USING GENETIC ALGORITHMS IN INDUSTRIAL ROBOT OPTIMAL TRAJECTORY GENERATION

Ing. Pavol Bezák, PhD.
Institute of Applied Informatics,
Automation and Mathematics MTF STU
Hajdúčzyho 1, 917 24 Trnava
e-mail: pavol.bezak@stuba.sk

Abstract
In this paper I deal with algorithms of path and trajectory planning and optimization of industrial robot motion trajectory using genetic algorithms. This problem is not completely solved because of its variability, complexity and growing computational complexity with the growing number of robot degrees of freedom. It is possible to generate optimal motion trajectory by various means - numeric and also by newer approaches, for example by evolution algorithms and specially genetic algorithms.

In this paper I present the approach of solving the dynamic optimization process of generating trajectory by genetic algorithms.

The contribution presents the solution and implementation in supporting software and also experimental results verification.

Key words: genetic algorithm, industrial robot, optimal trajectory

INTRODUCTION
Optimization problems occur in every area of human activity. Optimization problem arises in a situation where you must choose a solution. Of course we try to choose the solution that is best for us. Optimization problem can usually be solved by optimization methods. So that we can formulate the optimization problem mathematically, it is necessary to establish a mathematical model of the situation. Model of the real situation is always simplified, mathematically workable model does not describe accurately the reality of the situation and vice versa, model really close to reality may not be mathematically workable.

Finding the optimal solution using mathematical modeling is actually a search for extreme function by which the system is mathematically described. Specifically, the search for local extremes, which either are or are not also global extremes.

Evolutionary algorithms are among modern methods of optimization systems. Evolutionary mechanisms that were examined by nature long-term development can be very successfully applied to the technical problems, especially those that are complex or difficult described by mathematical methods.

Optimal industrial robot movement in certain trajectory requires the optimization criterion to define a trajectory generated adjusted so as to minimize the time the locomotive operation, and shall not exceed the maximum speed and acceleration that we do not reduce the life of the drive in view of the large joints and large load inertia [3].

THEORY
Genetic algorithm is stochastic adaptive algorithm including following operators and parameters:

\[ GA = (N, P, f, \Theta, \Omega, \Psi, \tau) \]

where \( P \) is population of \( N \) elements (individuals), \( P = \{S_1, S_2,..., S_N\} \). Every member \( S_i \), \( i = 1,..., N \) is string (or set) of integers of length \( n \), representing problem solution, it means

\[ S_i \in Z^n. \]

\( f \) is fitness function, that assigns each element a positive real number:

\[ f = S_i \rightarrow R^+; \ i = 1,...,N \]

\( \Theta \) is selection operator of parent elements, which selects the elements of \( P \):

\[ \Theta : P \rightarrow \{P_1,...,P_u\} \]

\( \Omega \) is a set of genetic operators, including the crossover operator \( \Omega_c \), mutation operator \( \Omega_m \) and any other problem-specific operators that together generate all the descendants (offsprings, children) from \( u \) parents:

\[ \Omega = \{\Omega_c, \Omega_m, \ldots\} : \{P_1,...,P_u\} \rightarrow \{O_1,...,O_v\} \]

\( \Psi \) is deletion operator, that deletes \( v \) selected elements in actual population \( P \). Then \( v \) children are added to the new population \( P(t+1) \):

\[ P(t+1) = P(t) - \Psi(P(t)) + \{O_1,...,O_v\} \]

\( \tau \) is ending criteria:

\[ \tau : P(t) \rightarrow \{true,false\} \]

Selection operator of parents \( \Theta \) and genetic operator \( \Omega \) are probabilistic, reduction operator \( \Psi \) can be deterministic [1].
PROBLEM REPRESENTATION

Let us have double-arm planar robotic manipulator to be moved from the starting position at tight end position. The aim is to find the optimal path that the manipulator moves in the least time.

To simplify the problem we establish the following assumptions:

1. The robot is considered a double-arm planar manipulator.
2. Individual kinematic limits and allowed trajectories of points should be set and allowed points of trajectory should be from the robot possible workplace.
3. The overall trajectory of the manipulator consists of intermediate points, nodes, which are obtained from genetic algorithm and are processed at regular intervals.
4. For each path between nodes is used approximation using spline curves.
5. It is assumed that the end effector of the manipulator begins its motion from zero velocity and ends at zero speed, with stops at intermediate nodes.

FORMULATION OF GENETIC TRAJECTORY PLANNING

Simple genetic algorithm usually uses binary coding to represent parameters. Let’s have a set of real parameters of number $N_p$, where $x = \{x_1, x_2, \ldots, x_{N_p}\}$ (it is defined as $x = \{U_{i=1}^{N_p} x_i\}$), it is coded to binary string $\hat{x} (=\{U_{i=1}^{N_p} \hat{x}_i\})$, that is also called chromosome. Every real parameter $x_i$ that has maximal boundary value $x_i^{U}$ and minimal boundary value $x_i^{L}$, is coded to binary string $\hat{x}_i$ using binary length $\hat{L}_i$.

For manipulator on Fig. 1 it is needed to optimize nine parameters in the form of the following chromosome:

$$\{q_1, q_2, q_3, q_4, \dot{q}_1, \dot{q}_2, \dot{q}_3, t_1, t_2\}$$

where $q_i$ are angles of rotation of links in transition points, $\dot{q}_i$ are velocities of joint $i$, $q_4$ is overall angle of the end configuration of manipulator, that is equal to the sum of angles, $t_1$ is the time from initial position to transition position, $t_2$ is the time from the transition position to the end position.

ALGORITHM OF THE PROCEDURE

Procedure of genetic trajectory planning for robotic manipulator is expressed as follows [2]:

$$\hat{x} = \{\hat{x}_i | i = 1, 2, \ldots, N_p\} : \text{trajectory chromosome}$$

$$k = 1, 2, \ldots, N_{pop} : \text{index of the } k \text{ individual of the population}$$

$$\hat{X} = \{\hat{x}^k | k = 1, 2, \ldots, N_{pop}\} : \text{population chromosome}$$

$$\text{Fit} = \{\text{fit}^k | k = 1, 2, \ldots, N_{pop}\} : \text{population fitness vector}$$

$$\Omega_s : \hat{X} \times R^{N_{pop}} \mapsto \hat{X} : \text{selection operator}$$

$$\Omega_c : \hat{X} \mapsto \hat{X} : \text{crossover operator}$$

$$\Omega_m : \hat{X} \mapsto \hat{X} : \text{mutation operator}$$

The initial conditions:

**robot:** link parameters $Q^L, Q^U, T^U$ : upper and lower boundaries of restrictive conditions

**GA:** $p_c, p_m \in [0, 1]$ : probability of crossover, mutation

$[\hat{x}^L, \hat{x}^U]$ : boundary coding for chromosome $\hat{x}$

$\hat{L} = \{\hat{L}_i | i = 1, 2, \ldots, N_p\}$ : coding lengths $\hat{x}$

$N_{pop}$ : size of population,

$N_{gen}$ : maximum number of generations

**other:** $dt$ : sampling period, $N$ : path part time

**input:**

$q_{io}, q_f$ : starting and ending position of the robot

**stop:** $\Delta : \hat{X} \mapsto \{\text{Yes, No}\}$; stop when the number of generation is $N_{gen}$

**output:**

elite chain trajectory satisfying stopping criteria
EXPERIMENTS

Trajectory optimization

Let me describe testing mechanism parameters. The lengths of links were $l_1 = 1$ m, $l_2 = 1$ m and $l_3 = 0.5$ m. The mass $m_1 = 1$ kg, $m_2 = 1$ kg a $m_3 = 0.5$ kg. Maximum permitted torques for actuators 1, 2 a 3 are 45Nm, 20Nm and 5Nm. Velocity and acceleration in the initial and final position is zero.

For genetic algorithm the following parameters were set: crossover probability $P_c = 0.8$ for chromosome (function $P_c$ defines, how often the chromosome will be crossovered), mutation probability $P_m = 0.05$ (function $P_m$ defines, how often the part of the chromosome will be mutated) and population of 40 individuals for angles in transition positions, velocities and link times, the size of chromosome string is 9. Number of crossover of chromosomes in each generation can be determined by multiplying $P_c$ and the size of population. Number of mutated genes in each generation can be determined by multiplying $P_m$, the size of population and the length of chromosome. Tournament selection was used, elitizm, the maximum number of 80th generation

Using the toolbox for optimization using genetic algorithms in MATLAB™, I conducted an experimental verification of the results.

CONCLUSION

The verification study was conducted for 3-arm planar moving mechanism in the free workspace. We found a reduction of time of movement while reducing the length of Cartesian trajectory.

Genetic algorithms have proven to be an appropriate optimization method. Moments during the drive motion did not exceed the permitted levels, while being fully utilized.

Literatúra

