Journal of Process Control 39 (2016) 100–110

Contents lists available at ScienceDirect

Journal of Process Control

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

Two-time dimensional recursive system identification incorporating priori pole and zero knowledge

Zhixing Cao [a], Ridong Zhang [a], Jingyi Lu [a], Furong Gao [a,b,*]

[a] Department of Chemical & Biomolecular Engineering, Hong Kong University of Science & Technology, Clear Water Bay, Kowloon, Hong Kong
[b] Fok Ying Tung Research Institute, Hong Kong University of Science & Technology, Hong Kong

ARTICLE INFO

Article history:
Received 16 March 2015
Received in revised form 2 December 2015
Accepted 17 December 2015
Available online 25 January 2016

Keywords:
Batch processes
Constrained recursive least squares
Minimum phase
Priori knowledge
Two-time dimensional

ABSTRACT

This paper studies an online identification algorithm for batch processes incorporating priori process knowledge of pole and zero positions. The knowledge is available to control engineers and can be exploited to improve the accuracy of the identified process model. To reduce the computation burden of directly invoking Lyapunov inequality, a bound on the identified parameters is imposed to enforce the match between the priori knowledge and identified model. The bound is recursively calculated according to the newly obtained model. The proposed identification method uses the information not only from the time direction but also along the batch direction to improve the identification performance from batch to batch. A filter is introduced to suppress the variation on the identified parameters. Finally, numerical simulations verify the performance and robustness of the proposed method.

1. Introduction

In the era of modern process control, model has an important role in lots of advanced control techniques, such as model predictive control (MPC) [1–5]and adaptive control [6–10]. Generally speaking, there are two schools of methods to establish a mathematical model – principle based and data based. In the first method, various physical conservation laws are used to link the input to the output with a bunch of partial differential equations (PDEs) and ordinary differential equations (ODEs). These methods require tedious procedures and sophisticated techniques to solve the differential equations for the control purpose. Moreover, these differential equations usually contain some unknown parameters that need to be further determined by experiments or other approaches. Unlike the methods above, system identification is an empirical method to extract model information from the process input and output data; thus there are not the aforementioned obstacles in this method. For the past few decades, this topic has been extensively studied [10–14]. Most of these studies focused on the conditions and methods to capture the dynamics of continuous processes asymptotically. Although these conventional identification methods can apply to batch processes, it often fails to yield satisfactory results, since the dynamics between continuous and batch processes is quite different [15]. For example, the ultimate goal for the online conventional identification is $\lim_{t\to\infty}\hat{\theta}(t)=\theta_0$, where $\hat{\theta}(t)$ stands for the estimated parameters of the process and θ_0 is the true parameters of the process, provided that the process can be delineated by the specific model structure and the parameters. But it is not that plausible for batch processes, owing to the finite duration and non-steady state of batch processes. Therefore, it is necessary to develop new identification methods for batch processes.

There are few literatures attributing to this topic. Ma and Braatz [16] developed a stopping criterion for off-line batch process identification. The identification terminates when the worst-case performance index satisfies certain specifications. Tayebi [17] proposed a continuous-time iterative learning adaptive control for robot manipulator with the Lagrangian dynamics assumption. Chi and his co-workers [18] extended the idea to discrete-time systems. The identification method included in both papers estimated the parameters in a two-time dimensional framework without its performance in the transient period discussed. Golshan and MacGregor [19] focused on the closed-loop identifiability condition for batch processes. They argued that adding a dither signal or shifting the control trajectory in a few batches is sufficient for the processes' identifiability. They also proposed to unfold the input and output data in a two-time dimensional fashion. Cao and his colleagues [15] studied the sufficient conditions for almost sure convergence of a two-time dimensional recursive least squares (2DRLS). It was also pointed out that the severe variation of the parameters identified

* Corresponding author at: Department of Chemical & Biomolecular Engineering, Hong Kong University of Science & Technology, Clear Water Bay, Kowloon, Hong Kong. Tel.: +852 23587139; fax: +852 23580054.

http://dx.doi.org/10.1016/j.jprocont.2015.12.011
0959-1524/

along the time direction was a defect of 2DRLS, which necessitates constraints on the estimated parameters into identification.

Constraints, a typical form of representing the priori knowledge of process, are ubiquitous in states estimation for refining the posterior estimates. Generally speaking, there are three types of approaches for constrained states estimation. The first method named clipping is to project the posterior estimates into the constrained regions [20,21]. The second method named acceptance/rejection is commonly adopted in constrained particle filtering [22]. Its underlying idea is to refine the posterior distribution by discarding the particles non-compliant with the constraints. The other method is to handle the constraints directly within the optimization. A typical example is moving horizon estimator [23,24]. Nevertheless, there are few papers addressing the constrained identification topic. Ikonen and Najim [25] studied using constraints to incorporate the priori process knowledge. They also pointed out that it was a difficult problem to consider the constraints on poles or zeros. Bruwer and MacGregor [26] studied the experimental design for process identification. They only addressed the input and output constraints via the D-optimal technique.

It is known that most processes in nature are stable, thanks to various conservation laws. Hence, this paper intends to merge the priori knowledge on zeros and poles into identification to further enhance the corresponding performance. The identification uses the current time input and output data to refine the estimated parameters of the previous batch. To circumvent the computation issue of directly applying Lyapunov inequality, a bound is introduced to allow the estimated parameters to achieve a suboptimal solution. The bound is also recursively calculated by simple linear matrix inequalities (LMIs). A filter is introduced to overcome the shortcoming of 2DRLS by smoothing the estimated parameters. The paper is organized as follows. Section 2 revisits the basics of recursive least squares (RLS) and 2DRLS and provides the motivation of this paper. Section 3 develops the algorithm. A detailed analysis associated with the algorithm is given in Section 4. Numerical simulations demonstrate the performance and robustness in Section 5. Conclusions are drawn in Section 6.

2. Revisit of RLS and 2DRLS

It has been well known that most batch processes possess certain nonlinearity. In most situations, the process variables are controlled to track a certain batch-wisely identical trajectory, which enables the control engineers to adopt a collection of linear models to approximate the dynamics. It has been reported that this idea has been successfully applied on the injection velocity control in injection molding, a classical example of batch processes [27–29]. To avoid obscuring the focus, only the autoregressive exogenous (ARX) model will be discussed in this paper. Consider the following ARX model:

$$\begin{aligned} y_k(t) + a_{1,0}y_k(t-1) + a_{2,0}y_k(t-2) + \cdots + a_{na,0}y_k(t-na) \\ = b_{1,0}u_k(t-d) + b_{2,0}u_k(t-d-1) + \cdots \\ + b_{nb,0}u_k(t-d-nb+1) + e_k(t) \end{aligned} \tag{1}$$

where $y_k(t)$ and $u_k(t)$ are, respectively, the process output and control input of the tth time instant and kth batch. d stands for the delay of the process. na, nb are the order of process output and input dynamics. $a_{1,0}$, $a_{2,0}$, ..., $a_{na,0}$ and $b_{1,0}$, $b_{2,0}$, ..., $b_{nb,0}$ are the output and input parameters. $\{a_{i,0}\}$ and $\{b_{i,0}\}$ are a function of time. It is also noted that the parameter sequence is batch invariant due to the invariance of input profile. Apparently, if the input profile is constant, the parameters $\{a_{i,0}\}$ and $\{b_{i,0}\}$ reduce to constants as well. $\{e_k(t)\}$ is subject to identical independent distribution and

$$\begin{cases} \mathbb{E}[e_k(t)] = 0 \\ \mathbb{E}[e_i(m)e_j(n)] = \sigma^2\delta_{i,j}\delta_{m,n} \end{cases} \tag{2}$$

Here $\delta_{i,j}$, $\delta_{m,n}$ are Kronecker delta, and $\delta_{i,j} = 1$ if and only if $i = j$. Eq. (1) can be rewritten as

$$y_k(t) = \phi_k^T(t)\theta_0(t) + e_k(t) \tag{3}$$

And $\phi_k(t)$ and $\theta_0(t)$ are denoted as follows:

$$\phi_k^T(t) = \begin{matrix} [-y_k(t-1) & -y_k(t-2) & \ldots & -y_k(t-na) & & \\ & u_k(t-d) & u_k(t-d-1) & \ldots & u_k(t-d-nb+1)] \end{matrix} \tag{4}$$

and

$$\theta_0^T(t) = \begin{matrix} [a_{1,0}(t) & a_{2,0}(t) & \ldots & a_{na,0}(t) & \\ & b_{1,0}(t) & b_{2,0}(t) & \ldots & b_{nb,0}(t)] \end{matrix} \tag{5}$$

The length of both vectors is $n_\theta = na + nb$. To resolve the parameters tracking problem, two identifying algorithms – RLS with forgetting factor and 2DRLS will be revisited.

RLS with forgetting factor [10]:

$$\hat{\theta}_k(t) = \hat{\theta}_k(t-1) + K_k(t)[y_k(t) - \phi_k^T(t)\hat{\theta}_k(t-1)] \tag{6a}$$

$$K_k(t) = \frac{P_k(t-1)\phi_k(t)}{\lambda + \phi_k^T(t)P_k(t-1)\phi_k(t)} \tag{6b}$$

$$P_k(t) = \frac{1}{\lambda}\left[P_k(t-1) - \frac{P_k(t-1)\phi_k(t)\phi_k^T(t)P_k(t-1)}{\lambda + \phi_k^T(t)P_k(t-1)\phi_k(t)}\right] \tag{6c}$$

2DRLS [15]:

$$\hat{\theta}_k(t) = \hat{\theta}_{k-1}(t) + K_k(t)[y_k(t) - \phi_k^T(t)\hat{\theta}_{k-1}(t)] \tag{7a}$$

$$K_k(t) = \frac{P_{k-1}(t)\phi_k(t)}{1 + \phi_k^T(t)P_{k-1}(t)\phi_k(t)} \tag{7b}$$

$$P_k(t) = P_{k-1}(t) - \frac{P_{k-1}(t)\phi_k(t)\phi_k^T(t)P_{k-1}(t)}{1 + \phi_k^T(t)P_{k-1}(t)\phi_k(t)} \tag{7c}$$

Here λ stands for the forgetting factor; usually selected between 0.98 (with memory size 50) and 0.995 (with memory size 200) [10]. 2DRLS distinguishes RLS with forgetting factor on two aspects. First, it is the way to update the three equations. Unlike RLS, 2DRLS updates the three equations from the batch direction ($k-1 \to k$) instead of the time direction ($t-1 \to t$). Second, there is a forgetting factor involved in RLS, while 2DRLS does not have such a parameter. As stated before, the reason is that the process is time-varying from the time direction, but batch invariant from the batch direction. To further compare these two methods, the following example will be examined [15].

$$G(z) = \begin{cases} \dfrac{1.69z^{-1} + 1.419z^{-2}}{1 - 1.582z^{-1} + 0.5916z^{-2}} & t \in [0, 150) \\[2ex] \dfrac{(1.69 - 0.2 * \dfrac{t-150}{150})z^{-1} + 1.419z^{-2}}{1 - 1.591z^{-1} + 0.5916z^{-2}} & t \in [150, 300) \\[2ex] \dfrac{1.49z^{-1} + 1.419z^{-2}}{1 - 1.591z^{-1} + 0.5916z^{-2}} & t \in [300, 400] \end{cases} \tag{8}$$

Fig. 1 shows the performance comparison between RLS with a forgetting factor 0.98 and the 50th batch identification of 2DRLS. It apparently shows that 2DRLS stays closer to the true parameter value than RLS with forgetting factor, except with the severe