Journal of Process Control 67 (2018) 56–68

Contents lists available at ScienceDirect

Journal of Process Control

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

Data-driven adaptive multiple model system utilizing growing self-organizing maps

Bo Lu[a,*], John Stuber[b], Thomas F. Edgar[c]

[a] Analytical Technology Center, The Dow Chemical Company, Freeport, TX, United States
[b] Smart Manufacturing Systems Group, VLSIP Technologies Inc., Richardson, TX, United States
[c] McKetta Department of Chemical Engineering, The University of Texas at Austin, Austin, TX, United States

ARTICLE INFO

Article history:
Received 8 April 2016
Received in revised form 1 June 2017
Accepted 5 June 2017
Available online 28 June 2017

Keywords:
Multiple model systems
Adaptive model
Growing self-organizing map
Growing structure multiple model systems
Soft sensor
Data-driven modeling
PLS
PCA
Projection to latent structure
Partial least squares

ABSTRACT

Data-driven soft sensors have seen tremendous development and adoption in both academia and industry. However, one of the challenges remaining is modeling process drifts, degradation and discontinuities in steady-state. Since processes are never truly operating at a steady-state, it is often difficult to assess how much and what types of process data are needed for training and model maintenance in the future. A purely adaptive model maintenance strategy struggles against discontinuities such as preventive maintenance or catalyst changes. In mixture modeling and multi-model systems, the overall modeling structure is fixed and only local coefficients are adapted. In addition, multiple model systems require large amount of training data to initialize. In this paper, we propose an adaptive multiple model system utilizing growing self organizing map to model processes with drifts and discontinuities. Simple model update mechanisms such as recursive model update or moving window model update is not sufficient to deal with discontinuities such as abrupt process changes or grade transitions. For these scenarios, our approach combines projection based local models (Partial Least Squares) with growing self-organizing maps to allow for flexible adjustments to model complexity during training, and also later in online adaptation. This flexible framework can also be used to explore new datasets and rapidly develop model prototypes. We demonstrate the effectiveness of our proposed method through a simulated test cases and an industrial case study in predicting etch rate of a plasma etch reactor.

1. Introduction

Industrial manufacturing systems are becoming increasingly complex and sophisticated due to more stringent requirements on efficiency, product quality and asset utilization. The additional system complexity is possible due to advancements in control, real-time optimization and continuous process improvement. As a result of these complex integrated systems, it becomes increasingly challenging to deal with monitoring and prediction of key quality parameters. Considerable work have gone into improving modern soft sensors performance by increasing their robustness against missing data or poor data [1–3], improved handling of system dynamics [4], and improved model updating [5,6]. Yet, many of these systems operate in multiple production modes which can be characterized by throughput, load, level, recipe and product grades [7]. Traditional methods such as static soft sensor models are ill-suited for these complex systems. Multivariate data-driven models using Principal Component Analysis (PCA) and Partial Least Squares (PLS) also face challenges due to inherent nonlinearity, process drifts, multiple operating modes and complex fault signatures [8].

Since the data generated from these systems naturally have multi-modal distributions [9], a "divide and conquer" approach is commonly taken to partition the data into sub-regions that can be approximated using simple local models. After satisfactory performance is achieved at the local level, selection of the current operating mode at a supervisory layer can then be used to smooth local model results for better prediction of the overall system. Liu proposed a method to detect changes in operating mode to select the best predictor in a multi-modal soft sensor application in [10]. Dunia and Edgar formulated a multi-state PLS framework where transitions between states are smoothed using a k-means clustering smoothing mechanism [11]. Kadlec et al. proposed an incremental learning adaptive soft sensor system that generates operating modes based on prediction residual performance and combines the local model predictions using a Bayesian-inference

* Corresponding author.
E-mail address: bo.lu@utexas.edu (B. Lu).

https://doi.org/10.1016/j.jprocont.2017.06.006
0959-1524/

based validity function approach [12]. There are also many other published works in this area [13–16]. Common challenges associated with these approaches are:

1. Partitioning of the operating space needs to be determined before local model training [12,11].
2. Operating modes that have limited training data can lead to numerical instability [12,16].
3. Local model and supervisory structural layer are often trained separately [11].
4. It is often difficult to implement these complex models online in an industrial environment.

On the other hand, mixture models and kernel density estimates based methods directly assumes that the underlying data distribution is non-Gaussian and multi-modal *without any easily identifiable class labels*. In these methods, mixture models do not require apriori definition of a multi-model structure. Yu and Qin [9] proposed using finite Gaussian Mixtures to monitor industrial process data. They obtained results superior to conventional multivariate monitoring schemes on the Tennessee–Eastman plant simulation. Thissen et al. applied finite Gaussian mixture models to industrial data of a fiber spinning process and found much higher sensitivity to faults using Gaussian Mixture Models [17]. Sammaknejad et al. modeled the transition among operating states as a Hidden Markov process where the transition probabilities are adapted online. Their approach was applied in monitoring an oil sand froth treatment process [18]. Some of the other works using mixtures of distributions are [19–22]. Through reviewing these works, we can observe that this class of methods face the following challenges:

1. difficult to troubleshoot due to the lack of transparency
2. difficult to implement online due to heavier computational requirements
3. stability of online adaptation of these Gaussian mixtures has not been thoroughly studied.

Batch process data from semiconductor manufacturing are particularly challenging to model due to the following characteristics:

- **high-mix manufacturing**, upstream and downstream processes could change depending on product and production thread. Therefore, unmeasured disturbances could affect process modeling results.
- **large number of measurements**, high dimensional data resulting from unfolding of batch trajectories
- **threaded production**, multiple recipes are ran on the same tool. As a result, some un-popular recipes might have little data available for model building.
- **nonlinearity**, input–output relationships are not linear
- **process degradation**, input–output relationships drift overtime

To improve upon existing works, our proposed new approach attempt to satisfy the following:

- Does not rely on class label information
- Does not require large amount of training data to initialize
- Can be easily extended for online adaptation
- Produces interpretable local models and diagnostics for troubleshooting

To this end, the work of growing structure multiple model systems proposed by Liu et al. [14] and Bleakie et al. [23] inspired the development of a hybrid multiple model system that combines PLS local models with a Growing self-organizing Map to form an integrated framework suitable for batch data from semiconductor manufacturing.

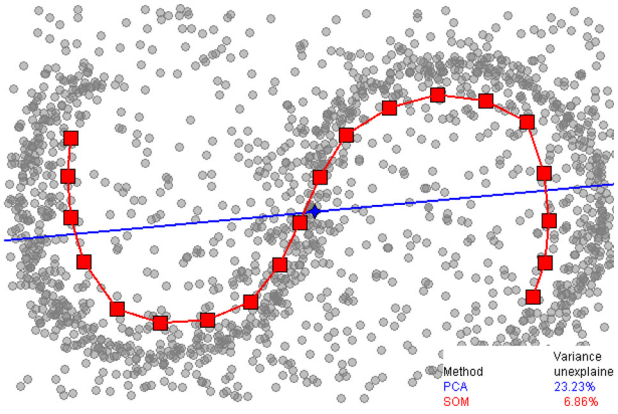

Fig. 1. Example of SOM vs GSOM under different tuning parameters, figure taken from [25].

2. Growing self-organizing map

Growing self-organizing map (GSOM) is a subclass of self-organizing maps (SOM), which is a type of unsupervised machine learning technique. SOM is also a special case of an artificial neural network. Its primary purpose is to map high dimensional data onto a lower-dimensional space. It is also called a Kohonen map [24]. Different from clustering algorithms and other artificial neural networks, SOMs attempt to preserve the topology of the input space by modeling the connections between clusters. As a result, these topological information can be used to improve prediction, training, and model update. GSOM makes no prior assumption about the size of the map and allows the algorithm to learn the model parameters at run-time. An example of the topology in a GSOM is given in Fig. 1, where GSOM is able to preserve the spiral geometry in the reduced dimensional space.

Just like other artificial neural networks; GSOM requires training before it becomes useful. The classic GSOM undergoes three stages of training:initializationwhere small number of nodes and the topological structure parameters are initialized.growingwhere training input is presented to GSOM to successively minimize the vector quantization error of the input data. Existing nodes will be relocated to new positions, and new nodes will be added as the algorithm sees fit.fine-tuningfinal adjustment of node positions according to training data.

In the proposed approach, since the structural learning phase (training the GSOM) is integrated with the local learning phase (training local PLS models), both training phases will be discussed together. First, the basic mathematical description of GSOM is introduced.

Given input and output data $\mathbf{X}(N \times P)$ and $\mathbf{y}(N \times 1)$, we can define a feature vector for each observation

$$\mathbf{s}_i = [s_1 \quad s_2 \quad s_3 \quad s_4 \ldots s_K]_i = [P(\mathbf{x}_i, \mathbf{x}_{i-1}, \ldots), Q(\mathbf{y}_i, \mathbf{y}_{i-1}, \ldots)] \qquad (1)$$

where $P(\cdot)$ and $Q(\cdot)$ are features selection functions. These functions can either reduce feature space dimension by projecting X onto a lower dimensional space using techniques such as Principal Component Analysis (PCA) or Independent Component Analysis (ICA), expand feature space by introducing nonlinear terms or lagged variables. Determining the appropriate feature selection function will depend on the dataset being modeled and also the prior knowledge of the process.

Once the feature vectors are defined, SOM is then trained on the features. A trained SOM has the location of its nodes and the