Reference model generation for tracking and ending in steady final state

Kitak Ahn, Wan Kyun Chung and Youngil Youm

Robotics & Bio-mechanics Lab., Department of Mechanical Engineering
Pohang University of Science & Technology(POSTECH), Pohang 790-784, Korea
(Tel: +82-54-279-5946; Fax: +82-54-279-5899; Email:termi@postech.ac.kr)

Abstract: In the area of tracking control, it is important to design not only the controllers but also the trajectories to which a system has to follow. 5^{th} order polynomial is often used with constraints of initial and final states. Smooth ending with possible minimum time is important for many systems because of vibration or jerky motions. Examples are increased with development of technology in smaller, more accurate systems. On the base of a polynomial like trajectory generation method from a paper in ACC2002 and RIC(Robust Internal-loop Compensator) control scheme of Robotics and Bio-mechanics lab. of POSTECH, generalized and expanded polynomial like trajectory generation method is showed.

Keywords: trajectory, model-following control, polynomial

1. Introduction

In the area of tracking control, model-following control is a widely used one. In practice, it is important to design not only the controllers but also the trajectories to which a system has to follow. There are some methods for trajectory generation. Polynomial, spline, sinusoidal and trapezoidal shape, etc. 5^{th} order polynomial is often used with constraints of initial and final states. The order 5 is from the optimal problem which minimize the integral of jerk square during run time in bio engineering [1], especially for the motion of the human hand and the guide of rehabilitation. But as for the non-bio systems, minimum jerk during full running time is not necessary. Smooth ending with possible minimum time is more important for many systems because of vibration or jerky motions. Examples are increased with development of technology in smaller, more accurate systems.

On the base of a polynomial like trajectory generation method from a paper in ACC2002 [2] and RIC (Robust Internal-loop Compensator) control scheme of Robotics and Bio-mechanics lab. of POSTECH [3], [4], [5] generalized and extended order of polynomial like trajectory generation method is showed. The method is based on the time varying state feedback and has the dynamics of double integrator. So with the model following controller and 2^{nd} order systems, it produces general order of polynomial like reference trajectory tracked by the system. The generated trajectory is the desired one and can then be used as set points to be tracked by the controller. The higher the order of the reference, the faster the system arrived to the final steady state. The constraints for the reference can be expanded to the n^{th} derivative of position. So, not only acceleration(3^{rd} derivative) but also jerk(4^{th} derivative) and even more concepts can be considered.

Suggested trajectory is applied to the simulation of a 2nd order system platform – chip mounter device, and shows a better tracking performance. It needs also much less computational load than other trajectory generation methods. When saturation of plant actuator has to be considered, a method using the controllability concept of discrete time-varying case can be predict the highest order of the trajectory with which the steady final state is achieved.

The rest of this paper is arranged as follows. Section 2 shows a brief introduction of the polynomial-like trajectory generation method. It is called "reference model generation" from the title of reference paper. Shortly RMG would be used here. In section 3 RMG is generalized and expanded to more minute concept and it is the main contribution of this paper. Section 4 shows the effects of RMG shortly. A simulation on the platform of chip mounting device which already has a robust controller designed with RIC concept is in section 5. The conclusion and future work are discussed in section 6.

2. RMG (Reference Model Generation)

2.1. Base platform

A model-following control method has 2 loops in the structure. One of the loop is inner loop and has a function to compensate the difference between the real and nominal model. After some reconstruction, it can be shown that actual plant with disturbance behaves like a given nominal model. Focused on this fact, a generalized disturbance compensating framework named robust internal-loop compensator (RIC) was proposed and unified analysis and design of robust controller were performed.[5]

A 2^{nd} order model is one of the most used model because of most real plant can be represented as a 2^{nd} order one with simplification and assumption procedures. RIC structure which has a feedback controller in the model-following feed-forward scheme can compensate for error due to the model mismatch. Fig. 1 shows the structure.

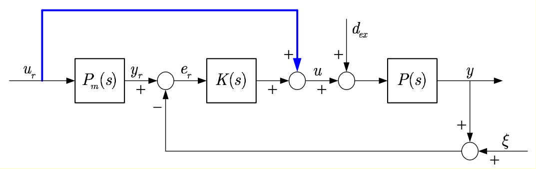


Fig. 1. RIC(Robust internal loop compensator) structure

2.2. Reference model generation

When control input has the type of a polynomial function of time,

$$u[n] = a_1 \cdot n + a_2 \cdot n^2 + a_3 \cdot n^3 + u[0] \qquad (1)$$

where n is the discrete time. We begin with the polynomial order 3. The plant model is regarded as a 2^{nd} order system and simply a double integrator. Then this model is described as the following simple equations.[6], [7]

$$\begin{aligned} v[n] &= v[n-1] + u[n-1] \\ x[n] &= x[n-1] + v[n-1] \end{aligned} \qquad (2)$$

The output of this model for input (1) is the desired output of the real plant. Therefore we can define each terms in (2) as follows.

$$\begin{array}{ll} u[n] & control\ input \\ v[n] & state(velocity) \\ x[n] & output(position) \end{array}$$

These can be rewritten after recursive expansion and arrangement with coefficients.

$$\begin{aligned} v[n] &= v[0] + \sum_{i=0}^{n-1} u[i] \\ x[n] &= x[0] + \sum_{i=0}^{n-1} v[i] \end{aligned}$$

The first input term $u[1]$ is

$$\begin{aligned} u[1] &= a_1 + a_2 + a_3 + u[0] \\ &= \tfrac{-9+N}{N} u[0] - \tfrac{36}{N(N+1)} v[0] - \tfrac{60}{N(N+1)(N+2)} x[0] \end{aligned}$$

For the steady motion at final time, that is to say, for the smooth settling, the states at final time N are all 0.

$$u[N] = v[N] = x[N] = 0$$

Using these conditions we can get the input state at time n with some tedious development procedure. Symbolic toolbox in the MATLAB is very helpful.

$$\begin{aligned} \text{u}[2] = & \tfrac{-9+N-1}{N-1} u[1] - \tfrac{36}{(N-1)(N-1+1)} v[1] \\ & - \tfrac{60}{(N-1)(N-1+1)(N-1+2)} x[1] \end{aligned}$$

$$\vdots \quad \begin{aligned} u[1] &\to u[n] \\ u[0] &\to u[n-1] \\ v[0] &\to v[n-1] \\ x[0] &\to x[n-1] \\ N &\to N-n+1 \end{aligned}$$

$$\begin{aligned} \text{u}[n] = & \tfrac{-9+N-(n-1)}{N-(n-1)} u[n-1] \\ & - \tfrac{36}{(N-(n-1))(N-(n-1)+1)} v[n-1] \\ & - \tfrac{60}{(N-(n-1))(N-(n-1)+1)(N-(n-1)+2)} x[n-1] \end{aligned}$$

This is the formula for the time-varying feedback for the states and input of the RMG model. Dividing this with constant parts and variable parts,

$$\begin{aligned} u[n] = & (1 + \alpha \cdot K[n-1]) \cdot u[n-1] \\ & + \beta \cdot K[n-1] \cdot K[n-2] \cdot v[n-1] \\ & l + \gamma \cdot K[n-1] \cdot K[n-2] \cdot K[n-3] \cdot x[n-1] \end{aligned} \qquad (3)$$

where

$$K[n-1] = \frac{1}{N-(n-1)}$$

And we can find the constants

$$\begin{bmatrix} \alpha & \beta & \gamma \end{bmatrix} = \begin{bmatrix} -9 & -36 & -60 \end{bmatrix}$$

In the reference [2], constants of several order input cases were shown.

up to 7th orders

$$\begin{aligned} [\alpha\ \beta\ \gamma] &= [\,-9 \quad -36 \quad -60\,] && 3^{rd}\ order \\ &= [\,-12 \quad -60 \quad -120\,] && 4^{th}\ order \\ &= [\,-15 \quad -90 \quad -210\,] && 5^{th}\ order \\ &= [\,-18 \quad -126 \quad -336\,] && 6^{th}\ order \\ &= [\,-21 \quad -168 \quad -504\,] && 7^{th}\ order \end{aligned}$$

RMG structure is including one-step ahead states and time-varying terms.

$$\begin{aligned} u[n] = & (1 + \alpha \cdot K[n-1]) \cdot u[n-1] \\ & + \beta \cdot K[n-1] \cdot K[n-2] \cdot v[n-1] \\ & + \gamma \cdot K[n-1] \cdot K[n-2] \cdot K[n-3] \cdot x[n-1] \\ v[n] = & v[n-1] + u[n-1] \\ x[n] = & x[n-1] + v[n-1] \end{aligned} \qquad (4)$$

where

$$K[n-1] = \frac{1}{N-(n-1)}$$

But because two of the three time-varying terms are obtained already at previous step as shown in Fig.2, the term to be calculated is just one.

$$\begin{aligned} u[n] = & (1 + \alpha \cdot N) \cdot M_{n-1} \\ & + \beta \cdot N \cdot M_{n-1} \cdot M_{n-1} \\ & + \gamma \cdot N \cdot M_{n-1} \cdot M_{n-1} \cdot M_{n-1} \\ v[n] = & M_{n-1} + M_{n-1} \\ x[n] = & M_{n-1} + M_{n-1} \end{aligned}$$

where

$$\begin{array}{ll} N & : New\ but\ just\ 1\ computation \\ M_{n-1} & : from\ the\ memory\ 1\ step\ before \end{array}$$

Therefore, for the trajectory generation on the system with its own CPU or real time OS, RMG significantly reduces the computational loads. In addition, RMG has the same computational load regardless of the order of polynomials.

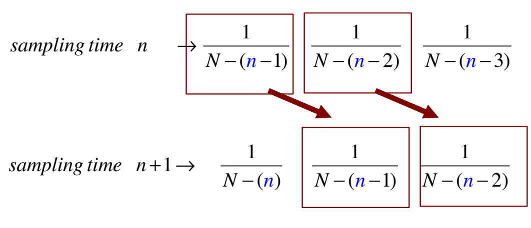


Fig. 2. Updated term at each sampling time :Time Varying part

3. Generalization and Expansion

3.1. Generalization

Although choosing a formula on the basis of a few observations or somewhat tedious calculations does not guarantee the validity of the formula, pattern recognition is useful to find some rules. Finite differences can help finding the pattern.

First differences of a sequence are found by subtracting consecutive terms. If the first differences are all the same, then the pattern is *linear*.

Second differences are found by subtracting consecutive first differences. If the second differences are all the same, then the pattern is *quadratic*.

From the fact that we can find a quadratic model by taking the equation $y = ax^2 + bx + c$ with 3 points, we can find the general n^{th} term

$$Y_n = a_0 n^2 + a_1 n + a_2$$

by substituting in three terms in the sequence for Y_n and their corresponding position in the n^{th} sequence. These are summarized in the table 1.

One of the advantages of RMG method is a simple and easy calculation to get a trajectory state at each sampling time when the order of the input polynomial is given. Using above finite difference method, it is shown to find a rule of RMG constants for general n^{th} order given reference trajectory. When the input to the double integrator RMG model is regarded as the system input, the constants to be determined are three : α, β, γ.

As shown in table 2 and Fig.3, α has a *linear* pattern.

From the general form and with two terms,

$$\alpha_n = a_0 + a_1 n$$

$$\alpha_3 = -9, \;\; \alpha_4 = -12$$

Table 1. Pattern from the finite differences

Difference	If all are same! Pattern is	General form
1^{st} Difference	Linear	$a_0 + a_1 n$
2^{nd}Difference	Quadratic	$a_0 + a_1 n + a_2 n^2$
3^{rd}Difference	Cubic	$a_0 + a_1 n + a_2 n^2 + a_3 n^3$
	⋮	
n^{th} N-tuple	N-tuple	$a_0 + \ldots + a_n n^n$

Table 2. Constants upon the order of input

Constants↓ Orders→	3	4	5	6	7
α	-9	-12	-15	-18	-21
β	-36	-60	-90	-126	-168
γ	-60	-120	-210	-336	-504

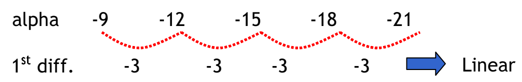


Fig. 3. Pattern of α

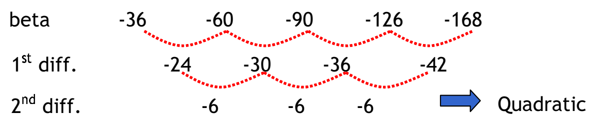


Fig. 4. Pattern of β

unknown parameters are determined.

$$a_0 = 0, \quad a_1 = -3$$

So, the general form of α is

$$\alpha_n = -3n$$

where n is the order of input polynomial.

As for β, the 2^{nd} differences are the same. So β can be generalized in the form of *quadratic*.(Fig.4)

From the general form and with three terms,

$$\beta_n = a_0 + a_1 n + a_2 n^2$$

$$\beta_3 = -36, \;\; \beta_4 = -60, \quad \beta_5 = -90$$

unknown parameters are determined.

$$a_0 = 0, \quad a_1 = -3, \quad a_2 = -3$$

So, the general form of β is

$$\beta_n = -3n - 3n^2 = -3n(n+1)$$

where n is the order of input polynomial.

In the same way, 3^{rd} constant γ has general *cubic* form. From the general form,

$$\gamma_n = a_0 + a_1 n + a_2 n^2 + a_3 n^3$$

the general form of γ is

$$\gamma_n = -2n - 3n^2 - n^3 = -n(n+1)(n+2)$$

Now the constants in equation (3) are given in the general form. To avoid the confusion due to the character n, refer m to the general discrete sampling time and n to the order of the input polynomial $u(m)$. So $u(m)$ has the highest order term m^n.

With total sampling number N,

$$\begin{aligned} u[m] &= (1 + \alpha \cdot K[m-1]) \cdot u[m-1] \\ &+ \beta \cdot K[m-1] \cdot K[m-2] \cdot v[m-1] \\ &+ \gamma \cdot K[m-1] \cdot K[m-2] \cdot K[m-3] \cdot x[m-1] \end{aligned} \tag{5}$$

where

$$K[m-1] = \frac{1}{N - (m-1)}$$

has the constants in the general form of

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} -3n \\ -3n(n+1) \\ -n(n+1)(n+2) \end{bmatrix} \tag{6}$$

The conditions for this RMG are as follows.

1. Three states(acceleration, velocity and position) are all 0 at final time. When continuous connection of trajectories is needed, offset and change of the sign will be applied.
2. Minimum order of the control input is three. In double integrator case here, control input is proportional to acceleration. So position has more than 5^{th} order polynomial equation.
3. The higher the order of input polynomial, the more constraints are needed. Let the accelerations are 0 at more steps before final time. $u[N-1] = 0, u[N-2] = 0, \ldots\ldots$ This means the higher the order of input polynomial the sooner the system become steady.

3.2. Expansion

Focused on the 3^{rd} condition above, it is a drawback to need more control input. As mentioned in section 1, the 5^{th} order polynomial trajectory minimize the integral of jerk square during run time. But most of the high technologies today need smooth motion not in full time but just at final stage. Fast moving but steady final ending make the system more stable and guarantee the accurate motion. Therefore, one step more strict concept than acceleration, jerk, can be considered.

Similar but expanded with RMG procedure, we begin with the expanded jerk equation.

$$j[n] = a_1 + 2a_2 \cdot n + 3a_3 \cdot n^2 + 4a_4 \cdot n^3 + j[0]$$

The results are

$$\begin{aligned} j[m] &= (1 + \alpha \cdot K[m-1]) \cdot j[m-1] \\ &+ \beta \cdot K[m-1] \cdot K[m-2] \cdot K[m-3] \cdot a[m-1] \\ &+ \gamma \cdot K[m-1] \cdot K[m-2] \cdot K[m-3] \cdot v[m-1] \\ &+ \delta \cdot K[m-1] \cdot K[m-2] \cdot K[m-3] \\ &\cdot K[m-4] \cdot x[m-1] \end{aligned} \tag{7}$$

where

$$K[m-1] = \frac{1}{N-(m-1)}$$

Now we have one more constant than the case of RMG based on the acceleration. (Table 3)

Pattern recognition using finite differences are performed in the same way shown already. Then the 4 constants are in

Table 3. Constants upon the order of jerk equation

	Coef. \ Order	4	5	6	7
	Alpha	-16	-20	-24	-28
	beta	-120	-180	-252	-336
	gamma	-480	-840	-1344	-2016
New term →	delta	-840	-1680	-3024	-5040

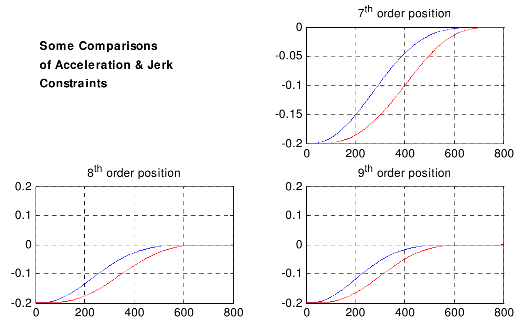


Fig. 5. Comparison of RMG_{acc} and RMG_{jerk}

the general form as follows.

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{bmatrix} = \begin{bmatrix} -4n \\ -6n(n+1) \\ -4n(n+1)(n+2) \\ -n(n+1)(n+2)(n+3) \end{bmatrix} \tag{8}$$

The conditions for this RMG are as follows.

1. Four states(jerk, acceleration, velocity and position) are all 0 at final time. When continuous connection of trajectories is needed, offset and change of the sign will be applied.
2. Minimum order of the control input, jerk in this case, is four. So position has more than 6^{th} order polynomial equation.
3. The higher the order of input polynomial, the more constraints are needed. Let the jerks are 0 at more steps before final time. $j[N-1] = 0, j[N-2] = 0, \ldots\ldots$ This means the higher the order of input polynomial the sooner the system become steady.

For the purpose of distinction, let the former be RMG_{acc}, the latter be RMG_{jerk}.

The comparison of RMG_{acc} and RMG_{jerk} under condition of the same order of the position polynomial.(Fig.5) The lower one is the position trajectory generated by RMG_{jerk}. Both of them satisfy the condition of steady settling at final time. But in the case of RMG_{acc}, it seems to be an over action which requires more input.

Nevil Hogan initially proposed that reaching movements of human arms are planned based on a maximum smoothness criterion that is equivalent to minimizing "jerk", which in turn is the third time derivative of position (or the first time derivative of acceleration). [1]

There are more minute concept, an official term for the first time derivative of jerk is "snap". The second and third time derivatives of jerk (i.e. the 5^{th} and 6^{th} derivatives of position) would naturally be referred to as "crackle" and "pop". High jerk (high changes in acceleration, and therefore high changes in force), can cause substantial damage to dynamic systems, and induce unwanted vibrations. It is also really hard to maneuver on a bus which operates with high jerk. Snap, crackle and pop are less easy to have a physical feel for, although they are considered in the study of human motion. The RMG method can be expanded to the concepts like snap, crackle, pop and more minute physical concepts. For example, the trajectory of the motion of Hubble space telescope is considered with "snap'. Just for showing the ability of RMG method, the snap trajectory generated by RMG_{snap} is followed.

$$\begin{aligned} u[m] = s[m] = & (1 + \alpha \cdot K[m-1]) \cdot s[m-1] \\ & + \beta \cdot K[m-1] \cdot K[m-2] \cdot j[m-1] \\ & + \gamma \cdot K[m-1] \cdot K[m-2] \cdot K[m-3] \cdot a[m-1] \\ & + \delta \cdot K[m-1] \cdot K[m-2] \cdot K[m-3] \cdot K[m-4] \\ & \cdot v[m-1] \\ & + \phi \cdot K[m-1] \cdot K[m-2] \cdot K[m-3] \cdot K[m-4] \\ & \cdot K[m-5] \cdot x[m-1] \end{aligned} \quad (9)$$

where

$$K[m-1] = \frac{1}{N-(m-1)}$$

After pattern recognition with finite differences is applied, the 5 constants are in the general form as follows.

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \\ \delta \\ \phi \end{bmatrix} = \begin{bmatrix} -5n \\ -10n(n+1) \\ -10n(n+1)(n+2) \\ -5n(n+1)(n+2)(n+3) \\ -n(n+1)(n+2)(n+3)(n+4) \end{bmatrix} \quad (10)$$

4. Effects

The trajectories by RMG shows more rapid steady settle with higher order of reference polynomial. (Fig.6) This is easily done just changing the constants in the equation as the wanted order n is changed.(6)
RMG method uses time-varying feedback during run time. This makes the steady settling condition be kept when the system is under saturation of the input. Hiroshi and Tetsuo [2] showed an experimental result about this. Here the simulation results are shown in the acceleration saturated cases.(Fig.7) Compared with not saturated trajectory4(line), the acceleration saturated trajectory(dot line) shows different maneuver but satisfies the final condition. There exists, however, a limit to rising the order of the control input polynomial. When saturation state exists somewhat long period of run time, steady settling states cannot be obtained.(Fig.8)

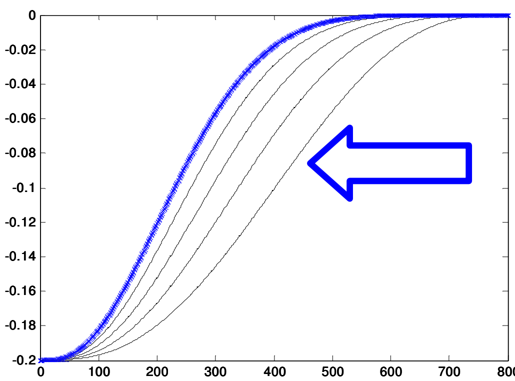


Fig. 6. Trajectory with varying order of RMG_{acc} polynomial

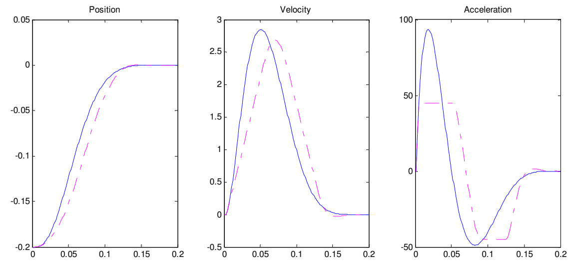


Fig. 7. State trajectories in saturation case (7^{th}order)

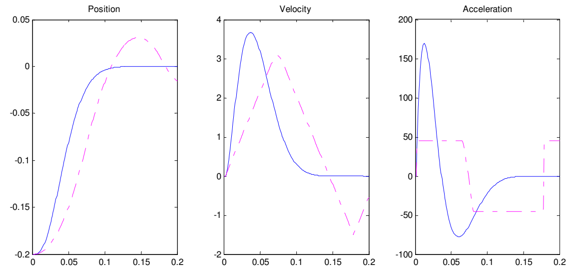


Fig. 8. State trajectories in saturation case (10^{th}order)

Fig. 9. High-accuracy positioning system

5. Simulation Result

The system we are dealing with is the one axis precision linear motor system used in the semiconductor chip mounting devices, which is shown in Fig. 9. The linear motor(ANORAD Corp., LEB–S–2–S–NC) is a direct drive motor with no backlash. The RIC based tracking controller was designed and experimented with the desired trajectory graph given as 5^{th} order polynomial function.[4] The control frequency is set to 1000 Hz and the position is measured by a linear encoder whose resolution is 5 μm.
The mathematical model has 2^{nd} order and the control block diagram is given in Fig.10. Therefore RMG could be placed instead of old 5^{th} order polynomial function and given as a reference trajectory.
3^{rd} order RMG_{acc} is applied because the order of position state is then equal to the case of the old 5^{th} order polynomial. The disturbance attenuation property of RIC is summarized that maximum magnitude of resulting errors with disturbances for each RIC controller gain is changed in the way where it can be verified that if the gain becomes N times, the error is reduced to its $1/N$, approximately. The simulation result obtained in [4] is shown Fig.11.

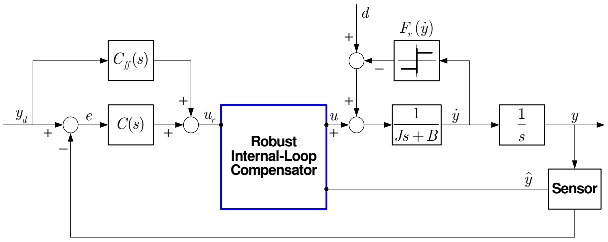


Fig. 10. Block diagram of control system

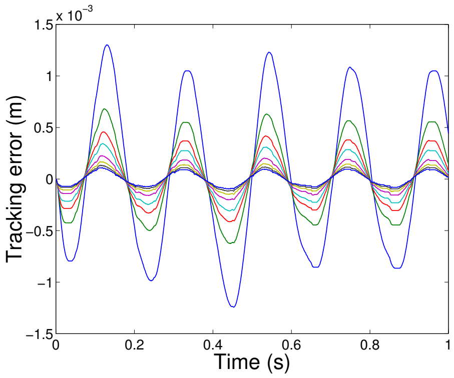
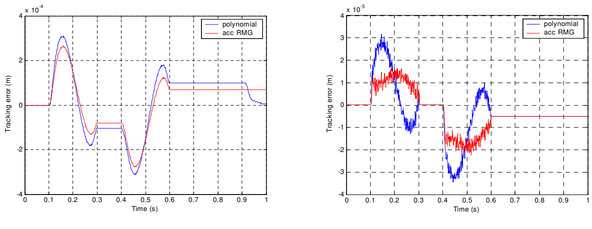
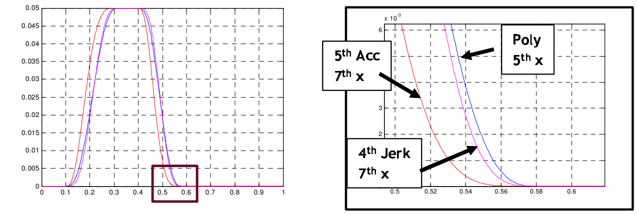


Fig. 11. Tracking error with d_{ex} (simulation)



(a)D=100 (b)D=800

Fig. 12. Error in tracking



(a)polynomial, RMG_{acc}, RMG_{jerk} (b)Enlarged view

Fig. 13. Comparison plot between three references

To estimate the effect of RMG with respect to the old 5^{th} order polynomial, RIC controller gain is fixed and the system was simulated again. As shown in Fig.12, the error is reduced. D means the RIC controller gain. This result means that smooth input effects the response of a control system in better way.

The trajectory generated by RMG_{jerk} is applied to the same system. But although the 5^{th} position trajectory is need to compare with other methods, the minimum order of RMG_{jerk} position state of is 7. So $5^{th}RMG_{acc}$ is used to match with the $4^{th}RMG_{acc}$. Fig.13 shows the references including 5^{th} order polynomial. When determining the order or the shape of the reference trajectory, consideration of required degree of system's steady state is more important to reduce the racking error.

6. Conclusions and future works

Only about trajectory generation method is shown in this paper. To design a good reference model is as important as to design a good controller for tracking problems.

Based on the time-varying state feedback structure, generalized and expanded RMG(Reference Model Generation) method is suggested. Under any steady settling condition in any physical concept, RMG can make a reference state trajectory of any high order. Because only constants are changed with desired order, it significantly reduces the computational load. This is a major advantage when RMG is used to a system of its own CPU or of rapid motion. The simulation result used the same model to design a model-following controller with RIC method for a high-speed high-accuracy system. These type of systems are good to apply RMG method.

There are something remained to study. Mathematically more strict method better than finite difference is needed to derive. To determine the usable highest order under saturation is on consideration.

References

[1] T.Flash and N.Hogan, "The coordination of arm movements: An experimentally confirmed mathematical model," *JNeuro*, vol. 5, pp. 1688–1703, 1985.

[2] H. Uvhida and T. Semba, "Reference model generation using time-varying feedback for model-following control and application to a hard disk drive," in *Proc. 2002 American Control Conference*, pp. 1389–1394, 2002.

[3] H. T. Choi, B. K. Kim, I. H. Suh, W. K. Chung, H. M. Bae, and H. S. Jeong, "Analysis and design of robust internal-loop compensator," in *Proc. 1999 KIEE Annual Summer Workshop*, pp. 586–588, 1999.

[4] B. K. Kim and W. K. Chung, "Performance predictable design of robust motion controllers for high-precision servo systems," *Proc. 2001 American Control Conference*, pp. 2249,2254, 2001.

[5] B. K. Kim, *Design and Performance Tuning of Robust Motion Controllers Based on Structure Analysis*. PhD thesis, POSTECH, 2001.

[6] K. J.Astrom and B. Wittenmark, *Computer-Controlled Systems*. Prentice-Hall,Inc., 3rd ed., 1997.

[7] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*. Addison Wesley, 1990.