UDC: 621.7/.9:004.67.021]:681.515.8
Original scientific paper

An algorithm design for fast tuning of non-linearity control

Ferit Idrizi [1], Jorgaq Kacani [2], Nderim Zeqiri[1]

[1] *Department of Mechatronics, Faculty of Applied Sciences*
[2]*Faculty of Mechanical Engineering, Polytechnic University of Tirana*
Corresponding Author: e-mail: ferit.idrizi@unite.edu.mk

Abstract

The first challenge for high precision and high-speed machining motion control is the presence of friction as a nonlinear phenomenon that exists in every mechanical system. On the CNC feed drive system of the vertical milling machine, the most important statics loads are the friction in the sideways and in the feed drive bearings. Another source of static loads is cutting forces, which usually have opposite direction of the moment of the feed drive where a servomotor is directly connected to lead-screw. The complexity of motion control is further enhanced by the fact that the milling process is a work shift that requires contour tracking at any moment of movement in order to achieve working tolerances under the limits. The Federate speed and feed drive position accuracy are directly dependent on the amount of torque and torque delivered by the servomotor and by the feed drive control algorithm executed by the Computer Numerical Control Unit.

To overcome all these nonlinearities, we have to design a control system that gives fast response with reasonable gains to respond properly to very frequent changes of the system. In this paper is presented a comparison between controlling algorithms and different tuning methods in order to find the faster solutions and the faster system response. Compensation for non-linearity is attempted through improvement of control and appropriate tuning methods. A flow chart for fast tuning algorithm and relevant results of simulation are presented.

Keywords: Fast, Tuning, Feed Drive, PID, FUZZY

Introductions and interest

In interest to design appropriate controllers, many attempts have been made by experts in order to sophisticate accuracy and preciseness of various processes. In many industrial processes, especially, on machine tools motion control processes and closed loop control systems, have been employed various controllers to control nonlinearity under their operating conditions.

Improving the performance means to overcome problems related to vibrations causes of the statics and dynamics load. These improvements generally have been made based on modification of conventional controllers. Researches has been focused on improving the systems time response which include eliminating undesirable overshoots and decreasing settling and rising time.

Conventional PID controllers are used also to design new intelligent controllers in combinations of AI techniques such as neural networks, fuzzy logic, genetic algorithm, etc.

The work conditions and specifications

Federate speed and feed drive position accuracy are directly dependent on the amount of torque and torque delivered by the servomotor and by the feed drive control algorithm executed by the Computer Numerical Control Unit. That's why the DC servomotor which is directly connected to lead-screw shaft drives the table and work piece and has to overcome both the static and dynamic loads. For the CNC milling machine each ax has separate servo control system for positioning purposes.
The main problem of these types of feed drive systems is vibration, which requires the use of linear ruler or ball bearings instead of traditional slip guides, thus reducing the damping ability by decreasing viscose friction coefficient.
For simulation purposes we used the equation as a transfer functions were being included also the equations the armature voltage of DC and mechanical part:

$$\frac{\Omega(s)}{V_a(s)} = \frac{K_t}{(L_a s + R_a)(Js + b) + K_t K_E}$$

After substitutions of the values for DC, we get the transfer function: G=2/(s^2+12*s+24);
The speed and accuracy of positioning the machine tool are affected by the trajectory generation and control algorithms, mechanical drives and guides, amplifiers, motors and sensors used in each feed drive.

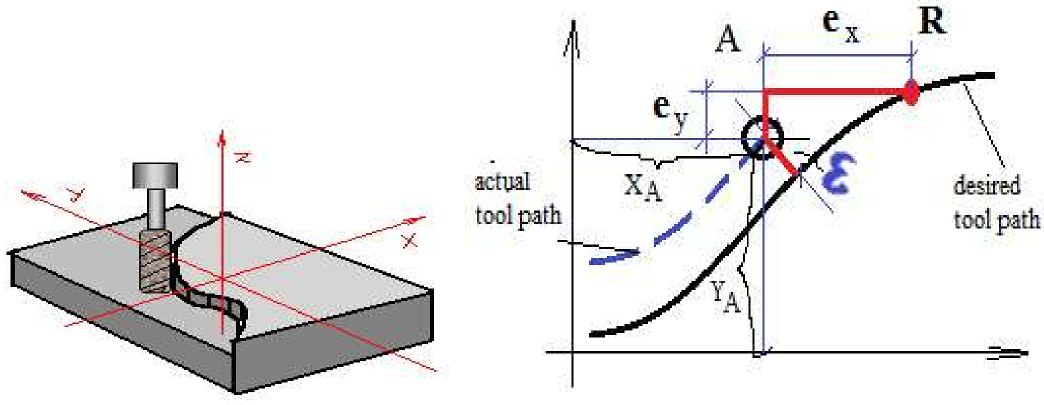

Figure 1. CNC milling and trajectory of motion

The PID Control

The mathematical expression of PID control strategy is based on a control algorithm that involves three separate parameters P, I and D, and on calculation of control action as a sum of these tree factors. In the practical view we should have a mathematical mechanism to make fast corrections, to take care of the error actual value, to consider the error accumulation effect and to predict possible error changes.

The proportional control it is a direct response to the error signal generated by the circuit. If the gain is too low, then the loop cannot respond to changes of the system properly. By increasing the gain, we can cause fast response to the circuit but too high values can cause oscillation of the value.
Integral control it is a sums of error which was not corrected in previous action. It goes a step further than proportional gain, it just the magnitude of the error and also the duration of the error. High gain values can cause significant overshoot and instability with oscillations. Too low and the circuit will be significantly slower in responding to changes in the system. Derivative control presents the damping action and attempts to reduce the overshoot and ringing potential from proportional and integral control. As a compensator it can reduce the overshoot and oscillations caused by integral and proportional control. Too high values of derivative control parameter will slow the response of the loop.

Tuning and Cause-effect relations

The main idea is based on the empirical method and the interest to reach quickly the approximate values of PID parameters. For this purpose, we use three operations: BIG step, Small step and Doubling

$$BIG\ STEP = 10^{i}, were\ i = 1, 2, 3, then\ \Delta K = 10, 100, 1000$$

$$SMALL\ STEP = 10^{i-1},\ were\ i = 1, 2, 3 \ \textit{(One level lower)}$$

$$\textit{Ex: if } i{=}3 \textit{ then }\ K = 10^{i} = 100, then\ \Delta k = 10^{i-1} = 10$$

Manual tuning of the gain settings is the simplest method but it does require some amount of experience and understanding. There is not a static set of rules for what the values should be for any specific system, following the general procedures should help in tuning a circuit to match one's system and environment. Based on the table below there are some rules for driving us to the best solution appropriate for the system:

Table 1. Performance relations with parameter increase

Parameter increase	Rise Time	Overshot	Settling time	SSE
Kp	decrease	increase	small change	decrease
Ki	decrease	increase	increase	eliminates
Kd	small change	decrease	decrease	small change

So, the objective of first Parameter circle will be the big values for increment until the firs overshot:

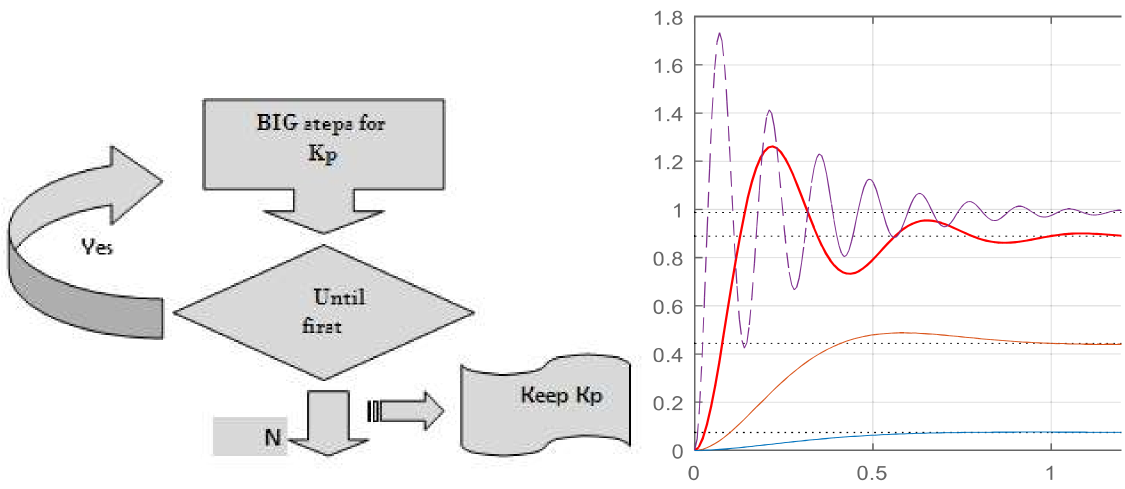

Figure 2. The big step circle and result

The second activity is the increase or decrease by low level values. SMALL STEPS means decrease with delta of lowered level of increasing:

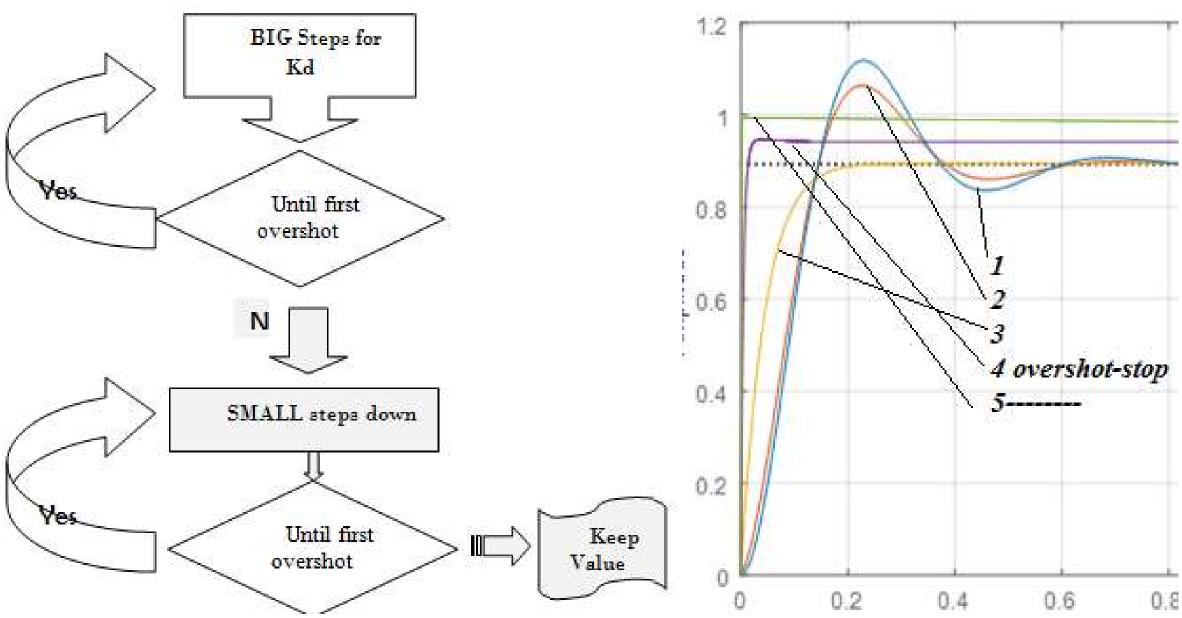

Figure 3. The small step circle

The third operation is Doubling and can be applicable if cause-effect relations have the same direction for both parameters (see effect of Kp and Kd increase in the table below):

Table 2. The same direction Cause-effect relation

Parameter	Rise Time	Overshot	Settling time	SSE
Kp-increase	decrease	increase	small change	decrease
Ki-increase	decrease	increase	increase	eliminates
Kd-increase	small change	decrease	decrease	small change

This is the ideal case for decreasing by doubling the both values:

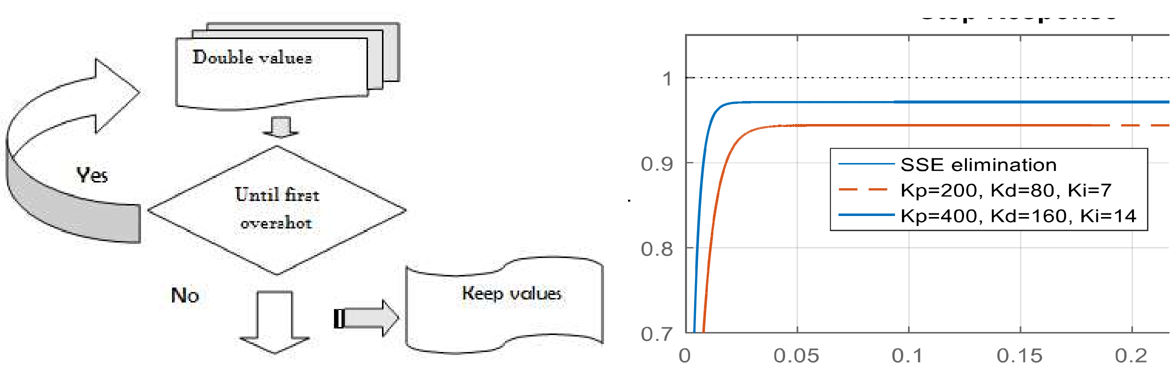

Figure 4. Decreasing the error by duplication

Usually when we are working with second order system it is almost best to start tuning the proportional gain and that with the derivative ones. The integral part has given the effect of destabilizing so it is better to left it in the end.

Conclusions

To see the performance score we have compared the genetic algorithm. The first one is GA1 with180 and GA2 after 300 interactions. The Mat Lab code is shown below:

```
function [GA]=pid_optimSH1(x)
G=2/(s^2+12*s+24);
Kp=x(1)
Ki=x(2)
Kd=x(3)
cont=Kp+Ki/s+Kd*s;
dt=0.01;
t=0:dt:1;
step(feedback(cont*plant,1));
e=1-step(feedback(G*cont,1),t);
%ITAE
GA=sum(t'.*abs(e)*dt);
end
```

Figure 4. Decreasing the error by duplication

-The first tendency was to see if there a tuning by rules of genetic algorithm could result the low levels of absolute error, so we tried with 180 interactions and got the result:

Kp =73.0117; Ki=99.4928; Kd=3.2217;

-The second variant was GA2 with 300 interactions and the result is elimination of overshot and smoother damping:

Kp =55.2628; Ki = 98.9995; Kd = 3.7851;

Table 1. Performance comparison

	Tr	Ts	Overshot	ESS
GA1	0.183	1.81	-	-
GA2	0.246	1.14	-	-
E	0.0116	0.0254	-	-

The comparison of these methods shows that genetic algorithm tuning method GA1 and GA2 gives better performances according to the requirements of the feed drive, which means faster response without overshoots.

References

[1]. Y. Altintas,A. Verl,C. Brecher, L. Uriarte, G. Pritschow, "Machine tool feed drives" ,CIRP Annals - Manufacturing Technology, 2012

[2]. Chien-Yu Lin and Ching-Hung Lee, Remote Servo Tuning System for Multi-Axis CNC, Machine Tools Using a Virtual Machine Tool Approach, 2017, www.mdpi.com/journal/applsci

[3]. Sou Watanabe · Ryuichi Oguro · Jun Kobayashi Fujio Ohkawa, Modeling and position control based on a two-mass system for a machine stand vibration system, Artif Life Robotics (2007)

[4]. Khalid A.Buragga, Essam Natesheh, “Comparison between conventional and fuzzy logic PID controllers for controlling DC Motors”, IJCSI, September, 2010.

[5]. Ferit Idrizi, Jorgaq Kacani" A Fast PID Tuning Algorithm for Feed Drive Servo Loop", ASRJETS, May,2018