\theta}_1^2 S(\theta_3+\theta_4))/256 - (l_4 l_2 m_4 \dot{\theta}_3^2 S(\theta_3+\theta_4))/2 - (l_4 l_2 m_4 \dot{\theta}_4^2 S(\theta_3+\theta_4))/2 - (l_3 l_2 m_3 \dot{\theta}_3^2 S(\theta_3))/2 - l_3 l_2 m_4 \dot{\theta}_3^2 S(\theta_3) - (l_3 l_4 m_4 \dot{\theta}_4^2 S(\theta_4))/2 + (l_3 l_4 m_4 \dot{\theta}_1^2 S(2\theta_2+2\theta_3+\theta_4))/2 + (l_3 l_2 m_3 \dot{\theta}_1^2 S(2\theta_2+\theta_3))/2 + l_3 l_2 m_4 \dot{\theta}_1^2 S(2\theta_2+\theta_3) - (l_3 l_2 m_4 \dot{\theta}_1^2 S(\theta_3-2\theta_4))/64 - (l_3 l_2 m_4 \dot{\theta}_1^2 S(\theta_3+2\theta_4))/64 - (3 l_4 l_2 m_4 \dot{\theta}_1^2 S(\theta_3-\theta_4))/256 - (3 l_4 l_2 m_4 \dot{\theta}_1^2 S(\theta_4-\theta_3))/256 - (3 l_4 l_2 m_4 \dot{\theta}_1^2 S(\theta_3-3\theta_4))/256 - (l_4 l_2 m_4 \dot{\theta}_1^2 S(\theta_3+3\theta_4))/256 - (l_4 l_2 m_4 \dot{\theta}_1^2 S(2\theta_3+\theta_4))/256 - (l_4 l_3 m_4 \dot{\theta}_1^2 S(2\theta_3+\theta_4))/128 + l_4 l_2 m_4 \ddot{\theta}_2 C(\theta_3+\theta_4) + (l_4 l_2 m_4 \ddot{\theta}_3 C(\theta_3+\theta_4))/2 + (l_4 l_2 m_4 \ddot{\theta}_4 C(\theta_3+\theta_4))/2 + l_3 l_2 m_3 \ddot{\theta}_2 C(\theta_3) + (l_3 l_2$

$m_3\ \ddot{\theta}_3\ C(\theta_3))/2\ +\ 2l_3\ l_2\ m_4\ \ddot{\theta}_2\ C(\theta_3)+l_3\ l_2\ m_4\ \ddot{\theta}_3\ C(\theta_3)+l_3\ l_4\ m_4\ \ddot{\theta}_2\ C(\theta_4)\ +\ l_3\ l_4\ m_4\ \ddot{\theta}_3\ C(\theta_4)\ +\ ($

$l_3\ l_4\ m_4\ \ddot{\theta}_4\ C(\theta_4))/2\ -\ (l_3\ l_2\ m_4\ \dot{\theta}_1^2\ S(-\theta_3-2\theta_4))/64\ -\ (l_3\ l_2\ m_4\ \dot{\theta}_1^2\ S(2\theta_4-\theta_3))/64\ -\ (7l_4\ l_2\ m_4$

$\dot{\theta}_1^2\ S(-\theta_4-\theta_3))/256\ -\ (l_4\ l_2\ m_4\ \dot{\theta}_1^2\ S(-3\theta_4-\theta_3))/256\ -\ ((3l_4\ l_2\ m_4\ \dot{\theta}_1^2\ S(3\theta_4-\theta_3))/256\ -\ (($

$l_4\ l_2\ m_4\ \dot{\theta}_1^2\ S(-\theta_4-3\theta_3))/256\ -\ ((l_4\ l_2\ m_4\ \dot{\theta}_1^2\ S(3\theta_4-3\theta_3))/256\ -\ ((l_4\ l_2\ m_4\ \dot{\theta}_1^2\ S(3\theta_3-3\theta_4))/256$

$-\ (\ (l_4\ l_3\ m_4\ \dot{\theta}_1^2\ S(-\theta_4-2\theta_3))/128\ -\ (l_4\ l_3\ m_4\ \dot{\theta}_1^2\ S(3\theta_4-2\theta_3))/128\ -\ (l_4\ l_3\ m_4\ \dot{\theta}_1^2\ S(2\theta_3-3\theta_4)$

$)/128\ -\ l_4\ l_2\ m_4\ \dot{\theta}_2\dot{\theta}_3\ S(\theta_3+\theta_4)\ -\ l_4\ l_2\ m_4\ \dot{\theta}_2\dot{\theta}_4\ S(\theta_3+\theta_4)\ -\ l_4\ l_2\ m_4\ \dot{\theta}_4\dot{\theta}_3\ S(\theta_3+\theta_4)\ -l_3\ l_2\ m_3$

$\dot{\theta}_2\dot{\theta}_3\ S(\theta_3)\ -\ 2l_3\ l_2\ m_4\ \dot{\theta}_2\dot{\theta}_3\ S(\theta_3)-\ l_4\ l_3\ m_4\ \dot{\theta}_2\dot{\theta}_4\ S(\theta_4)\ -l_4\ l_3\ m_4\ \dot{\theta}_4\dot{\theta}_3\ S(\theta_4)$

T3 $=(l_3^2\ m_3\ \ddot{\theta}_2)/3-5\sqrt{2}\,l_3C(\theta_4)\ -5\sqrt{2}\,l_4+\ (l_3^2\ m_3\ \ddot{\theta}_3)/3\ +\ l_3^2\ m_4\ \ddot{\theta}_2+\ (l_3^2\ m_4\ \ddot{\theta}_3)+\ ((l_4^2\ m_4\ \ddot{\theta}_2)/3\ +$

$((l_4^2\ m_4\ \ddot{\theta}_3)/3\ +\ ((l_4^2\ m_4\ \ddot{\theta}_4)/3\ -\ 5\sqrt{2}\,l_3S(\theta_4)\ +\ +Gl_3m_3C(\theta_2+\theta_3)/2+Gl_3m_4C(\theta_2+\theta_3)$

$+(Gl_4m_4C(\theta_2+\theta_3+\theta_4))/2\ +\ (l_4^2\ m_4\ \dot{\theta}_1^2\ S(2\theta_2+2\theta_3+2\theta_4))/6\ +$

$(l_3^2\ m_3\ \dot{\theta}_1^2\ S(2\theta_2+2\theta_3\ ))/6\ +\ (l_3^2\ m_4\ \dot{\theta}_1^2\ S(2\theta_2+2\theta_3\ ))/2\ +\ (l_4\ l_2\ m_4\ \dot{\theta}_1^2\ S(2\theta_2+\theta_3+\theta_4))/4\ +\ ($

$l_4\ l_2\ m_4\ \dot{\theta}_1^2\ S(\theta_3+\theta_4))/4\ +\ l_4\ l_2\ m_4\ \dot{\theta}_2^2\ S(\theta_3+\theta_4))/2\ +\ (l_3\ l_2\ m_3\ \dot{\theta}_1^2\ S(\theta_3))/4\ +\ (l_3\ l_2\ m_3\ \dot{\theta}_2^2\ S(\theta_3)$

$)/2\ +\ (l_3\ l_2\ m_4\ \dot{\theta}_1^2\ S(\theta_3))/2\ +\ l_3\ l_2\ m_4\ \dot{\theta}_2^2\ S(\theta_3)-(l_3\ l_4\ m_4\ \dot{\theta}_4^2\ S(\theta_4))/2\ +\ (l_3\ l_4\ m_4$

$\dot{\theta}_1^2\ S(2\theta_2+2\theta_3+\theta_4))/2\ +\ (l_3\ l_2\ m_3\ \dot{\theta}_1^2\ S(2\theta_2+\theta_3))/4\ +\ l_3\ l_2\ m_4\ \dot{\theta}_1^2\ S(2\theta_2+\theta_3)\ +\ l_4\ l_2\ m_4$

$\ddot{\theta}_2\ C(\theta_3+\theta_4)\ +\ (l_3\ l_2\ m_3\ \ddot{\theta}_2\ C(\theta_3))/2\ +\ l_3\ l_2\ m_4\ \ddot{\theta}_2\ C(\theta_3)\ +\ l_3\ l_4\ m_4\ \ddot{\theta}_2\ C(\theta_4)\ +\ l_3\ l_4\ m_4\ \ddot{\theta}_3\ C(\theta_4)+$

$(l_3\ l_4\ m_4\ \ddot{\theta}_4\ C(\theta_4))/2\ -\ l_4\ l_3\ m_4\ \dot{\theta}_2\dot{\theta}_4\ S(\theta_4)\ -l_4\ l_3\ m_4\ \dot{\theta}_4\dot{\theta}_3\ S(\theta_4)$

T4 $=(l_4(6Gm_4C(\theta_2+\theta_3+\theta_4)-60\sqrt{2}+4l_4\ m_4\ \ddot{\theta}_2\ +4l_4\ m_4\ \ddot{\theta}_3\ +4l_4\ m_4\ \ddot{\theta}_4\ +6l_2\ m_4\ \ddot{\theta}_2\ C(\theta_3+\theta_4)+2l_4\ m_4$

$\dot{\theta}_1^2\ S(2\theta_2+2\theta_3+2\theta_4)+6l_3\ m_4\ \ddot{\theta}_2\ C(\theta_4)+6l_3\ m_4\ \ddot{\theta}_3\ C(\theta_4)+3l_2\ m_4\ \dot{\theta}_1^2\ S(2\theta_2+\theta_3+\theta_4)+3l_2\ m_4$

$\dot{\theta}_1^2\ S(\theta_3+\theta_4)\ 6l_2\ m_4\ \dot{\theta}_2^2\ S(\theta_3+\theta_4)+3l_3\ m_4\ \dot{\theta}_1^2\ S(\theta_4)+6l_3\ m_4\ \dot{\theta}_2^2\ S(\theta_4)+6l_3\ m_4\ \dot{\theta}_3^2\ S(\theta_4)+3l_3\ m_4$

$\dot{\theta}_1^2\ S(2\theta_2+2\theta_3+\theta_4)\ 12l_3\ m_4\ \dot{\theta}_2\ \dot{\theta}_3\ S(\theta_4)))/12$

Above equations give expressions for the torque at the joint actuators as a function of joint position, velocity, and acceleration, link lengths.Obviously the equations are quite tedious.So it is often convenient to express the dynamic equations of a manipulator in a single equation that hides some of the details, but shows some of the structure of the equations.Evaluating above equations yieds a dynamic equation called as state space equation

$$\Gamma = M(\Theta)\ddot{\Theta} + V(\Theta,\dot{\Theta}) + G(\Theta)$$

Where $M(\Theta)$ is $n \times n$ **mass matrix** of manipulator, $V(\Theta,\dot{\Theta})$ is an $n \times 1$ **vector** of **Centrifugal and Coriolis** terms,and $G(\Theta)$ is an $n \times 1$ **vector** of **gravity terms**.For the proposed configuration we calculated the Mass matrix, $V(\Theta,\dot{\Theta})$, $G(\Theta)$ vectors which are useful in controlling the manipulator (for length considerations we haven’t written here)

5. CONTROLS AND PROGRAMMING

5.1 Actuator Controls

After the complete assembly of the proposed system, there was a need to check the system's capability to perform the desired functions. In the introductory phase we just controlled the system manually without any sort of automation for the demonstation purpose.In the proposed system,the 4 link motors providing 4 DoFs to the manipulator,4 base motors providing 3DoFs to the mobile base,The Vacuum cleaner,the Camera sensor are controlled independently. Following are the different types of manipulator controls that can be implemented.

1.Limited Sequence Control :
Pick and place operations using mechanical stops to set positions

2.Point to Point Control:
Records work cycle as a sequence of points, then playbacks the sequence during program execution.

3.Continuous Path Control:
Capability to execute paths (in addition to points)

4.Intelligent Control:
Responds to sensor inputs, make decisions and communicates with humans as well

For the proposed manipulator, The Intelligent Control has to be used .For the sake of demonstration purpose we have used the Limited Sequence Control for the time being.Here we have a fixed task and its fixed trajectory so the manipulator is able to sense the coordinates of the object using camera vision and our manipulator is able to reach to the object pick up the object and then put it into a basket.This task can be done iteratively.Three High Precision Quad Encoder Geared DC Motors are used for the links and are controlled using the Arduino and standard motor drives. Following are the features of the manipulator control:

- A **MIMO** problem involving joint and the end-effector locations, velocities and force vectors
- Consideration of each joint as independent for simplification
- Each joint is assumed to have a single input(set point) and single output(location ,velocity)
- 4 -DOF system controlled by 4-independent single-input, single-output control systems

The High Torque Encoder DC Servo Motors,used at the shoulder and at the omniwheels come with UART/I2C/PPM Drive.It means that the motor is an I2C slave device.In the system we are having five such kind of I2C devices. I2C is the easiest communication technique if multiple RMCS-220x or i2C devices are to be controlled from the same I2Ccontrol master.

The Inter-integrated Circuit (I^2C) Protocol is a communication protocol intended to allow multiple "slave" digital integrated circuits ("chips") to communicate with one or more "master" chips. I^2C uses only two bidirectional open-drain lines, Serial Data Line (**SDA**) and Serial Clock Line (**SCL**), pulled up with resistors to exchange information.

So in order to communicate with the I2C devices using Arduino, the library you need to use is the 'Wire Library'. The I2C pins are located on the Arduino boards as follows:

Board(Arduino)	**I2C pins**
Uno	A4 (SDA), A5 (SCL)
Mega2560	20 (SDA), 21 (SCL)

5.2 Vacuum Cleaner Control

Vacuum cleaner is an important feature of the system since it is being used to clean the area from dust,small bits of paper and the objects which cannot be picked up by the gripper of the manipulator. Hence the vacuum cleaner needs to be running throughout the operation.It can easily be achieved by an Arduino board and a motor driver.

5.3 Image Processing Algorithm

The next important thing is the object detection by the camera.The flow chart explaining the sequence we need to follow in order to detect the blue coloured body and show its coordinates is shown below.MATLAB has a very useful Image processing toolbox which is used here to accomplish the image processing.

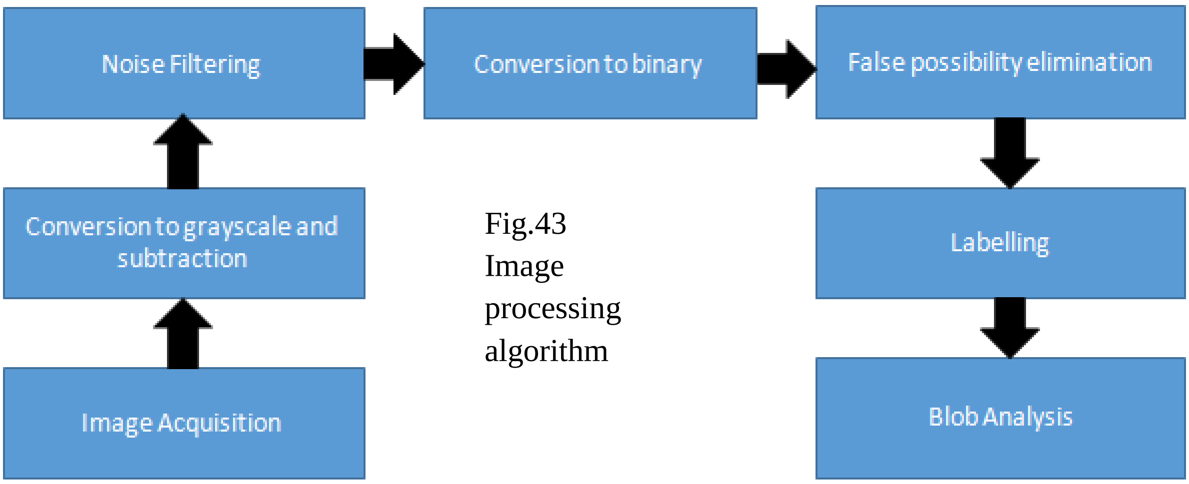


Fig.43 Image processing algorithm

The MATLAB commands that needs to be used are explained in brief herein.You can refer the code explained below.

a = imaqhwinfo('winvideo');

returns a structure,a, which contains information related to a particular adaptor(here it's 'winvideo').This information includes adaptor version and available hardware for the specified adaptor.

vid = videoinput(camera_name, camera_id, format) ;

constructs a video input object used to capture the video frames

set(vid, 'FramesPerTrigger', Inf)

The 'FramesPerTrigger' Specify number of frames to acquire per trigger using selected video source and when it is set to 'inf',the video input object keeps acquiring frames until an error occurs or you issue a stop command.

set(vid, 'ReturnedColorspace', 'rgb');

The 'ReturnedColorspace' property specifies the color space you want the toolbox to use when it returns image data to the MATLAB workspace

And here it is set to MATLAB RGB color space.

vid.FrameGrabInterval = 5;

The FrameGrabInterval property specifies how often the video input object acquires a frame from the video stream.Here it is set to 5 ,so the object acquires every 5^{th} frame from the video stream

Now to track blue objects in real time,we have to subtract the grayscale image from the image to extract the blue components in the image.The command to be used is imsubtract(data(:,:,3), rgb2gray(data));

medfilt2(diff_im, [3 3]);

In order to remove the salt & pepper noise from the image we need to do the image filtering

The command performs median filtering of the matrix 'diff_im' in two dimensions. Each output pixel contains the median value in the 3-by-3 neighborhood around the corresponding pixel in the input image.

in order to find a Region Of Interest - a portion of the image that is of interest for further processing and also to reduce the computational complexities we need to convert the image to binary.The command im2bw(diff_im,0.18);

converts the grayscale image ‘diff_im‘ to a binary image. The output image replaces all pixels in the input image with luminance greater than 0.18 with the value 1 (white) and replaces all other pixels with the value 0 (black).

Now in order to remove some small objects in the image in order to reduce the chances of detection of wrong object, we need to subtract from image the pixels less than a certain limit so that only the bigger objects are left, which are of our interest.hence we use the command

bwareaopen(diff_im,300);

which removes all the pixels less than 300px

In order to distinguish all the components in the image we need to label them which is done by the command bwlabel(diff_im, 8);

Which returns a matrix describing 8 connected objects and whose elements are integer values greater than or equal to 0. The pixels labeled 0 are the background. The pixels labeled 1 make up one object; the pixels labeled 2 make up a second object; and so on

Now once the objects are identified we need to know their properties like centroid,area,bounding box

The command regionprops(bw, 'BoundingBox', 'Centroid');

measures a properties 'BoundingBox', 'Centroid' for each connected component (object) in the binary image ‘bw’.This is called as Blob Analysis and

now we have the properties of the different regions,so we can bound the blue objects in a rectangular box and specify its coordinates which is done using the for-loop.

The code which is used in our system in order to detect the coordinates of the object to be picked up(for the time being we are just detecting a blue coloured object) is written below.The image shown below it clearly depicts the coordinates of the blue coloured object detected by the camera.These coordinates then are sent to MATLAB in order to find its joint space using the inverse kinematics,then the angles through which motor shaft need to rotate will be sent to microcontroller who will finally send the signals to the joint motors and then the manipulator will proceed to reach to the detected object.

```
//The Code
a = imaqhwinfo('winvideo');
[camera_name, camera_id, format] = getCameraInfo(a);
vid = videoinput(camera_name, camera_id, format);
set(vid, 'FramesPerTrigger', Inf);
set(vid, 'ReturnedColorspace', 'rgb')
vid.FrameGrabInterval = 5;
start(vid)
while(vid.FramesAcquired<=200)
    data = getsnapshot(vid);
    diff_im = imsubtract(data(:,:,3), rgb2gray(data));
    diff_im = medfilt2(diff_im, [3 3]);
    diff_im = im2bw(diff_im,0.18);
    diff_im = bwareaopen(diff_im,300);
    bw = bwlabel(diff_im, 8);
    stats = regionprops(bw, 'BoundingBox', 'Centroid');
    imshow(data)
    hold on
    for object = 1:length(stats)
        bb = stats(object).BoundingBox;
        bc = stats(object).Centroid;
        rectangle('Position',bb,'EdgeColor','r','LineWidth',2)
        plot(bc(1),bc(2), '-m+')
        a=text(bc(1)+15,bc(2), strcat('X: ', num2str(round(bc(1))), '    Y: ',
num2str(round(bc(2)))));
        set(a, 'FontName', 'Arial', 'FontWeight', 'bold', 'FontSize', 12, 'Color',
'yellow');
    end
     hold off
end
stop(vid);
flushdata(vid);
```

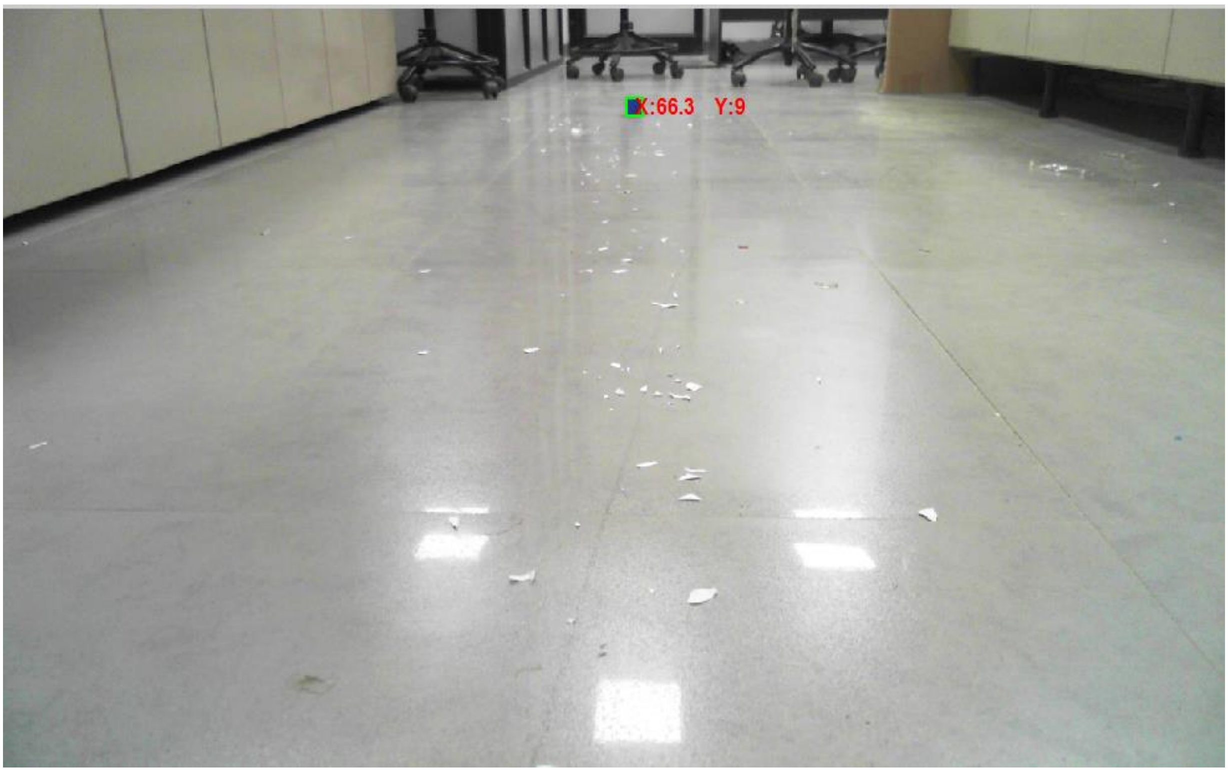


Fig.44 The detected object (bounded by yellow lines) with its coordinates

6. CONCLUSIONS AND FUTURE SCOPE

The paper presents design and development of a smart mobile manipulator which will use intelligent intervention methods to perform variety of tasks such as cleaning,monitoring etc.The mathematical modelling including the Workspace analysis,Dynamic analysis and the Static force analysis clearly suggested that the RRRR configuration of the manipulator is better than that of the RPRR for the desired task intend to perform.The next part of fabrication is one of the most important step including couplings,joints,proper alignments,dimensions which are some of the critical things need to be taken care of.Performing the limited sequence control of the manipulator validates the accurateness of virtual modelling as well as the physical prototyping and its controlling.

Next,the manipulator control can be made more advanced and fully automated.The Camera will be made able to detect some of the common waste matters such as dry leaves,thrown sachets,empty bottle and not just a specific coloured object .This can be achieved using advanced image processing functions in MATLAB toolbox.

When the mobile manipulator enters an unstructured environment,It should be able to sense the exact location of the waste matter as the mobile base proceeds on the terrain. For this we'll need IR distance sensors,Ultrasonic sensors which will feed input to the microcontroller regarding the proximity of the obstacle or the waste matter to be cleaned, accordingly the directions will be given to the base I2C slave motors.The object to be picked up or not will be checked by the Image processing algorithm.We can install a GPS on the base so that we are able to track the position of the base continuosly which has further advantages.We can attach a GSM module to the base so that if somehow or other the bot is not able to clean a particular spot it will send immediately a text message or a preprogrammed call to the care taker,so that the spot will be cleaned manually.We can put an accelerometer sensor in order to keep track of the motion and inclination of the mobile base that we can imrove functionalities like the quality of cleaning,base motor performance etc. In this way the bot will be able to autonomously clean an area without any human intervention.

but this product may need a deeper look on its controls aspect and structural optimisation aspects,because inaccurate mechanical design may lead to falure of its capability in certain conditions.Hence an all round precautions must be taken care of while designing.The controls should be such that the tasks are carried out smoothly.

Since the product is able to communicate with other devices and able to carry out the intended tasks , it can easily be used for IOT applications such smart systems,smart cities projects,etc. which can be the future scope of work on this project.

REFERENCES

- B. Hammer, S. Koterba ,J.Shi, R. Simmons ,S. Singh, "An autonomous mobile manipulator for assembly tasks", vol. 28. Springer,Autonomous robots , pp.131-149, 2010.
- M.Boukattaya,T.Damak,M.Jallouli, "Robust Adaptive Control for Mobile Manipulators," in International Journal of Automation and Computing, vol. 8(1), pp. 8-13 , 2011.
- S. Mishra, M.Santhakumar*, S.K. Vishvakarma, M.Vasanthakumar and B.Vinod "Task space motion control of a mobile manipulator using a nonlinear PID control along with an uncertainty estimator "in International Conference on Advancements in Automation Robotics & Sensing"(ICAARS)[June 23-24 2016](submitted)
- Yogesh Singh,V.Vinoth,Y.Ravi Kiran,Jayant Kumar Mohanta,Santhakumar Mohan Inverse dynamics and control of a3-DOF planar parallel (U-shaped 3-PPR) manipulator,India,2015.
- J. J. Craig, Introduction to Robotics: Mechanics and Control, Addison Wesley, Boston, MA, USA, 1986.
- R. Kelly, V. Santibanez, A. Loria, Control of Robot Manipulators in Joint Space, Springer Verlag, London, UK, 2005.
- A. Sabanovic, K. Ohnishi, Motion Control Systems, John Wiley and Sons, Singapore, 2011.
- J. J. E. Slotine, W. Li, Applied Nonlinear Control, Prentice-Hall, Englewood Cliffs, NJ, USA, 1991.
- M.Santhakumar,T.Asokan,Application of robust design techniques for underwater vehiclecontrol,in:Proceedings of ISOPE Ocean Mining Symposium. Chennai, India,2009,pp.285–289.
- http://spectrum.ieee.org/automaton/robotics/industrial-robots/fetch-robotics-introduces-fetch-and-freight-your-warehouse-is-now-automated
- D.L. Pieper. The kinematics of manipulators under computer control. PhD Thesis, Stanford University, Department of Mechanical Engineering, 1968
- P. Moubarak, et al., A Globally Converging Algorithm for Adaptive Manipulation and Trajectory Following for Mobile Robots with Serial Redundant Arms, Robotica, 31 (8) (2013) 1299 – 1311.
- Burdick JW (1991) A classification of 3R regional manipulator singularities and geometries. In: Proceedings of the IEEE International Conference on Robotics and Automation, Sacramento, CA, pp 2670–2675
- Tourassis VD, Ang MH (1992) Identification and analysis of robot manipulator singularities. Int J Robot Res 11:248–259