Design and Implementation of an Autonomous Robot Manipulator for Pick & Place Planning

Mohd. Nayab Zafar¹, J. C. Mohanta¹, Alok Sanyal¹
¹Motilal Nehru National Institute of Technology Allahabad, Prayagraj-211004, India
E-mail: rme1506@mnnit.ac.in, jcmohanta@mnnit.ac.in and nayab.zafar7@gmail.com

Abstract
This paper focuses on advancement of an autonomous robot manipulator for pick and place applications using Artificial Intelligence Technique. The mobile robot carries a four Degree of Freedom (DOF) robotic arm for picking a particular object from a given initial position to the required target position in an unknown static environment, while tackling obstacles and implementing path planning algorithm using artificial intelligence technique. The proposed artificial intelligence technique employs the optimization strategy, based on a novel meta-heuristic approach for mobile robot path planning. The mobile manipulator involves the use of infrared, ultrasonic sensors for detecting obstacles, microcontroller for artificial intelligence software and geared motors & servo motors for motion locomotion. The effectiveness of the method was tested and verified by via simulation mode on four different trajectories. From the simulation results, it was found that approximately path length and elapsed time of triangular shape, conical shape, cubical shape&S- shape obstacles environments are 70.22, 70.92, 71.29 & 70.36 pixels and 117.93, 73.94, 122.86 & 117.87 seconds respectively.

Keywords: Mobile Robot Manipulator, Artificial Intelligence Technique, Path Planning, Obstacle Avoidance.

1. Introduction
In the field of robotics, mobile robot manipulators are generally combination of manipulator arm integrated with a movable base for performing remarkable jobs, dexterous tasks routinely in industries. In research community/industry of robotics, robotic manipulators are seeking attention due its multifaceted nature of obstacle avoidance and kinematic singularity avoidance in a jumbled situation type arena. Mounting of a manipulator over mobile base overcomes the workspace limitations, joint angle limitations redundancies and failure of joints that are incompatible for human interference in operating arena. In Cartesian coordinates, inverse kinematics simplifies the finding of correct coordinate of base and joint positions for a given end effectors coordinates [1]. Basically in manipulation a widely range of products are assembled as cars to microprocessors with highly precise speed and accuracy. The lack of abundance of robots in our homes might seems puzzling that they can assemble the cars but not clearing the kitchen stuffs after cooking, researchers claim that industries are well structured but homes are unstructured by muddle and disorderliness which creates difference that industries are structured for robots and homes are structured for humans. Finally, researchers are working to enhance the understanding, adaptability and utilizing the structure scenario of our homes, due to clutter and messiness in car assembling industry. In contrast, a car mechanics workshop is
small, with a few skilled mechanics spread over a few square feet, each performing a multitude of tasks which might create confusion related to home tasks.

According to Raghavan et al. [2], problems arise in designing series and parallel manipulators for applying prescribed wrenches on the environment. Methodology is basically depending on the basic principle of statics, which yields polynomial design equations and are easier to deal with in comparison to the differential equations, when inertial effects are modeled. According to Shiller et al. [3], optimal dynamic performance is measured by designing manipulators for selecting the link lengths and actuator sizes for least time motions along specified paths. Exact methods are applied for optimizing parameters and cost functions as the motion time along the path serves as a benchmark for a best efficient approximation, by selecting system parameters to explore the acceleration along specified path. According to Mermertas et al. [4], in parallel manipulators only single location is described between force and motion (i.e., mobility index=1), maximum mobility (local) index depends upon the input link position and design charts of manipulator, based on link measuring parameters performance of manipulator can be maximized for not only a certain position, also for position intervals. Where, local mobility index is not equivalent to one, there it gives better conditions in terms of force-motion relationship in compared with parallel manipulator. According to Harada et al. [5], planner of dual arm manipulators for objects pick and place motion in both offline and online phases is developed, object with set of regions and environment surfaces are developed in offline phase and several parameters needed in the online phase is calculated, then planner selects a grasping position of the robot and a putting posture of the object by searching regions found in offline phase is basically the online phase. It also plans robot’s trajectory and dual arm rewrapping strategy. According to Begum et al. [6], obtaining performance of pick and place robot using object detection application by android and PIC microcontroller basically programmed in java language is designed, input of voice is received by RF receiver and processed to the controller by HM2007 (RF module) microcontroller. Picking of object using camera of mobile which overcomes the drawback of sensors by capturing the objects via android mobile applications. Further, Bluetooth sends output received to the controller. Electromagnetic induction concept is applied for wireless charging, allowing the robot to charge itself whenever battery is low on-board. According to Patel et al. [7], development of framework for generating optimal geometric structure of manipulator for various task necessities and performance restraints in terms of the required end-effector locations, orientations along the task route. Under set of operating restraints this methodology, which can be used analytically or numerically in module of inverse kinematics guaranteed the task performance of the developed manipulator structure. According to Aboti et al. [8], development of a general framework for prototyping of robotic arm with 5 degrees of freedom, SCARA configuration to pick and place numerous quantities of components from a horizontal stack into a loading crate based on task descriptions and operating constraints. Denavit-Hartenberg convention and generalized algorithm is applied for joint link parameters, validation is done using MATLAB® gives desirable value with error less than 1% of the real values. According to McTaggart et al. [9], Cartesian novel manipulator named Cartman (Amazon Robotics Challenge 2017 winner) can be examined for evaluation of the design and performance for undertaking the challenges associated with motion planning and control problem for multiple degree of freedom in robotic manipulation. The aforesaid manipulator having linear speed limited to 1 m/s, angular speed to 1.5 radian/s, payload of 2 kg, and static sub-millimetre accuracy, due to belt slipping in system target speed is not achieved its yields limited speed up to 0.57 m/s. According to Ropo et al. [10], framework for optimal utilization of machine and manpower with desired speed and highly desirable accuracy for performing repetitive tasks in industries except Nigerian country, where availability of cheap labours is very high creating major issue in terms of cost. A direct drive serial manipulator having 3 degrees of freedom is designed (Blender software), fabricated (3D printing) and controlled (servo motors) for performing tasks (i.e., movements of ferromagnetic metals), direct drive mechanism incorporates to convey heavy loads lacking speed of system. As per requirement the payload capacity and required torque at each joint is 15N and 10.787N-m respectively, which is sufficient as required at elbow joint, shoulder joint and base joints is 4.3N-m, 7.2N-m and 8.7N-m, respectively. According to Ram et al.
[1] robotic manipulator working under obstruction free arena and cluttered arena is solved using bidirectional PSO for obtaining inverse kinematics solution. Bidirectional method gives quicker result compared to unidirectional method, line type obstructions tackling problem in collision detection and obstruction avoidance tasks are undertaken via particle swarm technique considering the objective function as movement of joints and constraints as collision avoidance, illustration of systems effectiveness experiments are performed numerically on 4 degree of freedom manipulator system.

The aforesaid literature review shows that there is a scope of improvement in the techniques given above for path optimization and use of mobile robot for pick and drop applications. In the above techniques there are too many local optima’s obtained [12-14]. Most of the paths developed by researchers especially for the situation like edge navigation at the turning points having sharp edges that may be difficult to obtain in real robot. So, we have chosen grey wolf optimization as the Artificial Intelligence Technique for mobile robot manipulator. The main objectives of the paper are:
- To pick and place the object in minimum possible time with the help of motion planning.
- Finding the most optimized path from starting location to target location.
- Selection of the most optimized path and movement through artificial intelligence technique.
- Avoiding obstacles (Static/Dynamic) while moving from starting location to target location.

2. Analysis of Mobile Manipulator

2.1. Problem Description

Consider the following map as shown in figure 1, we have to pick and place an object in an unknown environment thereby avoiding obstacles and following an optimized path. For implementing this a robotic arm is mounted on a mobile robot, which picks an object from a given start point and then implement path planning using artificial intelligence technique to reach to a given target point after which it places the object.

![Figure 1. Description of Problem](image_url)

2.2. Motion Planning Steps:
- General Goal: Compute motion commands to achieve a goal arrangement of physical objects.
- Basic problem: Collision-free path planning among static obstacles.

*Inputs:* (a). Geometric descriptions of the obstacles and the robot.
(b). Initial and goal positions (configurations) of the robot.

Output: Collision-free continuous path connecting the initial and goal configurations.

3. Artificial Intelligence Technique

Artificial Intelligence techniques is basically making a machine intelligent alike natural intelligence in humans and other animals for achieving better efficacy as per the human’s requirement and needs. In department of computer science, Artificial intelligence technique is viewed as an intelligent agent, which identifies its arena and performs for maximizing the chance of achieving the goal successfully. In this paper grey wolf optimization technique is applied for optimizing the path of mobile robot, performing pick and place application.

3.1. Grey Wolf Optimization Technique

It is a novel approach proposed by Mirjalili et al. [11] in 2014. It is fairly simple, high flexibility in nature and has superior abilities to avoid local optima situations. This technique defines the hierarchy of leadership in group and system applied for hunting the prey for survival of wolves. A mathematical model of grey wolf optimization technique by Mirjalili et al. [11] is as follows:

\[ \bar{D} = |\bar{C} \cdot \bar{X}_p(t) - \bar{X}(t)| \]  
\[ \bar{X}(t+1) = \bar{X}_p(t) - \bar{A} \bar{D} \]  
\[ \bar{A} = 2\bar{a} \bar{n} - \bar{a} \]  
\[ \bar{C} = 2, \bar{r}_2 \]  

where, \( D \) is the displacement vector, current iteration is \( t \), \( X \) & \( X_p \) is wolf position and prey position respectively, \( C \& A \) are vector coefficients, \( a \& r \) are random vectors.

3.2. Implementation of Grey Wolf Artificial Intelligence Technique

Methodology Proposed:

- Understanding towards the approach, optimization and manipulator basics.
- Development of MATLAB® software code for grey wolf optimization (AI Technique).
- Simulation of mobile robot implementing artificial intelligence software in MATLAB®.
- Designing robotic arm and calculations of payload.
- Procurement of mobile manipulator hardware.
- Development and assembly of robotic arm and mobile robot, using its parts to form a mobile robot manipulator.
- Testing of mobile robot manipulator for pick and drop applications implemented with artificial intelligence in unknown static environment consisting of obstacles.

4. Robotic Arm Manipulator

An arm, mechanically reprogrammable having alike functions of human’s arm is termed as robotic arm manipulator, allowing motions like rotational and translational/linear [15]. The links forms a kinematic chain; terminus of the chain is termed as end effector. The foremost application of robotic arm manipulator is assembling of parts, painting, welding, material handling, palletizing, spinning, gripping etc.
4.1. Robotic Arm Design

The key factor for achieving higher efficacy of system is designing, designing of servo controlled robotics arm manipulator is done on designing software SOLIDWORKS® 2016 as shown in figure 2:

![Figure 2. Design of Servo Controlled Robotic Arm](image)

4.2. Robotic Arm Payload Calculations

To calculate the weight of an object \(W\) to be lifted or payload, using the concept of frictional force that acts perpendicular to the normal force as shown in figure 3, having servo motor torque in the range of 3-5N-cm, torque in the range of 1.5-5N-cm, factor of safety 2.0 (assumed), but practically we have friction of coefficient for wood in the range of 0.3-0.5 & Clamping Length=10 cm, then:

Normal Reaction Force \(N = \frac{1.5}{\left(2 \times 10\right)} = 1.5N\)

Assuming, friction of coefficient \(u = 0.3\) and weight of mechanism =45gm (includes weight of links and gears).

weight of object and weight of mechanism \((W_i) = \frac{2 \times u \times N}{g} = \frac{2 \times 0.3 \times 1.5}{9.81} = 95gm\)

So, the weight of object \(W = W_i - 45 = 50gm\).

5. Simulation Results

Simulation result for four different type of domain are shown in figures and optimized path length with elapsed time during performance with obstacle and without obstacle is incorporated in tables as shown below.

5.1. Navigation in Triangular Shaped Slab Obstacles:

In this simulation three triangular shaped slab obstacle are placed in such a way that the target is not directly visible from the robot’s start position. The robot has to start the navigation from bottom left corner (i.e. red square) to the target point (i.e. green square) in the chosen arena as shown in figure 4. The robot follows the wall following rule during locomotion and navigate in an approximately straight path till the end of the first slab and turn left to reach the target, while avoiding the obstacle.
5.2. Navigation in Conical Shaped Obstacles:

In second simulation a more complex arena has been selected as compared to first situation, the second simulation consist of conical shaped obstacles which are placed in such a way that they form a complex maze as shown in figure 5. The robot has to starts its journey from bottom left corner to the middle of the arena where the target is surrounded by many obstacles in its periphery. The robot had chosen the approximately near optimal path to reach the target by getting the systematic sensory information from the target and the obstacles without colliding with them.

5.3. Navigation in Cubical Shaped Obstacles:

In third simulation the robot has to cope up with dissimilar obstacles shapes and size as compared to the previous simulation. In this simulation the target is kept very close to one of the obstacles. The robot has to start its journey from bottom left corner of the chosen arena and has to reach the target point by following shortest path without colliding with the obstacles as shown in figure 6.
5.4. Navigation in S-Shaped Obstacles:

In the fourth simulation a S-shaped obstacle has been considered. The robot first moves diagonally to reach the wall of the S-shape boundary and follows the wall following rule to reach the target by navigating along the near optimal path as shown in figure 7 without colliding with the obstacles.

![Figure 7.2D & 3DS-Shaped Obstacle Simulation](image)

5.5. Path without obstacles:

| Domain | Elapsed Time (seconds) | Path length (pixels) | Result |
|--------|------------------------|----------------------|--------|
| 1      | 55.596324              | 59.5642              | Robot reached target point |

5.6. Path with obstacles:

| Domains | Elapsed Time (seconds) | Path Length (pixels*) | Maximum Heading Angle (degrees*) | Results |
|---------|------------------------|-----------------------|----------------------------------|---------|
| 1       | 117.926889             | 70.2210               | 85                               | Robot reached target point |
| 2       | 73.937600              | 70.9244               | 60                               | Robot reached target point |
| 3       | 122.856923             | 71.2924               | 35                               | Robot reached target point |
| 4       | 117.872310             | 70.3603               | 70                               | Robot reached target point |

6. Conclusion and Future Scope

The key objectives of this paper are modelling and simulation section of an autonomous mobile robot manipulator for pick and place applications; it clearly shows that its movement is precise, accurate, easy to program and user friendly in nature. The GWO technique was implemented for trying to successfully develop an optimized path for the robot to perform its desired task. The programming and interfacing of microcontroller has been mastered during the implementation of objectives. In real time applications we can use a superior grade of electronic components for high end performance. If compared with real-time situation, there are some deviation due to the fact that ambient conditions are continuously changing and the simulation results does not take dissipative factors like friction, heat developed in electronics etc. we solved this difficulty by using a small sensing radius but when compared to the real-time situations our assumptions created some erroneous results. Also the algorithm works in a way that, though it reaches to the optimum value really fastest one but can’t guarantee that the path taken by the robot to reach to the target point is best.
The paper basically depicts the concept of a robot interacting with a day-to-day environment and performing the desired operations. The research in the field can be taken to further levels like involving more complicated environment and more complicated dynamic conditions. This paper depicts basic level condition that involved the use of only simple sensors and basic microcontroller. With more investment in this field much broader and advance level research can be done. With finding advanced methods for robot path planning and making our robot smarter, we are moving towards a better future with artificial intelligence.

References

[1] Ram R V, Pathak PM, Junco SJ. Inverse kinematics of mobile manipulator using bidirectional particle swarm optimization by manipulator decoupling. Mech Mach Theory 2019;131:385–405.
[2] Raghavan M, Roth B. On the design of manipulator for applying wrenches. 1989.
[3] Shiller Z, Sundar S. Design of Robotic Manipulators for Optimal Dynamic Performance. 1991 IEEE Int Conf Robot Autom.
[4] Mermertas V. Mechanism and Machine Theory Optimal design of manipulator with four-bar mechanism. 2004;39:545–54.
[5] Harada K, Foissotte T, Tsai J, Nagata K, Yamanobe N, Nakamura A, et al. Pick and Place Planning for Dual-Arm Manipulators. 2012 IEEE Int Conf Robot Autom. 2012;2281–6.
[6] Begum NF, Vignesh P. Design and Implementation of Pick and Place Robot with Wireless Charging Application. 2015;4(2):2013–6.
[7] Patel S, Sobh T. Goal Directed Design of Serial Robotic Manipulators. 2014.
[8] Abotii A, Acharya S, Anand A, Chintale R, Ruiwale V. Design of a Prototype of a Pick and Place Robotic Arm. 2016;5(3):74–81.
[9] McTaggart M, Smith R, Erskine J, Grinover R, Gurman A, Hunn T. Mechanical Design of a Cartesian Manipulator for Warehouse Pick and Place. Mechanical Design of a Cartesian Manipulator for Warehouse Pick and Place. 2017;(September).
[10] Ropo OO, Adekunle OF, Adetola O. Design of a Pick and Place Serial Manipulator Design of a Pick and Place Serial Manipulator. 2018; IOP Conf. Series: Materials Science and Engineering 413 (2018) 012056.
[11] Mirjalili S, Mohammad S, Lewis A. Grey Wolf Optimizer. Adv Eng Software. 2014;69:46–61. http://dx.doi.org/10.1016/j.advengsoft.2013.12.007
[12] Zafar MN, Mohanta JC. Methodology for Path Planning and Optimization of Mobile Robots: A Review. Procedia Comput Sci. 2018;133:141–52.
[13] Parhi DR, Mohanta JC. Navigational control of several mobile robotic agents using Petri-potential-fuzzy hybrid controller. Appl Soft Comput J. 2011;11(4):3546–57.
[14] Mohanta JC, Parhi DR, Patel SK. Path planning strategy for autonomous mobile robot navigation using Petri-GA optimisation. Comput Electr Eng. 2011;37(6):1058–70.
[15] Dhote PK, Mohanta JC. Motion Analysis of Articulated Robotic Arm for. 2016;(4):20–4.