RUNOUT COMPENSATION IN PERIPHERAL MILLING USING REPETITIVE CONTROL

By

MIN-CHING HORNG

Bachelor of Science

Oklahoma State University

Stillwater, Oklahoma

1988

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the degree of
MASTER OF SCIENCE
December, 1989

RUNOUT COMPENSATION IN PERIPHERAL MILLING USING REPETITIVE CONTROL

Thesis Approved:

Steven Y Liang

Thesis Adviser

R L Lowery

P M Moretti

Norman N. Durham

Dean of the Graduate College

PREFACE

Runout due to the eccentricity of the rotating machine tool, the spindle axis, and / or the bearings of a spindle axis reduces the precision of machined parts. When the required precision of machined products is not high, runout effect can be tolerated. However, when high precision is required, runout effect needs to be eliminated or reduced to a tolerable level. Conventional controllers such as Proportional-Plus-Integral (PI) and feedback controllers are not adequate for runout compensation, especially when the frequency of runout is high. A new controller, the repetitive controller, had been proposed by other researchers for use in repeated jobs like the tracking of robot manipulators, non-circular machining, and the positioning of computer storage read / write heads. The success of these applications suggests the possibility of applying the repetitive controller on runout compensation for a peripheral milling process. Results are encouraging: They show that, using appropriately selected controller gains and sampling rate, the application is successful.

The research results had been written and sent out as a technical paper and been accepted for publication by Symposium on Advances in Manufacturing

Systems Engineering, the Winter Annual Meeting of the American Society of Mechanical Engineers, San Francisco, December 1989. This thesis is written based on the framework of that paper which has the same title as this thesis. In addition, more simulation results, more functions and derivations, more detailed discussion and explanation, and more references are presented in this thesis. Such additional information will make this thesis much more easier to read and understand.

The paper represents not only my effort but also the collective work of all the individuals who had assisted me and deserve my sincere gratitude. At first, I wish to thank my principal adviser, Dr. Steven Y. Liang, who helped me acquiring knowledge in automatic control and preparing the paper and this thesis. The publication of such paper and the writing of this thesis would be impossible without his help. I also wish to thank the other committee members, Dr. Peter M. Moretti and Dr. Richard L. Lowery, for their helpful instruction, advisement, and supporting teaching assistantships. Special thanks to Professor Emeritus Dr. Jerald D. Parker who admitted me to the School of Mechanical and Aerospace Engineering and provided me helpful advisement and teaching assistantships. I would also like to thank my undergraduate adviser, Mr. Howard E. Conlon, for his kindly help in my undergraduate study here. I am especially grateful to Mr. Mu-Sheng Kow, my senior high class counselor in Taiwan, the Republic of China, who has continuously giving me strong spiritual and part of my financial support since I was at senior high school. A thank-you is also extended to all faculty members who had instructed and passed me their knowledge.

The help of Miss Man Liu, a research assistant at Manufacturing Engineering Center, is especially appreciated. She helped performing the peripheral milling cutting and dynamic modelling presented in Chapter II.

TABLE OF CONTENTS

Chapter Page

LIST OF FIGURES

Figure Page

Figure Page

CHAPTER I

INTRODUCTION

Runout is the dimensional error on a machined part due to the eccentricity of the rotating machine tool, the spindle axis, and / or the bearings of a spindle axis. It is commonly encountered in cutting operations involving rotational tools or workpieces such as turning, drilling, and milling [1-3]. Because of the rotation of

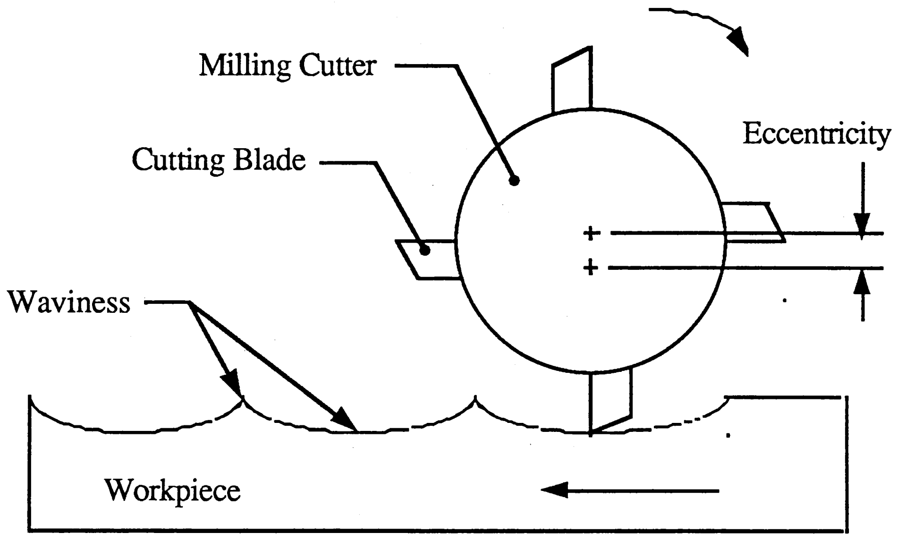

Fig. 1 Waviness Generated on Surface of a Machined part as a Result of Runout

the tools or workpieces, runout is periodic in nature. It generates variation in depth of cut and forms waviness, as illustrated in Figure 1, on machined surfaces. Such waviness reduces the dimensional precision of the parts. The variation in the depth of cut in turn induces vibration and variation in cutting force which may cause fatigue in a cutting tool and shorten its life. If the variation in cutting force is too large, the cutting tool may even be broken.

Runout and other unwanted noise are usually treated together as disturbance [4]. It is a common practice to use a filter to filter or smooth out disturbance signals before they are fed back to a controller [5-6]. Doing this, disturbance signals are simply ignored. Although they are invisible to the controller, their effect is still on the machined parts. When the required precision of a machined part is not very high or the runout is small, filtering is an acceptable practice. However, in the case of a precision machining or large runout, runout needs to be actually compensated for to achieve good dimensional accuracy. Some researchers proposed to deal with runout by using integral, feedforward, and feedback controllers [4-7]. So far, the best results is the reduction of runout effect by one order. To further reduce or eliminate runout effect, new controllers need to be implemented.

A repetitive controller is a controller capable of adjusting the controller controlling commands in the current period according to the tracking error signals in previous periods so that it can reduce the tracking error period by period. The adjusting action is repeated until the tracking error vanishes. Repetitive controllers had been successfully applied on the tracking of robot manipulators [8], non-circular turnings [9], and the positioning of computer storage read / write heads [10]. They have not been applied on a milling process yet. In this thesis the application of a repetitive controller on a peripheral milling process is studied. This Thesis is devoted exclusively on the design of a digital repetitive controller for runout compensation. Readers who are interested in continuous-time repetitive controllers are referred to listed references [11-14].

In the next chapter, the experimental set-up for conducting a peripheral milling and the procedures in deriving the milling plant dynamics are described. Chapter III presents a PI controller design for enhancing the stability and transient performance of the peripheral milling process. In Chapter IV, the characteristics of the milling cutter runout and its effect on the performance of a PI controller is studied. In Chapter V, a working repetitive control scheme along with its constraints for successful design are presented. In Chapter VI, the milling force responses to different repetitive controller gains and sampling rates are presented. Simulation results are presented in graphical forms throughout Chapter III, IV, V, and VI for easier understanding and comparison. Conclusions are given in Chapter VII.

CHAPTER II

EXPERIMENTAL SET-UP AND DYNAMIC MODELLING

Introduction

In order to design an appropriate controller, it is required to identify the dynamic model of an object process it is designed to control. A dynamic model is the mathematical equation which represents or, in most cases, approximates the dynamics of a process. It can be presented in either continuous-time or discrete-time format. For digital control, a discrete time model is easier to implement. Because in this study a personal computers is used to implement the repetitive controller, the digital model is used. To obtain a dynamic model with sufficient accuracy, actual cutting operation should be performed. During cutting, useful output signals such as force output signals and controller controlling commands are measured and recorded by using a dynamometer. The recorded data are then fitted

The experimental work and dynamic modelling was performed by Miss Man Liu. This chapter is presented for continuity and clarity of this thesis.

through a proper statistical algorithm to get the parameters of the required dynamic model.

There are many methods available for identifying the dynamic model. Least squares, maximum likelyhood, dynamic data system, impulse response, and step response methods are part of the list [15-17]. The least square method is the simplest and the most frequently used algorithm. It is also the method used for identification in this study. This thesis is not intended to describe and compare all these identification methods. Interested readers are referred to other papers [15-22]. An identified model can be presented in first order, second order, or other higher order forms [23-25]. It had been proposed that a second-order model is appropriate for a slot milling process. Because a peripheral milling process is very similar to the slot milling, the second-order model is adopted with the model parameters modified to fit the specific cutting conditions set in this study.

Experimental Set-up

In this study, the machine used for performing the experimental peripheral milling operation is a Bridgeport Interact 412V CNC machine center located in the Manufacturing Engineering Center of Mechanical and Aerospace Engineering Research Laboratory. Figure 2 shows a schematic of the peripheral milling process. The milling operation was carried out on a 7075 aluminum workpiece with a four-flute, 3 / 8 inch diameter, high speed steel milling cutter. The body of the milling cutter and the four cutting flutes form an integral milling cutter. The spindle speed of the milling cutter was 600 rpm (10 HZ). The nominal feedrate of the milling table was 4.5 inches per minute, while the nominal depth of cut was 0.03 inch. No cutting fluid was used.

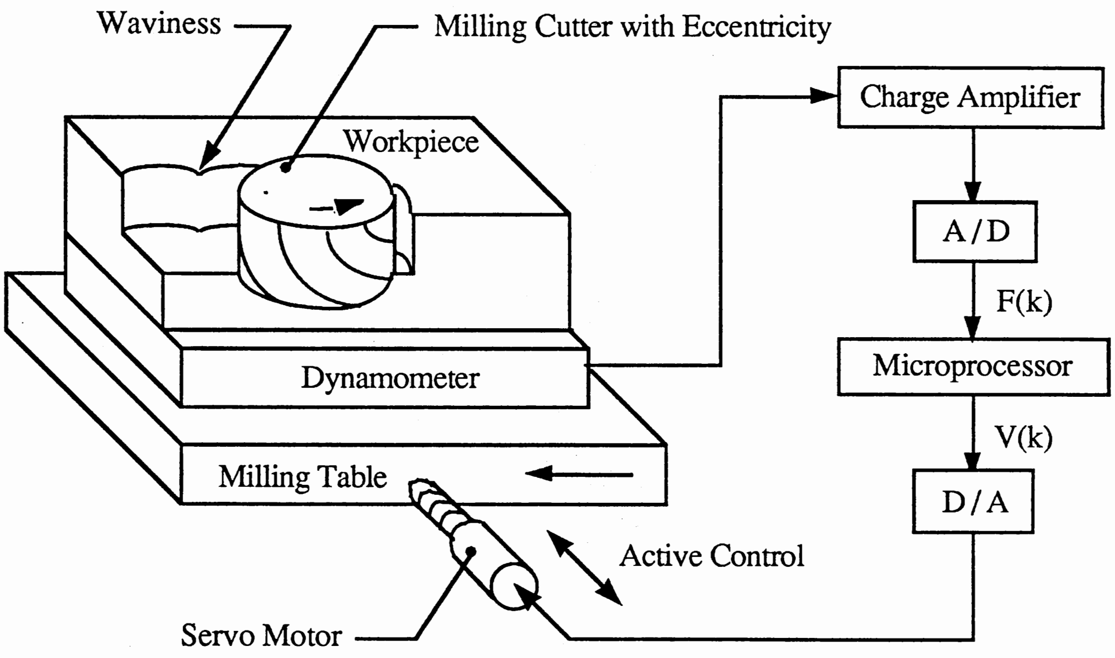

Fig. 2 Schematic of the Experimental Set-up for Peripheral Milling

Force signals were measured by using a Kistler 9257A dynamometer mounted between the milling table and the workpiece. The measured force signals were filtered by a first-order low pass filter with a 30 HZ corner frequency to reduce the signals due to the engagement and disengagement of the individual cutting flute with the workpiece and other unwanted high frequency noise. Because runout signals are at a frequency of 10 HZ which is lower than 30 HZ, they will not be filtered out. The filtered signals and the measured CNC feedrate commands, in millivolts (mV), were then sampled by a computer at a sampling rate of 60 HZ. Such a sampling rate is appropriate for the following reasons. First, it is three times the minimum sampling rate requirement, the Nyquist frequency, which is 20 HZ in this study. Second, the sampling rate is well within the capacity of the computer for the simple data acquisition task in this experimental cutting.

Dynamic Modelling

To ensure adequate accuracy of an identified process dynamics, the input signals to a controlled process should be able to excite all modes of the process [15]. In other words, the input signals should be as random as possible. In this study, CNC feedrate commands in the direction of the depth of cut are the input signals. The requirement of randomness is fulfilled by using a pseudo random binary sequence (PRBS) as the CNC feedrate commands.

In a peripheral milling process, the process dynamics is primarily due to the dynamics of chip formation, the compliance of the machine tool, and the elasticity of the workpiece [26-27]. For real-time control, the process dynamic model should be kept as simple as possible especially when the cutting speed is high or the workpiece geometry is complex. A simple second order digital dynamic model for a slot milling process had been proposed by Lauderbaugh and Ulsoy [25] as following:

$$G(q^{-1}) = \frac{F(k)}{V(k)} = \frac{q^{-1}(b_0 + b_1 q^{-1})}{1 + a_1 q^{-1} + a_2 q^{-2}} = \frac{q^{-1} B(q^{-1})}{A(q^{-1})} \tag{1}$$

where

$G(q^{-1})$ is the transfer function between F(k) and V(k);

F(k) is the force output at the kth step;

V(k) is the feedrate command at the kth step;

q^{-i} is the i steps delay operator;

a_1, a_2, b_0, and b_1 are the process parameters; and

$A(q^{-1})$ and $B(q^{-1})$ are the denominator and numerator polynomials of the transfer function.

To identify the parameters a_1, a_2, b_0, and b_1, the least-squares algorithm is applied. A least-squares algorithm is an algorithm that identifies the unknown process parameters in such a way that "the sum of the squares of the differences between the actually observed and computed values multiplied by numbers that measured the degree of precision is a minimum [15]." In equation form, the least-square algorithm is an algorithm that minimizes the penalty function J:

$$J = \sum_{k=1}^{n} [\, F(k) - \underline{\phi}^{T}(k\text{-}1)\, \hat{\underline{\theta}}\,]^{2} \qquad (2)$$

where

n is the total number of samples;

F(k) is the force measured at the kth step;

$\underline{\Phi}^{T}$(k-1) is the transpose of $\underline{\phi}$ matrix at the (k-1)th step;

$$\underline{\phi}\,(k\text{-}1) = \begin{bmatrix} F(k\text{-}1) \\ F(k\text{-}2) \\ V(k\text{-}1) \\ V(k\text{-}2) \end{bmatrix}; \qquad (3)$$

and

$$\hat{\underline{\theta}} = \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \\ \widehat{b}_0 \\ \widehat{b}_1 \end{bmatrix}. \qquad (4)$$

The solution for $\hat{\underline{\theta}}$ is given [28] as:

$$\hat{\underline{\theta}} = [\sum_{k=1}^{n} \underline{\phi}(k-1)\,\underline{\phi}^{T}(k-1)\,]^{-1} \sum_{k=1}^{n} F\,(k-1)\,\underline{\phi}\,(k-1) \qquad (5)$$

The structure of the plant dynamic model in this study is expected to be similar to that of equation (1). However, because the cutting conditions are different, the parameters in the plant model for the peripheral milling process are expected to be different from those of equation (1). After performing the experimental peripheral milling and fitting the obtained data by using equations (2), (3), (4), and (5), the parameters for the plant model are identified as:

$$\hat{\underline{\theta}} = \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \\ \hat{b}_0 \\ \hat{b}_1 \end{bmatrix} = \begin{bmatrix} -0.7537 \\ -0.2404 \\ 9.0452 \\ -7.8975 \end{bmatrix} \qquad (6)$$

Using these parameters, equation (1) can be rewritten as:

$$G(q^{-1}) = \frac{F(k)}{V(k)} = \frac{q^{-1}(9.0452 - 7.8975\,q^{-1})}{1 - 0.7537\,q^{-1} - 0.2404\,q^{-2}} \qquad (7)$$

or

$$F(k) = 0.7537\,F(k-1)+0.2404\,F(k-2)+9.0452\,V(k-1) - 7.8975\,V(k-2) \qquad (8)$$

The two zeros of the plant model are $z_1 = 0$ and $z_2 = 0.8731$. The two poles are $z_3 = 0.9952$ and $z_4 = -0.2415$. Because $z_3 = 0.9952$ is very close to unity, it indicates that the stability margin of the peripheral milling process is small. This pole is likely to be contributed by the integration from feedrate to chip formation and is expected to affect the asymptotic stability of the milling process. This is verified by Figure 3 which shows the simulated force response of the identified model to a unit step feedrate command.

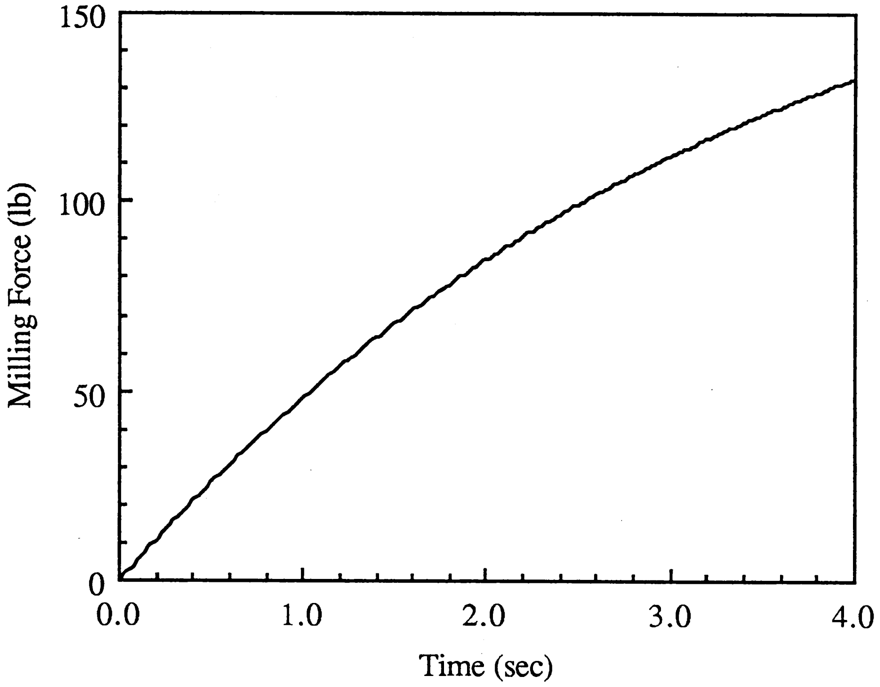

Fig. 3 Milling Force Response to a Step Feedrate Command

Because of the nearly unstable pole, the response shows no sign of convergence. This is a problem which needs special attention when designing a controller.

CHAPTER III

PERIPHERAL MILLING PROCESSES UNDER PI CONTROL

Introduction

Before designing a repetitive controller, the identified model represented by equation (7), is put under PI control. The purpose of using the PI controller is to increase the stability margin of the process by doing an appropriate pole placement. A pole placement is the assigning of the closed-loop poles of a control system by properly choosing its controller gains. For a PI control system, the controller gains are K_p and K_I which is equals to α times K_p. To design a PI controller by pole placement, one first selects a set of poles, gets the corresponding transfer function or functions, and then simulates the system with proper input to get simulation responses. Usually, either a discrete-time or continuous-time root locus is plotted to help locating the poles of the system. All these procedures are repeated until satisfactory responses are obtained.

PI Controller Design

A block diagram for the peripheral milling process under PI feedback control is shown in Figure 4.

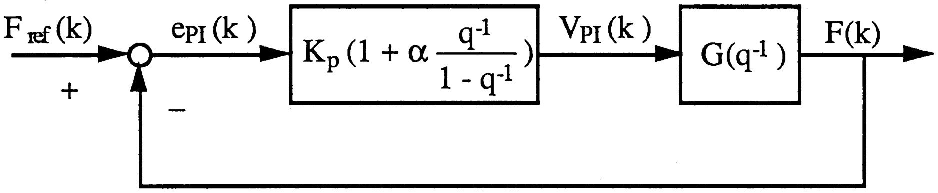

Fig. 4 Block Diagram of Peripheral Milling Process under PI Control

The transfer function for the PI controller can be expressed as:

$$G_{PI}(q^{-1}) = \frac{V_{PI}(k)}{e_{PI}(k)} = K_P\,(1 + \alpha \frac{q^{-1}}{1 - q^{-1}}) \qquad (9)$$

where

$V_{pI}(k)$ is the PI controlling command at the kth step;

$e_{pI}(k)$ is the error between $F_{ref}(k)$ and F(k) at the kth step;

$F_{ref}(k)$ is the reference force at the kth step;

F(k) is the force output at the kth step;

K_p is the proportional gain of the PI controller; and

α is the proportional constant between the integral and the proportional controllers.

For computer simulation, an equation in causal form is preferred. The causal form of equation (9) is:

$$V_{PI}(k+1) = V_{PI}(k) + K_P(\alpha - 1)\, e_{PI}(k) + K_P\, e_{PI}(k+1) \tag{10}$$

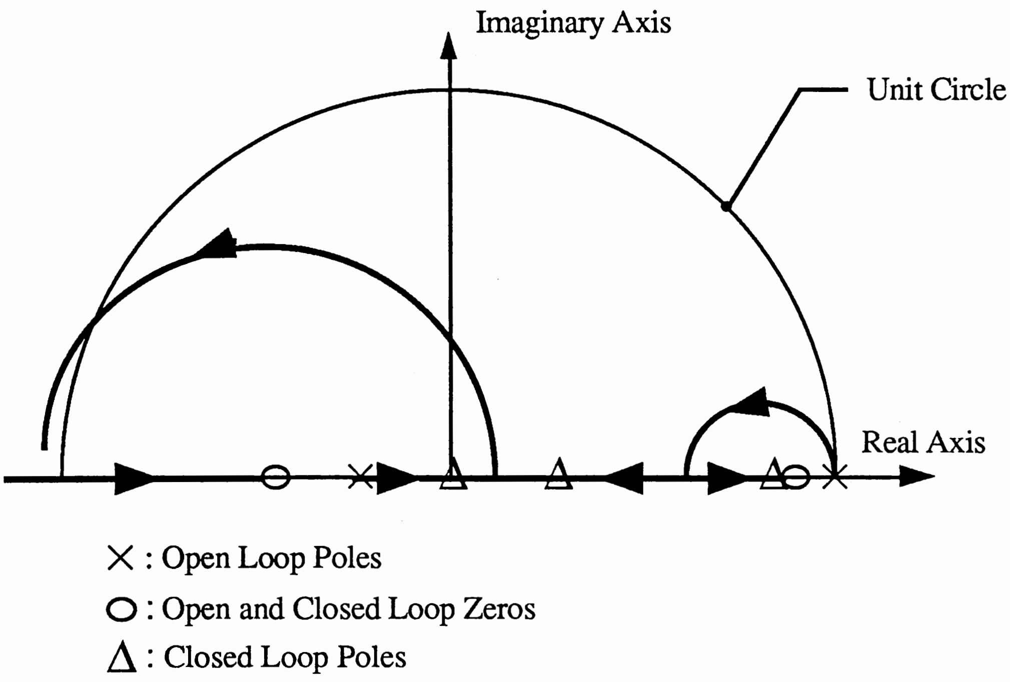

Fig. 5 Discrete-Time Root Locus Plot for Peripheral Milling Process under PI Control

Assigning a value of 1.5 for α , the discrete-time root locus can be plotted as Figure 5. Figure 5 is only the top portion of the original root locus plot. Because the bottom half of the root locus plot is symmetric to the top one, it is a common practice to draw only the top one for simplicity. Choosing $K_p = 0.064$, the causal equation, equation (10), becomes:

$$V_{PI}(k+1) = V_{PI}(k) + 0.032\ e_{PI}(k) + 0.064\ e_{PI}(k+1) \tag{11}$$

The overall transfer function becomes:

$$G(q^{-1}) = \frac{F(k)}{F_{ref}(k)} = \frac{z^{d}Q(z^{-1})}{P(z^{-1})} = \frac{G_{PI}(z^{-1})\,G(z^{-1})}{1 + G_{PI}(z^{-1})\,G(z^{-1})}$$

$$= \frac{z^{-1}[\,0.5789\,(1 - 0.3731\,z^{-1} - 0.4365\,z^{-2})\,]}{1 - \ 1.1748\,z^{-1} + 0.2973\,z^{-2} - 0.0123\,z^{-3}} \tag{12}$$

Equation (12) has two zeros which are $z_1 = 0.8731$ and $z_2 = -\ 0.5000$. It has three poles which are $z_3 = 0.8373$, $z_4 = 0.2862$, and $z_5 = 0.0513$. Adding the PI controller increases the order of the original process. The original process has only one zero and two poles, while the PI controlled one has two zeros and three poles. Adding one additional order, will introduce one additional delay step and thus delay the response for one step. However, because the new poles are now away from the stability limit, the unit circle, the stability of the peripheral milling process is expected to be improved.

Simulation Results and Discussion

Figure 6 is the simulated force response of the new system to a 20 lb step reference cutting force.

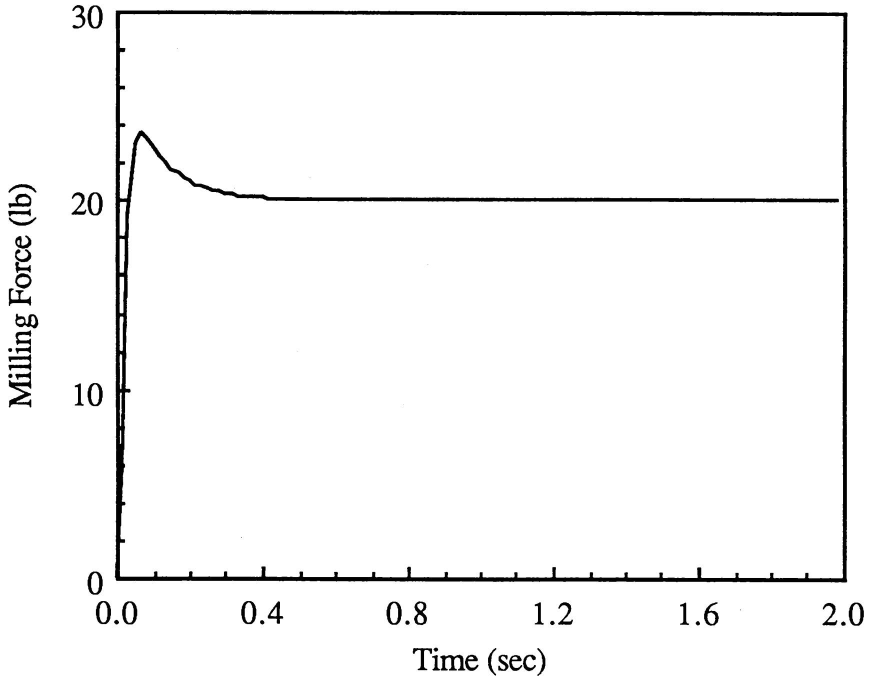

Fig. 6 PI Controlled Milling Force Response to a 20 lb Step Input

As expected, Figure 6 shows that PI controlled peripheral milling process is stable with no steady-state error. The response has an overshoot less than 20% and a 10% settling time within 0.2 second. Such a response is satisfactory for this study. In case that lower overshoot or smaller settling time is required, parameters K_p and α can be readjusted by repeating procedures depicted in last section. However, lower overshoot is accompanied by longer settling time, while smaller settling time leads to larger overshoot. It is impossible to achieve smaller overshoot and smaller settling time at the same time. Compromise has to be made between overshoot and settling time.

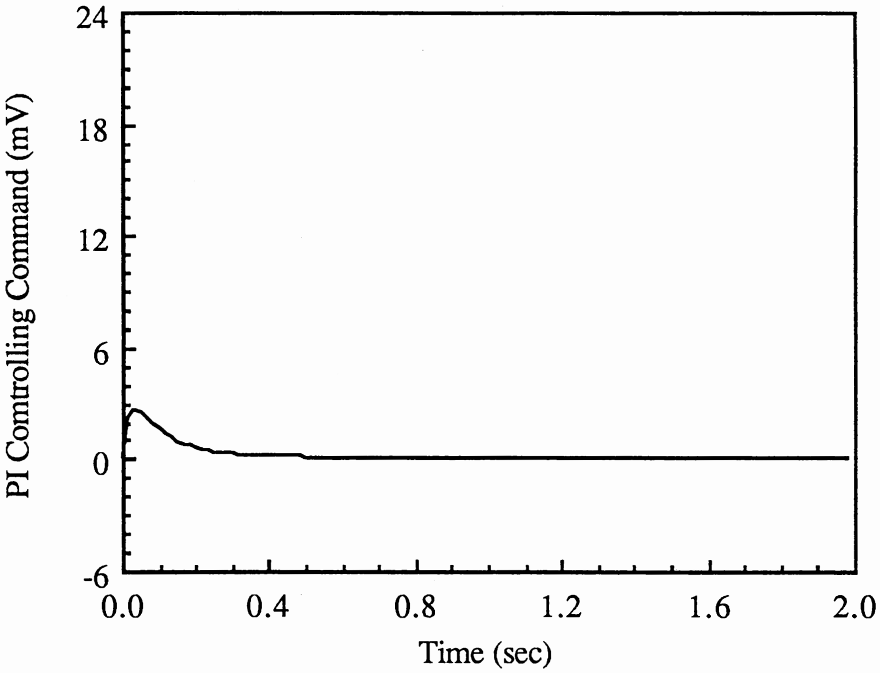

Fig. 7 PI Feedrate Controlling Command in Response to a 20 lb Step Input

Feedrate commands issued by the PI controller on the time basis are plotted in

Figure 7. Figure 7 shows that the feedrate command is zero at the very beginning. It reaches the maximum, 3 mV, in less than 0.03 second then gradually dies down as the tracking error, e_{pI} (k), decays. For the time interval beyond 0.4 second, the PI controlling commands all equal to 0.1. Because the tracking error has all dies down to zero, there is nothing to be corrected. Therefore, no change in the controlling commands is expected.

There is one important thing which needs to be mentioned. When performing simulation, one should pay attention to the maximum and minimum output voltage limits of the PI controller. When either of its limit is exceeded, the controlling commands issued by the PI controller will be saturated at this limit. All intended controlling voltages higher than the maximum or lower than the minimum limit will be sent out at the limiting value instead. This gives rise to significant error in the simulated results if such limits are not accounted for. When this happens, an IF statement like:

```
IF controlling command > upper limit
THEN controlling command = upper limit
ELSE IF controlling command < lower limit
THEN controlling command = lower limit
```

can be inserted into proper position inside the control algorithm to take care of such a saturation problem. The digital port of the personal computer used for performing the PI control in this study has a output voltage capacity of +/- 50 mV. Figure 7 shows that the predicted PI feedrate controlling commands are all less than 3 mV which is well within the capacity of the computer. Therefore, no serious error in the simulation results are expected.

CHAPTER IV

PERIPHERAL MILLING PROCESS WITH CUTTER RUNOUT

Introduction

In a peripheral milling process, runout is caused by either the mispositioning of cutting blades or the spindle eccentricity. Runout caused by the spindle eccentricity has the same frequency as that of the spindle rotation. Runout caused by the mispositioning of cutting blades has a frequency equivalent to the number of cutting blades times the spindle frequency. As mentioned in Chapter II, the milling cutter used for the experimental peripheral milling for this study has four flutes on its hub. No inserts are used. Therefore, runout created by the mispositioning of cutting blades is not existed. The actual relationship between runout and cutting force is very complicated and is not explored here. However, even without the precise knowledge of such a relationship, sufficient general characteristics of runout still can be drawn for the design of a successful repetitive controller for runout compensation. Some useful general characteristics of runout due to the eccentricity of a milling cutter can be understood by simulating and studying the effect of runout on the PI controlled peripheral milling process discussed in last chapter.

Characteristic of Runout in Peripheral Milling

In the experimental cutting, the spindle speed of the milling cutter was set at 600 rpm. Therefore, runout due to the eccentricity of the spindle also has a frequency of 600 rpm. The amplitude of runout is twice the spindle eccentricity. Since the frequency of the variation in cutting force is the same as that of runout, the frequency of the variation in cutting force should also be 10 HZ. There is a phase lag between runout and cutting force. This phase lag depend on the actual cutting conditions. In general, the magnitude of the variation in cutting force is proportional to runout signals [29]. The precise model explaining how the magnitude of the cutting force varies in accordance with runout is very complicated. It requires detailed analysis of the cutter topology, workpiece material properties, and related cutting conditions. This is not the object of this study. Interested readers are referred to Shaw [30] for further explanation. Indeed, as will be proved later, the precise waveform and phase angle of runout and its induced varying forces are not critical as long as they are kept periodic. This characteristic enables the design of a working repetitive controller without the precise knowledge of the mechanism creating the variation in cutting force.

Since the precise shape and phase angle of runout is not critical, a sinusoidal signal, $\sin(\omega kT + 2\pi / 3)$, is used for simplicity to simulate the variation in feedrate due to runout. In this given sinusoidal signal, ω represents the spindle speed (10 HZ in this study), k is the step number, T is the sampling period (16.6666667 ms in this study), while $2\pi / 3$ is the arbitrarily selected phase lag of runout with respect to spindle rotation. Runout signals are assumed to have an amplitude of 0.30 mV which stirs up a variation in force output with an amplitude of about 6 lb, Figure 9,

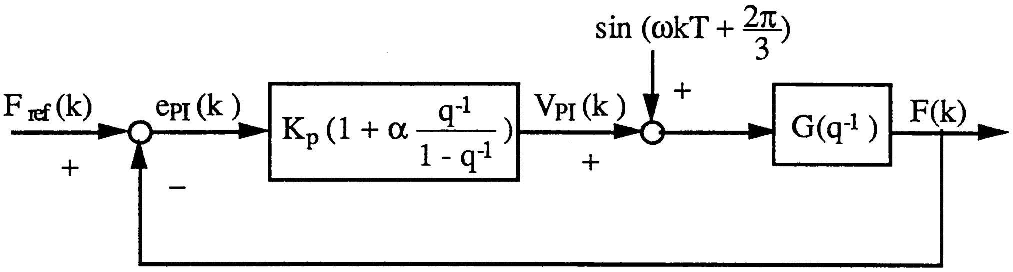
$\sin(\omega kT + \frac{2\pi}{3})$
$F_{ref}(k)$
+
$e_{PI}(k)$
−
$K_p(1 + \alpha \frac{q^{-1}}{1 - q^{-1}})$
$V_{PI}(k)$
+
+
$G(q^{-1})$
$F(k)$

Fig. 8 Block Diagram of Peripheral Milling Process under PI Control with Sinusoidal Runout

which is reasonable for this study. Adding this prescribed sinusoidal runout signal to the PI controlled milling process shown in Figure 4, the new control system is plotted in Figure 8.

Simulation Results and Discussion

To further understand the effect of the added runout to the stabilized PI controlled milling process, another simulation is performed. Comparing Figure 4 and Figure 8, it is found that equation (10) is still valid. However, equation (8) needs to be modified as:

$$F(k) = 0.7537\ F(k\text{-}1) + 0.2404\ F(k\text{-}2) \\ + 9.0452\ [\ V(k\text{-}1) + W(k\text{-}1)\ ] - 7.8975\ [\ V(k\text{-}2) + W(k\text{-}2)\ ] \qquad (13)$$

Simulate the new system with a 20 lb step reference cutting force by using equations (11) and (13). The milling force response is shown in Figure 9 and the feedrate controlling commands is plotted in Figure 10. Comparison of Figure 9 with Figure 6 reveals several informative features. First, the PI controller is unable to eliminate the effect of runout because the figure shows that the steady state force response vibrates with an amplitude of about 6 lb. Second, the mean values of the force response of the new system do conform with those of the original system. Third, the frequency of the variation of the cutting force is the same as the imposing runout. This confirms previous argument that the two frequencies are the same. The PI controller is busy in issuing the oscillatory feedrate controlling commands all the time trying to compensate for incoming runout.

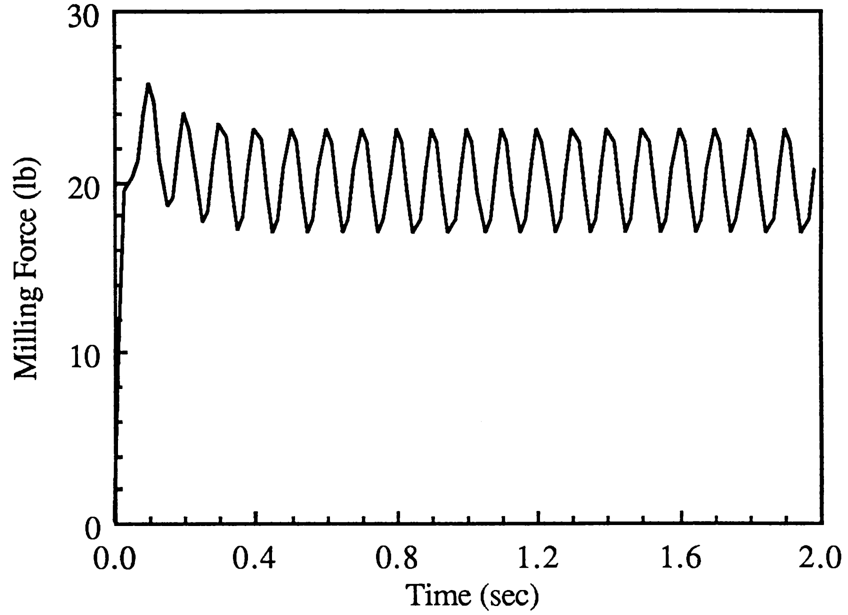

Fig. 9 PI Controlled Milling Force Response
to a 20 lb Step Input with
Sinusoidal Cutter Runout

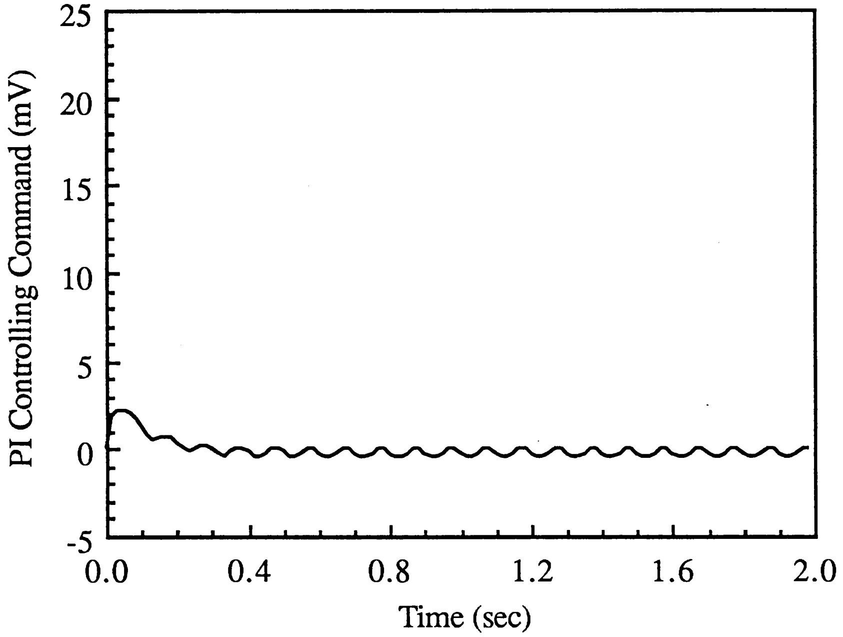

Fig. 10 PI Feedrate Controlling Command
in Response to a 20 lb Step Input
with Sinusoidal Cutter Runout

However, the controlling commands never die out. This is another confirmation that a PI controller alone is not sufficient for runout compensation. Comparison of Figure 9 and Figure 10 reveals that although the oscillatory PI controlling commands do have a frequency of 10 HZ, they are in opposite phase to that of the force response. This is due to the effort of the PI controller trying to counteract the effect of runout.

CHAPTER V

REPETITIVE CONTROLLER DESIGN FOR RUNOUT COMPENSATION

Introduction

A repetitive controller is a controller capable of adjusting the controlling commands in the current period according to the tracking error signals in previous periods so that it can reduce the tracking error period by period. The feature of making use of information from previous periods is called "leaning" [31-32]. The learning and correction action is repeated until the tracking error vanishes. The learning capability of a repetitive controller is due to the embedded internal signal generator [33] which has the same frequency as the process it is designed to control. The internal signal generator is also a requirement for obtaining the asymptotic stability [10-11]. To ensure the asymptotic stability of a repetitive controller, other constraints should also be complied in selecting the controller parameters [9-12]. Selecting parameters according to those constraints, the transfer function and causal equation for computer simulation can be derived. As usual, the goodness of the controller design can be verified by performing simulations.

The Internal Signal Generator

A repetitive controller is equipped with an internal signal generator which has the same frequency as that of the controlled process [33]. Such an internal signal generator is essential for ensuring the asymptotic stability, because when tracking error is finally reduced to zero the repetitive controller must be able to generate repetitive signals to exactly cancel out runout signals. The internal signal generator is also the mechanism enabling the learning capability. An internal signal generator is of the following form:

$$G_{\text{internal}}(Z^{-1}) = \frac{k_r z^{-N}}{1 - z^{-N}} \tag{14}$$

where

k_r is the internal signal generator gain;

N is the number of sampling steps in one period.

Incorporating such a transfer function of an internal signal generator, the transfer function of a repetitive controller for a process without unacceptable zero can be expressed as:

$$G_r(z^{-1}) = \frac{k_r\, z^{-N+d}\, P(z^{-1})}{(1 - z^{-N})\, Q(z^{-1})} \tag{15}$$

where

k_r is now called the repetitive controller gain;

d is the number of delay steps in the controlled process;

$P(z^{-1})$ and $z^{-d}\, Q(z^{-1})$ are the denominator and numerator polynomials of the controlled process seen by the repetitive controller.

It has been proved [9-12] that for asymptotic stability, it is required that:

$$0 < k_r < 2 \tag{16}$$

Repetitive Controller Design

A schematic of a repetitive control system is shown in Figure 11.

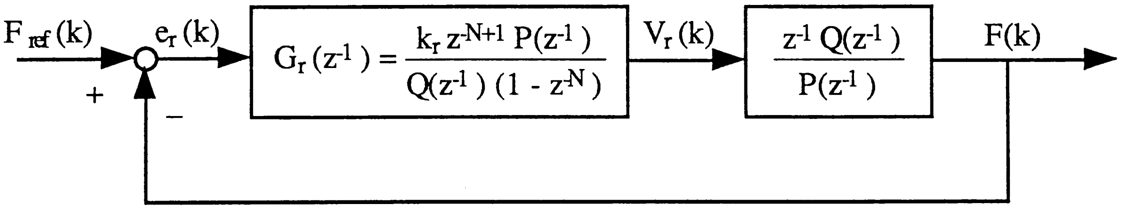
$F_{ref}(k)$
$e_r(k)$
$G_r(z^{-1}) = \frac{k_r z^{-N+1} P(z^{-1})}{Q(z^{-1})(1 - z^{-N})}$
$V_r(k)$
$\frac{z^{-1} Q(z^{-1})}{P(z^{-1})}$
$F(k)$
+
–

Fig. 11 Block diagram of Peripheral Milling Process under Repetitive Control

The open loop transfer function for Figure 11 is:

$$G_{open}(z^{-1}) = [\frac{k_r z^{-N+d} P(z^{-1})}{(1 - z^{-N}) Q(z^{-1})}] = \frac{k_r z^{-N}}{1 - z^{-N}} \tag{17}$$

which is exactly the same as equation (14). Therefore, one of the requirement for achieving asymptotic stability of the system is met.

Equation (17) implies that pole-zero cancellation is involved in a repetitive control system. A control system involving pole-zero cancellation is very sensitive to unstable zero and model-plant mismatch [34]. The unstable zero becomes the unstable pole in the repetitive controller transfer function. It causes the whole control system to become unstable. The problem that model-plant mismatch may cause could be seen by analyzing Equation (17). The right hand side expression in Equation (17) is derived by canceling the P's and Q's. If there is model-plant mismatch, the P and Q in the transfer function of the repetitive controller are not the same as those of the plant. They can not be canceled out. Therefore, the use of the right hand expression in equation (17) as the transfer function of the open loop is incorrect. This will results in unexpected or even unstable responses depending on how serious the model-plant mismatch is. The problem of model-plant mismatch can be solved by using an adaptive control algorithm. Whenever there is any unstable zero in the controlled plant, a prototype repetitive controller

[8-10] of the following form can be used.

$$G_r(z^{-1}) = \frac{V_r(k)}{e_r(k)} = \frac{k_r z^{-N+d} P(z^{-1})\, Q(z)}{(1 - z^{-N})\, Q^+(z^{-1})\, b} \tag{18}$$

where

Q(z) is an expression obtained by replacing z^{-1} in $Q(z^{-1})$ by z in the following equation;

$$Q(z^{-1}) = Q^+(z^{-1})\, Q(z^{-1}) \tag{19}$$

and

$V_r(k)$ is the repetitive controlling command at the kth step;

$e_r(k)$ is the tracking error at the kth step;

$Q^+(z^{-1})$ is the part of $Q(z^{-1})$ without unstable zero;

$Q^-(z^{-1})$ is the part of $Q(z^{-1})$ with unstable zero; and

$$b \geq \max_{\omega \in [0,\, \pi]} |B^-(e^{-j\omega})|^2 \tag{20}$$

The schematic of the repetitive control process represented by equation (18) can still be represented by Figure 12. The open loop transfer function for Figure 12 becomes:

$$G_{open}(z^{-1}) = [\frac{k_r z^{-N+d} P(z^{-1}) Q(z)}{(1 - z^{-N}) Q^{+}(z^{-1}) b}] [\frac{z^{-d} Q^{+}(z^{-1}) Q^{-}(z^{-1})}{P(z^{-1})}]$$

$$= \frac{k_r z^{-N} Q(z) Q^{-}(z^{-1})}{(1 - z^{-N}) b} \qquad (21)$$

As seen from equation (21), the internal signal generator expressed by equation (14), which is essential for the asymptotic stability, is still left untouched. The denominator b in the following expression:

$$\frac{Q(z) Q^{-}(z^{-1})}{b}$$

is for compensating the effect of unstable zero in $Q(z^{-1})$ and Q(z) so that the tracking error will eventually vanish. In this study, no unacceptable zero is involved. Besides, even when unacceptable zeros are existed, a PI controller can be applied first to remove the unacceptable zeros. Therefore, the use of equation (18) is not required. However, as discussed in Chapter III, using a PI controller will increase the order of the overall system and thus increase the settling time. This is one factor needs to be considered before making the choice between using equation (15) or equation (18). The procedures about the design of a PI controller for removing the unacceptable zero are the same as those discussed in Chapter III. Because no unacceptable zeros are involved, equation (15) is appropriate for designing the repetitive controller for this study.

Replacing $\frac{z^{-1}Q(z^{-1})}{P(z^{-1})}$ in Figure 11 by equation (11), Figure 11 and Figure 8 now can be combined as Figure 12.

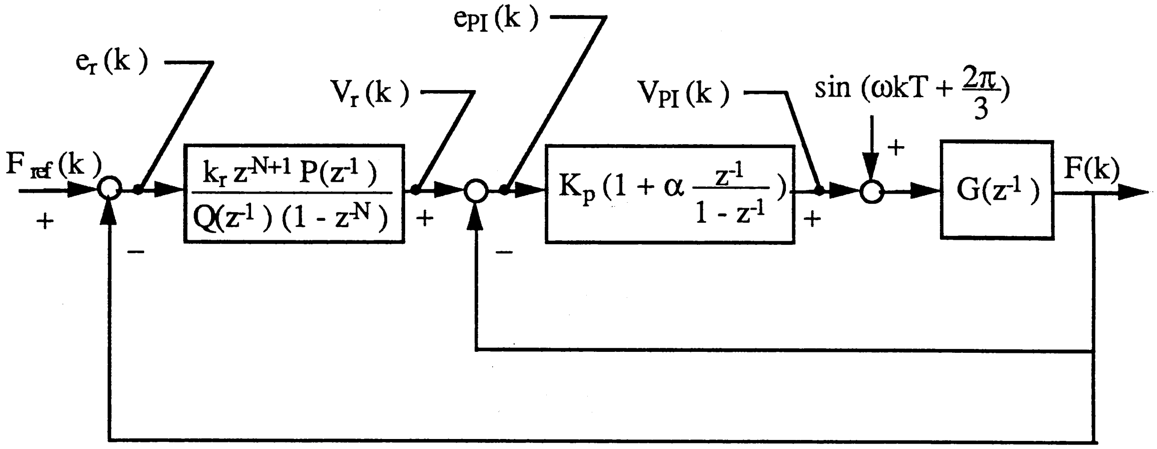
$e_r(k)$
$V_r(k)$
$e_{PI}(k)$
$V_{PI}(k)$
$\sin(\omega kT + \frac{2\pi}{3})$
$F_{ref}(k)$
$\frac{k_r z^{-N+1} P(z^{-1})}{Q(z^{-1})(1 - z^{-N})}$
$K_p(1 + \alpha \frac{z^{-1}}{1 - z^{-1}})$
$G(z^{-1})$
$F(k)$
+
−
+
−
+
+

Fig. 12 Block Diagram of Peripheral Milling Process under Combined Repetitive and PI Control with Sinusoidal Runout

Using equation (11), equation (15) can be rewritten as:

$$G_r(z^{-1}) = \frac{V_r(k)}{e_r(k)} = \frac{k_r z^{-N+1}(1 - 1.1748\, z^{-1} + 0.2973\, z^{-2} - 0.0123\, z^{-3})}{(1 - z^{-N})\,[\,0.5789\,(\,1 - 0.3731\, z^{-1} - 0.4365\, z^{-2})\,]} \qquad (22)$$

According to equation (16), choosing $k_r = 0.8$ is legal. If choosing $N = 6$, the sampling frequency is higher than the required lowest frequency, the Nyquist frequency, which is 20 HZ in this study.

Equation (22) now can be rewritten in the transfer function form as:

$$G_r(z^{-1}) = \frac{V_r(k)}{e_r(k)} = \frac{0.8\ z^{-5}(1 - 1.1748\ z^{-1} + 0.2973\ z^{-2} - 0.0123\ z^{-3})}{(1 -\ z^{-N})\ [\ 0.5789\ (\ 1 - 0.3731\ z^{-1} - 0.4365\ z^{-2})\ ]} \tag{23}$$

or in causal form as:

$$\begin{aligned} V_r\,(k+1) &= 0.3731\ V_r\,(k) + 0.4365\ V_r\,(k-1) + V_r\,(k-5) \\ &\quad - 0.3731\ V_r\,(k-6) - 0.4365\ V_r\,(k-7) + 1.3819\ e_r\,(k-4) \\ &\quad - 1.6234\ e_r\,(k-5) + 0.4108\ e_r\,(k-6) - 0.0169\ e_r\,(k-7) \end{aligned} \tag{24}$$

Equation (24) clearly shows that the current repetitive controlling command, V_r (k+1), is based not only on the results in the kth and the (k+1)th steps but also those in the (k-4)th, (k-5)th, (k-6)th, and (k-7)th steps. This demonstrates that the adjusting action of a repetitive controller is based on its "learning" capability.

According to Figure 12, the equation to be used for simulation are equation (24) and the followings:

$$\begin{aligned} F(k+1) &= 0.7537\ F(k) + 0.2404\ F(k-1) \\ &\quad + 9.0452\ [\ V(k-1) + W(k-1)\ ] - 7.8975\ [\ V(k-2) + W(k-2)\ ] \end{aligned} \tag{25}$$

$$e_r\,(k+1) = F_{ref}\,(k+1) - F(k+1) \tag{26}$$

$$e_{PI}(k+1) = V_r(k+1) - F(k+1) \qquad (27)$$

$$V_{PI}(k+1) = V_{PI}(k) + 0\ 064\ e_{PI}(k+1) + 0.032\ e_{PI}(k) \qquad (28)$$

Equations (26) and (27) can be easily obtained by simply observing Figure 12. Equation (28) is the same as equation (10), while equation (25) is derived from equation (13) by shifting each term one step forward. In case of doing simulation for zero runout disturbance, equation (25) can be rewritten as:

$$F(k+1) = 0.7537\ F(k) + 0.2404\ F(k-1) + 9.0452\ V(k-1) - 7.8975\ V(k-2) \qquad (29)$$

This equation is derived by setting all w(k) terms in equation (25) to zero or shifting each term in equation (8) one step forward. The shifting of terms in equation (8) and equation (12) is necessary in order to keep all the equations at the same (k+1)th step for performing simulations.

Simulation Results and Discussion

Figure 13, 14, and 15 are the simulation results of the repetitive controlled peripheral milling process represented by Figure 12 using equations (25), (26), (27), and (28). The step reference force is still 20 lb. Figure 13 shows that at the beginning of the milling, there are periodic variation in cutting forces induced

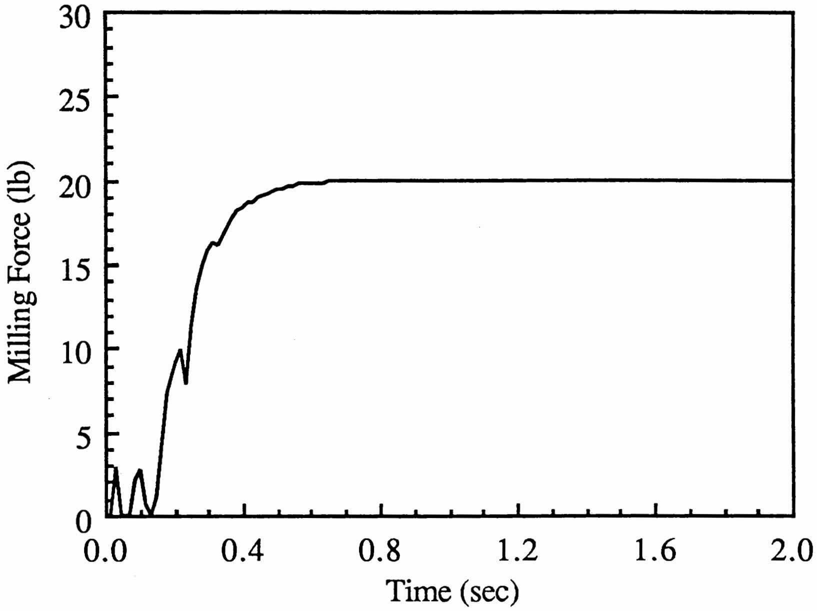

Fig. 13 Milling Force Response to a 20 lb Step Input under Combined Repetitive and PI Control

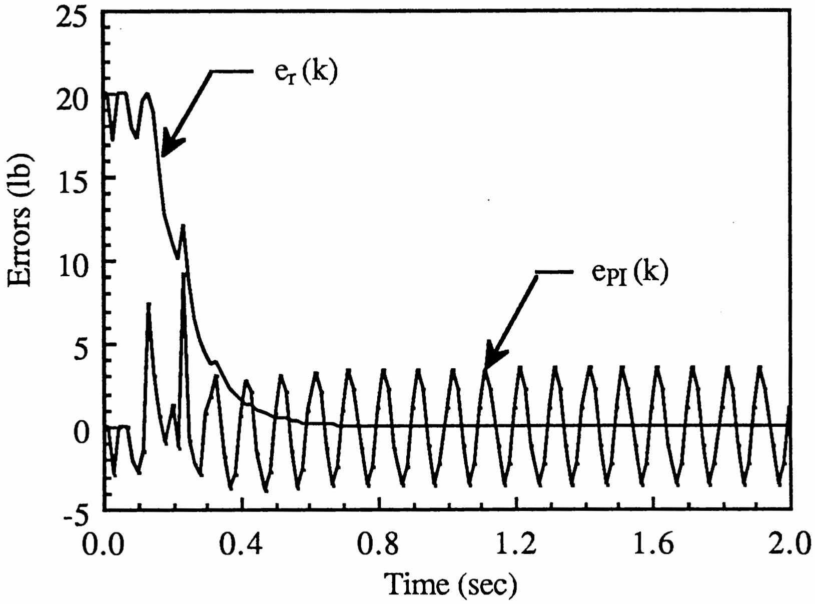

Fig. 14 Error Signals as Seen by the Repetitive Controller and the PI Controller

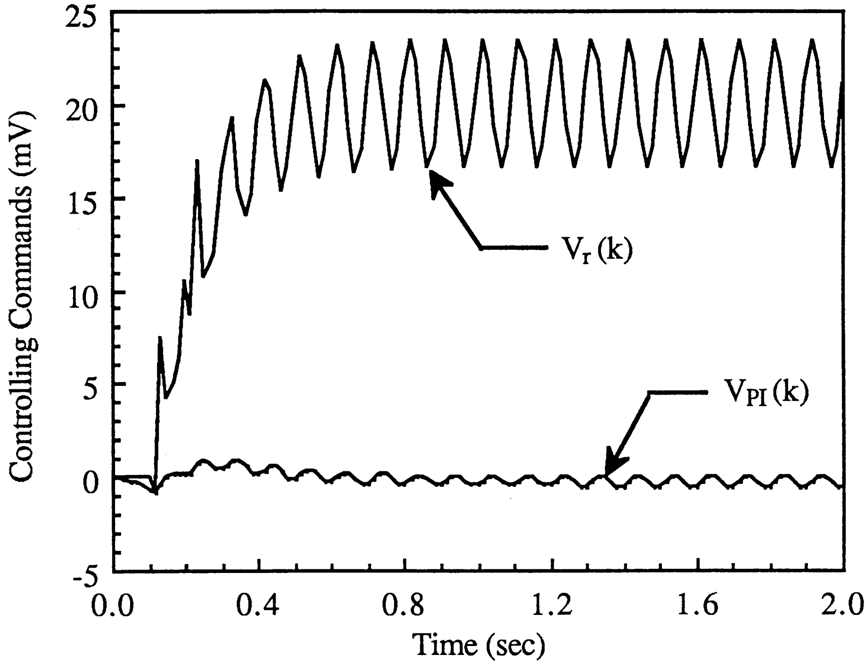

Fig. 15 Feedrate Controlling Commands in the Presence of a Sinusoidal Cutter Runout

by runout. The effect of runout is reduced period by period even when the system is still in transient state. Runout effect is completely eliminated in about 600 ms which corresponds to 6 spindle revolutions. There is no runout effect when the system reaches steady state. The tracking error, $e_r(k)$, should decay to zero when the runout effect is eventually eliminated. This is exactly what Figure 14 shows. The PI controller uses error signals, $e_{pI}(k)$, to counteract the runout disturbance. Therefore, after reaching steady state, $e_{pI}(k)$ should have the same frequency but opposite phase with those of runout. This is also what Figure (14) shows. Figure 15 shows that most correcting action is done by the repetitive controller, because the PI controlling commands is less than one tenth of the repetitive controlling commands. The proportion of correcting action assumed by the two different controllers can changed by adjusting parameters k_r and K_p by method of trial and error.

Figure 16 shows the 20 lb step force response of the same milling process with runout turned off. Figure 16 and 6 reveal that the addition of the repetitive controller help suppressing the overshoot while increasing settling time. The increase in settling time is due to the increase in delay steps introduced by the added controller.

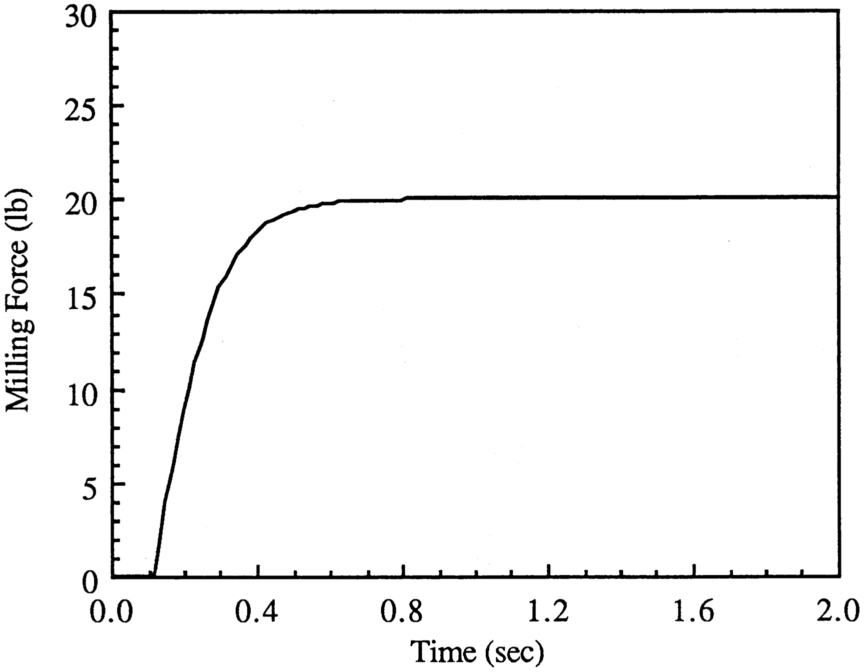

Fig. 16 Milling Force Response to a 20 lb Step Input without Cutter Runout

Results in Figure 16 suggest several useful features. First, the repetitive controller is a new option for overshoot suppression. Second, a repetitive controller can be applied on systems without runout. Increased damping is usually accompanied by increased settling time. This can also be seen by comparing Figure 6 and Figure 16. The 10% settling time for Figure 6 is less than 0.2 second, while that for Figure 16 is about 0.4 second. Therefore, a repetitive controller may be applied to damp out the vibration in a system through active control although it may not be an economic way to do so.

Runout control using output feedback performed by Bifano et al. [7] shows that the dimensional error due to runout could only be reduced by one order (from 25 μm to 2.5 μm). Results in Figure 9 also shows that a conventional PI controller cannot completely eliminate runout effect. As Figure 13 shows, a repetitive controller is the only controller that can completely reduce runout effect to zero.

The repetitive controller design in this section can be easily modified to be applied in other machining operations involving runout by changing the plant transfer function, equation (7), and the sampling rate, N.

CHAPTER VI

PROCESS RESPONSES TO ARBITRARY SHAPED RUNOUT AND DIFFERENT CONTROLLER PARAMETERS

Introduction

In previous chapters, only the system responses to $k_r = 0.8$ and $N = 6$ are studied. Runout signals were assumed to have a sinusoidal shape and an amplitude of 0.30 mV. In this chapter, simulations for different values of N and k_r are performed. An arbitrarily shaped periodic runout is used to replace the sinusoidal runout to verify previous argument that the precise waveform of a periodic runout is not critical. Comparing simulation results presented in this chapter and those in previous ones will give the readers clearer pictures on how a repetitive controller works.

Process Responses to Arbitrarily Shaped Runout

Figure 17 shows an arbitrary shaped runout signals which has the same frequency as the sinusoidal runout used for simulations in previous chapters.

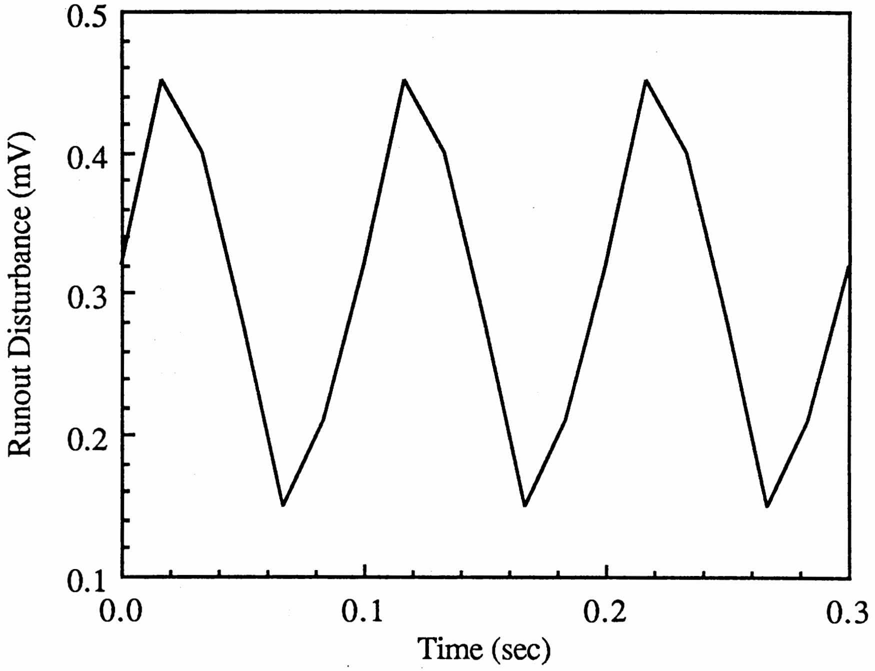

Fig. 17 Variation of Feedrate Command due to Arbitrary Shaped Runout

The new runout signals has an amplitude of 0.25 mV. The new system is the same as that depicted by Figure 12 except the runout signals represented by the expression $\sin(\omega kT + 2\pi / 3)$ are replaced by the new arbitrary shaped runout signals. Therefore, equations (24), (25), (26), (27), and (28) are still good for the new system except that the W(k-1) and W(k-2) terms need to use new runout signals. Figure 18 shows the force response of the system with new runout.

Qualitatively, the milling force response in Figure 18 is similar to that in Figure 13. However, because the new runout signals have an amplitude smaller than the sinusoidal one, the induced variation in cutting forces in the transient state is smaller. In steady state, the milling force outputs of both processes are able to exactly follow the reference step force which is 20 lb in this study. This proves previous argument that the precise waveform of runout signals are not critical as long as they are kept periodic.

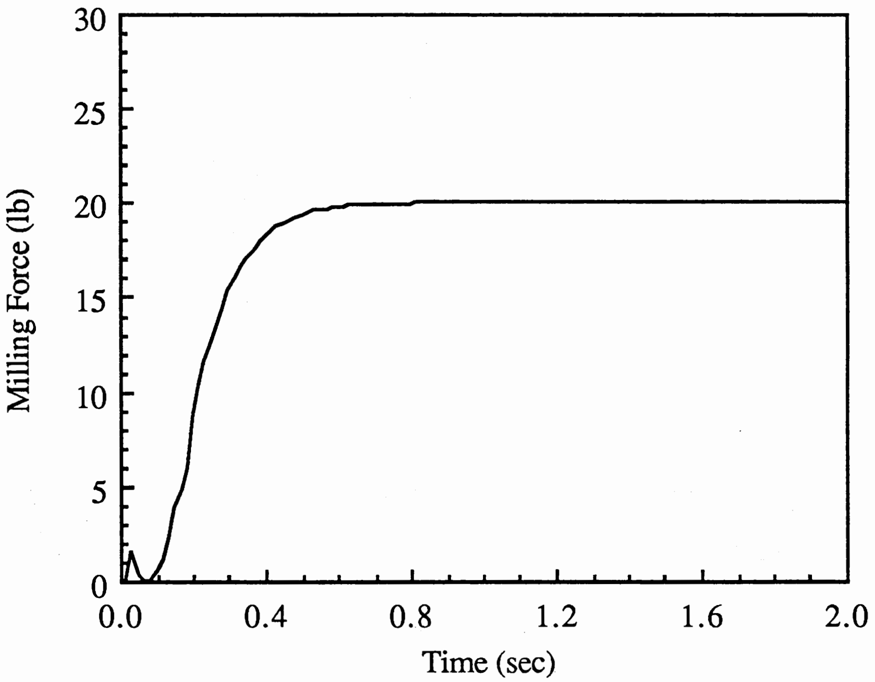

Fig. 18 Milling Force Response to a 20 lb Step Input with Arbitrary Shaped Runout

Process Responses to Different Repetitive Controller Gains

Equation (16) shows that for asymptotic stability for the repetitive controlled peripheral milling process, it is required that $0 < k_r < 2$. In Chapter V, simulations are performed by choosing $k_r = 0.8$. To fully understand the effect of changing k_r on the responses of the repetitive controlled process, simulations are performed with $k_r = 0.4$, 0.6, 0.8, 1.0, 1.2, 1.4, 2.0, and 2.4. The results are plotted in Figure 19, 20, and 21.

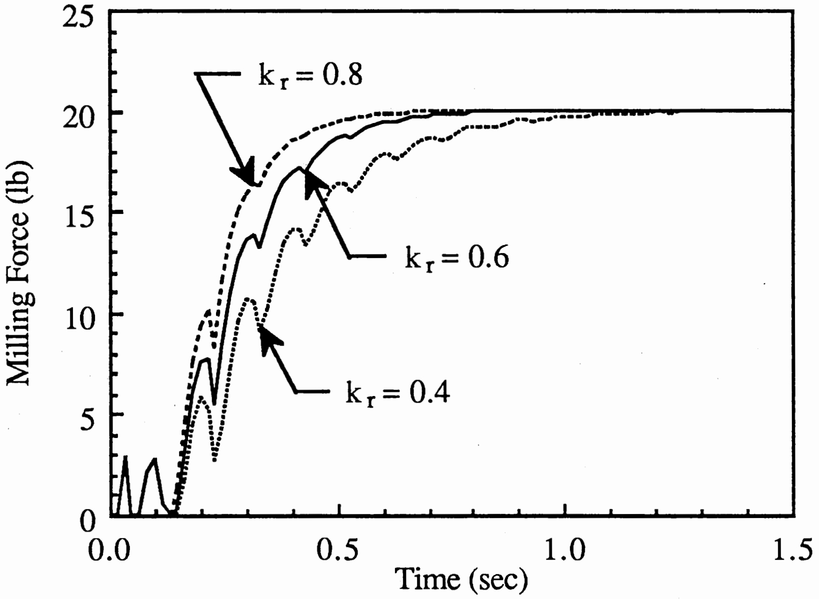

Fig. 19 Milling Force Responses for $k_r = 0.4$, 0.6, and 0.8

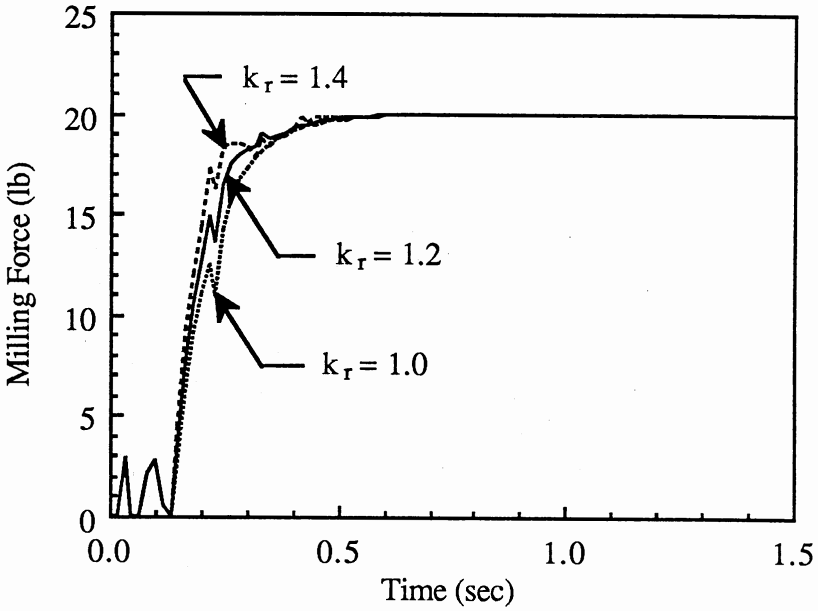

Fig. 20 Milling Force Responses for $k_r = 1.0$, 1.2, and 1.4

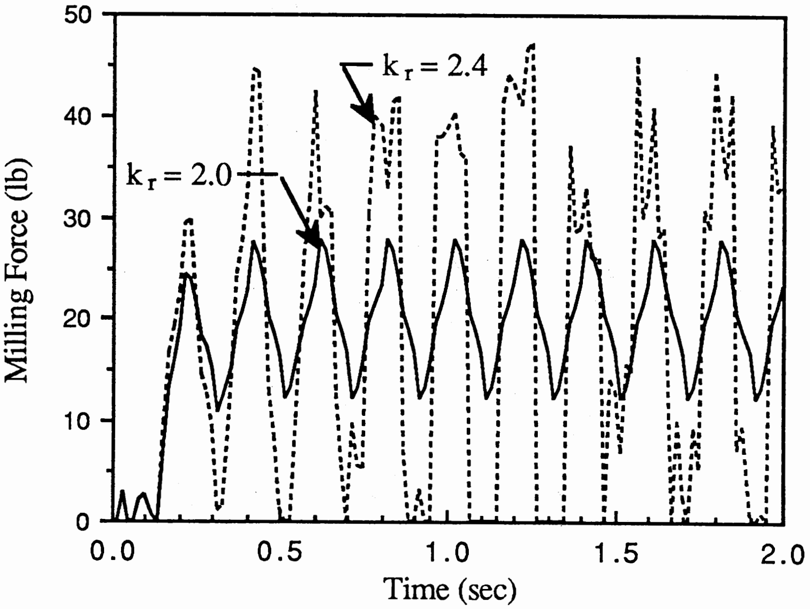

Fig. 21 Milling Force Response for $k_r = 2.0$ and 2.4

Figure 19 shows the milling force responses for $k_r = 0.4$, 0.6, and 0.8, while Figure 20 shows the milling force responses for $k_r = 1.0$, 1.2, and 1.4. Figure 19 shows that when $k_r = 0.4$, the system response is sluggish, because the correction action is slow. It takes 1.2 second for the force output to track the the step reference force, 20 lb in this study, without error. Figure 20 shows that when $k_r = 1.4$, the system is quite active, because the correction action is fast. It takes only 0.6 second for the force output to track the step reference force without error. However, the transient force response start to vibrate with large amplitude. Figure 19 and 20 together show that for $0.6 < k_r < 1.4$, the milling force responses are satisfactory. Figure 21 shows the milling force response for $k_r = 2.0$ and 2.4. For $k_r = 2.0$, the milling force response starts to lose its asymptotic stability. They vibrates even in the steady state and never calms down. For $k_r = 2.4$, the steady state milling force response vibrates with an amplitude as high as 47 lb. The milling force response looks very bad. This confirms that for asymptotic stability, equation (16) should be followed.

Process Responses to Different
Sampling Rate

Theoretically, there are no restrictions on the selection of the sampling rate, N, as long as the selected sampling rate gives a sampling frequency higher than the

Nyquist frequency which is 20 HZ in this study. However, for real time computer control, there is an upper limit for N depending on the computer used for performing the control. The sampling rate should be within the capacity of the computer. In previous chapter, simulations were performed by choosing $N = 6$. To fully understand the effect of changing N on the force responses of the repetitive controlled peripheral milling process, simulations are performed for $N = 2$, 4, 6, 8, 10, and 12. $N = 2$ is the minimum required sampling rate for this study because it corresponds to the Nyquist frequency. $N = 12$ is still within the capacity of the computer used in this study. The computer had been tried for a sampling rate up to $N = 18$ without loosing its accuracy. Simulations results are plotted in Figure 22 and 23. Figure 22 shows the simulation results for $N = 2$, 4, and 12, while Figure 23 shows the simulation results for $N = 6$, 8, and 10. These two figures reveal the following trend. When N is too small, for example $N = 2$, the milling force responses are sluggish. It takes more than 1.5 second for the force output to track the step reference force, 20 lb in this study, without error. The process stays idle for three periods before it starts to rise to track the reference force.

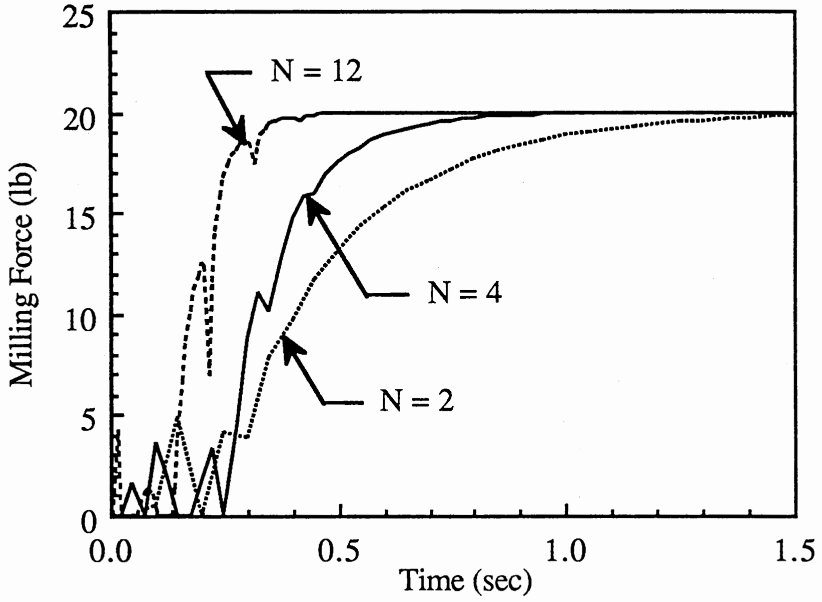

Fig. 22 Milling Force Responses for N = 2, 4, and 12

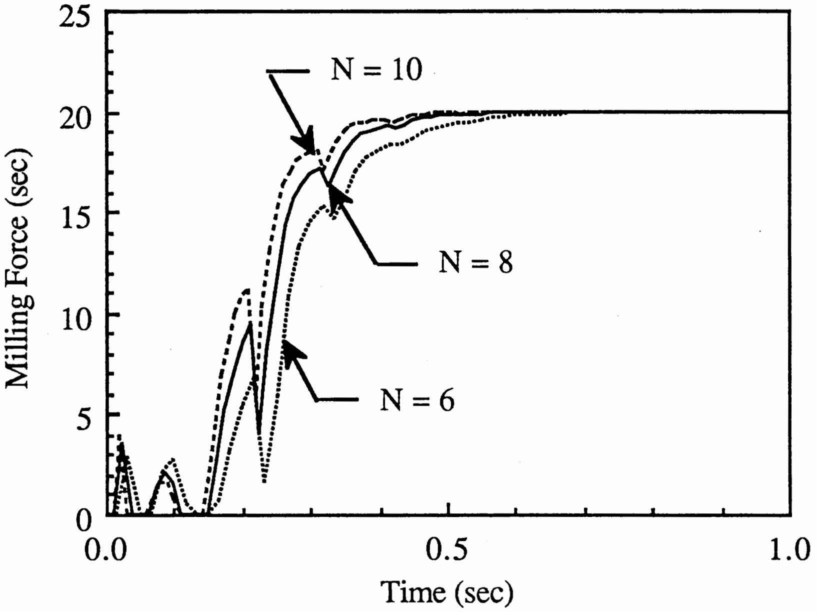

Fig. 23 Milling Force Responses for N = 6, 8, and 10

For simulations in previous chapters which has $N = 6$, there are only two such idle periods. On the other hand, when N is too large, for example $N = 12$, the milling force responses are too active. Although the 10% settling is less than 0.3 second, the transient force response start to vibrate with large amplitude. Therefore, choosing a large N is not necessarily good. It adds more sampling burden on the computer while degrading the force responses. Figure 21 and 22 together show that for $6 <= N <= 10$, the milling force responses are satisfactory.

CHAPTER VII

CONCLUSIONS

This thesis presents works on the digital repetitive control for runout compensation. Results show that although a PI controller alone is good for removing the unstable poles and unacceptable zeros, it it not capable of runout compensation. A combined repetitive and PI controller with properly selected controller gain and sampling rate is good for eliminating runout effect. For asymptotic stability, it is generally required that $0 < k_r < 2$. Simulation results show that the milling force responses are satisfactory when $0.6 < k_r < 1.4$. For values of k_r smaller than 0.6, the milling force responses are too sluggish. On the other hand, when k_r is greater than 1.4, the milling force responses start to vibrates with large amplitude in the transient state. To avoid the problem of aliasing, it is generally required that the sampling frequency is higher than the Nyquist frequency which corresponds to $N = 2$ in this study. For real-time computer control, there is an upper limit for the sampling rate. Such a limit depends on the computer used for performing the control. Results show that the milling force responses are satisfactory for $6 <= N <= 10$. If the sampling rate is lower than six, the milling force responses are too sluggish. On the other hand, if the sampling rate is greater than 10, the milling force responses start to vibrates with large amplitude in the transient state. Large sampling rate also adds sampling burden on the controlling computer.

Results also show that the precise waveform of runout is not critical as long as they are periodic. Because the precise model of runout is hard to obtain, such a phenomenon is very important. It enables the design of a repetitive controller for runout compensation without knowing the precise model represents runout. Although the repetitive controller is designed for a peripheral milling process in this study, the design can be easily modified for use in other machining operations involving runout.

BIBLIOGRAPHY

1. A. M. Figatner, "Accuracy of Spindle Units in Precision Machine Tools," Machines and Tooling, V. 35, No. 7, 1964, pp 25-29.

2. I. G. Zharkov and A. N. Volkov, "Vibration and Surface Waviness in Slot Milling," Machines and Tooling, V. 39, No. 12, 1968, pp 36 - 38.

3. A. KH. Razdobreev, "How Milling-Cutter Eccentricity Affects Cutting Rates," Machines and Tooling, V. 43, No. 2, 1972, pp 55-56.

4. C. D. Johnson, "Disturbance-Accomodating Control; An Overview," American Control Conference, June 1986, pp 526-536.

5. B. D. Anderson and J. B. Moore, Optimal Filtering, Prentice-Hall, New Jersey, 1979.

6. R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," Journal of Basic Engineering, March 1982, pp 34-35.

7. T. G. Bifano, and T. A. Dow, "Real Time Control of Spindle Runout," Optical Engineering, September 1985, V. 24, No. 5, pp 888-892.

8. T. Omata, S. Hara, and M. Nakano, "Nonlinear Repetitive Control with Application to Trajectory Control of Manipulators," Journal of Robotic Systems, 1987, pp 631-652.

9. T. Tsao, and M. Tomizuka, "Adaptive and Repetitive Digital Algorithms for Noncircular Machining," American Control Conference, June 1988, pp 115-120.

10. M. Tomizuka, T. Tsao, and K. Chew, "Discrete-Time Domain Analysis and Synthesis of Repetitive Controllers," American Control Conference, June 1988, pp 860-866.

11. S. Hara, and Y. Yamamoto, "Stability of Repetitive Control Systems," Proceedings of the 24th Conference on Decision and Control, December 1985, pp 326-327.

12. S. Hara, T. Omata, and M. Nakano, "Synthesis of Repetitive Control Systems and its Application," Proceedings of the 24th Conference on Decision and Control, December 1985, pp 1387-1392.

13. S. Hara, Y. Yamamoto, and M. Nakano, "Repetitive Control System: A New Type Servo System for Periodic Exogenous Signals," IEEE Transactions on Automatic Control, V. 33, No. 7, July 1988, pp 659-668.

14. S. Hara and Y. Yamamoto, "Relationships between Internal and External Stability for Infinite-Dimensional Systems with Applications to a Servo Problem," IEEE Transactions on Automatic Control, V. 33, No. 11, November 1988, pp 1044-1052.

15. K. J. Astrom and B. Wittenmark, Computer Controlled Systems-Theory and Design, Prentice-Hall, New Jersey, 1984.

16. K. J. Astrom and P. Eykhoff, "System Identification - A Survey," Automatica, V. 7, 1971, pp 123-162.

17. J. Chery and A. Menendez, "A Comparison on Three Dynamic Identification Methods," Joint Automatic Control Conference, 1969, pp 982-983.

18. A. V. Balakrishnan and V. Peterka, "Identification in Automatic Control Systems," Automatica, V. 5, 1969, pp 817-829.

19. T. Kailath, "An Innovations Approach to Least Squares Estimation. Part I: Linear Filtering in Additive White Noise," IEEE Transactions on Automatic Control, AC-13, 1968, pp 646-655.

20. R. L. Kashyap, "Maximum Likelyhood Identification of Stochastic Linear Systems," IEEE Transactions on Automatic Control, AC-15, 1970, pp 25-34.

21. T. Y. Ahn, K. F. Eman, and S. M. Wu, "Cutting Dynamics Identification by dynamic Data System (DDS) Modeling Approach," Journal of Engineering

for Industry, May 1985, V. 107, pp 91-94.

22. I. Gustavsson, "Comparison of Different Methods for Identification of Linear Models for Industrial Processes," the 2nd International Federation of Automatic Control, Symposium on Identification and Process Parameter Estimation, Prague, 1970, paper 11.4.

23. M. Tomizuka, J. H. Oh, and D. A. Dornfeld, "Model Reference Adaptive Control for the Milling Process," Control of Manufacturing Processes and Robotic Systems, ed. D. Hardt and W. Book, New York, 1983, pp 75-90.

24. J. G. Bollinger and N. A. Duffie, Computer Control of Machines and Processes, Addison-Wesley Publishing, New York, 1988.

25. L. K. Lauderbaugh and A. G. Ulsoy, "Dynamic Modeling for Control of the Milling Process," ASME Journal of Engineering for Industry, V. 110, November 1988, pp 367-374.

26. D. W. Wu and C. R. Liu, "An Analytical Model of Cutting Dynamics, Part 1: Model Building," ASME Journal of Engineering for Industry, V. 107, No. 2, May 1985, pp 107-111.

27. T. Y. Ahn, K. F. Eman, and S. M. Wu, "Identification of the Transfer Function of Dynamic Cutting Processes - A Comparative Assessment," International Journal of Machine Design and Research, V. 25, No. 1, 1985, pp 75-90.

28. G. C. Goodwin and K. S. Sin, Adaptive Filtering Prediction and Control, Prentice-Hall, New Jersey, 1984.

29. J. Tlusty, "Criteria for Static and Dynamic Stiffness of Structures," Chapter 8.5, V. 3, Machine Tool Task Force Research Conference, Lawrence Livermore Laboratory, 1980.

30. M. C. Shaw, Metal Cutting Principles, Clarendon Press, Oxford, 1984.

31. S. Arimoto, S. Kawamura, F. Miyazaki, and S. Tamaki, "Learning Control Theory for Dynamical Systems," Proceedings of the 24th Conference on Decision and Control, December 1985, pp 1375-1380.

32. R. Shoureshi, R. Evans, and D. Swedes, "Learning Control for Autonomous Machines," ASME 88-WA / DSC-29, 1988, pp 1-7.

33. B. A. Francis and W. M. Wonham, "The Internal Model Principle for Linear Multivariable Regulators," Applied Mathematics and Optimization, V. 2, No. 2, 1975, pp 171-194.

34. B. C. Kuo, Automatic Control System, Prentice-Hall, New Jersey, 1987.

VITA

Min-Ching Horng

Candidate for the Degree of

Master of Science

Thesis : RUNOUT COMPENSATION IN PERIPHERAL MILLING USING REPETITIVE CONTROL

Major Field : Mechanical Engineering

Biographical :

Personal Data : Born in Ping-Tung, Taiwan, the Republic of China, November 16, 1957, the son of Tsin-Tsu Horng and Tsai-Lan Day.

Education : Graduated from Provincial Kaohsiung Technical High School, Kaohsiung, Taiwan, in June 1977; received diploma in Industrial Products Design from the National Taipei Institute of Technology, Taipei, Taiwan, the Republic of China, in June 1980; received Bachelor of Science Degree in Mechanical Engineering from Oklahoma State University in May 1988; completed requirements for the Master of Science degree at Oklahoma State University in December 1989.

Professional Experience : Engineering Machinery Maintenance Officer (Second Lieutenant), the Army of the Republic of China, July 1980, to May 1982; Mechanical Drawing Teacher, Kai-Nan and Hsing-Ming Technical High School, July 1982, to January 1984; Teaching Assistant, Department of Mechanical Engineering, Oklahoma State University, August 1987, to December 1989.