International Journal of Machine Tools & Manufacture 117 (2017) 1–10

Contents lists available at ScienceDirect

International Journal of Machine Tools & Manufacture

journal homepage: www.elsevier.com/locate/ijmactool

Servo performance improvement through iterative tuning feedforward controller with disturbance compensator

Jianhua Wu*, Yong Han, Zhenhua Xiong, Han Ding

Robot Institute, School of Mechanical Engineering, Shanghai Jiao Tong University, No.800, Dong Chuan Rd., Shanghai 200240, PR China

ARTICLE INFO

Key words:
Iterative tuning
Servo control
Feedforward controller
Tracking
Contour following

ABSTRACT

Servo system is widely used in NC machines and its performance directly determines the precision of the machines. In most situations, the control structure for the servo system usually contains a cascaded P-PI feedback controller and a feedforward controller. This paper focuses on the feedforward controller parameters tuning to improve the servo performance. The feedforward controller consists of a model inversion and a parameterized disturbance model. Its parameters are tuned iteratively using the last cycle motion results. This method has the good extrapolation capability to the references and the performance improvement capacity. Moreover, it is easy to implement in real machines due to the simplicity and thus is of interest to control engineers. Experiments are carried out on an industrial prototype system. The results show that the proposed tuning method can improve the servo performance rapidly and the references are not required to keep the same during the tuning process.

1. Introduction

NC machines usually utilize servo motors as their actuators to realize spatial motions and the machines' precision is highly dependent on the servo performance. Given the mechanism and the control hardware, the servo performance is determined by its control algorithm. Till now, a great number of papers on controller designing have been published to decrease the tracking error [1–4] and contour error [4–8], which are the widely used indices to characterize the servo performance. Between the above two indices, tracking error is more fundamental since contour error becomes meaningless when tracking error is large. Actually, many of the existing contour control algorithms focusing on coupling all axes to match their dynamics through various strategies [6,9,10] were based on that the tracking errors are small enough. Hence, the primary task of servo control is to improve the tracking performance.

In the published tracking control algorithms, most of them usually had a feedback plus feedforward structure, where the feedforward controller was to cancel the effects of the servo dynamics and the known disturbance while the feedback one suppressed the rest disturbances. The feedforward controller plays a vital role in the servo performance since it can compensate for the servo error caused by the reference in a predetermined way [11]. There are two main strategies to design the feedforward controller: model-based and model-free. In model-based methods, the feedforward controller is the inversion of the servo model and can achieve good performance provided that the model inversion is accurate enough [12,13]. The model inversion can be identified off-line or on-line. The off-line method has the merit that it will not bring stable problems but it has difficulty in achieving optimal results and is not able to adapt to the servo changes during the motions [14–16]. As the alternative, though the on-line method like adaptive algorithms endues the feedforward controller with the ability to adjust its parameters to handle the variations [1,17,18], it is much sensitive to disturbances and easy to destabilize the system [19]. In model-free methods, the feedforward controller uses advanced algorithms instead of the model inversion. The advanced algorithms either approximate the model inversion or directly generate the control effort. Neural network is a good approximation of the model inversion [20]. With its weights trained or adapted, it has the similar property as the model-based feedforward controllers. That is, it cannot adapt to the servo variations when the weights are trained off-line while is much sensitive to external disturbance when the weights are adjusted on-line. As the alternative, iterative learning control (ILC) algorithm generates the control efforts directly according to the past motion cycle results [21,22]. It generally improves the performance gradually and gets better results than the model-based feedforward controller. But it is only suitable for repetitive motions and thus lacks the extrapolation capabilities with respect to different references. Additionally, it is much sensitive to the external disturbances since the control effort is adjusted at each sampling time and any disturbance will directly affect the

* Corresponding author.
E-mail address: wujh@sjtu.edu.cn (J. Wu).

http://dx.doi.org/10.1016/j.ijmachtools.2017.02.002
Received 7 December 2016; Received in revised form 7 February 2017; Accepted 10 February 2017
Available online 13 February 2017
0890-6955/

adjustment.

Recently, the literature [23] presented the combination of the model-based feedforward controller and ILC. The idea was to decompose the feedforward controller into a sum of basis functions and tune the coefficients iteratively according to the past results. It inherited the good extrapolation capabilities from the model-based feedforward controller and the performance improvement capacity from ILC. Besides, this method became less sensitive to disturbances since only a few coefficients were adapted according to a whole cycle motion results and thus the disturbance effect was filtered out in some sense. The paper [24] was a specific example of [23], where the basis function were 1st-order, 2nd-order, 3rd-order and 4th-order derivatives of the reference, i.e., velocity, acceleration, jerk and snap feedforward controller. The authors utilized the Newton's method to optimize the feedforward controller coefficients through model-based or data-based approaches. The experiments showed the performance improvement was obtained iteratively in a wafer stage. In [25], Bolder tried to expand the model-based feedforward controller from the polynomial into rational functions, which could handle the flexible dynamics. In order to deal with the parametric nonlinearity, the authors transferred the optimization criterion into a quadratic function through considering the nonlinear terms as a priori unknown weighting functions. But the optimization results were not an analytic solution and the stability were not guaranteed. The paper [26] handled the rational basis functions through designing an input shaper and a polynomial feedforward controller. The optimization criterion became a quadratic function with respect to the coefficients of the input shaper and the feedforward controller. Using the past motion errors and control efforts, the coefficients could be optimized iteratively. But this method was only suitable for the point-to-point motions in theory.

From [23–26], the iteratively tuning methods for the feedforward controller based on the polynomial basis functions are theoretically perfect but those based on the rational basis functions need to be studied further in tracking fields. That is, the methods are preferable to apply in the systems without flexible dynamics. In real manufacturing, NC machines are usually operated below some bandwidth where they behave as rigid bodies. However, the above methods cannot be directly used because the servo systems in NC machines are under the cascaded feedback loop control [27] while the above papers only discussed the single feedback loop control. Additionally, it is noted that the above feedforward controllers were only to cancel the effect of the servo dynamics and none of them considered the external disturbances. But, external disturbances, especially the friction, degrade the servo performance greatly [28]. It is expected that better performance can be obtained if more disturbances are compensated in the feedforward way. How to combine the disturbance with the model inversion in the feedforward controller and tune the parameters iteratively remains open.

This paper will focus on iteratively tuning the feedforward controller for servo systems which are under the P-PI cascaded feedback control. The feedforward controller includes both the model inversion and external disturbances. The tuning method has the advantage that only tracking errors are required and hence it is very easy to apply for control engineers. The remainder of this paper is as follows. Section 2 gives the cascaded feedback loop plus the feedforward control structure. The analysis on the tracking performance improvement will be described. In Section 3, the feedforward controller's parameters are tuned iteratively. Section 4 demonstrates the proposed method's effectiveness through experiments where the friction is considered as the external disturbance. At last, conclusions are drawn in Section 5.

2. Feedforward controller in cascaded feedback control structure

In NC machines, servo systems usually utilize cascaded PID controller in the feedback way due to its robustness and simplicity.

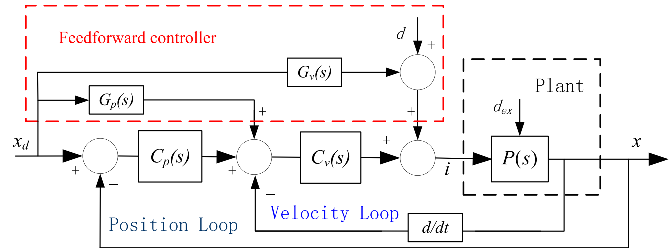

Fig. 1. The feedback and feedforward control structure.

The cascaded control structure contains three loops, i.e., the position, velocity and current loop. In such structure, the current loop controller handles electrical part of the servo system and its parameters are tuned to make the current loop have much higher bandwidth than the velocity and position loop. Thus, the transfer function of the electrical component is regarded as 1 when considering the velocity loop and position loop controller design.

The cascaded feedback plus the feedforward structure is shown as Fig. 1, where x_d and x, are the reference and the output, i is the current command, $P(s)$ is the controlled plant model, d_{ex} is the lumped disturbance, C_p and G_p are the position loop feedback and feedforward controllers, C_v and G_v are the velocity loop ones, d is the feedforward disturbance compensator. In this structure, there are two feedforward control components, G_p and G_v, which differs with the single loop structure where only the model inversion is the feedforward controller. The output in Fig. 1 is

$$X(s) = G(s)X_d(s) + G_d(D(s) + D_{ex}(s)) \tag{1}$$

$$G(s) = \frac{G_vP + (C_p + G_p)C_vP}{1+(C_p + s)C_vP} \tag{2}$$

$$G_d(s) = \frac{P}{1+(C_p + s)C_vP} \tag{3}$$

Where $X(s)$, $X_d(s)$, $D(s)$ and $D_{ex}(s)$ are the Laplace transforms of $x(t)$, $x_d(t)$, d and d_{ex} respectively.

In order to achieve good tracking performance, it is preferable that the following equations hold

$$1+sC_vP = G_vP + G_pC_vP \tag{4}$$

$$D(s) + D_{ex}(s)=0 \tag{5}$$

There are many solutions to Eq. (4) in theory. In this paper, a simple solution is considered as

$$G_p(s) = s \tag{6}$$

$$G_v(s) = P^{-1}(s) \tag{7}$$

Since the NC machines are often operated in a bandwidth where the flexible modes won't be excited, the controlled plant for the velocity and position loop controlled can be modeled as a rigid system

$$m\ddot{x}(t) + c\dot{x}(t) = ki(t) + d_e(t) \tag{8}$$

where m is the moving inertia, c is the viscous coefficient, k is the thrust coefficient and $d_e(t)$ is the disturbance. Then, the model $P(s)$ in Fig. 1 is

$$P(s) = \frac{1}{as^2 + bs} \tag{9}$$

where $a=m/k$ and $b=c/k$. Then, the velocity feedforward controller (7) becomes

$$G_v(s) = as^2 + bs \tag{10}$$

which is an acceleration plus velocity feedforward controller.

The lumped disturbance d_{ex} in Fig. 1 is