Mechatronics 20 (2010) 53–58

Contents lists available at ScienceDirect

Mechatronics

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

An Iterative Learning Control design for Self-ServoWriting in Hard Disk Drives

Shang-Chen Wu [a], Masayoshi Tomizuka [b,*]

[a] Western Digital Corp., San Jose, CA 95138, USA
[b] Department of Mechanical Engineering, University of California at Berkeley, Berkeley, CA 94720, USA

ARTICLE INFO

Keywords:
Iterative Learning Control
Self-ServoWriting
Hard Disk Drive

ABSTRACT

This paper considers the modeling and compensator design for Self-ServoWriting (SSW) process in disk drives. An Iterative Learning Control (ILC) based scheme is established to deal with radial error propagation and improve the quality of written tracks. In the proposed scheme, a feedback controller for track following is first designed to achieve good disturbance attenuation. Then, an ILC structure is applied to generate an external signal, which is injected into the feedback loop in order to compensate for the written-in errors in the previous track while the next track is written. As a result, the error propagation can be contained. The learning controller is synthesized by solving Linear Matrix Inequality (LMI) equations to ensure the stability and monotonic convergence of the control algorithm. Simulation results show the effectiveness of the proposed scheme on the error containment which results in good quality written tracks.

1. Introduction

In the sector servo system for Hard Disk Drives (HDDs), a circular disk is divided into equally sized angular pieces, called servo sectors. Laid on the boundaries of the servo sectors are servo fields on which special servo patterns are embedded. Whenever the Read/Write head (R/W head) passes over the servo patterns, a waveform is read back and decoded to generate a position error signal (PES) which indicates the deviation of the R/W head from the center of the track. PES is then utilized for the control of the head position. With the continuously increasing track density, the required accuracy on the placement of servo patterns is proportionally increased and becomes a crucial issue.

Servowriting is the process to write servo patterns onto the disk. The goal is to place the servo patterns in centric circles with minimum variation of track spacing for all tracks. Conventionally, the process is accomplished by a high precision device called a servowriter, which uses external positional references to accurately position the heads in the disk drive for writing the servo patterns. During the process, in order for the device to access the heads, the drive cover has to be removed. Hence, a clean-room environment is required to servowrite a disk drive. With nowadays high density drives, the servowriting process is extremely time consuming. Consequently, long hours spent on expensive servowriters in clean-room space result in substantial manufacturing cost. Recently, a technique called Self-ServoWriting (SSW) has been developed in order to cut down the cost.

For one type of SSW process [1], the basic concept is to propagate the servo patterns from a few seed tracks by the drives itself. Only the seed tracks are written by an external device. Thereafter, the majority portion of the servowriting process is done after the drive is assembled. The time spent in the clean-room and thus the manufacturing cost can be dramatically reduced. However, in SSW, an absolute position reference is not available. Instead, a previously written track is used as a relative position reference. As a result, the written-in error may be re-produced and even amplified when successive tracks are written. The phenomenon is called radial error propagation, the avoidance of which is the major challenge in SSW. Therefore, the motivation of this research is to prevent radial error propagation and improve the quality of tracks in terms of written-in errors.

In control theories, Iterative Learning Control (ILC) and Repetitive Control (RC) both have the capability in learning through iterations to reject periodic disturbances. The major difference between them is the setting of the initial conditions. ILC is intended for discontinues operation, where the initial conditions are reset in each iteration. On the other hand, RC aims for continuous operation, where the initial conditions are set to the final conditions of the previous iteration [5]. In SSW process, after one track is written, the read head seeks to the written track for the write head to start servowriting the next track. Therefore, ILC is suitable for the application because of the reset natural of the process. It can be an effective approach to mitigate radial error propagation. Some researchers (see [2,3]) have formulated the problem into a similar manner and provided heuristic design

* Corresponding author.
E-mail addresses: shang-chen.wu@wdc.com (S.-C. Wu), tomizuka@me.berkeley.edu (M. Tomizuka).

0957-4158/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mechatronics.2009.06.004

methods. In this paper, we propose an ILC based scheme and establish a systematic framework for the compensator design. The learning controller in the ILC structure is synthesized by solving Linear Matrix Inequalities (LMIs) to ensure the stability and monotonic convergence of the control algorithm. The disturbance and noise rejection is also taken into consideration. The remainder of this paper is organized as follows. In Section 2, the modeling of the SSW process and the root cause of radial error propagation are briefly reviewed. Then Section 3 describes the proposed ILC scheme. LMI conditions are derived for controller synthesis. In Section 4, the proposed method is applied to a HDD benchmark problem. Finally, conclusions are given in Section 5.

2. Problem formulation

2.1. Modeling of the SSW process

Consider a simplified setting for a SSW process, which contains the following steps:

- Step 1: A seed track which may be written by a servo track writer is available on the disk. PES can be obtained when the read head is track following on the seed track.
- Step 2: Assume that there is a constant read-head-to-write-head offset of one track width. Make the read head follows the seed track in the usual track following mode with the sensed PES, while the write head writes servo patterns to create next servo track.
- Step 3: The read head uses the newly written track as a following reference while the write head writes the next track.
- Step 4: Continue Step 3 until all tracks are written.

Fig. 1 illustrates the described SSW process, where $y_i(k)$ represents the track profile of track i at servo sector k. In other words, its value indicates the deviation of the servo written track center from the perfect circular track center. It is the so called written-in error or Written-In Repeatable Runout (WIRRO). The block diagram representation of the SSW system is shown in Fig. 2. The system contains a standard track following servo loop with Voice Coil Motor (VCM) as plant $P(z)$ and the feedback controller $C(z)$. The read head follows on track i with track profile $y_i(k)$ as the reference and generates PES signal for feedback control. Due to the constant one track offset between read and write heads, the plant output is also the write head position where the write head places the servo patterns for the next track and generate the track profile $y_{i+1}(k)$. From the block diagram, $y_{i+1}(k)$ is related to $y_i(k)$ by the complimentary sensitivity function $T(z) = P(z)C(z)/[1 + P(z)C(z)]$, i.e.,

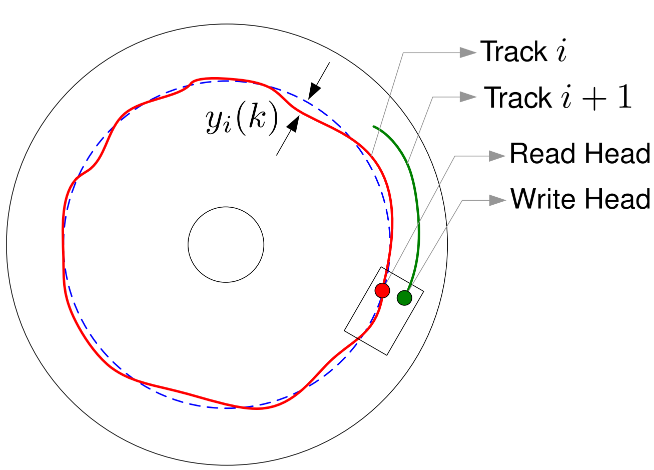

Fig. 1. An illustration of Self-ServoWriting process.

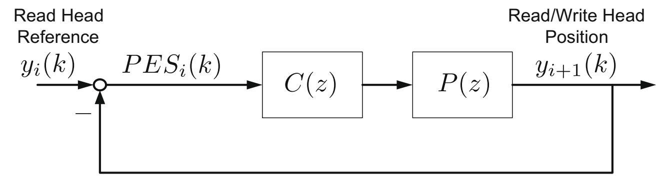

Fig. 2. Block diagram of Self-ServoWriting servo loop.

$$y_{i+1}(k) = T(z)y_i(k) \tag{1}$$

Eq. (1) governs the evolution of the track profile for every servowriting step.

2.2. Radial error propagation

In an ideal situation, if $T(z) = 1$, then from (1), $y_{i+1}(k)$ will be the exact copy of $y_i(k)$, which means every written tracks will have equal spacing which is the desired one track width. However, in the actual case, the gain of $T(z)$ must be larger than 1 at some frequencies. The compound effect causes the written-in error build up in the radial direction across the disk. The root cause of the radial error propagation is due to the integral constraint on the complimentary sensitivity function. According to Bode's integral theorem, given a stable plant $P(z)$ and a stable controller $C(z)$ such that open loop transfer function $P(z)C(z)$ has relative degree larger or equal to two, the integral of the log magnitude of the sensitivity function over the entire frequency from zero to infinity must be equal to zero. In most cases, the HDD servo design usually satisfies the above condition [4]. Furthermore, as an extension of the theorem, the complimentary sensitivity function of the system has to obey the same control limitation. A typical complimentary sensitivity function for a HDD servo system is shown in Fig. 3. It is observed that less than unity gain is required at high frequencies for noise attenuation. Therefore, the complimentary sensitivity hump appears inevitable due to the zero integral constraint. Since the gain of $T(z)$ is larger than one at certain frequencies, according to (1), the frequency components of $y_i(k)$ may be amplified and transferred to $y_{i+1}(k)$. In addition, external disturbances and noise could cause extra off-track motion on the R/W head. As a result, compound errors propagate across the tracks and lead to excessive track non-circularity.

3. ILC compensation scheme

In this section, a compensation scheme will be proposed to deal with radial error propagation in the SSW process. First, assume a high bandwidth feedback controller $C(z)$ is designed for track following purpose, so that the performance for disturbance rejection meets the design specifications. To prevent radial error propagation, one way is to inject a compensation signal $u_i(k)$ into the servo loop as shown in Fig. 4. Ideally, if $u_i(k) = y_i(k)$, then $y_{i+1}(k) = 0$ may be possible. That means zero WIRRO in the next track is achievable

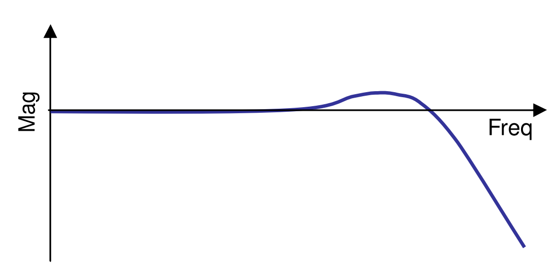

Fig. 3. A typical complimentary sensitivity function for HDD servo systems.