TRAJECTORY PLANNING USING GENETIC ALGORITHM FOR THREE JOINTS ROBOT MANIPULATOR

N. Andelić¹, S. Blažević¹ and Z. Car¹

¹ Faculty of Engineering Rijeka, Vukovarska 58, 51000 Rijeka, Croatia
Email: nandelic@riteh.hr, sbazevic@riteh.hr, car@riteh.hr

Keywords: genetic algorithm, path planning algorithm, robot manipulator.

Abstract: In this paper, the dynamic governing equations are obtained for three joints robot manipulator. Using obtained governing equations, the trajectory of manipulator is planned using genetic algorithm. The resulting trajectory is then compared with conventional trajectory planning algorithm.

Introduction

Today robot manipulators are widely used in manufacturing. They are automated, programmable and capable of movement in two or more axes. The typical application of these robots in manufacturing include: welding, painting, assembly, pick and place, packaging and labeling, product inspection and testing. All these operations are performed with high endurance, precision and speed. Before they are put in exploitation robot manipulators must be programmed or to be more precise their trajectory must be planned. There are two main approaches in trajectory planning depending on performed operation and these approaches are: point to point or continuous trajectory planning. For example, if operation of robot manipulator is pick and place then the point to point trajectory is more suitable since we’re interested only in initial and final position of the robot. If, however operation is welding or painting then the continuous path planning trajectory approach is suitable since robot must equally apply paint or equally apply welding material. In order to optimize trajectory of robot manipulator, artificial intelligence algorithm such as genetic algorithm could be applied to minimize energy applied to execution of trajectory while achieving the same results. In this paper the governing equations for planar three joint manipulator are derived and trajectory was planned using point to point approach. Based on these results the genetic algorithm was developed in order to optimize trajectory of robot manipulator.

Approach in trajectory planning

In this paper each joint trajectory is assumed to be time polynomial of 4th degree. Angle between links \( q(t) \) is fourth order polynomial and can be expressed in the form:

\[
q(t) = at^4 + bt^3 + ct^2 + dt + e
\]

where \( a, b, c \) and \( d \) are parameters based on boundary conditions. Angle \( q(t) \) is positive in counterclockwise direction.

Lagrangian-Euler dynamics algorithm

In Fig. 1. the planar three joint robot manipulator is shown. The governing equations of planar three joint robot manipulator are derived using Lagrange-Euler algorithm [1]. The procedure of L-E algorithm is given in Table 1.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Using direct/forward kinematics procedure (Denavit – Hartenberg procedure) assign the coordinate system to joints of robot manipulator for base up to end-effector and obtain the matrix of robot manipulator.</td>
</tr>
<tr>
<td>2.</td>
<td>Set values ( T_{i-1}^i ), ( i = 1 \ldots , n ), ( D(q) = 0 )</td>
</tr>
<tr>
<td>3.</td>
<td>Calculate homogeneous coordinates of center of mass of ( i )-th member in relation to the coordinate system ( L_i ).</td>
</tr>
<tr>
<td>4.</td>
<td>Determine the inertia tensor for center of mass of ( i )-th member.</td>
</tr>
<tr>
<td>5.</td>
<td>Calculate vector: ( \mathbf{z}^{i-1}(q) = R^{-1}_i(q) \mathbf{h}<em>i \cdot I^i ), calculate matrix of complex homogenous transformation: ( T(q) = T</em>{i-1}^{-1}(q) R^{-1}_i(q) ), find coordinates of mass center for ( i )-th member in relation to the coordinate base system: ( c_i(q) = ).</td>
</tr>
</tbody>
</table>
\[
\beta_i(q) = R_i(q)[R_i(q)\mathbf{D}(q)]^T
\]

6. Determine the Jacobian matrix of i-th member $f_i(q)$ using the expression:

\[
\mathbf{J}(q) = \begin{bmatrix}
\frac{\partial \mathbf{c}(q)}{\partial \mathbf{q}}, & \ldots, & \frac{\partial \mathbf{c}(q)}{\partial \mathbf{q}_n}, & 0 & \ldots & 0
\end{bmatrix}
\]

7. Divide the Jacobian matrix on submatrices A and B and determine an inertia tensor of robot manipulator $\mathbf{D}(q)$ using following mathematical expression:

\[
\mathbf{D}(q) = \sum_{k=1}^{n} ([\mathbf{A}(q)]^T \mathbf{m}_k \mathbf{A}(q) + [\mathbf{B}(q)]^T \mathbf{n}_k \mathbf{n}(q) \mathbf{B}(q))
\]

8. Increase the value of the variable $i$ by 1 and if the relation $i < n$ is “True” go back to 3rd step, if its “False” set value of $i = 1$ and go to the step 9.

9. Determine velocity linking matrix, gravitational action vector and friction force.

10. 1st Lagrange Euler equation:

\[
\tau_i = \sum_{j=1}^{n} \left[ \mathbf{m}_{ij}(q) \mathbf{q}_{ij} \right] + \sum_{j=1}^{n} \left[ \mathbf{r}_{ij}(q) \mathbf{q}_{ij} \right] + \mathbf{b}_i(q) + \mathbf{n}_i(q)
\]

11. Raise value of $i$ for 1 and if the relation $i < n$ is True go back to the 9th step if however, the relation is False the procedure is completed.

Based on Denavit-Hartenberg procedure the matrix of robot manipulator is generated which is represented by Eq. (1).

\[
\mathbf{T}_2(q) = \begin{bmatrix}
\mathbf{C}_{123} & -\mathbf{S}_{123} & 0 & \frac{a_1}{d_4} \\
\mathbf{S}_{123} & \mathbf{C}_{123} & 0 & \frac{a_2}{d_4} \\
0 & 0 & 1 & d_4 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

With utilization of algorithm defined in Table 1 the following dynamic equation for planar three joint robot manipulator were obtained. The dynamic equations are given below:

\[
\tau_1 = (K_1 + K_2 + K_3)q_1 + (K_2 + K_3)q_1^2 - 2K_3S_0 q_1 q_2 - g_0(K_1 + K_3)q_3 - b_1^2 q_1 + \text{sgn}(q_1) \left[ b_1^2 + (b_1^2 - b_2^2) \exp \left( \frac{-\pi q_1}{\alpha} \right) \right]
\]

\[
\tau_2 = (K_2 + K_3)q_1 + K_2q_2 + K_3q_1q_2 + g_0K_1C_1 + b_2^2 q_2 + \text{sgn}(q_2) \left[ b_2^2 + (b_1^2 - b_2^2) \exp \left( \frac{-\pi q_1}{\alpha} \right) \right]
\]

\[
\tau_3 = b_3(\epsilon_3) = b_3^2 q_1 + \text{sgn}(q_3) \left[ b_3^2 + (b_1^2 - b_2^2) \exp \left( \frac{-\pi q_3}{\alpha} \right) \right]
\]

Genetic algorithm

Genetic algorithm is artificial intelligence tool inspired by the process of natural selection that belongs to the larger class of evolutionary algorithms [2,3]. They are commonly used to generate high-quality solutions in optimization and search problems by relying on bio-inspired operations such as mutation, crossover and selection.

Given the initial and final position of a robot end-effector, the problem of finding optimal path is solved using genetic algorithm. The problem consists of finding specific path that requires the least amount of torque. It is evident that the end effector, moving between any two specific end points, can follow variety of paths. All those paths require different amounts of torque depending on the distance covered, the velocity and acceleration achieved, and the payload carried.

Genetic algorithm consists of several steps and these steps are: initial population, fitness function, selection, crossover and mutation, and termination condition. At the start of genetic algorithm, the initial population must be created. Since each joint trajectory is assumed to be 4th order polynomial, with application of boundary condition only first coefficient in each polynomial function must be determined. So each member of the initial population consists of two genes and that is a1 and a2. The third joint has only one degree of freedom and can be neglected from the optimization problem. After the initial population is created the fitness function must be applied. The fitness function can be measure of solution quality and in this case is given in the form of:

\[
ff = \Sigma \left[ \sqrt{\tau_i^2 + \tau_j^2} \right]
\]

Based on quality of solution the selection process must be performed. In selection process members of population are selected for crossover and mutation. Mutation operation is performed on single population member. In this operation the random genes from population member are changed with randomly selected genes from gene set (pool). In crossover operation two or more population members are
required and the genes from these members create new population members. Since the number of population members is constant the members that give lowest value of fitness function will be thrown from the population.

**Results**

For the simulation the starting position of robot manipulator in terms of angles for joint 1 and 2 is 0 radians and for final position is \( \frac{\pi}{2} \) radians respectively. The simulation was performed with 1000 population members and with 1000 iterations. Fig.2 is graphical representation of best solution versus number of iterations. As seen from Fig. 2 the value of best solution (fitness function) drops below 35 after just 200 iterations. Fig. 3 represents the change of Torque 1 and 2 versus time. As seen from Fig. 3 the Torque 1 and 2 are minimized during trajectory execution.

![Graph of Best Solution vs Number of Iterations](image1)

**Fig. 2 Best solution versus number of iterations**

![Graph of Torque vs Time](image2)

**Fig. 3 Torque versus time**

**Conclusion**

In this paper the governing equation of robot manipulator were obtained using L-E method. Based on these equations the genetic algorithm was implemented in order to optimize the trajectory of robot manipulator in terms of torque minimization. The results are promising for this fitness function. Further development and testing of genetic algorithm is required such as fitness function formula, crossover and mutation. Beside genetic algorithm, the possibility of implementing other algorithms such as different types of evolutionary algorithms or simulated annealing should be investigated.

**Acknowledgment**

This study is supported by University of Rijeka within Development of Cloud Manufacturing intelligent systems for controlling, managing and automating the production process project.

**References**

