Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/BA08.20250091
Published: December 20, 2025

Analyzing Algorithms for Optimal Point-to-Point Trajectory Generation of a 3-DOF Manipulator

Md. Shirajul Islam
Department of Mechatronics Engineering
Khulna University of Engineering & Technology
Khulna-9203, Bangladesh
shirajulkuet@gmail.com

Asief Javed
Assistant Professor
Department of Mechatronics Engineering
Khulna University of Engineering & Technology
Khulna-9203, Bangladesh
asiefjaved@mte.kuet.ac.bd

Md. Helal-An-Nahiyan
Professor
Department of Mechanical Engineering
Khulna University of Engineering & Technology
Khulna-9203, Bangladesh
nahiyan@me.kuet.ac.bd

Abstract

The study focuses on the trajectory generation of a 3-degree-of-freedom (3-DOF) articulated manipulator, aiming to reduce the end-effector position inaccuracy at each intermediate point along the trajectory. A Genetic Algorithm (GA) combined with the Random Search Method (RSM), B-Spline, and Cubic Polynomial has been applied to a Simulink-designed 3-DOF articulated manipulator. Our outcomes show that the GA and RSM outperform other methods, gaining a point-to-point accuracy of 99.97%. For B-Spline and Cubic Polynomial, the accuracies are 75.40% and 79.59%, respectively. The hybridized framework introduces a search technique that gives optimum Genetic Algorithm parameters and these optimum parameters produce a trajectory with minimum path deviation compared to other algorithms. Hence, these results emphasize the excellence of the Genetic Algorithm and Random Search Method in providing an optimized trajectory.

Keywords
End-effector Position, Optimal Trajectory Generation, Genetic Algorithm, Genetic Algorithm's Parameters and Random Search Method.

1. Introduction
Manipulators are robotic arms that accomplish various industrial tasks such as welding, assembly, pick-and-place tasks, plastic injection molding, robotic surgery, and much more. To execute the above processes in a defined path within a certain duration, trajectory generation plays an important role. Trajectory generation is performed based on

some criteria, which are minimum path deviation, minimum time, minimum energy, etc. (Ata 2007). Trajectory can be classified based on the manipulator's position and orientation, i.e., Joint Space Trajectory and Cartesian Space Trajectory. The Joint Space trajectory follows the system arm's joints, while the Cartesian Space trajectory follows the end-effector's path. The other type of trajectory is based on the movement by the manipulator, i.e., Point-to-Point Trajectory and Continuous Trajectory. The manipulator moves through specific points in a point-to-point trajectory, while in a continuous trajectory, it moves the entire points of the defined path (Control - What are the differences between trajectory planning, trajectory tracking, path planning, path following and motion planning? 2019). B-Spline, Cubic Polynomial, Quantic Polynomial, Genetic Algorithm, Rapidly-exploring Random Trees, A* Algorithm, D* Algorithm, Gradient-based Trajectory Optimization, etc., are commonly used in trajectory generation for manipulators (Generate polynomial trajectories using B-splines - MATLAB bsplinepolytraj, Mathworks).

Trajectory generation algorithms are used to track the manipulator's trajectory, but their output may differ from user input. Many researchers have used a single trajectory generation algorithm to minimize the trajectory variation. Tuning of the algorithm's parameters is recommended for optimal results.

The proposed system uses a hybridized framework that combines an optimization algorithm and a search method and gives a generated trajectory that is very similar to the input trajectory with a mathematical explanation.

1.1 Objectives

The primary objective of this work is to develop a novel hybrid optimization framework that integrates an optimization technique and a search method to generate the optimum point-to-point trajectory, minimizing path deviation. The proposed approach is fully implemented in Simulink. The generated trajectory is then compared against the Simulink implementation of classical-level trajectory generation algorithms.

2. Literature Review

Trajectory plays a vital role in performing the task of industrial machinery such as Industrial robots, CNC machines, High Speed machine tools, etc. The generation of trajectory is very crucial for the execution of any specific task. Below, some literature is provided of the recent work in this respect.

Elbanhawi et al. (2015) applied B-Spline in non-holonomic vehicles. The trajectory produced by the B-Spline results in smooth traveling and turning, but this trajectory eliminates any abrupt changes. Moreover, the work enforces only smoothing the trajectory while not focusing on kinematic or dynamic constraints and not conducting full scale steering experiments for different vehicles on a variety of paths. Li (2018) used Cubic-Polynomial in a robot manipulator, generating a feasible trajectory, but the work could not balance the input parameters, i.e., linear velocity and steering angle velocity, with the output parameters, i.e., position and orientation. Abo-Hammour et al. (2011) used the Continuous Genetic Algorithm for the point-to-point trajectory generation in Cartesian space, an optimization algorithm that is inspired by natural selection, applying it to PUMA560 in a simulation environment, proving it superior to classical algorithms. However, they did not discuss its real-world implementation. Liu et al. (2022) applied Genetic Algorithm for the path planning in a Digital Twin Robot, where the trajectory was trained in a virtual environment, reducing error in trajectory movement while dealing with virtual environment data and the real environment, and addressing real-time execution and sensor accuracy issues. Wang et al. (2022) employed Genetic Algorithm with Model Predictive Control in autonomous vehicle path tracking, minimizing path deviation while addressing high computational complexity and manual parameter tuning. Esfandiar and Korayem (2016) applied the Harmony Search Method for the point-to-point trajectory generation for flexible manipulators, addressing non-linear dynamics, non-linear optimization problems, and optimization-related problems, but the real-world implementation of the proposed method was not discussed. Bagheri and Naseradinmousavi (2017) applied Genetic Algorithm and Gradient Based Search to the 7-DOF Baxter Manipulator to generate a smooth "S-shaped" trajectory. The approach generates a smooth trajectory while avoiding the jerky motions. Karahan et al. (2022), used Cubic spline, trigonometric spline, and a combination of cubic spline and 7th-order polynomial for the trajectory generation in joint space of a PUMA 560. Cubic spline achieved the best result for smooth and accurate trajectory generation. Moreover, performance was improved when a combination of cubic spline and 7th-order polynomial was applied in the PUMA 560. However, the work was not applied to other robot configurations.

The above literatures focus on the smoothness of the trajectory and does not give any mathematical explanation of their achievement. Through our work, the smoothness and the mathematical explanation can be drawn.

3. Trajectory Design Algorithms

This work has used three trajectory generation algorithms. In one algorithmic approach, GA and a search method called RSM have been merged to give the optimum result. Below is a short description of the three algorithmic approaches.

3.1 B-Spline Curve

A B-Spline curve (short for Basis-Spline) is a piecewise polynomial that represents complex curves and surfaces in robotics, automation, etc. It has a set of control points that facilitate the B- Spline curve to produce any complex shape of trajectory. To generate a trajectory by the B-Spline, first, the desired waypoints are set for the motion. Then the knot vector (which defines the parametric domain of the curve that influences each control point) is chosen. After that, the B- Spline curve creates a smooth trajectory. Finally, the shape of the trajectory can be adjusted just by adjusting the control points. The mathematical formula is defined as

$$X(t) = \Sigma \left(P_i * B_i, d(t)\right).$$

Where X(t) is the position of the curve at t, P_i are the control points in the desired trajectory, B_i,d(t) are the basis functions of degree d evaluated at parameter t, and Σ represents summation across all control points with non-zero values determined by the knot vector (Holliig and Horner 2013).

3.2 Cubic Polynomial

It is a third-degree polynomial, which generates a smooth and precise trajectory for the manipulator. For unseen activity, its small response time ensures safety. The mathematical formula is defined as

$$p(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3.$$

Where P(t) is the output of the function, which can be position, displacement, or any other quantity that varies with t(time), and a_0, a_1, a_2, and a_3 are the coefficients in real numbers (Sidobre and Desormeaux 2018).

3.3 Genetic Algorithm

Genetic Algorithm (GA) is an evolutionary algorithm that uses random populations to start the program and iteratively processes them to determine the optimal solution. This algorithm provides a random number of solutions rather than a single solution. Population Size, Max Generation, Crossover Probability, and Mutation Probability are the parameters of the GA. The proper selection of GA parameters plays a vital role in providing an optimal solution (Set Maximum Number of Generations and Stall Generations - MATLAB & Simulink).From Figure 1., it is noticed that the algorithm starts operation with an initializing population size, with the fitness value from the fitness function (which measures how good a solution is in GA), performs crossover and mutation operations. It stops, if the stopping criterion is met, and if not, then it resumes the fitness calculation. A selection process identifies the fittest individuals for reproduction. The algorithm randomly generates candidate solutions within defined constraints. New generations are formed using crossover and mutation to maintain diversity. Elitism may be used to retain top solutions from each generation. Across iterations, the population gradually evolves towards improved solutions as the algorithm converges.

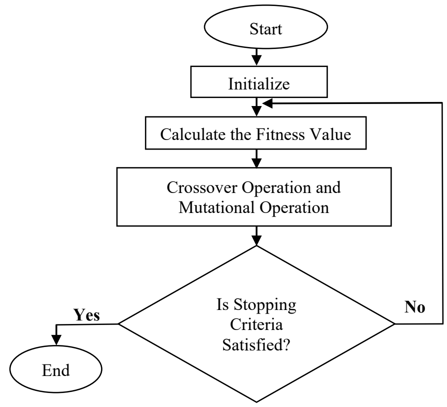


Figure 1. Flow Chart of Genetic Algorithm Working Procedure

3.4 Random Search Method

Random Search Method (RSM) is a search technique that randomly chooses hyperparameters and assesses their performance. Before performing GA, GA parameters need to be defined. Selection of the values of GA parameters is time-consuming, and occasionally, the desired outcomes are not achievable. To come out of this problem, the random search method searches the GA parameters in a three-dimensional space within a defined range of GA parameters given by the researchers (Gaille 2017).

The Random Search method begins by generating a random solution within a defined range, ensuring that each attempt is unconstrained by prior choices. This initial solution is then evaluated using the fitness function, which measures the quality of the proposed answer to the optimization problem. After evaluation, a decision is made as to whether this solution is better than any previously discovered candidates, updating the best found solution if necessary. If the current solution is not superior, the process repeats, continually searching for improvement through randomized exploration. The algorithm checks whether the maximum number of iterations has been reached to determine if the search should conclude; if not, it persists in trying new solutions from scratch. Upon reaching the iteration limit, the algorithm examines whether an acceptable solution has been found; if so, the search ends; otherwise, the whole process is restarted to continue seeking better answers, as depicted in Figure 2.

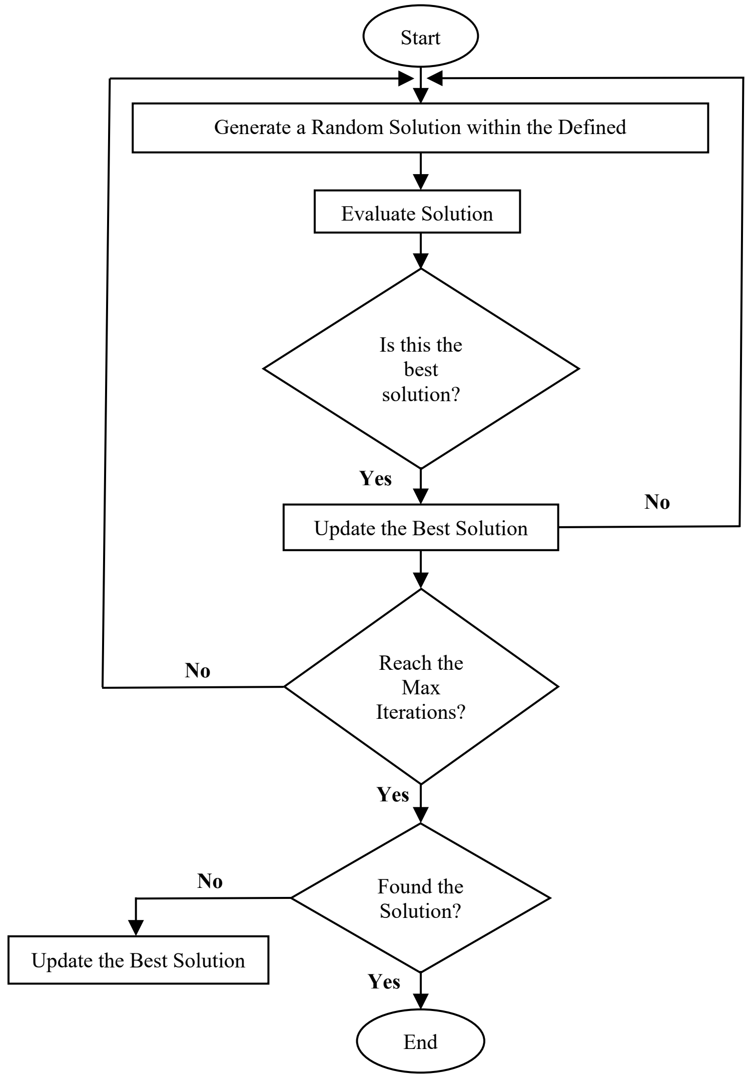
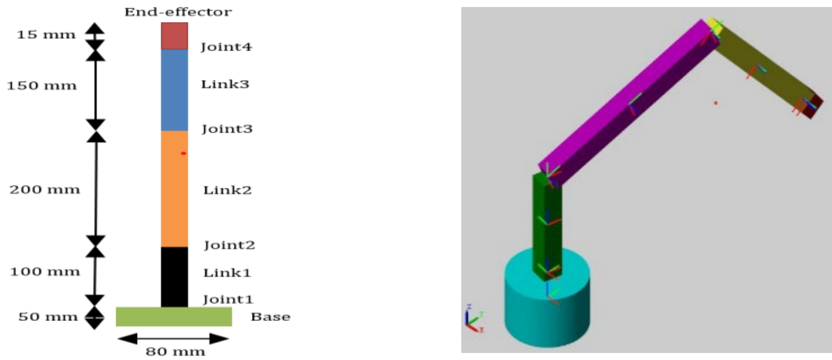


Figure 2. Flow Chart of Random Search Method Working Procedure

4. Implementation of Trajectory Design Algorithms

4.1 Design

The designed 3-DOF articulated manipulator's first three joints are Revolute joints, where Joint2 and Joint3 are perpendicular to the Joint 1, and Joint 4 is a Butt Joint, and the blue indicator is the Z-axis direction, as illustrated in Figure 3(b). The Joint2 and Joint3 have a joint limit of -90 degrees to +90 degrees, and Joint1 has a limit of -180 degrees to +180 degrees. From its dimension and orientation, the axis limit was calculated. X-axis, Y-axis, and Z-axis limits are 365, 365, and (151-500) mm, respectively, as Joint2 is 150 mm above the base and random waypoints were selected within these limits, as depicted in Figure 3(a).



(a) (b)

Figure 3. Designed 3-DOF Articulated Manipulator. (a) Dimension Parameter (b) Simulink Model and Dimensions

4.2 B-Spline Curve or Cubic Polynomial Trajectory Generation Process in Simulink

Figure 4. shows the process involves selecting desired waypoints, using the Polynomial Trajectory Generation block in Simulink, and setting time interval and velocity boundary conditions. The output of the Polynomial Trajectory Generation block, which are Joint Angle (θ), Joint Velocity (ω), and Joint Acceleration (α), are provided to the Inverse Dynamics block, which generates Torque (τ) for a 3-DOF articulated manipulator to follow the generated trajectory.

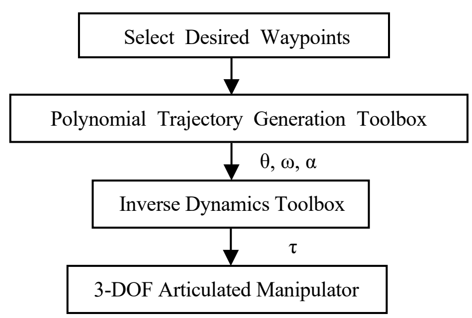


Figure 4. Flow Chart of the Trajectory Generation Process Using Cubic Polynomial or B-Spline Curve

4.3 Genetic Algorithm and Random Search Method Trajectory Generation Process in Simulink

Random waypoints are selected from the workspace limit, and provided to the MATLAB code combined GA and RSM. Optimized waypoints are provided to the Inverse-Kinematics block, and Joint Angle (θ), Joint Velocity (ω), and Joint Acceleration (α) are obtained. These are saved in the To Workspace Block and passed to the Inverse Dynamics block using the From Workspace block. The torque (τ) is generated for the 3-DOF articulated manipulator to follow the generated trajectory, as illustrated in Figure 5.

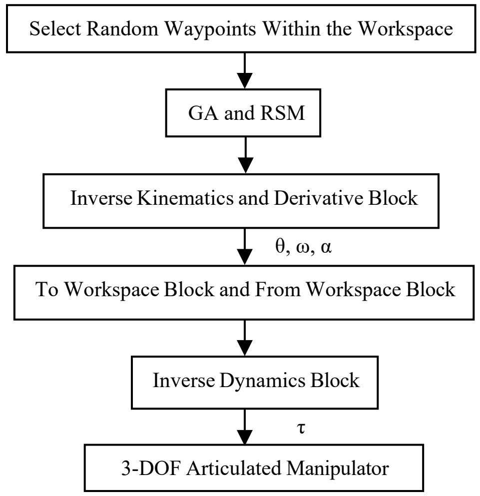


Figure 5. Flow Chart of the Trajectory Generation Process Using GA and RSM

5. Simulation Results and Discussion

5.1 Optimal GA Parameters Calculation

The Genetic Algorithm Parameters (GAP), which are Population Size (PS), Max Generation (MG), Crossover Probability (CP), and Mutational Probability (MP), play a vital role in getting the optimum waypoints.
Table 1. and Table 2. show the random value of GAP fed to the MATLAB code of the Genetic Algorithm and the input data and output data of the MATLAB code of the Genetic Algorithm (GA), respectively. Here, all the values are represented in millimeters. Here the Target Points (TP) are the desired waypoints, and the Given Points (GP) are the randomly provided input points to the MATLAB code of GA. The Optimized Points (OP) are obtained from GA MATLAB code. From Table 2, it is seen that OP is not equal to the TP due to not providing the optimum GAP.

Table 1. GAP for GA Test Result

Genetic Algorithm Parameter	**Value**
PS	1
MG	30
CP	0.2
MP	0.016

Table 2. GA Test Result

Input	**GP (mm)**	220, 90, 330	210, 110, 300.10	30.06, 309.90, 60.05	10.09, 333.10, 56.13
	TP (mm)	220, 90, 330	210, 110, 300	30, 310, 60	10, 330, 30
Output	**OP (mm)**	220.10, 90.30, 330.20	210, 109.9, 300.5	30.10, 310.10, 60.10	10, 330, 30

So, the Random Search Method (RSM) is introduced with GA to get the optimum GAP. Here, the limit value of Genetic Algorithm Parameters is fed to the combined MATLAB code of GA and RSM. RSM provides the optimum value of GAP from the limit value of Genetic Algorithm Parameters. Therefore, the GA produces the OP that are near or equal to the TP.

The data in Table 3. and input data of Table 4. are fed to the MATLAB code. From Table 4 and Table 5, it is found that the OP are nearly equal to the TP, and the required value of GAP are shown in Table 5. for the above findings.

Table 3. Limit Value of Genetic Algorithm Parameters for the combined MATLAB code of GA and RSM

Genetic Algorithm Parameter	Value
PS	[4,150]
MG	[50,1000]
CP	[0.80, 0.95]
MP	[0.010, 0.049]

Table 4. GA and RSM Test Result

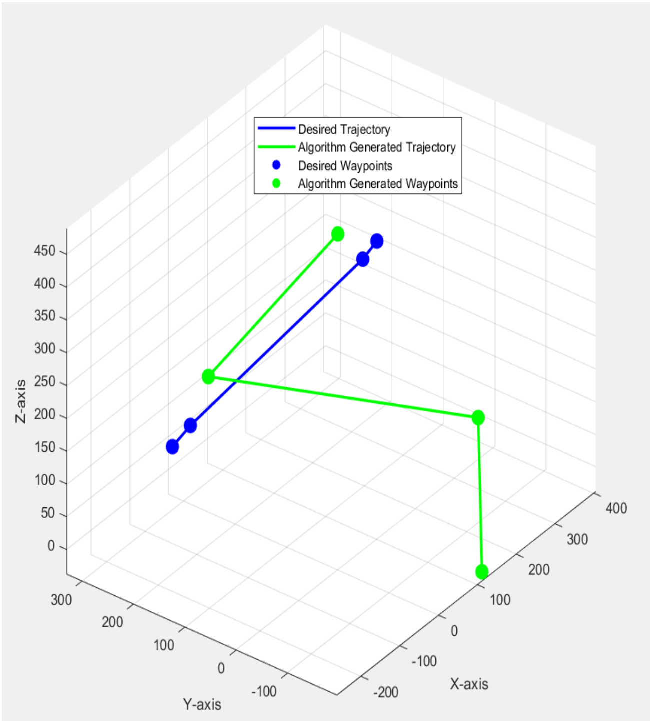
Input	**GP (mm)**	220, 90, 330	210, 110, 300.10	30.06, 309.90, 60.05	10.09, 333.10, 56.13
	TP (mm)	220, 90, 330	210, 110, 300	30, 310, 60	10, 330, 30
Output	**OP (mm)**	220, 90, 330	210, 110, 300.10	30, 310, 60	10, 330, 30

Table 5. Optimized Genetic Algorithm Parameters

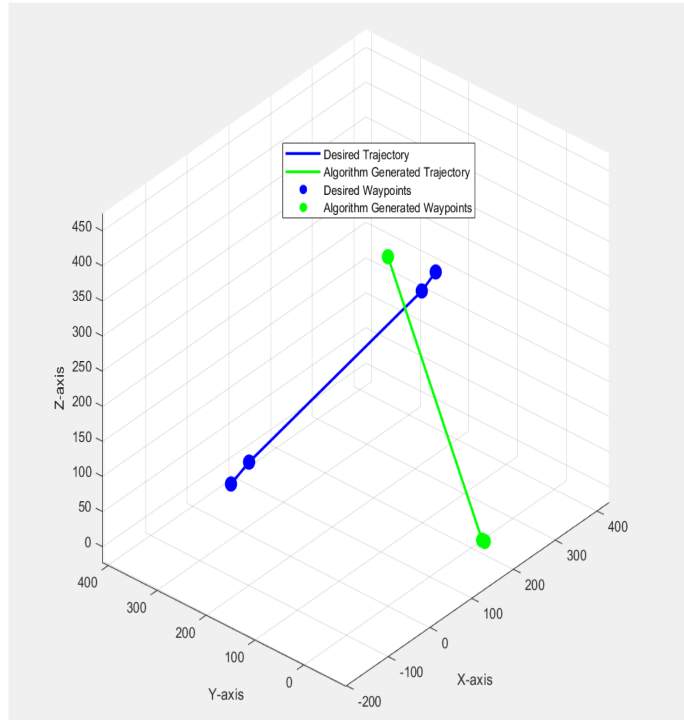
Genetic Algorithm Parameter	Value
PS	100
MG	300
CP	0.82
MP	0.049

5.2 Visualization of Generated Trajectory in Three-Dimensional Space

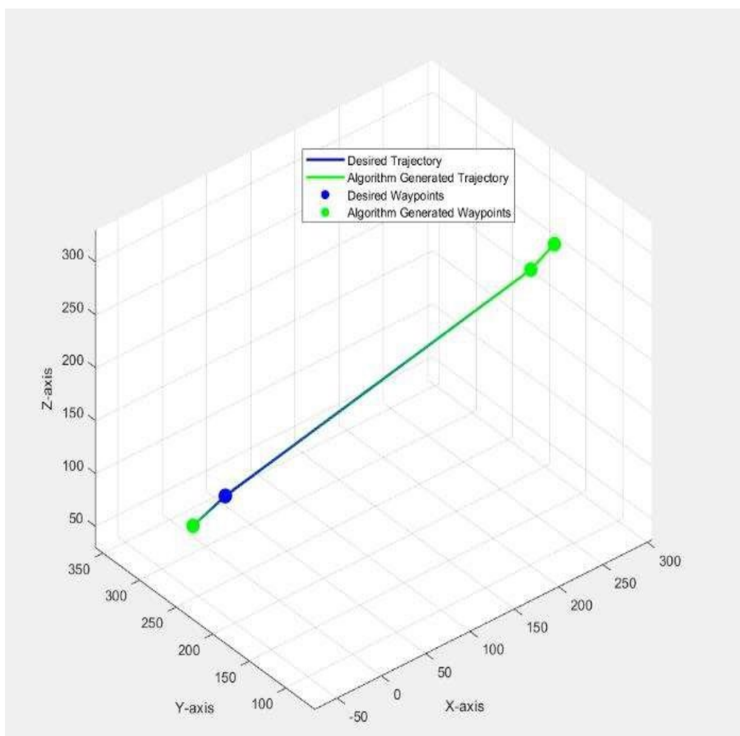
The three-dimensional trajectories are represented in millimeters. From Figures 6(a), 6(b) and 6(C), the desired-trajectory and algorithm-generated trajectory for GA and RSM are similar, while deviation exists for the B-Spline Curve and Cubic Polynomial due to a lack of search techniques such as RSM, no hybridized approaches, no fine-tuning of the mathematical parameters, no introducing of additional control points (in the case of B-Spline Curve), and not adjusting boundary conditions and continuity requirements (in the case of Cubic Polynomial).



(a)



(b)



(c)

Figure 6. Comparison between Desired-Trajectory and Algorithm-Generated Trajectory. (a) B-Spline Curve. (b) Cubic Polynomial. (c) Genetic Algorithm and Random Search Method.

Table 6. Point-to-Point Accuracy Between Input Position (P_1) and Operational Position (P_2) for Applying Algorithms

Algorithm	Point-to-Point Accuracy (%)
GA and RSM	99.97
B-Spline	75.40
Cubic Polynomial	79.59

The work is compared with Z. S. Abo-Hammour et al., where the authors applied the Continuous Genetic Algorithm to the PUMA 560 manipulator, where the maximum path deviation in case of absolute error was 0.7 mm (Abo-Hammour et al. 2011), but the GA and RSM shows an absolute error of 0.01mm for the designed system. So, the designed 3-DOF articulated manipulator shows excellent results in the case of minimum path deviation.

6. Conclusion

The study analyzed the performance of different trajectory generation algorithms with the GA and RSM on the Simscape-designed 3-DOF articulated manipulator. The GA and RSM produces the optimum trajectory, which is very similar to the given trajectory. Therefore, the work has successfully achieved its goal. But the B-Spline curve and the Cubic Polynomial show less performance than the GA and RSM. In the future, tunning of the mathematical parameters of the B-Spline curve and the Cubic Polynomial will be introduced to improve its trajectory generation performance. Moreover, the Reinforcement Learning Toolbox, and Navigation Toolbox will be incorporated with the designed manipulator to work in the obstacle environment using the sound facility of reinforcement learning algorithms.

References

Abo-Hammour, Z. S., Continuous Genetic Algorithms for Collision-Free Cartesian Path Planning of Robot
advantages-and-disadvantages-of-random-sampling, June 16, 2017.
algorithm, *Intelligent Service Robotics*, vol. 15, no. 5, pp. 627–648, September 6, 2022,
and motion planning? - Robotics Stack Exchange,
Ata, A., Optimal Trajectory Planning of Manipulators: A Review, *Journal of Engineering Science and*
Autonomous Vehicle, *Mathematical Problems in Engineering*, vol. 2022, no. 1, p. 4661401, September 20, 2022,
Available: https://robotics.stackexchange.com/questions/18761/what-are-the-differences-between-trajectory-
Avaliable: https://robotcanwrite.com/wpcontent/uploads/2018/01/Yuliang-Trajectory-Generation-Using-Cubic-Curvature- Polynomials.pdf, April 18, 2017.
Avaliable: https://www.mathworks.com/help/gads/setting-the-maximum-number-of-generations.html.
Avaliable: https://www.mathworks.com/help/robotics/ref/bsplinepolytraj.html.
Bagheri, M. and Naseradinmousavi, P., Novel analytical and experimental trajectory optimization of a 7-DOF
baxter robot: global design sensitivity and step size analyses, *International Journal of Advanced Manufacturing*
Control - What are the differences between trajectory planning, trajectory tracking, path planning, path following
Deformation by Using Harmony Search Method, *Journal of Theoretical and Applied Mechanics*, vol.
doi: 10.1007/s11370-022-00440-8.
doi: 10.1155/2022/4661401.
Elbanhawi, M., Continuous Path Smoothing for Car-Like Robots Using B- Spline Curves, *Journal of*
Esfandiar, H. and Korayem, M. H., Optimal Point to Point Path Planning of Flexible Manipulator Under Large
Gaille, L., 17 Advantages and Disadvantages of Random Sampling – Vittana.org, Avaliable: https://vittana.org/17-
Generate polynomial trajectories using B-splines - MATLAB bsplinepolytraj, Mathworks,
Holliig, K. and Horner, J., *Approximation and Modeling with B-Splines,* Saim, 2013.
Intelligent & Robotic Systems, vol. 95, pp. 851–869, October 25, 2018, doi: 10.1007/s10846-018-0936-z.
Intelligent &Robotic Systems, vol. 80, no. s1, pp. 23–56, January 8, 2015, doi: 10.1007/S10846-014-0172-0.
Karahan, O., Optimal trajectory generation in joint space for 6R industrial serial robots using cuckoo search
Li, Y., Trajectory Generation Using Cubic Curvature Polynomials,
Liu, X., Genetic Algorithm-Based Trajectory Optimization for Digital Twin Robots, *Front Bioeng Biotechnol*, vol. 9, p. 793782, January 10, 2022, doi: 10.3389/fbioe.2021.793782.

Manipulators, *International Journal of Advanced Robotic Systems*, vol. 8, no. 6, December, 2011, doi: planning-trajectory-tracking-path, March 8, 2019.
Set Maximum Number of Generations and Stall Generations - MATLAB & Simulink,
Sidobre, D. and Desormeaux, K., Smooth Cubic Polynomial Trajectories for Human-Robot Interactions, *Journal of Technology,* vol. 2, no. 1, pp. 32–54, April 2007, https://www.researchgate.com/publication/49596198. *Technology*, vol. 93, pp. 4153-4167, August 03, 2017, doi: 10.1007/s00170-017-0877-x.
Wang, M., Path Tracking Method Based on Model Predictive Control and Genetic Algorithm for 10.5772/50902. 54, no. 1, pp. 179–193, 2016, doi: 10.15632/JTAM-PL.54.1.179.

Biographies

Md. Shirajul Islam is currently purchasing an MS in Mechatronics Engineering from Khulna University of Engineering & Technology, Khulna. He received his BS in the same department from the same institution. Now he serves as a Teaching Assistant in the same place where he is purchasing his post-graduation degree and also serving as adjunct faculty in the Mechanical Engineering Department at Imperial College of Engineering (Affiliated with University of Rajshahi), Khulna. His interest lies in Robotics, Control System, Automotive System, Bio-embedded System, and Image Processing.

Dr. Asief Javed is working as an Assistant Professor in the Department of Mechatronics Engineering. He joined Khulna University of Engineering & Technology (KUET) in 2020. He completed his PhD on vibration reduction in magnetic suspension system in 2018 at Saitama University, Japan. He completed his Master's from the same university in 2015. He was awarded MEXT scholarship to pursue both of the postgraduate programs. During his master's and PhD program, he worked at the Control Engineering Lab (Seigyo Kenkyushitsu). He finished his B.Sc. in Mechanical Engineering from Rajshahi University of Engineering & Technology (RUET) in 2012. His research interest include implementation of control in different systems, vibration suppression, robotics, etc.

Dr. Md. Helal-An-Nahiyan is working as a Professor in the Department of Mechanical Engineering and also serving as a Department Head of Mechatronics Engineering from September, 2025 to the present. He joined Khulna University of Engineering & Technology (KUET) in 2009. He completed his PhD in Control Engineering in 2018 from Saitama University, Japan. He completed his MS from Korea Advanced Institute of Science & Technology (KAIST), South Korea. He finished his B.Sc. in Mechanical Engineering from Bangladesh University of Engineering & Technology (BUET). His research interest include control, robotics, mechatronics, etc.