MP6-4 5:20

Proceedings of the 1999 IEEE
International Conference on Control Applications
Kohala Coast-Island of Hawai'i, Hawai'i, USA • August 22-27, 1999

A Minimum-time Motion Planning Method based on Phase Space Analysis

K.C. Koh* , H.S. Aum*, H.S.Cho**

* Sun Moon University, Mechanical and Control Engineering, Galsan-ri 100,
Asan-shi, Chungnam-Do, 336-840, Korea
** KAIST, Mechanical Engineering, 373-1,
Yusong-gu, Taejon, 305-701, Korea

Abstract— In this paper, we propose a minimum time motion planning algorithm considering jerk and acceleration constraints. The conventional planning methods are valid only for zero boundary conditions, thus they have limitations in real time applications. The proposed method is computationally efficient and can deal with a problem with non-zero boundary conditions so that it is adequate for real time motion planning. For the realtime implementation, the planning algorithm is derived in the phase space. In order to determine the jerk, a landing surface is introduced through bang-bang principle. To show the validity of the proposed method, a series of simulations are conducted. The algorithm has merits of providing a consistent and unified numeric solution for motion control system requiring the real time motion planning with jerk constraint and suffering from strong disturbance.

I. INTRODUCTION

In an inertial system with one-dimensional motion, the velocity, acceleration and jerk(the rate of acceleration change) are constrained by considering mechanical stability and the actuator capability. In particular, a robotic assembly system with high speed motion needs the constraint for maximum jerk to improve the positional accuracy and to prevent the mechanical system from the vibration([1],[2]). The anti-vibration is the key factor for determining the life cycle of the mechanism. In robotic systems, the jerk constrained motion guarantees a smooth and stable motion[3]. For given jerk and acceleration constraints, many schemes on minimum time motion planning were proposed. R.D. Peters[5] solved the motion equation for position, velocity and acceleration according to the distance to travel. He classified the motion into three cases:(A) both the velocity and the acceleration are lower than each maximum, (b) the acceleration reaches its maximum, but the velocity is lower than the maximum, (c) both the velocity and the acceleration reach each maximum value . In this method, the solutions are valid only for zero boundary conditions on both starting and target positions. Thus, the planning is accomplished by offline. However, as in most robotic systems, an unpredictable positional error occurs due to the external disturbances, the structural deformation arising from long-time run, and servo control error in the actuators, etc.. To cope with this problem, two kinds of position sensors are being used. One is a motor encoder used for servo systems and the other is an absolute position sensor for compensation of error due to the disturbances. The motor encoder is for guaranteeing the stability of position control loop but does not include direct information about the real position. Thus, the off-line motion planning is meaningless in case of the updating position commands based on an absolute sensor for compensating such indirect sensor. The compensation requires the real time motion planning which should consider non-zero boundary conditions. The consideration of the acceleration and jerk constraints is important for guaranteeing the smooth motion and avoiding the mechanical shock. In this context, the real time path planing should satisfy two conditions. One is to solve the minimum time problem with non-zero boundary condition at the starting point. The other is the mathematical simplicity and consistency in its numeric solution which is very important for software implementation in the embedded systems. The offline planning method needs to solve the sixth order polynomial equation. Its heavy computational burden makes the real time implementation difficult. In this study, a real time motion planning method based on phase error space is proposed. It guarantees the minimum time motion and provides the general non-zero boundary condition solution. A landing surface plane is defined in the three-dimensional space of velocity(x), acceleration(y) and position(z) coordinates. Most robotic control system involves the motion planner generating the position and velocity commands, and then the motion controller following these commands for every sampling instant. The proposed method determines the jerk from the current position, velocity and acceleration at every sampling time and new commands for the next sampling time are computed by real-time integration of the jerk. Since this planning method is structurally compatible with the motion controller, it is possible to determine the jerk in real time for moving target point. Furthermore, the computational simple structure makes it possible to implement it on an practical embedded system. This paper is organised as follows: Section II explains the main idea of the method and Section III details the proposed algorithm. In Section IV, the computer simulation results are discussed and conclusion is remarked in the Section V with our further study plan.

II. THEORETICAL BACK GROUND

2.1 Time optimal Motion Planning

Let us consider a virtual phase space with X-Y-Z axis as velocity, acceleration and position, respectively. A current point P_c is represented by $(v_c, a_c, p_c)^T$ and has the following equation of motion:

$$\dot{p}_c = v_c \quad (1)$$
$$\dot{v}_c = a_c \quad (2)$$
$$\dot{a}_c = j_c \quad (3)$$

where the velocity, acceleration and jerk are constrained by each maximum values as follows;

$$|v_c| \le v_m \quad (4)$$
$$|a_c| \le a_m \quad (5)$$
$$|j_c| \le j_m \quad (6)$$

where v_m, a_m and j_m are maximum values for velocity, acceleration and jerk, respectively. Now, consider the minimum time problem to find the jerk, j_c so that the current position, P_c reach the target $P_c=(v_t,a_t,p_t)^T$ in minimum time.

2.2 offline planning method

The time optimal path can be obtained by taking a jerk trajectory which switches twice as follows: For the forward motion$(p_t > p_c)$, the current point leaves the starting point, P_c along the $+j_m$ path, then the jerk is switched to negative maximum, $-j_m$ at $t = t_1$ and finally reaches the target by switching again to positive maximum, $+j_m$ at $t = t_2 > t_1$. On the other hand, for the backward motion$(p_t < p_c)$, the direction of the jerk from the starting to target point is switched in the reversed order as $-j_m, +j_m$, and $-j_m$. In the conventional off-line methods, once boundary conditions for starting and target points are given, the switching time instants, t_1, t_2, and the final reaching time t_f are computed from the boundary conditions and their continuity conditions. The final solutions for position, velocity and acceleration are given by:

$$P_c(t) = (v_c(t), a_c(t), p_c(t))^t$$

where $0 \le t \le t_f$. This path planing with time-based solution is difficult to be applied to a real-time motion system in which the target should be updated on-line using an absolute position sensor. When it is implemented on a digital system, the error due to a finite sampling time makes the algorithm complex. As an approach to cope with this problem, a jerk determining algorithm based on phase space which has the *3D*- axis of velocity(x), acceleration(y) and position(z) is proposed.

23 Phase space and landing surface

Let us consider a motion planing with zero target velocity and acceleration which is common to robot or elevator control. Once the target point is given by $P_t = (0,0,p_t)$, a virtual plane guiding all the points nearby a target point towards the target point is assumed to exist. We call this plane the landing surface and denote it as $p_s = \phi_s(v_s, a_s)$. This landing surface can be defined from path design to connect a given point $P_s = (v_s, a_s, p_s)$ in the phase plane to the target $P_t = (v_t, a_t, p_t)$ in minimum time. The landing surface is composed of two parts of which domain is separated by the landing curve $v_s + \frac{a_s^2}{2j_{max}}\mathrm{sgn}(a_s) = 0$ as in Fig.1.

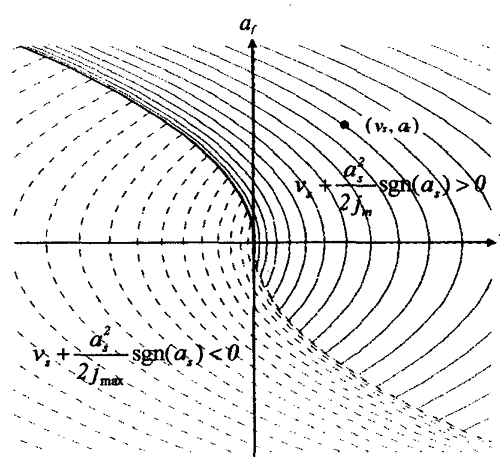

Fig.1 Positive landing domain and negative landing domain

(i) positive landing domain: $v_s + \frac{a_s^2}{2j_{max}}\mathrm{sgn}(a_s) \ge 0$.

As seen from Fig.1, the optimal path in a domain (v_s, a_s) of $v_s + \frac{a_s^2}{2j_m}\mathrm{sgn}(a_s) > 0$ which is defined as positive landing domain, the point departing the P_s, moves along a curve γ_s^- which has negative maximum jerk, $-j_m$ and reaches the target P_t by switching to a curve γ_t^+ which has the positive maximum jerk, $+j_m$. When the switching occurs at the point, $P_1 = (v_1, a_1, p_1)$, the set of curves, γ_s-γ_t is uniquely determined for any arbitrary point (v_s, a_s) given on the perspective plane of Fig.1. The aggregation of these infinite set of curves, γ_s-γ_t makes a landing surface, $p_s = \phi_s(v_s, a_s)$ which leads the initial point of (v_s, a_s) to target point of (v_t, a_t). The surface equation can be derived through a mathematical process with parametric curve design in 3D space. Omitting the detail derivation procedures, the final result is presented by the following equation.

$$p_s^+ = \phi_s^+(v_s, a_s) = p_t - \frac{a_s^3}{3j_m^2} - \frac{v_s a_s}{j_m} + \frac{a_1^3}{j_m^2} \quad (7)$$

where $a_1 = -(v_s j_m + \frac{a_s^2}{2})^{1/2}$. (8)

(ii) negative landing domain: $v_s + \frac{a_s^2}{2j_{max}} \operatorname{sgn}(a_s) < 0$.

Next, let us consider the landing surface for the negative domain of $v_s + \frac{a_s^2}{2j_{max}} \operatorname{sgn}(a_s) < 0$. As seen from Fig.1, the path reaching the target $P_t = (v_t, a_t, p_t)$ starting from P_s in the negative landing domain, becomes symmetrically opponent to that of the positive landing domain. Thus, it consists of the curve γ_s^+ with positive maximum jerk, $+j_m$, starting from P_s, and the curve γ_t^- with negative maximum jerk, $-j_m$ reaching the target P_t. From phase space analysis, the landing surface equation in the negative landing domains can be given by

$$p_s^- = \phi_s^-(v_s, a_s) = p_t - \frac{a_s^3}{3j_m^2} + \frac{v_s a_s}{j_m} + \frac{a_1^3}{j_m^2} \tag{9}$$

$$a_1 = (-v_s j_m + \frac{a_s^2}{2})^{1/2}. \tag{10}$$

By defining j_m^* as,

$$j_m^* = j_m \operatorname{sgn}(v_s + \frac{a_s^2}{2j_m} \operatorname{sgn}(a_s)) \tag{11}$$

the equations from (7) to (10) can be unified as

$$\phi_s(v_s, a_s) = p_t - \frac{a_s^3}{3j_m^2} - \frac{v_s a_s}{j_m^*} + \frac{a_1^3}{j_m^2} \tag{12}$$

where $a_1 = -(v_s j_m^* + \frac{a_s^2}{2})^{1/2} \operatorname{sgn}(j_m^*)$. (13)

Fig.2 shows the landing surface drawn in 3D phase space.

III. Proposed Motion Planning Algorithms

The proposed motion planning method is implemented as iterative algorithm which determines the velocity, acceleration and position commands at every control instant. For the *k*-th sampling time, the given target point $P_t^k = (0, 0, p_t^k)$ with $(v_t^k = a_t^k = 0)$ and the given current point, $P_c^k = (v_c^k, a_c^k, p_c^k)$, the reference point for k+1th sampling time, $P_c^{k+1} = (v_c^{k+1}, a_c^{k+1}, p_c^{k+1})$ is computed by the algorithm explained below.

3.1 Jerk Determination with the landing surface t

Once current point $P_c^k = (v_c^k, a_c^k, p_c^k)$ is given in the phase space, the distance from the point to the landing surface, Δp_s is computed by

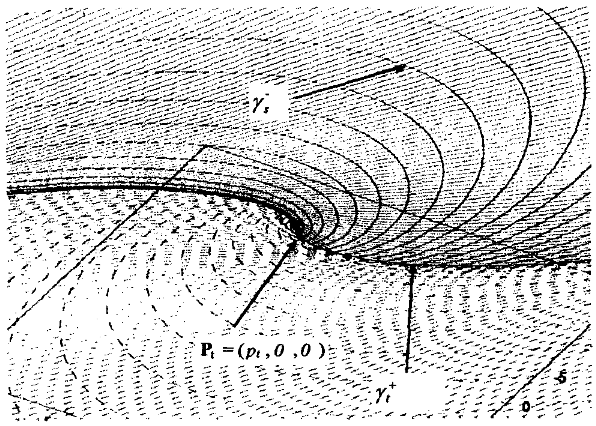

Fig.2 Landing surface defined in the phase space

$$\Delta p_s = \phi_s(v_c^k, a_c^k) - p_c^k \tag{14}$$

where Δp_s is the distance from the current point $P_c^k = (v_c^k, a_c^k, p_c^k)$ to the landing surface in z-direction. The jerk is determined to make the point approach the landing surface, in direction that Δp_s becomes zero. If $\Delta p_s > 0$, the forward motion follows the direction of making Δp_s zero. Therefore the jerk is determined to be positive maximum($+j_{max}$). On the contrary, if $\Delta p_s < 0$, the jerk is set to be negative maximum($-j_{max}$) and the current point, P_c^k goes down to landing surface. The following equation represents this idea:

$$j_c^{k+1} = j_m \operatorname{sgn}(\Delta p_s) \tag{15}$$

With this jerk action, P_c^k approached the landing surface. If it reaches the negative landing surface, ϕ_s^-, the direction of jerk is switched to $+j_{max}$ by Eq.(15) and then, P_c^k moves along the landing surface to the target point. If P_c^k reaches the positive landing surface, ϕ_s^+, the sign of the jerk is changed $(-j_{max})$ and finally approaches the target P_t along with the positive landing surface. This following motion along with the landing surface becomes the time optimal since it is based on bang-bang principle and switches the jerk sign twice as $+j_{max}$, $-j_{max}$, and $+j_{max}$ for forward motion or $-j_{max}$, $+j_{max}$, and $-j_{max}$ for backward motion. Due to the jerk determined in this manner, the point in the next sampling instant, $P_c^{k+1} = (v_c^{k+1}, a_c^{k+1}, p_c^{k+1})$ are obtained by the following iterative equations:

$$a_c^{k+1} = a_c^k + j_c^{k+1} \Delta t_s \tag{16}$$

$$v_c^{k+1} = v_c^k + a_c^k \Delta t_s + j_c^k \Delta t_s^2 / 2 \tag{17}$$

$$p_c^{k+1} = p_c^k + v_c^k \Delta t_s + a_c^k \Delta t_s^2 / 2 + j_c^k \Delta t_s^3 / 6 . \quad (18)$$

3.2 Jerk Determination with Maximum Acceleration

In this section, the acceleration constraint given by Eq.(5) is considered. First, the landing surface should be re-derived by adding the acceleration saturation into Eqs.(12) and (13). The condition for the acceleration saturation can be obtained from the phase analysis of the equation of the motion as:

$$|v_c| \geq \frac{|a_c^2 - 2a_{\max}^2|}{2j_{\max}} . \quad (19)$$

The landing surface for this saturated zone is also composed of two parts defined in positive and negative landing domain, respectively. For each domain, new equations of the landing surfaces can be driven. In the positive landing domain, the point departing the $P_s = (v_s, a_s, p_s)$, moves along a curve, γ_s^- with negative maximum jerk, $-j_m$ and then switched to the curve γ_0^- with zero jerk($j_s = 0$) when it meets the saturation condition of (19). Finally, the point rides the curve γ_t^+ with positive maximum jerk, $+j_m$ guiding to the target point, In the negative domain, the path from starting point to the target becomes symmetrically opponent to the positive landing case. It follows γ_s^+ with positive maximum jerk, $+j_m$ passing P_s, switches to the curve γ_0^+ with zero jerk, and rides the curve γ_t^- with negative maximum jerk, $-j_m$ approaching the target P_t. Using this concept, The derivation of the saturated landing surface equation can be also performed and here the resultant equation is presented by

$$\phi_s(v_s, a_s) = p_t - \frac{a_s^3}{3j_m^{*2}} - \frac{v_s a_s}{j_m^*} - \frac{v_I^2}{2a_m^*} - \frac{a_m}{j_m}(v_I + \frac{3a_m^2}{8j_m^*}) \quad (20)$$

where $v_I = v_s + \dfrac{a_s^2 - a_m^2}{2j_m^*}$. (21)

The jerk determination rule considering the acceleration constraints is modified by using clamp function as follows:

$$j_c^{k+1} = clamp(\frac{a_m \operatorname{sgn}(\Delta p_s) - a_c^k}{\Delta t_s}, j_m)$$

(22)

where the clamp function, $clam(a,b)$ is the function which limits the magnitude of a to b.

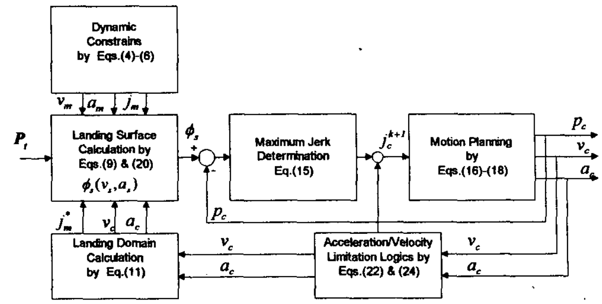

Fig.3 S/W block diagram of proposed planning algorithm

3.3 Jerk Determination with Maximum Velocity

Finally, the velocity constraint given by Eq.(4) is considered. The condition for the velocity saturation can be driven from the motion analysis in the phase space as:

$$|v_c| + \frac{a_c^2}{2j_m} \leq v_m . \quad (23)$$

Considering this condition, the jerk determination for $|v_c^k| + \dfrac{a_c^{k+1\,2}}{2j_m} \geq v_m$ can be written as the following equation:

$$j_c^{k+1} = clamp(-\frac{|a_c^k|}{\Delta t_s} \operatorname{sgn}(\Delta p_s), j_m) \cdot \quad (24)$$

The jerk determination algorithm is now summarised as the following five steps for a given current point, $P_c^k = (v_c^k, a_c^k, p_c^k)$,

<step 1> *Landing domain*: calculate j_m^* using Eq.(11)

<step 2> *Landing surface* :If $|v_c^k| < \dfrac{|a_c^{k2} - 2a_m^2|}{2j_m}$, then select the landing surface $\phi_s(v_c^k, a_c^k)$ of Eqs.(12)-(13), else select saturated landing surface $\phi_s(v_c^k, a_c^k)$ of Eqs.(20)-(21).

<step 3> *Jerk determination:* If $|v_c^k| < \dfrac{|a_c^{k2} - 2a_m^2|}{2j_m}$, compute the jerk by Eqs.(14) and (15), else compute the jerk by Eqs.(14) and (22)

<step 4> *Consideration of Velocity Constraint:* If $|v_c^k| + \dfrac{a_c^{k2}}{2j_m} \geq v_m$, calculate j_c^{k+1} using Eq.(24)

<step 5> *Integration:* Compute next current point $P_c^{k+1} = (p_c^{k+1}, v_c^{k+1}, a_c^{k+1})$ by Eq.(16)-(18).

IV. Simulation Results

To demonstrate the proposed algorithm, a series of computer simulations are conducted for the various cases of the motion. The motion planning software based on the proposed algorithm presented in the Section III, is implemented in discrete version so as to be applicable for digital control system. Fig.3 shows the overall block diagram of the motion planing system design for the simulation. The conditions for the simulation are:

(1) *Maximum velocity:* $v_m = 2m / \sec$

(2) *Maximum acceleration:* $a_m = 0.8m / \sec^2$

(3) *Maximum jerk :* $j_m = 0.8m / \sec^3$

(4) *Sampling time:* $\Delta t_s = 1 \times 10^{-3} \sec$

The simulation was repeated for three targets with difference distances from the starting point. Fig.4 shows the motion planning result for (a) $p_t = 1m$. As seen from the figure, this case shows the motion with both acceleration and velocity below each maximum value. Fig.5 shows the motion planning result for (b) $p_t = 5m$, which leads to the case of reaching the maximum acceleration and below the maximum velocity. Fig.6 shows the motion planning result for (c) $p_t = 10m$, which leads to the case of reaching both the maximum acceleration and maximum velocity. Fig.7 shows the motion trajectories depicted in the velocity-acceleration phase plane for previous three cases. This switching pattern can be viewed more clearly in the 3-D phase space with landing surface as in Fig.8. This figure well explains where the jerk switching occurs for each case. The figure shows that the *Case(a)* has the pattern of reaching the landing surface on a middle point, the *Case(b)* has the pattern of reaching the landing surface on an acceleration edge point, and the *Case(c)* has the pattern of reaching the landing surface on the velocity edge point. From this 3D visualisation, the time optimal motion with dynamic constraints can be intuitively understood. Fig.9 demonstrate that the algorithm can be applied to the case of changing target position during control.

V. Summary

In this paper, the real-time minimum time motion planning algorithm considering jerk, acceleration and velocity constraints is presented. The proposed motion planning algorithm provides motion reaching the target from any non-zero velocity and acceleration initial point The proposed algorithm is computationally simple and can be effectively implemented in real-time embedded system. The algorithm is effective for following the changing target which is frequent in real robotic system. The simulation results show the proposed method shows the validity of the proposed algorithm and the possibility of its realistic implementation from the computational view of point. The planning software based on the proposed algorithm is designed for digital control system and is going to be applied to the mobile robot path control in the future study.

Ackowledgments

We would like to acknowledge partial support received from KOSEF and the facilities provided by RRC at the Sun Moon University.

References

1. R.G. White and J.G.Walker, "Noise and Vibration",1982, John Willey & Sons
2. K.C. Koh and H.S. Cho, "A path tracking control system for autonomous mobile robots: an experimental investigation", *Mechatronics*, Vol. 4, No. 8, pp. 799-820, 1994.
3. E.A.Croft. et al."Near-time Optimal Robot Motion Planning for On-line Applications", J. of Robotic Systems 12(8), 553-567,1995
4. C.M.Chen et al."Minimum Time Trajectory Planning of Robotic Manipulators, Int. J. of Robotics and Automation, Vol.7, No.3, 117-125, 1990
5. R.D. Peter,"Ideal Lift Kinematics", Proceeding of ELECON'95, pp.175-184,1995
6. Larry Mittag,"Multitasking Design and Implementation Issues in Embedded Systems", Proceedings of ESC'97,pp.527-532, 1997

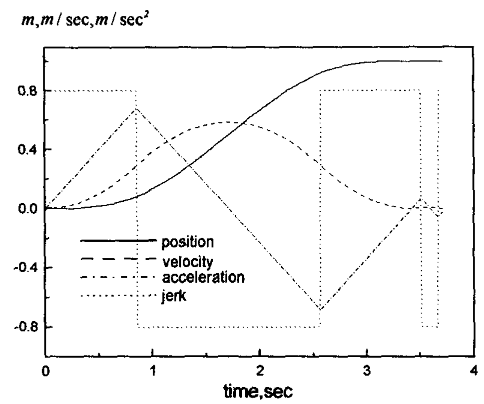

Fig.4 Simulation result below both maximum velocity and maximum acceleration (case A)

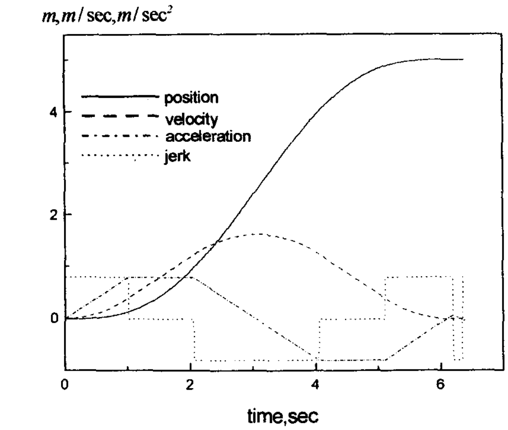

Fig.5 Simulation result below reaching only maximum acceleration but below maximum velocity(case B)

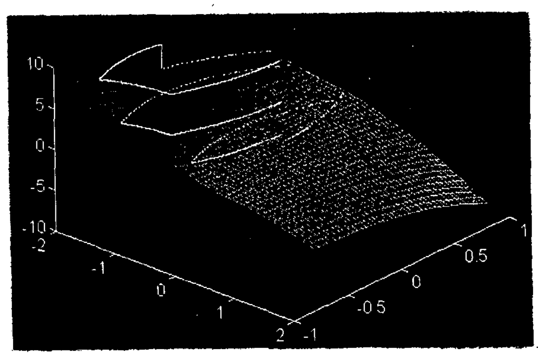

Fig.8 Planned motion trajectories in 3-D phase space

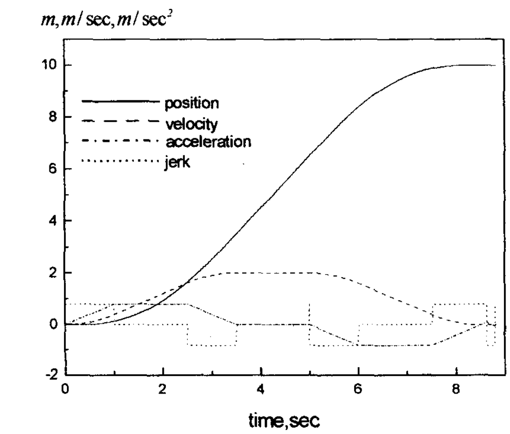

Fig.6 Simulation result reaching both maximum acceleration and maximum velocity(case c)

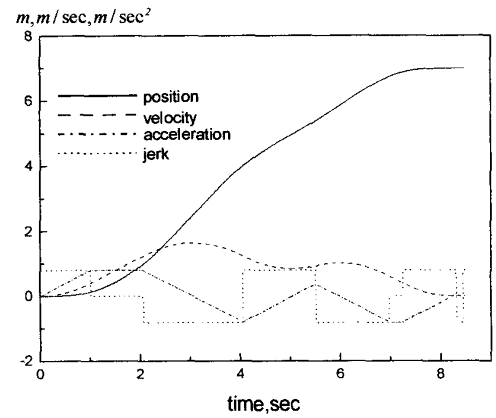

Fig.9 Simulation result when the target was changed at t=4.2sec (Pf=5->7m,Vf=Af=0,Pi=Vi=Ai=0,Ts=1mS,Vmax=2m/sec, Amax=0.8m/sec**2, Jmax=0.8m/sec**3)

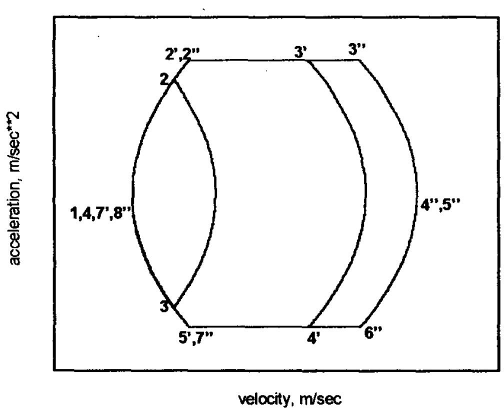

Fig.7 phase plane diagram