e-ISSN: 2249-8303, p-ISSN: 2319-2208 © VSRD International Journals: www.vsrdjournals.com

REVIEW PAPER

A REVIEW ON TRAJECTORY PLANNING AND CONTROLLING OF ROBOT MANIPULATOR ARM

[1]Chandra Shekhar* and [2]S.C. Jayswal
[1]Research Scholar, [2]Head of Department, [1,2]Department of Mechanical Engineering,
Madan Mohan Malaviya University of Technology, Gorakhpur, Uttar Pradesh, INDIA.
*Corresponding Author's Email ID: chandrashekher35@yahoo.in

ABSTRACT

As the automation increases in the industries for achieving the global needs, robots play a very important role to reach the level of automation. Robots are used for many tasks like packaging, part holding, relay line operation, loading/ Unloading etc. These all tasks have been performed by robot manipulator arm by a very basic operation which is a pick and place operation. This simple operation requires a trajectory through which robot manipulator move from initial to goal position. In this paper, review of different trajectory optimization algorithms, algorithms includes the different path curves (Cubic polynomial curve, B-spline curve, Non Uniform Rational B- spline (NURBS) curve) their properties and mathematics and different objective functions (Minimum travelling time, Maximum manipulability, Minimum joint accelerations, Minimum joint jerks, Minimum actuator torques or minimum energy, Obstacle avoidance criteria) is presented.

Keywords: Manipulator,Objective Function,Path Curves, Trajectory.

1. INTRODUCTION

In the present era, a large number of robots are used in industries for simple pick and place operation and other operations which greatly improving the production efficiency. The repetitive tasks performed by the robots and in that situation the minimization of energy consumption will become significantly important.[1] Autonomous manipulation requires algorithms that rapidly and reliably compute collision free motion for robotic manipulators with many degrees of freedom. As there are many ways of the real-world planning task that require further research, the central problem of reliable real-world planning is trajectory planning. The classic trajectory planning problem can be described roughly as follows: to find a path that connects both initial and final configurations of the robot and satisfies some objectives (Minimization of time, torque, mechanical energy, actuator efforts, joint accelerations, jerks and vibrations, singularity avoidance, obstacle avoidance criteria, etc.) and robots kinematics constraints, dynamics constraints, obstacle avoidance constraints and payload constraints, etc.

In the figure.1 shown, the robot manipulator arm pick up an object at point A (initial point) and place at goal position B (goal position or final position). Suppose time taken by manipulator is tf for motion of robot arm from A to B. A path is defined as a sequence of robot configurations in a particular order without regard to the timing of these configurations. So, if a robot goes from point (and thus configuration) A to point B as in Figure 1, the configurations between A and B constitutes a path. However, a trajectory is concerned about when each part of the path must be attained during specific time interval. As a result, in Figure 1, the path is the same, while a trajectory, depending on the velocities and accelerations. In trajectory, the concern is not only about the path a robot takes, but also its velocities, accelerations and jerks.

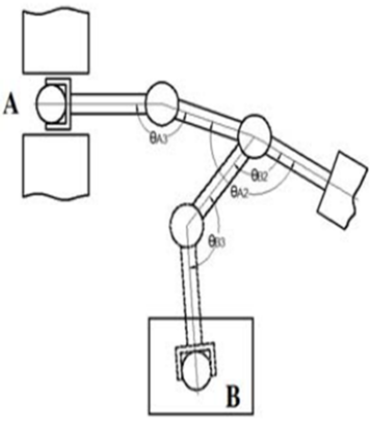

Fig.1: Diagrammatic Sketch of Robotic Arm Initial Position A and Terminal Position B.[2]

2. DIFFERENT PATH CURVES TAKEN FOR TRAJECTORY PLANNING

The trajectory planning based on the type of curve used to represent the trajectory is given below:

- Cubic polynomial curve
- B-spline curve
- Non Uniform Rational B-spline (NURBS) curve

Cubic Polynomial Curve: Generally Polynomial curves are very flexible and useful where a model is developed empirically. They fit a wide range of curvatures. Polynomial curves have certain advantages:

They possess the necessary features that allow them to be

- Easily computed and
- Infinitely differentiable.

A cubic polynomial curve is represented mathematically as:

$$\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \qquad \ldots (1)$$

Where θ is the function of time t and a_0,a_1,a_2,a_3 are the arbitrary constants.

As there is four arbitrary constant in cubic polynomial require four boundary conditions to generate the trajectory.

Suppose, equation (1) shows the position function then one time differentiation will give velocity as:

$$v(t) = a_1 + 2a_2 t + 3a_3 t^2 \qquad \ldots(2)$$

Again, differentiation will give acceleration, as

$$a(t) = 2a_2 + 6a_3 t \qquad \ldots (3)$$

For a robot manipulator motion takes t_f time from initial point A(say) to the goal point B(say) the four conditions may be such that:

- At t=0, $\theta(t) = \theta_0$ (initial position)
- At t=t_f, $\theta(t_f) = \theta_f$(final position)
- At t=0, v(t)=0
- At t=t_f, v(t)=0

On applying these four boundary conditions in Equation (1), and (2), we get the arbitrary constants as:

$$a_0 = \theta_0$$

$$a_1 = 0$$

$$a_2 = \frac{3}{t_f^2}\left(\theta_f - \theta_0\right)$$

$$a_3 = \frac{-2}{t_f^3}\left(\theta_f - \theta_0\right)$$

By putting these values, in equation (1) a relation between time and position can be obtained for trajectory planning.[3]

The simplest polynomial curve that is not generically convex is a cubic curve. In fact, one of the reasons that cubic curves are significantly used in geometric modeling is because they are the lowest degree polynomial curves, which are not generically convex. Another advantage of cubic curves is that they can be joined in a C2 (second order) manner, which means the first and second derivatives at the point where they are joined agree, while quadratic polynomial curves (parabolic arcs) cannot. This extra smoothness factor allows the user to create a vast variety of smooth curves by joining simple polynomial curves. This smoothness allows the user to model a physical phenomenon very easily with cubic curves. The position, velocity and acceleration at one point completely determine a parabolic trajectory. This is a direct result of motion under constant acceleration and Taylor polynomials. [4].

Chettibi et al. (2004) used a cubic polynomial curve to represent the trajectory, while doing optimum trajectory planning for a 3-link articulated robot and PUMA 560 robot. They used Sequential Quadratic Programming technique (SQP). Rana and Zalzala (1996) developed a method to plan a near time-optimal, collision-free motion in the case of multi arm manipulators. The planning is carried out in the joint space and the path is represented as a string of via-points connected through cubic polynomial curves. Shintaku (1999) proposed a simple technique based on genetic algorithm to approximate the joint trajectory as a polynomial curve for underwater manipulator.

B-spline Curve: When a thin elastic wooden or metal strip that is used to draw curves through certain fixed points (called nodes). The resulting curve minimizes the internal strain energy in the splines and hence it is considered to be smooth. The mathematical equivalent of spline curves is the cubic polynomial spline. The piecewise joining of polynomials gives splines. Polynomials are preferred as a blending function due to the following reasons:

- Easier to control
- Easier to check for the continuity

For better control of curve shape the splines selected which give minimum span or maximum possible control over the curve is known as B-spline.

Advantages of B-splines over Bezier curves:

- The degrees of B-spline polynomial can be set independently of the number of control points.
- B-spline Function allows local control over the shape of the function.
- B-spline curves have the ability to interpolate or approximate the given set of control points.
- B-spline curves delink the degrees of resulting curves from the number of control points.
- B-spline curves have C^2 continuity, similar to the natural splines, but do not interpolate their control points.

In B-spline curves, the splines are used as blending functions. In blending function formulation, the general expression for the calculation of coordinate positions P(t) on a B-spline curve may be expressed as:

$$P(t) = \sum_{i=1}^{n+1} P_i \, N_{i,m}(t) t_{min} \leq t \leq t_{max},$$

$$2 \leq m \leq n+1 \qquad \ldots (4)$$

Where, P_i = position vector (coordinates) of (n+1) vertices defining the polygon control points

$N_{i,m}(t)$ =Normalized B-spline blending (basis) functions
m= order of B-spline curve

m-1= Degree of polynomials of B-spline blending function.

The blending function for B-spline curve is defined by the recursive formula:

$$N_{i,m}(t) = \frac{(t-x_i)}{(x_{i+m-1}-x_i)} N_{i,m-1}(t) + \frac{(x_{i+m}-t)}{(x_{i+m}-x_{i+1})} N_{i+1,m-1}(t) \quad \dots (5)$$

To start the recursive calculations, we must define the B-spline blending function of order one (m=1) by Cox-deBoor as:

$$N_{i,1}(t) = \begin{cases} 1 \; if x_i \leq t \leq x_{i+1} \\ 0 \; otherwise \end{cases}$$

The Cox-deBoor formula is used to calculate B-spline blending function in a recursive relation.[4].

Sezimaria F.P. Saramago and Valder Stefen Junior (1998, 1999, 2000, 2001) and Sezimaria F.P. Saramago and Marco Ceccarelli (2002, 2004) have used B-spline curve to represent the trajectory while doing optimum trajectory planning of 2-link planar robot, 3-link articulated robot, PUMA 560 robot and STANFORD manipulator. They have used Sequential unconstrained minimization techniques (SUMT).

Non Uniform Rational B-spline (NURBS) curve: It is little bit hard to produce a perfect circle with B-Splines. Circles and arcs serve as significant curve examples and as major structure for making surfaces such as surface of revolution. Thus, a generalization of B-spline, NURBS is required. NURBS are nearly ubiquitous for Computer-Aided Design (CAD), Manufacturing (CAM), and Engineering (CAE) and are part of numerous industry wide used standards such as IGES, STEP, ACIS, and PHIGS. In general, it can be said that editing NURBS curves is highly intuitive and predictable.

There is only one use of NURBS curve in robot field. The task is difficult since the mobile robot has non-holonomic constraint. Considering the constraint of motion of mobile robot, planning of the trajectory, which the robot can follow, is indispensable for a wheeled type mobile robot control. The NURBS curve is used for trajectory planning. The continuity to second derivative and convex hull, which are the characteristics of the NURBS curve, augurs well for trajectory planning of mobile robots. By this method, efficient tracking is realized. Simulations confirm the validity of the proposed method.[4]

Cubic NURBS functions have been used for constructing joint trajectories due to the following advantages:

- They represent free form shapes with a little data and can be well defined in mathematic form.
- They allow local controllability, which implies the local changes in shape are confined to the NURBS parameters local to that change.
- They possess the ability to control smoothness and enhance curvature continuity.
- They got a characteristic of shape invariance under affine transformation, which means the affine transformed curve is still a NURBS curve whose control points and weights are related to the original curve control points and weights through this transformation.

The choice of NURBS as a shape descriptor, not only offers a common mathematical form for representing free-form shapes but also geometric shapes. The difference between NURBS and B-spline is that the former includes a non-uniform control point vector and an additional parameter, which is weight. Inclusion of weight as an additional parameter adds an extra degree of freedom to NURBS and facilitates the representation of a wide variety of shapes. Further, the use of non-uniform control point vectors allows a better shape control and modeling of a larger class of shapes than uniform knot vector used in B-spline. With these additional parameters, NURBS allows a compact representation that effectively reduces the original number of the boundary points required to represent the robot trajectory. [Tatematsu and Ohnishi (2003)]

Therefore, it is firmly believed that NURBS is a good and accurate trajectory shape descriptor.

3. TRAJECTORY CONTROLLING BASED ON DIFFERENT OBJECTIVE FUNCTION

The trajectory planning may be based on any one or more objective functions as follows:

- Minimum travelling time
- Maximum manipulability
- Minimum joint accelerations
- Minimum joint jerks
- Minimum actuator torques or minimum energy
- Obstacle avoidance criteria

Minimum Travelling Time: In order to maximize the speed of operation, which affects productivity in industrial applications, it is necessary to minimize the total travelling time for robots (Bobrow et al 1985, Shin and McKay 1985, Shin and McKay 1986, Balkan 1998, Chen 1991, Bianco and Guarino 2002, Furukawa 2002, Amar Khoukhi et al 2008, Hui-Fang WANG et al 2008).

Maximum Manipulability: A measure of manipulability for robot manipulators is very useful for manipulator designing, task planning and enables robot the ability to recover fast from escapable singular points. To obtain a practical trajectory (the robot need not lose any degree of freedom at any stage), the manipulability measure can be used as a design criterion. Chih-Jer Lin (2004) used perturbation method to experiment an optimal path planning approach for minimizing the cost of moving a RV-M2 robot (5 degrees of freedom) of MITSUBISHI manipulators along a specified geometric path subject to angle change constraints. The objective function included the obstacle avoidance and singularity avoidance. Mayorga and Wong (1987) dealt the importance of singularities avoidance method for trajectory planning of redundant and non-redundant robot manipulators. Lloyd and Vincent Hayward (2001) presented a robust trajectory generator for PUMA 560 robot to generate singularity-free trajectories by taking the objective function as time minimization using path time scaling method.

Minimum Joint Accelerations: Elnagar and Hussein (2000) proposed an approach to generate acceleration-based optimal smooth piecewise trajectories. Given two configurations (position and orientation) in 3D, the algorithm searches for the minimal energy trajectory that minimizes the integral of the squared acceleration. A numerical iterative procedure is used for computing the optimal piecewise trajectory as a solution of a constrained boundary value problem. The resulting trajectories are not only smoother but also safer with optimal velocity (acceleration) profiles and therefore suitable for robot trajectory planning applications. They demonstrated this fact with experimental results.

Minimum Joint Jerks: Fields of research namely computer graphics, geometric design and robotics (motion planning) prefer smooth trajectories, which are achieved by minimizing the robot joint jerks and accelerations.

The positive effects induced by jerk minimization are:

- Errors during trajectory tracking are reduced.
- Stresses to the actuators and to the structure of the manipulator are reduced.
- Excitation of resonance frequencies of the robot is limited.
- A better coordinated and natural robot motion is obtained.

There are several algorithms for getting smooth trajectories where the value of the jerk appears in the objective function (Simon and Isik 1993, Piazzi and Visioli 1997, Gasparetto and Zanotto 2007). For example, Simon and Isik (1993) minimized the integral of the squared jerk, while Piazzi and Visioli (1997) used a minimax approach to minimize the maximum value of the jerk along the trajectory. They used the interval analysis to develop an algorithm that globally minimizes the maximum absolute value of the jerk along a trajectory whose execution time is set a priori: hence, an approach of the type minimax is used. The trajectories are expressed by means of cubic splines and the intervals between the via-points are computed so that the lowest possible jerk peak is produced. However, these techniques consider the execution time as known (and set a priori); moreover, it is not possible to set any kind of kinematic constraint on the trajectory, because they are not taken into account. Gasparetto and Zanotto (2007) presented a method based on SQP for smooth trajectory planning of robot manipulators. Their objective function is a weighted balance function between the travelling time and integral of squared joint jerks. But their method considered only kinematic constraints. They did not consider dynamic constraints such as joint torques.

Minimum Actuator Torques or Minimum Energy: In certain fields consumed energy rather than time is considered as a primal criterion. This can be the case where the amount of available energy is scarce. Planning of robot trajectory using energy criteria provides several advantages. It yields smooth trajectories easier to track and reduces the stresses to the actuators and to the manipulator structure. Moreover, saving energy may be desirable in several applications, such as those making use of with a limited capacity energy source (e.g. robots for spatial or submarine exploration, material handling). Examples of energy optimal trajectory planning are provided in some literatures (Von Stryk and Schlemmer 1994, Field and Stepanenko 1996, Martin and Bobrow 1999). In few literatures (Von Stryk and Schlemmer 1994, Field and Stepanenko 1996), point-to-point trajectories with minimal energy were considered with upper bounds on the amplitude of the control signals and the joint velocities. Martin and Bobrow (1999) optimized a trajectory with motion constraints set on the end effector. Ata et al (2003) investigated the problem of optimal joint trajectory for a rigid flexible manipulator during constrained motion. Three trajectories, cubic polynomial, sine and Gaussian profiles are proposed and checked based on the minimum energy consumption. These three profiles represent generic pick and place operation and also possess starting and ending values, but they differ in the rate of increase of velocity. They concluded that Gaussian profile gives the lowest energy followed by sine and cubic polynomial profiles. The genetic algorithms can also be applied to humanoid robot to find the minimum consumed energy during specific tasks. Capi et al (2001) proposed a genetic algorithm gait synthesis method to generate angle trajectories for walking and gait upstairs. They applied a Radial Basis Neural Network for real world applications. Lei and Su (2004) proposed an adaptive algorithm based on the genetic algorithm for stable, minimum energy trajectory. The proposed trajectory can be applied to other activities like walking and overcoming obstacles as well. Both optimal travelling time and minimum mechanical energy of actuators are considered together as objective function in some literatures (Shiller 1996, Chettibi et al 2004, Sezimaria F.P. Saramago and Valder Stefen Junior 1998, Sezimaria F.P. Saramago and Marco Ceccarelli 2002). Shiller (1996) dealt the time-energy optimal trajectories, i.e. the cost function has a term linked to the execution time and a term expressing the energy spent. The trade-off between the two needs can be adjusted by changing the respective weights. Chettibi et al (2004) used a Sequential Quadratic Programming (SQP) method for getting optimal motion planning for a PUMA560 manipulator. Their objective function was a weighted balance of transfer time and mean average of actuators efforts and power. Sezimaria F.P. Saramago and ValderStefen Junior (2000) proposed a method based on Sequential Unconstrained Minimization Techniques (SUMT) to get an optimal motion planner for a STANFORD manipulator in the presence of fixed, moving and oscillating obstacles. Their objective function was a weighted balance of transfer time, mechanical energy of the actuators and penalty for collision free motion. They have taken physical constraints like joints displacements, velocities, accelerations, jerks and torques and collision avoidance.

Obstacle Avoidance Criteria: The obstacle avoidance motion problem has been largely studied over a great time. The main task of path planning for robot manipulators is to find an optimal collision-free trajectory from an initial to a goal position. Many important contributions to this problem have been made in the recent years (Henrich et al 1998, Saab and VanPutte1999, Helguera and Zegloul 2000, Volpe and Khosla 1997, Muniz et al 1995, Yang and Meng2000, Rogério R. dos Santos et al 2007, FarbodFahimi et al 2008).

4. CONCLUSION

From the review of literatures, criteria are identified for the purpose of comparing and evaluating trajectories of robots:

- The trajectory should be safer, considers limiting parameters of the robot, takes less time, economical one and needs only minimum actuator efforts.
- Trajectories should be predictable and accurate and should not degenerate unacceptably near a singularity.
- The robot joints positions, velocities, accelerations and jerks should be smoother functions of time.
- Trajectories should be effective both to compute and execute.

To get all these benefits, we have to consider decision criteria viz., minimization of travelling time, mechanical energy of the actuators, joint jerks, joint accelerations, penalty function to guarantee free collision-motion and maximization of manipulability in a combined manner.

The trajectory planning and controlling by using different path curves like cubic polynomial curve, B-spline curve, Non- rational B-spline (NURBS) curve with different objective functions like Minimum travelling time, Maximum manipulability, Minimum joint accelerations, Minimum joint jerks, Minimum actuator torques or minimum energy, Obstacle avoidance criteria has been performed by different researchers. The choice of the trajectory depends on the objective function and also on the accuracy needed for the robot manipulator motion. As the application needs more accuracy and have a complex (means mixing of two or more than two objective function discussed previously) type of objective function, the optimization of trajectory become uneconomic in terms of time required by the system for calculation as well as the cost of controller also increases on increase of complexity. Hence the choice of path curve and objective function should be selected according to the application for which the robot manipulator is designed.

5. REFERENCES

[1] Hirakawa, A.R., Kawamura, A., Trajectory generation for redundant manipulators under optimization of consumed electrical energy. Industry Application Conference, 1996.Thirty-first IAS Annual Meeting, Conference record of the 1996 IEEE, pp. 1626-1632.

[2] Du.H, Du.M, Chen .L.A., Mai Z.W., Liu.X.H., and Cai.H.Z., Multi-DOF Robotic Manipulator Trajectory Controlling Based on Minimum Energy Optimization., International Conference on Artificial Intelligence and Industrial Engineering (AIIE-2015)

[3] Craig.John.J.,Introduction to Robotics mechanics and control, third edition, Dorling Kindersley (India) Pvt.Ltd., licensees of Pearson Education in South Asia.

[4] Srivastav. S.K., Computer Aided Design, First edition, I.K. International Publishing House Pvt. Ltd., New Delhi.

[5] Shinkatu E. (1999), Minimum energy trajectory for an underwater manipulator and its simple planning method by using genetic algorithm , Advanced Robotics, Vol. 13, pp. 115-138.

[6] Rana A. and Zalzala A. (1996), An evolutionary planner for near time-optimal collision-free motion of multi-arm robotic manipulators , UKACC International Conference on Control, Vol. 1, pp. 29-35.

[7] Tatematsu N. and Ohnishi K. (2003), Tracking motion of mobile robot for moving target using NURBS curve , Proceedings of the IEEE International Conference on Industrial Technology, Vol. 1, pp. 245- 249.

[8] Bobrow J.E., Dubowsky S. and Gibson J.S. (1985), 'Time-optimal control of robotic manipulators along specified paths', International Journal of Robotics Research, Vol. 4, No. 3, pp. 554-561.

[9] Bianco L. and Guarino C. (2002), 'Minimum-time trajectory planning of manipulators under dynamic constraints', International Journal of Control, Vol. 75, No. 13, pp. 967-980.

[10] Chen Y.C. (1991), 'Solving robot trajectory planning problems with uniform cubic B-splines', Optimal Control Applications and Methods, Vol. 12, pp. 247-262.

[11] Shin K.G. and McKay N.D. (1985), 'Minimum-time control of robotic manipulators with geometric path constraints', IEEE Transactions on Automatic Control, Vol. 30, No. 6, pp. 531-541.

[12] Shin K.G. and McKay N.D. (1986), A dynamic programming approach to trajectory planning of robotic manipulators, IEEE Transactions on Automatic Control, Vol. 31, No. 6, pp. 491-500.

[13] Balkan T. (1998), A dynamic programming approach to optimal control of robotic manipulators', Mechanics Research Communications, Vol. 25, No. 2, pp. 225-230.

[14] Amar Khoukhi, Luc Baron, Marek Balazinski and KudretDemirli (2008), 'A Hierarchical Neuro-Fuzzy System to Near Optimal-Time Trajectory Planning of Redundant Manipulators', Engineering Applications of Artificial Intelligence, Vol. 21, No. 7, pp. 974-984.

[15] Furukawa T. (2002), Time-sub minimal trajectory planning for discrete non-linear systems, ww.acfr.usyd.edu.au/people/academic/tfurukawa/papers/E 02002-1.pdf.

[16] Hui-Fang WANG and Yang-Zhou CHEN (2008), Time-optimal Trajectories for a Car-like Robot, Acta Automatica Sinica, Vol. 34, No. 4, pp. 445-452.

[17] Chih-Jer Lin (2004), 'Motion planning of redundant robots by perturbation method', Mechatronics, Vol. 14, pp. 281-297.

[18] Mayorga R.V. and Wong A.K.C. (1987), A singularities avoidance method for the trajectory planning of redundant and non-redundant robot manipulators, Proceedings of IEEE International Conference on Robotics and Automation, pp. 1707-1712.

[19] Lloyd J.E. and Vincent Hayward (2001), Singularity-Robust Trajectory Generation', International Journal of Robotics Research, Vol. 20, No. 1, pp. 38-56.

[20] Elnagar A. and Hussein A. (2000), On optimal constrained trajectory planning in 3D environments, Robotics and Autonomous Systems, Vol. 33, No. 4, pp. 195-206.

[21] Gasparetto A. and Zanotto V. (2007), A new method for smooth trajectory planning of robot manipulators', Mechanism and Machine Theory, Vol. 42, No. 4, pp. 455-471.

[22] Piazzi A. and Visioli A. (1997), 'An interval algorithm for minimum jerk trajectory planning of robot manipulators', Proceedings of the 36th Conference on Decision and Control, San Diego, California, December, pp. 1924-1927.

[23] Simon D. and Isik C. (1993), A trigonometric trajectory generator for robotic arms', International Journal of Control, Vol. 57, No. 3, pp. 505-517.

[24] Von Stryk O. and Schlemmer M. (1994), 'Optimal control of the industrial robot, Computational Optimal Control', International Series of Numerical Mathematics, Basel, Birkhauser, Vol. 115, pp. 367-382.

[25] Field G. and Stepanenko Y. (1996), 'Iterative dynamic programming: an approach to minimum energy trajectory planning for robotic manipulators, Proceedings of the IEEE International Conference on Robotics and Automation, pp. 2755-2760.

[26] Martin B.J. and Bobrow J.E. (1999), 'Minimum effort motions for open chain manipulators with task-dependent end-effector constraints, International Journal of Robotics Research, Vol. 18, No. 2, pp. 213- 224.

[27] Capi G. Nasu Y., Barolli L., Mitobe K. and Takeda K. (2001), 'Application of genetic algorithm for biped robot gait synthesis optimization during walking and going up-stairs', Advanced Robotics,Vol. 15, No. 6, pp. 675-694.

[28] Shiller Z. (1996), 'Time-energy optimal control of articulated systems with geometric path constraints', Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 118, pp. 139-143.

[29] Chettibi T., Lehtihet H.E., Haddad M. and Hanchi S. (2004), 'Minimum cost trajectory planning for industrial robots', European Journal of Mechanics A/Solids, Vol. 23, pp. 703-715.

[30] Sezimaria F.P. Saramago and ValderStefen Junior (1998), 'Optimization of the Trajectory Planning of Robot Manipulators taking into account the Dynamics of the System', Mechanism and Machine Theory, Vol. 33, No. 7, pp. 883-894.

[31] Sezimaria F.P. Saramago and Marco Ceccarelli(2002), 'An optimum robot path planning with payload constraints', Robotica, Vol. 20,pp. 395-404.

[32] Sezimaria F.P. Saramago and ValderStefen Junior (2000), 'Optimal trajectory planning of robot manipulators in the presence of moving obstacles', Mechanism and Machine Theory, Vol. 35, pp. 1079-1094.

[33] Henrich D., Wurll C. and Worn H. (1998), 'On-line path planning by heuristic hierarchical search', IECON'98 Proceedings of the 24thAnnual Conference of the IEEE Industrial Electronics Society, New York, NY, USA, pp. 2239-2244.

[34] Saab Y. and VanPutte M. (1999), 'Shortest path planning on topographical maps', IEEE Transaction on Systems, Man and Cybernetics Part-A., Vol. 29, No. 1, pp. 139-150.

[35] Helguera C. and Zegloul S. (2000), 'A local-based method for manipulators path planning in heavy cluttered environments', Proceedings of the 2000 IEEE International Conference on Robotics and Automation, San Francisco, CA, USA, pp. 3467-3472.

[36] Volpe J.R. and Khosla P. (1997), 'Artificial potentials with elliptical contours for obstacle avoidance', IEEE Proceedings of the 26thConference on Decision and Control, Los Angeles, CA, USA,pp. 180-185.

[37] Yang S.X. and Meng M. (2000), 'An efficient neural network approach to dynamic robot motion planning', Neural Networks, Vol. 13, pp. 143-148.

[38] Rogério R. dos Santos, Valder Steffen Jr. and Sezimária de F.P. Saramago (2007), 'Robot path planning in a constrained workspace by using optimal control techniques', Multibody System Dynamics, DOI 10.1007/s11044-007-9059-1.

[39] FarbodFahimi, Nataraj C. and Hashem Ashrafiuon (2008), 'Real-time obstacle avoidance for multiple mobile robots', Robotica, Article in press, Corrected proof published in online, DOI: 10.1017/S0263574708004438.

●●●