A COMPARATIVE STUDY OF OPTIMAL CONTROL ALGORITHMS FOR ROBOT CONTINUOUS PATH PLANNING

Patrick PLEDEL, Yasmina BESTAOUI
Laboratoire d'Automatique de Nantes, URA CNRS 823
Ecole Centrale de Nantes / Université de Nantes
1 rue de la Noë, 44072 Nantes, France
Tel : (33) 40-37-25-12 ; fax : (33) 40-74-74-06
email : bestaoui@lan.ec-nantes.fr ;

Abstract :
The optimal motion generation problem is solved subject to actuator constraints while the motion is constrained to an arbitrary path. The considered objective function is a weighted time energy function. Existing methods consider only the time-optimal problem. We present some simulation results using a mathematical programming technique (Sequential Quadratic Programming) existing in the software NPSOL. Then a comparative study is made between existing methods and the proposed technique.

I - INTRODUCTION

Motion along a predefined path is common in robotics, for instance in the case of spraying, cutting or welding. It is natural to look for an optimal solution along the path [VANW]. The path is given from the application and a first step is to obtain a nominal motion specification. A required path is normally expressed analytically in cartesian space. The resulting trajectory can be expressed either in cartesian space or joint space and the advantages and disadvantages of either representation are consistent with the interpolation method [PLED]. The geometric path does not contain any timing information but includes only spatial positions and orientations. When a continuous correspondence is made between a trajectory described in cartesian space and joint space, problems may appear related to workspace and singularities. The trajectory must be planned so as to remain in the manipulator workspace. In this case, path generation in joint space could be easily executed but a cartesian trajectory would fail. In this paper, the emphasis is put on the optimal planning of manipulators trajectories, in joint space. The trajectory must be optimal with respect to a weighted time-energy performance index.

For rigid robots, the minimal time optimization along a predefined path can be solved using phase-plane techniques. The optimization algorithm ([BOBR], [SHIL]) is a tractable way to obtain the minimum-time solution. This algorithm however cannot be extended to the case under interest, the objective function being a weighted time energy function. The determination of the desired robot motion as a function of time involves a nonlinear optimal control problem. The optimization problem considers the generally non linear robot and actuators dynamics. We will focus on manipulators actuated by DC motors. DC motors operate over a wide speed range and have excellent control characteristics.

The remainder of this paper is divided into five sections. Modelling is introduced in the following paragraph. Then some existing methods are presented in the third section, while a nonlinear programming technique is introduced in the fourth paragraph. After that, some simulation results are presented in the fifth section and some discussions given. Finally, general conclusions are given in the last paragraph.

II - MODELLING

2-1 manipulator model

For a manipulator with n joints, the dynamic model can be expressed using the Lagrangian equation as:

$$\Gamma_i = \sum_{j=1}^{n} A_{ij}(q)\,\ddot{q}_j + G_{i}(q) + \sum_{j=1}^{n} F_{ij}(q)\,\dot{q}_j + \sum_{j=1,\,m=1}^{n} B_{ijm}(q)\,\dot{q}_j\,\dot{q}_m \quad i=1,\dots,n \tag{1}$$

where the n x 1 vectors q, $\dot{q}$ and $\ddot{q}$ are respectively the joint position, velocity and acceleration, the n x 1 vector Γ is the joint input torque, G is the n x 1 gravitational force vector, B is the n x n x n Coriolis and Centrifugal force matrix, F is the viscous friction and A is the n x n inertial matrix.

2-2 actuator model

In a permanent magnet DC motor [TAHB], the magnetic field is developed by permanent magnets. For a non-redundant multi-degrees-of-freedom robot, there are usually as many actuators as the number of degrees-of-freedom. The actuator dynamics, giving the voltage U as a function of the current I, can be characterized in a matrix form as:

$$U = L\frac{dI}{dt} + R\,I + K\,\dot{q} \tag{2}$$

where L, R and K are square regular diagonal matrices with respectively inductance, resistance and torque constant elements at their diagonals.

2-3 path description

The path describes the robot motion in space. In practice, analytical function are seldom directly available. Usually, the path is specified as a finite number of points which have to be crossed in a given order [PLED]. Assume the path is represented as a parameterized curve:

$$q = q(s) \tag{3}$$

where s is a scalar path parameter. A trajectory is obtained from the path $g(q(s)) = 0$, by specifying the path parameter as a function of time. The function s(t) is defined on the interval [0,T] where $s(0)=s_0$ and $s(T)=s_f$. Since we assume that the path is fixed, i.e the function q(s) is given, the trajectory q(s(t)) can be represented by the path parameter

0-7803-2129-4/94 $3.00 © 1994 IEEE

$s(t)$. We assume that the path parameter $s(t)$ is piecewise twice differentiable with respect to t. Further, we only consider trajectories which represent forward motion along the path. The assumption $\dot{s}(t) > 0$ implies that $\dot{s}(t)$ can be expressed as a function of $s(t)$. This function is called the velocity profile and is denoted $v(s)$.

III EXISTING METHODS

3.1 Introduction

Most motion generation laws are developed based on kinematical constraints, obtained for the most unfavorable configurations. Thus, to define maximal torques, accelerations and velocities, we may write:

$\Gamma_{max} = K\, I_{max}$ and neglecting all the dynamic terms apart inertial ones in (1), $\ddot{q}_{max} = (K\, I_{max}) / |A|_{max}$. Also, neglecting L in (2) gives:

$$\dot{q}_{max} = (U_{max} - R\, I_{max}) / K \qquad (4)$$

The determination of theses values allows to propose simple motion generation laws. Some examples are given in [DOMB] : bang-bang or polynomial laws for point to point motion or a trajectory crossing some given points.

[PLUM] has proposed a suboptimal law with respect to maximal torques. From a given trajectory, a temporal scaling is applied to obtain a new trajectory respecting the maximal torques constraints.

3.2 Maximum velocity limit curve method

3.2.1 rewriting the robot dynamics

Equation (1) can be written as:

$$\Gamma = A_1(s)\ddot{s} + B_1(s)\dot{s}^2 + F_1(s)\dot{s} + G_1(s) = A_1(s)\,\ddot{s} + A_2(s, \dot{s})$$

where : $A_1(s) = A(q(s))\, q_s$ $\quad F_1(s) = F_v(q(s))\, q_s$

$B_1(s) = q_s^T B(q(s))\, q_s + A(q(s))\, q_{ss}$ $\quad G_1(s) = G(q(s))$

$$A_2(s, \dot{s}) = B_1(s)\,\dot{s}^2 + F_1(s)\,\dot{s} + G_1(s) \qquad (5)$$

where q_s is the unit vector tangent to the path and q_{ss} is the curvature vector obtained by differentiating q_s with respect to s. The parameters A_1, B_1, F_1 and G_1 are path specific, representing the inertia and centrifugal-Coriolis, gravity and friction forces reflected at the joints for a given point along the path.

3.2.2 maximum velocity limit curve method

The constraints which are obtained by combining the torque constraints, the robot dynamics and the path constraint, provide useful information for the design of the path velocity controller. Ignoring the dynamics of manipulators during trajectory planning results either in reduced path accuracy or a less than minimal traverse.

The constraints on the torques ($\Gamma_{min} \le \Gamma \le \Gamma_{max}$) may be written as velocities and acceleration constraints:

$$\ddot{s}_{min}(s,\dot{s}) \le \ddot{s} \le \ddot{s}_{max}(s,\dot{s}) \quad \text{and} \quad 0 \le \dot{s} \le \dot{s}_{max}(s) \qquad (6)$$

where:

$$\ddot{s}_{min} = \max_i \left\{ \min_{\Gamma^i}\left[[\Gamma^i - A^i_2(s,\dot{s})] \right] / A^i_1(s) \right\}$$

$$\ddot{s}_{max} = \min_i \left\{ \max_{\Gamma^i}\left[[\Gamma^i - A^i_2(s,\dot{s})] / A^i_1(s) \right] \right\}$$

The velocity limit $\dot{s}_{max}(s)$ is defined as the velocity at which $\ddot{s}_{min} = \ddot{s}_{max}$. The resolved problem may be written as :

$$\text{Min } (J = \int_0^T dt)$$

$$\text{subject to :} \begin{cases} u = \ddot{s} \\ \ddot{s}_{min}(s, \dot{s}) \le u \le \ddot{s}_{max}(s, \dot{s}) \\ 0 \le \dot{s} \le \dot{s}_{max}(s) \end{cases} \qquad (7)$$

If $A^i_1(s) = 0$ for some i, then this actuator does not bound the acceleration. Thus, there is another state constraint $\Gamma_{min} \le A_2(s, \dot{s}) \le \Gamma_{max}$. These points are called singular points and the corresponding trajectory, singular arc. Their existence is tied to the nature of the robot and the path. When there exist singular arcs, solutions are bang-coast-bang. A switch occurs at tangency, discontinuity and critical points.

The following algorithm was proposed by [SHIL]:

1- From the initial point, integrate forward the maximum acceleration, if it reaches the final point, go to step 5. If the trajectory hits the limit curve at some point S_h, go to step 2.

2 - Search forward for the nearest critical or tangency point, $S_t \ge S_h$.

3 - From S_t, integrate backward the maximum feasible deceleration until the trajectory crosses the previous trajectory at some point, $S_{cr} \le S_h$. At this point, the trajectory switches from acceleration to deceleration.

4 - From S_t, integrate forward the maximum feasible acceleration until the trajectory hits the limit curve again. If it passes the final point, then proceed to step 5, otherwise go to step 2.

5 - From the final point, integrate backward the maximum deceleration until crossing the previous trajectory.

The result shows that for conditions that are typically satisfied, the time-optimal solution is bang-bang in the sense that at least one torque is always at the limit.

IV - GENERAL MOTION GENERATION PROBLEM

4-1 optimal problem formulation

The capabilities of a DC motor are mainly limited by the heat generation and dissipation characteristics. The actuator constraint limits the torque (or force) applied to each link. This limitation will result in bounds on joint speeds and accelerations. The state and control variables are respectively chosen to be : $x = [s, \dot{s}, \ddot{s}]^T$ and $u = \dddot{s}$, so the differential equation describing the system is given by:

$$\dot{x} = A\,x + B\,u \text{ where } A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

the controlled system is a linear one. However, the other functions involved are highly nonlinear ones.
The optimization of the motion along a specified path can be stated as the following problem:

$$\text{Min} \qquad J = \int_0^T [\,(1-\pi) + \pi\, U(t)\, I(t)\,]\, dt \tag{8}$$

Subject to

- dynamical model: $\dot{x} = A\,x + B\,u$ (9)
- state (position) constraint :
$g(q(s)) = 0$; $s(0)=0$; $s(T)=s_f$ (10)
- actuators limitations:
$-I_{max} \le I \le I_{max}$ $\qquad -U_{max} \le U \le I_{max}$ (11)

$$-Dimax \le \frac{dI}{dt} \le Dimax \tag{12}$$

$$\sqrt{\int_0^T I^2(t)dt} \le Ieff \tag{13}$$

where Imax, Dimax and Ieff are respectively the maximum absolute values of the motor current instantaneous value, slew rate and square mean value and Umax is the maximum absolute value of the voltage supply. The parameter π is chosen by the user to give more or less weight to time or energy. All the actuators limitations may have a unified representation:
$S(x(t),u(t)) \le 0$ (14)

4-2 resolution method

4-2-1 introduction

In theory, any optimal control problem can be solved analytically by employing Pontryagin's minimum principle. However, it is impracticable to do so if the dimension of the state vector is higher than two. In optimal control, there are direct and indirect solution methods. Direct means to affect a control history directly by varying a finite set of defining parameters. Indirect means to solve the two point boundary value problems constituted by the necessary conditions of optimality. Such methods are for example: Differential Dynamic Programming technique [BEST], or multiple shooting technique [BRYS]. However, they require an a priori identification of singular arcs.
Therefore as practical alternatives, many numerical methods have been developed. All of the available numerical methods first discretize the given continuous system with a fixed sampling period. The period should be small enough so that no significant discretization error is introduced. Then the minimum count of the control steps are searched for, under control constraints, such that the system reaches the desired final state. Since in the discrete domain, the number of variables (control steps) is itself a variable, this problem can only be considered by solving exhaustively a sequence of fixed time problems. The difficulties of this method lie in the mechanism of sequencing the fixed time problems and in knowing when to stop ([CHUN]).

Another difficulty of the discretize-then-search method is the conflicting effect of the sampling period on the discretization error and on the complexity of the optimization problem. Since the final time is fixed for the original continuous system, although it is unknown yet, it should be equal to the product of the sampling period used in the discretization and the count of the control steps found in the optimization. Thus, if the sampling period is reduced for better accuracy, then the minimum count of the control steps increases inversely. This makes the search iteration count long, and the problem size becomes larger near the end of the search process. In order to avoid the exhaustive iteration and to overcome the conflicting effect of the sampling period for the discretization error and on the computational complexity, this paper fixes the count of the control steps and treats the sampling period for the discretization error as an optimization variable. The optimization process minimizes the sampling period. Any change of the sampling period during the optimization changes the dynamics of the discrete system which is a counterpart of the original continuous system. Since the usual optimization process is performed on a system whose dynamics are fixed, the proposed approach seems unreasonable. However, the goal of optimization is not only to reduce the sampling period itself, thus reducing the control time (the product of the count of the control steps and the minimum sampling period obtained in the optimization), but for the discrete counterpart to best approximate the original continuous system.

4-2-2 Discrete counterpart

With a period τ being a very small constant, Euler's first order approximation of the time derivative of the state vector is given as: $\dot{x}(t)=(x(t+\tau)-x(t))/\tau$. This gives a discrete state equation : $x_{i+1} = (I + \tau A)\, x_i + \tau B\, u_i$
where $x_i= x(i\,\tau)$ and $u_i= u(i\,\tau)$. Solving it gives the state expression at general time k in terms of the initial state and the intermediate inputs as :

$$x_k=(I + \tau A)^k x_0 + \tau \sum_{i=0}^{k-1} (I + \tau A)^{k-1-i} B u_i \tag{15}$$

With this discretization, the original problem is restated in a new manner. If the original continuous system has a (theoretical) optimal time T, the minimum count of the discrete control steps will be: N = Int(T/τ)+1. Considering the sampling period as a variable, this problem is transformed into a discrete form as follows:

$X = [u_0, u_1, ..., u_{N-1}, \tau]^T$ with $\dim(X)= N+1$ (16)

The performance index: $(1-\pi)\,(N-2)\tau + \pi\,\tau \sum_{i=0}^{N-1} U_i\, I_i$ (17)

The equality constraints:

$$x_f - (I + \tau A)^{N-2} x_0 - \tau \sum_{i=0}^{N-2} (I + \tau A)^{N-2-i} B u_i = 0 \tag{18}$$

The inequality constraints:

$$S(X(k)) \le 0 \; ; \; -\dot{s}_k \le 0 \;\; \text{for all } k \le N-1 \; ; \; -\tau \le 0 \qquad (19)$$

Numerous simulations were performed using the NPSOL software [GILL]. NPSOL uses the Sequential Quadratic Programming (SQP) technique which belongs to the class of Projected Lagrangian methods [GILL]. This class includes algorithms that contain a sequence of linearly constrained subproblems based on the Lagrangian method. The idea of linearizing nonlinear constraints occurs in many algorithms for non linearly constrained optimization including the reduced gradient type methods. The subproblems involves the minimization of a general non linear function subject to linear equality constraints and can be solved using an appropriate technique [GILL]. The choice of solution method will depend on the information available about the problem functions (i.e the level and cost of derivatives information) and on the problem size. For our concrete problem, we have given the gradients of the functions computed with the MAPLE software, to the NPSOL program.

V - SIMULATION RESULTS

5-1 simulations

The algorithm was applied to a two-degree-of-freedom robot (see fig1).

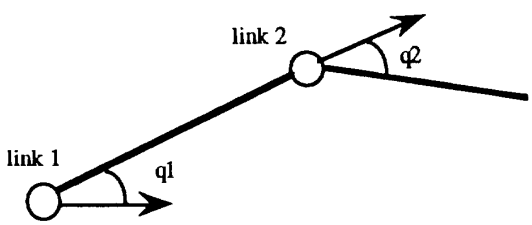

figure 1

The proposed path, defined by four crossing points, is represented by parts of fifth degree s-polynomials. Table 1 gives the crossing points coordinates:

Points	1	2	3	4
Axis X (m)	0.5	0.2	0.0	0.0
Axis Y (m)	0.3	0.5	0.5	0.4

Table 1

The dimension of vector X is chosen to be 91, There exists then 633 constraints. The bounds of the voltage and current are assumed respectively to be:

$Umax=(30,20)^T V$ $\qquad$ $Imax= (10,10)^T A$

$Ieff=(12,10)^T A$ $\qquad$ $Dimax=(10^4,10^4) A/s$

This gives the following torques bounds:

$\Gamma max=(14.38,11.26)^T Nm$ and $\Gamma min=-\Gamma max$

Most of the nonlinear programming algorithms require a feasible solution set of the optimization variables to start the optimization. However, in many cases, it is difficult to find a feasible solution. It is known from SQP that it converges quadratically if the initial estimate is sufficiently close to the solution and the hessian of the Lagrangian is positive semi-definite. When the restrictive conditions mentioned above, are not verified, SQP algorithm will in general fail to converge. One reason is that the correct active constraints set must somehow be determined. Another is that the subproblems may be defective because of incompatible constraints. SQP can be viewed as a Newton-method and it is known that an unsafeguarded Newton method is not a robust algorithm. SQP is good in terms of its local convergence properties but not in terms of guaranteed convergence.

We describe a way to deal with this problem. As the different solutions of general minimum-time problems are usually made of phases with positive and negative accelerations, we choose an initial solution of this shape for $[\ddot{s}_k]_{k \le N-1}$. Moreover, experiences show that convergence was easier if the initial solution is continuous and satisfy $u_0 = 0$ and $u_f = u_0$. So, we finally choose an initial solution X0 of fifth order polynomial shape, varying from umax to -umax. The number of discretization points and umax are user dependable parameters. τ is obtained such as $s(T)=s_f$.

The optimal times obtained respectively for the maximal velocity limit curve method and the one we propose, are T=1.2 s and T=1.4 s. Moreover, the suboptimal method of [PLUM] gives T=2.4s. When the objective function is a weighted time energy function, the maximal velocity limit curve method cannot be used. Using our method, the final time for π=0.1 is T=3.4 s and for π=0.5 is T=5.9 s.

Figures 2,4,6,8 represent respectively the torques for the maximal velocity limit curve method, the one we propose for π=0 and π=0.1, and [PLUM] method while figures 3,5, 7,9 represent the related joint accelerations versus time. All these figures show the importance of the DC actuators constraints, particularly for the primary degrees of freedom, for which the motors are more powerful, with non negligeable resistance and inductance .

Although the boundary conditions are specified for zero speed and acceleration at the end points, the method is applicable to any arbitrary boundary conditions on these states.

5-2 Discussions

The maximal velocity limit curve method gives good results for simple cases. The forward and backward integration must be done with a very low period. Moreover, on a singular arc, the search must be very precise. The nature of possible switching points being very different (tangency, discontinuity and critical), the behaviour of the robot in these points is also very different. The implementation of the algorithm involves the determination of many precision parameters, depending on each followed path. This determination takes a certain amount of time and needs some experience. Velocity and torques constraints are not sufficient to ensure a safe behavior of the robot, in some cases, this algorithm gives an infinite curvilinear acceleration and thus the real actuators constraints are not fulfilled.

In the authors knowledge, weighted time-energy problem is solved for the first time and the obtained results are original. Solutions are smoother than for minimal time approach. As π grows from 0 to 1, the motion is slower as it takes more time to be performed. The minimal energy approach may be very interesting in some cases. The method is applied to a two link manipulator driven by DC motors. It is however applicable to general multi degrees of freedom systems, driven by actuators such as pneumatic and hydraulic actuators and internal combustion engines. Moreover, the proposed technique is very general and some other performance indices may be chosen.

VI - CONCLUSIONS

This paper considers a solution to the problem of moving a manipulator, with weighted time-energy performance index along a specified geometric path subject to voltage and current constraints, taking into account the viscous friction. The motion generation algorithm uses the solution of an optimal problem to find the predicted arrival time as well as the joint acceleration, velocity and position versus time. The obtained trajectory is twice continuously differentiable. In contrast to traditional methods in which the count of the control steps is chosen as the variable and an exhaustive sequential search is used to find the minimum time, the proposed approach considers the sampling period as a variable. The optimization problem is solved using NPSOL software. A weighted time-energy performance index is of great interest since it allows the use of smooth controls while existing methods are only time-optimal ones. We propose some comparison remarks.

REFERENCES

[BEST] Y.Bestaoui "Design of a robotic discrete minimum energy regulator" IEE Proceedings on Control, Theory and Applications, 1991, vol 138, n°6, pp509-514

[BOBR] J.E.Bobrow, S.Dubowski, J.S.Gibson "Time optimal control of robotic manipulators along prespecified paths" International Journal of Robotics Research, vol4, n°3, 1985

[BRYS] A.Bryson, Y.Ho "Applied optimal control " Blaisdell ed, 1969

[CHUN]T.Chung, C.Wu "A computationally efficient numerical algorithm for the minimum time control problem of continuous systems" Automatica, vol 28, n°4, pp841-847, 1992

[DAHL] O.Dahl "Path constrained robot control " PhD thesis, June 1992, Lund Institute of Technology, Sweden

[DOMB] E.Dombre, W.Khalil " Modélisation et commande des robots" Hermès Publ, 1988, Paris

[FOUR] J.Y.Fourquet "Mouvement en temps minimal pour les robots manipulateurs en tenant compte de leur dynamique non linéaire" PhD Thesis, LAAS, Univ of Toulouse, France, Dec 1990

[GILL] P.Gill, W.Murray, M.Wright "Practical optimization " Academic Press, 1981

[PLED] P.Plédel " Optimisation de la génération de mouvements de robots manipulateurs en suivi de trajectoires " DEA report, Sept 1993, Univ. of Nantes, France

[PLUM] F.Plumet "Contrôle dynamique et placement de tâches optimal pour manipulateurs rapides" PhD thesis, Nov 1992, Paris, France

[SHIL] Z.Shiller, W.Lu "Computation of path constrained time optimal motions with dynamic singularities" ASME, vol 114, 1992, pp34-41

[TAHB] K.Tahboub "Modeling and control of constrained robots" PhD thesis, Univ. Of Wuppertal, Germany, sept 1993

[VANW] L.G.Van Willigenburg "Digital optimal control of rigid manipulators" PhD thesis, 1991, Univ. Of Delft, The Netherlands

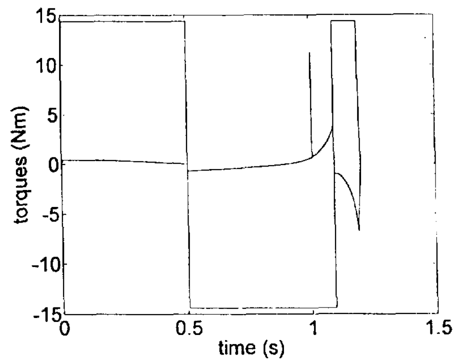

figure 2

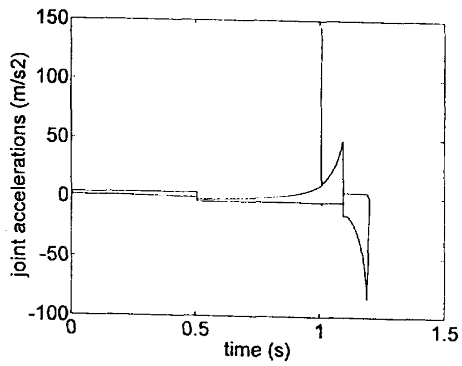

figure 3

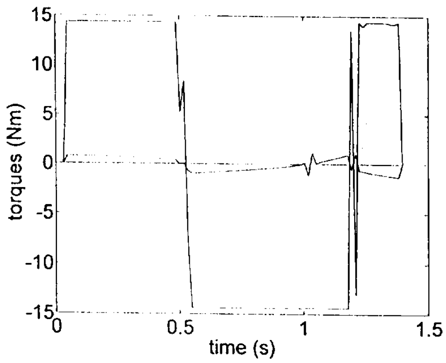

figure 4

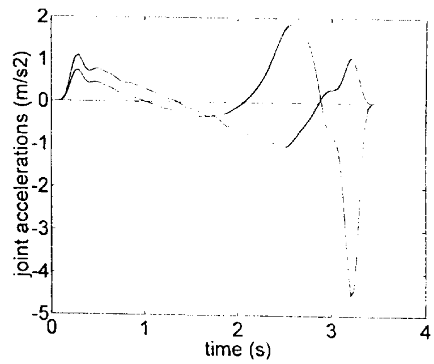

figure 7

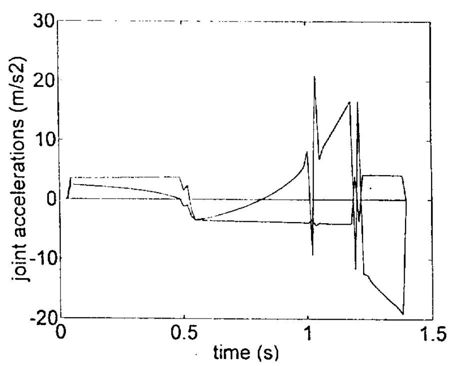

figure 5

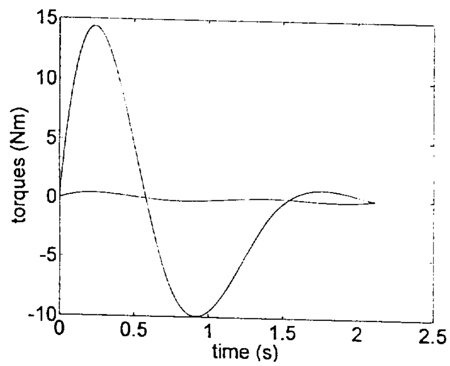

figure 8

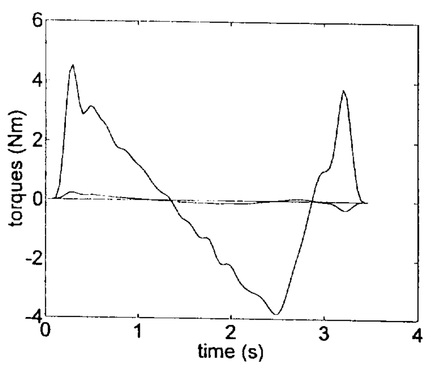

figure 6

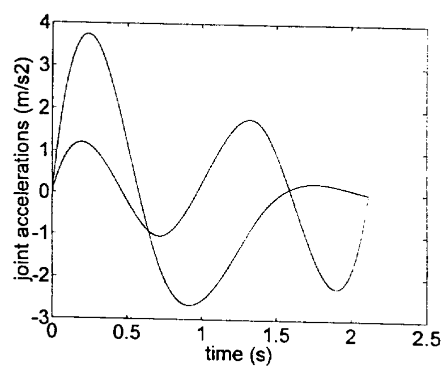

figure 9