Korean J. Chem. Eng., **26**(1), 14-20 (2009)

A recursive PLS-based soft sensor for prediction of the melt index during grade change operations in HDPE plant

Faisal Ahmed, Salman Nazir, and Yeong Koo Yeo†

Department of Chemical Engineering, Hanyang University, Seoul 133-791, Korea
(*Received 8 May 2008 • accepted 22 July 2008*)

Abstract−An empirical model has been developed for the successful prediction of the melt index (MI) during grade change operations in a high density polyethylene plant. To efficiently capture the nonlinearity and grade-changing characteristics of the polymerization process, the plant operation data is treated with the recursive partial least square (RPLS) scheme combined with model output bias updating. In this work two different schemes have been proposed. The first scheme makes use of an arbitrary threshold value which selects one of the two updating methods according to the process requirement so as to minimize the root mean square error (RMSE). In the second scheme, the number of RPLS updating runs is minimized to make the soft sensor time efficient, while reducing, maintaining or normally increasing the RMSE obtained from first scheme up to some extent. These schemes are compared with other techniques to exhibit their superiority.

Key words: Online Updating Scheme, Recursive Partial Least Square, Model Bias Updating, High Density Polyethylene (HDPE), Melt Index (MI)

INTRODUCTION

In this era of emerging technologies, model-based monitoring and control methods have greatly facilitated the online prediction of quality variables of complex chemical processes. In the chemical industry, efficient monitoring and control of process variables are essential industrial practices for reduction of the amount of off-specification product and the production cost. Off-line analytical measurement of the product quality variables often takes a long time to efficiently control the process, and this delay forces the operator to make quick decisions in nonlinear and dynamic processes. Hence, a number of first principle models, black box models using neural networks, statistical data modeling (SDM) techniques and hybrid models have been developed for monitoring, control and optimization of complex processes. However, practical infeasibility of the development of a first principle model for some highly complex nonlinear processes and increasing demand of low cost products are the issues that have motivated many researchers towards the SDM of complex nonlinear processes. SDM is based on the historical relations among the process and quality variables, and prevents one from the laborious study of complicated chemical and physical phenomena involved. Sensors developed by SDM are called “Soft” or “Virtual” sensors and find their places quite frequently in the industry for quality control purposes.

Under the category of SDM techniques, PLS has been shown to be a proficient and powerful multivariate regression technique for addressing highly correlated process variables [1,2] using lesser number of latent variables than principle component analysis (PCA). Most chemical processes, and in particular polymerization processes, involve high dimensionality, collinearity and nonlinearity. Except for nonlinearity, the other two problems can be overcome by the linear PLS framework, which can only capture the linear relationship between the process and quality variable latent vectors. To capture the nonlinearity, many nonlinear techniques have been proposed both within the PLS framework (Quadratic PLS, Spline-PLS, Fuzzy-PLS and Neural network PLS) and without the PLS framework (Black box model that includes Neural network model). In addition to capturing nonlinearity, the time-varying nature of a process has been addressed by Helland et al. [3], Dayal and MacGregor [4] and Qin [5].

In industrial practices, different grades of HDPE are usually produced in the same reactor with irregular intervals. Use of soft sensors without update causes a major difficulty in predicting MI when the polymerization process changes its grades. To properly cope with this grade-changing characteristic of HDPE process, recursive adaptive data models can be employed. In this work, inspired by recursive update of PLS by new process data points [5] with mean and variance update [6] as well as model bias update [7], two different modeling schemes are developed for MI prediction of HDPE by a combination of these updating methods. These updating schemes with inherent selection criteria keep track of grade changing characteristic by adding new process data point(s) and removing the oldest one(s) to update the model recursively with the update of mean and variance each time the PLS model is selected. These schemes take benefits of model bias update simultaneously with RPLS update.

PLS UPDATE

1. Partial Least Square (PLS)

The PLS method overcomes the ill-conditioning problem of ordinary PLS by projecting the process variables matrix X and response variable(s) matrix Y onto the orthogonal latent variable subspace.

†To whom correspondence should be addressed.
E-mail: ykyeo@hanyang.ac.kr
‡This paper is dedicated to Professor Chang Kyun Choi to celebrate his retirement from the school of chemical and biological engineering of Seoul National University.

In this way PLS decomposes the input and output matrices X and Y, respectively, into a number of uncorrelated univariate regression problems. Many PLS algorithms have been proposed [1,4] to project the input matrices to the latent variable space through different iterative manners of calculating eigenvectors. In this work the nonlinear iterative partial least squares (NIPALS) algorithm [1] is used for the projection:

$$\mathbf{X}=\mathbf{t}_1\mathbf{p}_1^T+\mathbf{E}_1 \quad (1)$$

$$\mathbf{Y}=\mathbf{u}_1\mathbf{q}_1^T+\mathbf{F}_1 \quad (2)$$

where $\mathbf{t}_1$ and $\mathbf{u}_1$ are the score vectors of the first factor, $\mathbf{p}_1$ and $\mathbf{q}_1$ are the loading vectors, and $\mathbf{E}_1$ and $\mathbf{F}_1$ are the residuals. Inner relation between $\mathbf{u}_1$ and $\mathbf{t}_1$ is obtained through the univariate regression of $\mathbf{u}_1$ on $\mathbf{t}_1$ as:

$$\mathbf{u}_1=\mathbf{b}_1\mathbf{t}_1+\mathbf{r}_1 \quad (3)$$

where $\mathbf{b}_1$ is the regression coefficient determined by the minimization of the residual $\mathbf{r}_1$. If $\mathbf{t}_1$ and $\mathbf{u}_1$ do not extract sufficient information, further score and loading vectors are calculated iteratively by extracting the information from deflated matrices for a specifically determined number of factors. The original input and output matrices are deflated as follows:

$$\mathbf{E}_1=\mathbf{X}-\mathbf{t}_1\mathbf{p}_1^T \quad (4)$$

$$\mathbf{F}_1=\mathbf{Y}-\mathbf{b}_1\mathbf{t}_1\mathbf{q}_1^T \quad (5)$$

Once the specified number of factors is extracted $\mathbf{C}_{pls}$ is calculated as:

$$\mathbf{C}_{pls}=\mathbf{W}^*\mathbf{B}\mathbf{Q}^T \quad (6)$$

with

$$\mathbf{B}=\mathrm{diag}(\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_a), \mathbf{Q}=[\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_a],$$

$$\mathbf{W}^*=[\mathbf{w}_1^*, \mathbf{w}_2^*, \ldots, \mathbf{w}_a^*], (\mathbf{w}_1^*=\mathbf{w}_1),$$

$$\mathbf{w}_a^*=\prod_{k=1}^{a-1}(\mathbf{I}_K-\mathbf{w}_k\mathbf{p}_k^T)\mathbf{w}_a$$

where $\mathbf{w}_i$ (i=1, …, a) are weighting factors and $\mathbf{I}_K$ is the identity matrix of dimension K. The number of factors for information extraction is usually determined by the cross validation [1]. The prediction error of sum of squares (PRESS) is calculated for different numbers of factors, and the number of factors giving the least PRESS is chosen for information extraction from process variables data set. The NIPALS algorithm is given in Table 1.

Table 1. PLS algorithm

1. Take the mean-centered and variance-scaled training matrices **X** and **Y**; let **E**=**X**; **F**=**Y**.
2. Start the iteration with **u**=any column of **Y**.
3. Calculate input weight and score vectors:
   $\mathbf{w}=\mathbf{E}\mathbf{u}/\mathbf{u}'\mathbf{u}$
   $\mathbf{w}=\mathbf{w}/\|\mathbf{w}\|$
   $\mathbf{t}=\mathbf{E}\mathbf{w}$
4. Calculate output loading and score vectors:
   $\mathbf{q}=\mathbf{F}'\mathbf{t}/\mathbf{t}'\mathbf{t}$
   $\mathbf{q}=\mathbf{q}/\|\mathbf{q}\|$
   $\mathbf{u}=\mathbf{F}\mathbf{q}/\mathbf{q}'\mathbf{q}$
5. Check for convergence:
6. Calculate input loading vector:
   $\mathbf{p}=\mathbf{E}'\mathbf{t}/\mathbf{t}'\mathbf{t}$
   $\mathbf{p}=\mathbf{p}/\|\mathbf{p}\|$
7. Find coefficients of regression:
   $b=\mathbf{u}'\mathbf{t}/\mathbf{t}'\mathbf{t}$
8. Deflate **X** and **Y**:
   $\mathbf{E}=\mathbf{E}-\mathbf{t}\mathbf{p}'$
   $\mathbf{F}=\mathbf{F}-b\mathbf{t}\mathbf{q}'$
9. Go to step 2 if next factor is to be extracted.

2. Mean and Variance Update

Since the PLS method depends upon how the process data is scaled, process and response variables are often symmetrically transformed to give equal weight to each process variable. Experience-based scaling is also possible which can give biased weight to relatively more correlated process variables. However, if this relative correlation is unknown, a common approach is the mean centering and scaling to unit variance. Mean centering is carried out by subtracting the mean of the variable from each data point of that variable. Compared to the mean centering scheme, the scaling to unit variance is obtained by dividing each data point of the variable by its standard deviation.

$$\mathrm{x}_{i,ms}=\frac{\mathrm{x}_i-\mathrm{m}}{\mathrm{s}} \quad (7)$$

where $\mathrm{x}_{i,ms}$ is the transformed value of x_i, i=1, 2, …, N, andrepresents the mean and standard deviation of the corresponding variable, respectively. Since the recursive PLS (see next section) updates the model by renewing the process data set to follow the process trend, mean and variance are also updated to keep providing the current scaling factors. This updating method is given as [6]:

$$\mathrm{m}_{h+1}=\frac{\mathrm{N}-1}{\mathrm{N}}\mathrm{m}_h+\frac{1}{\mathrm{N}}\mathrm{x}_{h+1} \quad (8)$$

$$\mathrm{s}_{h+1}^2=\frac{\mathrm{N}-2}{\mathrm{N}-1}\mathrm{s}_h^2+\frac{1}{\mathrm{N}-1}(\mathrm{x}_{h+1}-\mathrm{m}_{h+1})^2 \quad (9)$$

where m_h and s_h are the mean and variance of training data at the h^{th} addition of the new measurements, and m_{h+1} and s_{h+1} represent the corresponding values at $(h+1)^{th}$ addition.

3. Recursive PLS Update

Many industrial processes are vulnerable to the process environment drift that occurs due to aging and efficiency degradation of plant as well as grade changing events especially in polymerization processes. An adaptive model is vital to address these time-varying effects of processes. Helland et al. [3] developed a recursive PLS model by updating the training data set recursively and simultaneously

Table 2. Recursive PLS algorithm

Transform the training data using Eq. (7).
1. Calculate the coefficient of regression $\mathbf{C}_{pls}$ by Eq. (6)
2. Use the model to predict the response variable: $\mathbf{Y}_{pred}=\mathbf{X}\mathbf{C}_{pls}$.
3. On the arrival of each new measurement, add it to the training data and remove the oldest sample, update the mean and variance of data set using Eqs. (8) and (9).
4. Repeat the algorithm from 1.

removing the oldest data samples. In this way, the size of the matrices can be kept constant, the model can be adapted with new events, and process history can be partially retained. Qin [5] extended Helland's algorithm to give similar results to the PLS. A summary of the recursive PLS is given in Table 2.

4. Model Output Bias Update

Doubt about the reliability of process data and/or grade-changing events may result in RPLS giving deviated and unacceptable predictions. A soft sensor needs to be reliable and robustly adaptive, and for this reason, along with RPLS, model bias updating method is incorporated in the updating schemes. This method uses the difference of model prediction and the corresponding measurements to correct the model's upcoming predictions [7].

At t=0 (where t is the index of model update run: t=0 represents before the update),

$$\mathbf{Y}_{pred}=\mathbf{X}(t)\times\mathbf{C}_{pls} \quad (10)$$

And at $t=t^{th}$ run,

$$\text{bias}(t)=Y_{lab}(t-1)-Y_{pred}(t-1) \quad (11)$$

where $\mathbf{Y}_{lab}$ and $\mathbf{Y}_{pred}$ are the offline measurements and model predictions, respectively. Since the offline measurement of MI often takes several hours to complete, time lagging must be considered before calculating bias. Finally, the weighted bias is added to the predicted value to give the modified final prediction calculated by Eq. (12):

$$Y_{mod}(t)=Y_{pred}(t)+\text{bias}(t) \quad (12)$$

where $\mathbf{Y}_{mod}$ is the modified value of $\mathbf{Y}_{pred}$ by the (weighted) bias term.

Mu et al. [6] used an arbitrary weight ω to the bias of previous runs in calculation of the bias of the current run. It takes a value from 0 to 1 and is optimized according to the process behavior.

$$\text{bias}_0(t)=Y_{lab}(t-1)-Y_{pred}(t-1) \quad (13)$$

$$\text{bias}(t)=\omega\times\text{bias}_0(t)+(1-\omega)\times\text{bias}(t-1) \quad (14)$$

where $\text{bias}_0(t)$ is the current bias, bias(t) is the final bias to be used for the model output modification by Eq. (12). We have bias(0)=0.

ONLINE MODEL PREDICTION AND UPDATING SCHEMES

In this work two schemes are proposed to identify the slow and rapid changes of the process and to deal with them separately by the combination of the recursive PLS and model bias updating methods. Proposed online updating schemes take advantage of the two different updating methods in a fashion so as to minimize the prediction error. These two online updating methods are activated one at a time on the arrival of a new offline measurement. The activation sequence depends on significant changes within the same grade as well as grade-changing event detection. The measurements are used for calibration of the PLS model to adapt it with grade-changing characteristics of the process. Although both schemes are based on the same basic idea, they slightly differ from each other with respect to the parameters. In proposed schemes, the model bias update not only detects the grade-changing events but is used within the same grade as well, which helps the minimization of the number of RPLS update runs and making the soft sensor time efficient.

1. Proposed Scheme 1

The first scheme (scheme 1) takes the process variable data set as the input and builds a model through PLS. In several industrial processes, offline measurements of responses or quality variables are available after a certain interval. Therefore, the initial PLS model is used for the predictions during the interval until a new measurement is available. As soon as a new measurement arrives and is added to the process data, the model updates its parameters and generates predictions for the next interval. For the selection of update method for a certain instance, a threshold constant d is used, which decides the suitable and efficient updating method for the future predictions. The selection of the updating methods for this scheme relies on the threshold constant below which there is no effect on the RMSE of prediction. This constant acts as a switch between PLS update and model bias update; it chooses the PLS update when the absolute difference of current and previous measurements is less than d; otherwise the model bias update is selected for update of the model. The value of d is arbitrary and is dependent upon the quality variable responses and changing behavior. It affects the model predictions by selecting the update method not only at the occurrence of grade change but also during the same grade as well.

The PLS update is carried out for the prediction until the differ-

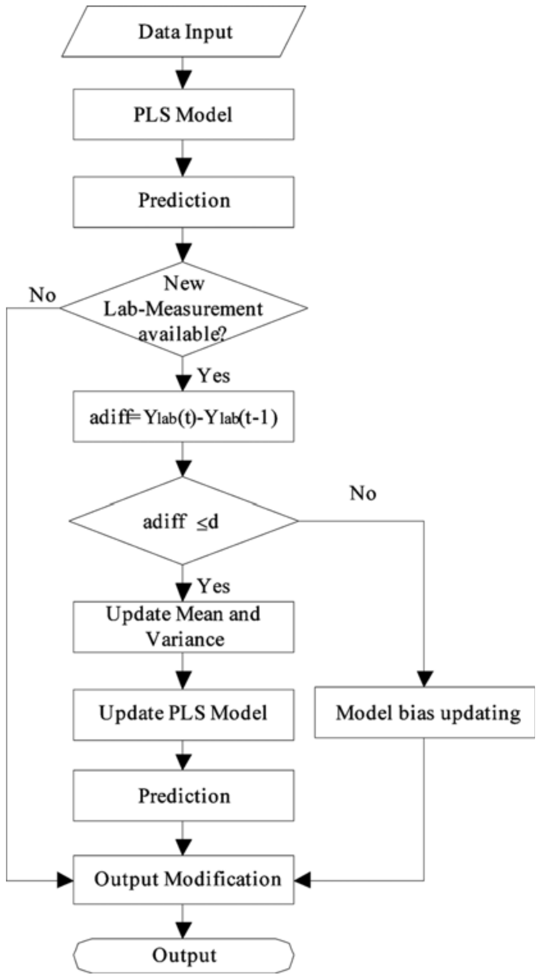

Fig. 1. Flowchart of update scheme 1.

ence of two consecutive measurements within the same grade reaches the threshold value of d or the process shifts to another grade. Whenever the difference reaches or exceeds the threshold d, the model bias update is activated and updates the bias term. Eventually, predictions from both update methods are modified by the bias term by using Eq. (12). The procedure is illustrated in Fig. 1.

When the process shifts from one grade to another, the difference between the first value of the shifted grade and the old value belonging to the previous grade is termed as grade-change-defining-value (GCDV). When the process consists of several grades, GCDV may have different values for various grades; in this case the minimum GCDV, which is 1.94 for this application, is selected for the optimization. The threshold constant is obtained from the optimization; one can start with the GCDV proceeding towards lower values. The threshold value giving the lowest RMSE is used as the updating method selector d (see section 4.5).

The determination of threshold constant for scheme 1 can be considered as an unconstrained optimization problem with the RMSE as objective function to be minimized subject to d having values from zero to GCDV. The optimization problem takes the following form:

Minimize:	RMSE(d)=Scheme 1	(15.1)
Subject to:	$0 \le d \le GCDV$	(15.2)

where the threshold constant d becomes an unknown variable in the optimization and is related to the RMSE through the update scheme 1.

2. Proposed Scheme 2

Scheme 2 differs from scheme 1 in that it prevents the unnecessary update and hence reduces the time taken by the soft sensor during online predictions. Our main goal is to develop a model with maximum prediction power and minimum time taken by the soft sensor. To follow the latter criterion, preference is given to no update at all (when there is no need of update) followed by the model bias update and lastly PLS update. To make the soft sensor time efficient, a lower bound value d_1 is identified. Below d_1 a system without update does not exhibit any remarkable change and has no or negligible effect on the RMSE. The model does not require any update in such cases. An upper bound d_2 is selected. Above d_2 the model bias update updates the bias and captures the sharp and rapid changes. The value of d_2 is optimized so as to make use of the model bias update during the significant changes within the same grade and drastic changes of grade shifting as well. The range between d_1 and d_2 represents the activation of PLS update.

For the soft sensor to be accurate and efficient, the prediction power of the model needs to be emphasized along with time efficiency. The trade-off between the former and latter criteria gives the range that provides the optimized number of runs for both the model bias update and the PLS update, leaving the remaining number of runs without update. For the optimization of the placement and the range between threshold values, one can start with zero and the GCDV as the lower and upper bounds, respectively, proceeding towards each other.

The lower and upper bounds placement along with the range are obtained by the optimization using Eq. (16). For scheme 2, the optimization problem is constrained with RMSE with the number of PLS updating run (NPR) as the objective function to be minimized and bounds placement and range as the unknown variables. The

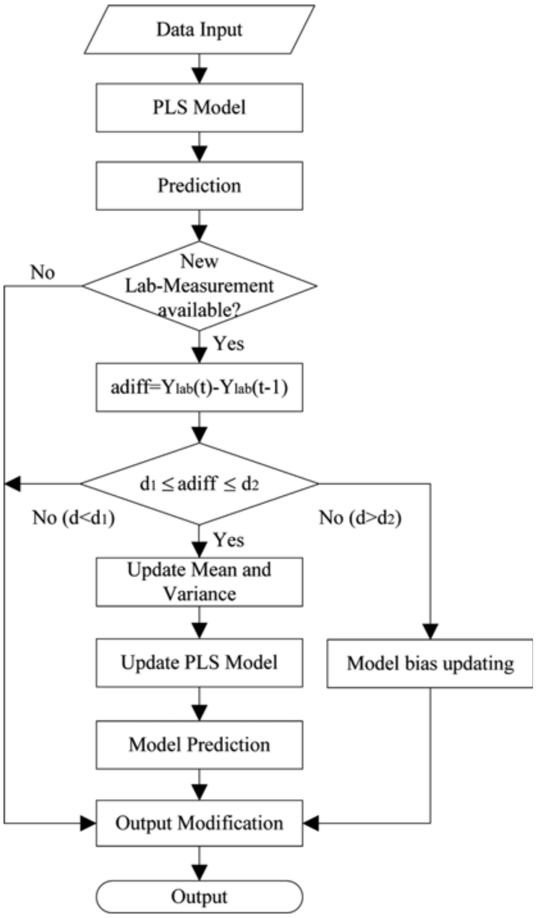

Fig. 2. Flowchart of update scheme 2.

optimization problem takes the following form:

Minimize:	NPR=Scheme 2	(16.1)
Subject to:	RMSE 2≤RMSE 1+compensation factor	(16.2)
	$0 \le d_1 \le GCDV$; $d < d_2 \le GCDV$	(16.3)

where the compensation factor is 0.0001 for this application, and RMSE 1 and RMSE 2 represent the RMSE obtained by scheme 1 and scheme 2, respectively.

The bound placement and range giving the least RMSE are used as the update method selection criteria (see section 4.5). Eventually, as in scheme 1, predictions from both update methods are modified by the bias terms. The procedure is illustrated in Fig. 2.

PREDICTION OF MI IN HDPE PROCESS

1. Process Description

In the present work, the High Density Polyethylene (HDPE) plant located in LG Petrochemicals, Korea, is studied. The plant is involved in the edge-cutting low-pressure polymerization manufacturing process. In the HDPE plant there are two polymerization processes, named K1 and K2. Two parallel reactors are operated in K1, whereas a cascade arrangement is used for K2. These reactions are highly exothermic that evolve 1,000 kcal/1 kg ethylene. Therefore, an efficient cooling system is required to remove polymerization heat from the reactor. The reactant feed of the reactor includes ethylene co-monomer, hydrogen, activator, catalyst, co-catalyst, hexane and mother liquor which is recycled continuously. The reactor volume is filled up to 90-95% with reaction slurry, which is transferred to the subsequent equipment with the rise in pressure to maintain the slurry level in the reactor. The K1 process operates under 8-10 kg/cm^2 and the K2 process operates under 2-4 kg/cm^2 with a temperature of range of 74-85 °C.

2. Numerical Simulation

For the input variable selection among 43 input variables, we first removed a variable denoting the activator feed rate which was constant throughout the process. Using interval PLS for input variable selection, 14 input variables were then selected from the remaining 42 process input variables. These variables include feed rates of co-monomer, hydrogen, catalyst and recycled hexane, partial pressure of C_2H_3 and H_2 and reactor pressure, etc. For the simulation purpose, 505 offline measurements were collected with an interval of 2 hours in 42 days. Two update methods (recursive PLS update and model bias update) as well as dual update method with a selection parameter "period" [6] were used for the comparison with the proposed schemes. Primarily, the PLS model was set up (see Table 1) with 300 data points for all the update strategies using one latent factor except for solo RPLS update for which four latent factors were used. The remaining samples were used to test the model updating prediction accuracy. It takes about two hours to get the offline measurement of MI in LG petrochemicals. Thus initially built model was used for the predictions with an interval of five minutes for two hours. As soon as a new measurement arrived and was added to the process data, a certain updating method was activated to update the model parameters, and predictions were taken for the next two hours. In this way, model updating and predictions were performed sequentially with the interval of two hours. As a performance criterion, the RMSE is used for comparison purpose and is given by

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{N}_t}\sum_{i=1}^{N_t}\left(\frac{\mathrm{Y}_{i,actual}-\mathrm{Y}_{i,mod}}{\mathrm{Y}_{i,actual}}\right)^2} \quad (17)$$

where N_t is the number of observations in test data set, whereas $\mathrm{Y}_{i,actual}$ and $\mathrm{Y}_{i,mod}$ are the actual and modified (after prediction) values of MI, respectively. The distinct features of the methods are described below:

(1) For the recursive PLS updating method, the PLS model is set up with five latent factors. The model is updated recursively with mean and variance updated by Eqs. (8) and (9) whenever a new measurement is added to the training data set.

(2) The model bias update is developed with one latent factor and the bias term is updated by Eq. (11) with each arrival of a new measurement.

(3) To develop the "period" strategy, parameters giving the least RMSE are selected, i.e., period P=13, 1 latent factor for PLS model, and ω=1 where ω is the weighted bias giving respective weights to the model bias at previous and current run. The bias term using ω can be calculated by Eq. (14). To follow the trend of rapidly changing effects closely, no weight is given to the bias at previous run for the predictions at current run.

(4) The proposed scheme 1 makes use of the threshold value d, which is 0.01 in this application, along with latent factor equal to 1.

(5) For the scheme 2 parameters, latent factors d1 and d2 with the values 1, 0.02 and 0.14 are selected as the optimized values, respectively. As mentioned earlier, the polymerization process operation consists of rapid grade change sequences which require the model bias update only with the current bias at that moment. Therefore, schemes 1 and 2 do not use weighted bias ω; it can be interpreted as if ω=1 was used in this application. However, depending on the process application response, it can be used to enhance the prediction accuracy.

RESULTS AND DISCUSSIONS

Fig. 3 shows the results for melt index predictions as a comparison among the five aforementioned update strategies. RPLS update method (Fig. 3A) updates the model and predicts the MI reasonably at the early stage, but starts to get deviated from actual MI values and tends to over-fit the model. Like Recursive PLS update, the predictions from model bias update (Fig. 3B) were found closer

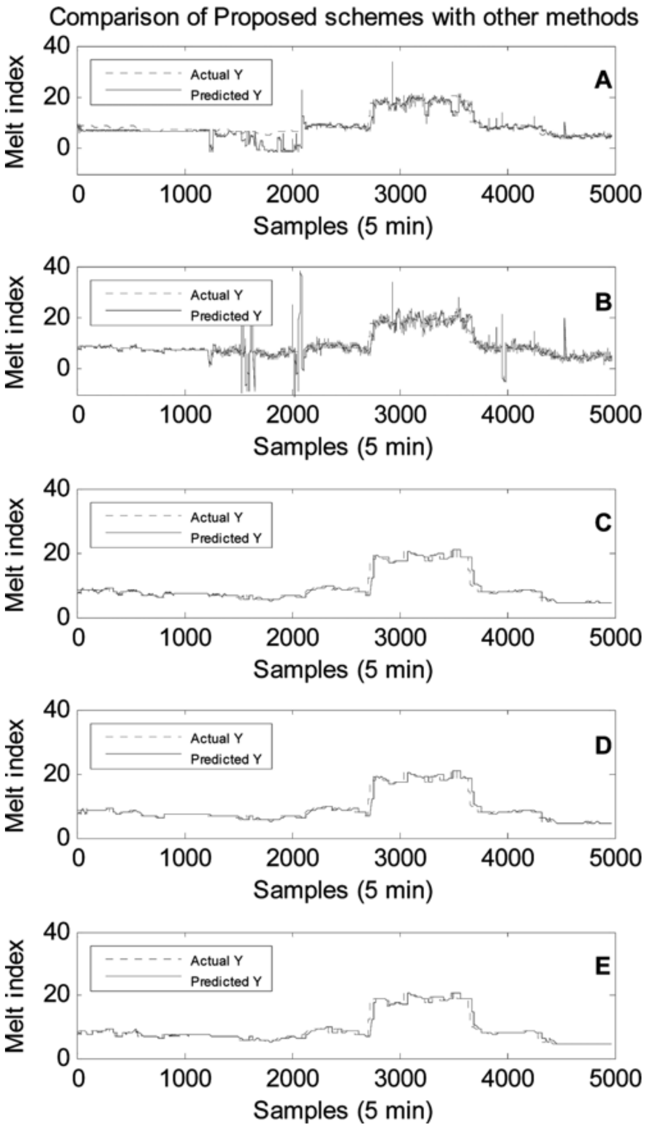

Fig. 3. Comparison of actual and predicted MI from five strategies. A. RPLS method; B. Model bias update method; C. Dual updating strategy by Mu; D. Proposed scheme 1; E. Proposed scheme 2.

Table 3. RMSE comparison of five updating procedures

No.	Procedure	RMSE
1	Solo RPLS updating	0.3557
2	Solo model Bias updating	0.5197
3	Period strategy	0.096
4	Scheme 1	0.0941
5	Scheme 2	0.0944

to the actual MI at the early stage. But this method also starts over-fitting the model much earlier than the grade is changed rapidly in the process. The reason for these large overshoots is the sudden change in certain process variables by the operator to change the grade, which unnecessarily enhances the effect of update. However, in comparison to the single model bias update, we can see that the RPLS update provides better predictions. Fig. 3C represents Mu's strategy [6], which follows the actual MI measurements closely and overcomes the deviation and large overshoots by activating the PLS at a period of 7. The proposed scheme 1 (Fig. 3D) gives better results with the reliability of selecting the suitable update method at a certain instance. And finally, the proposed scheme 2 (Fig. 3E) minimizes the number of PLS updating runs as compared to the scheme 1 and compensates it with a negligible increase in RMSE, as shown in Table 3.

In the polymerization process, an abrupt change occurs in process data when the process is near to shift to other grade. The single RPLS updates the PLS parameters recursively and tries to overcome the deviation caused by abrupt changes in process data. It also gives some small overshoots during and after grade change operations. As far as the model bias update is concerned, it is also vulnerable to these abrupt changes, and at the instances of these abrupt changes the model updates the bias term accordingly, which in turn over-modifies the predictions. However, it is observed that whenever the large bias is obtained due to the grade changing event, it tends to follow the predictions of MI. Mu's strategy [6] circumvents this problem of over-modification robustly by using the RPLS and model bias update with a period of 7. On the other hand, since the natural choice of update at certain instances makes it possible to keep track of minimization of the RMSE, the proposed schemes are observed to be robust against the abrupt changes in process data.

The instances of changing the grades as well as the duration for which the grade holds on to its position may differ from month to month. Since Mu's updating strategy [6] utilizes an arbitrarily fixed parameter "period" which is optimized on the basis of lowest RMSE for the specific grade changing instances of the test data of specific month, the method can fit the model with acceptable accuracy but may not keep track of the requirement of appropriate updating method at a certain instance. For this reason, the period P may need to be updated with the passage of time. Whereas the criterion of utilizing the threshold constant is quite inherent and depends upon the process response behavior, no changes or slow and rapid changes are identified, categorized and treated separately, by no updating, RPLS update and model bias update, respectively.

The main feature in the proposed scheme 2 is to shrink and place the threshold range (the distance between lower and upper threshold bounds) aptly to minimize the number of PLS updating runs so as to decrease the algorithm running time, and to maintain the RMSE with some compensation factor, or sometimes even to decrease the RMSE as well. By converting some updating instances to no updating, a small increase in RMSE may be observed. The value up to which the increase in RMSE is acceptable is termed as the compensation factor. The decrease in RMSE by scheme 2 is a rare case, which may be caused by inappropriate input variable selection and/or relative over-fitting (by the unnecessary update where no update is required) due to the presence of noise in input data since noise enhances the effect of over-fitting.

The value of the threshold constant d, which is equal to zero or close to zero, shrinks the number of runs of recursive PLS update. Therefore, the model bias update will be activated too frequently,

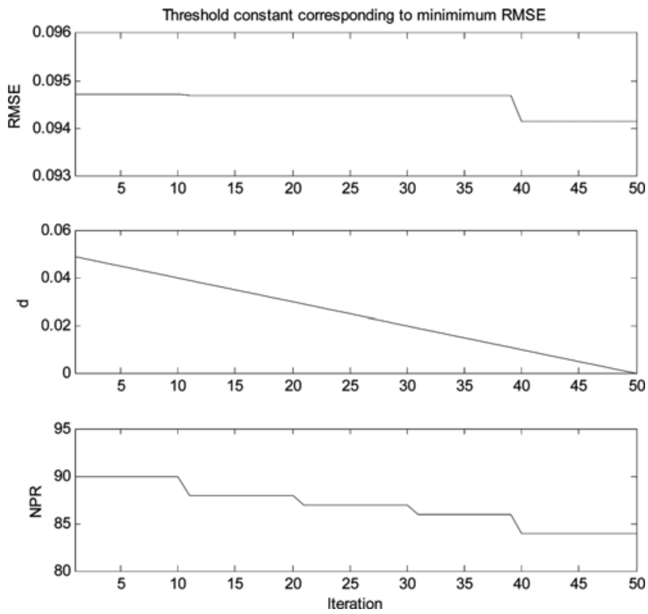

Fig. 4. Threshold constant corresponding to minimized RMSE.

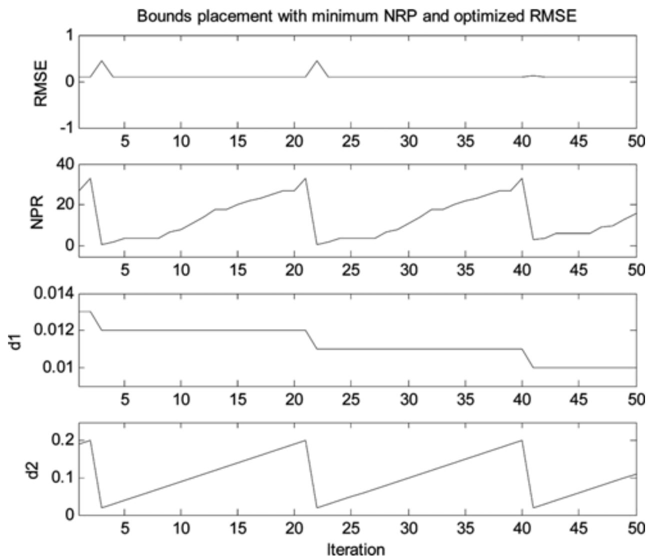

Fig. 5. Bounds placement and range corresponding to optimized RMSE and NRP.

Table 4. Parameter comparison among updating schemes and period strategy

Updating scheme	RMSE	NPR	NMBR	NNP	Parameters
Period strategy	0.096	29	176	-	Period=7
Scheme 1	0.0941	84	121	-	d=0.01
Scheme 2	0.0944	4	117	84	d1=0.01, d2=0.03

NMBR=Number of model bias updating run
NNP=Number of no updating

resulting in the possibility of over-modification. On the other hand, a value closer to GCDV will result in frequently activated recursive PLS runs and finally exhibiting over-fitting. Optimization was carried out for both the schemes and the results were truncated from both left and right sides to focus on the minimum RMSE value as shown in Figs. 4 and 5. Minimum RMSE=0.094 was observed at d=0.01 with the NPR=84 by conducting the optimization for the scheme 1 as shown at the 40^{th} iteration in Fig. 4. Utilizing an analogous approach, the bounds placement and range corresponding to the minimum NRP were investigated, where the optimization converges at the NRP=9 with RMSE=0.0941, d1=0.02, d2=0.014 and range=0.12 at the 13^{th} iteration (Fig. 5). Table 3 summarizes the parameter comparison among various updating schemes and period strategies, and Table 4 summarizes the optimization results from both proposed schemes as well as the results of comparison between the NPR of the period-strategy with proposed schemes.

CONCLUSION

In this work, we developed a soft sensor for predicting the product quality variable MI with grade changing characteristics by combining the recursive PLS updating and model bias updating. We formulated two model updating schemes: Scheme 1 and scheme 2. Scheme 1 intelligently decides the appropriate updating method from recursive PLS updating and model bias updating to minimize the RMSE; whereas scheme 2 focuses on minimizing the number of recursive PLS updating runs as well as maintaining the RMSE with a little compensation or sometimes even reducing it a little, in comparison with that obtained by scheme 1. The proposed schemes guarantee reliability and provide robust updating selection criteria with the passage of time along with satisfactory prediction of the melt index of high density polyethylene products. Superiority of the proposed schemes is shown by comparison with other existing update methods. A detailed study of determining the parameters is also conducted and presented with optimized results for both schemes.

The proposed schemes not only cope with the known and planned changes in process output variables (MI) but can also deal with unintentional and accidental changes due to unreliability of process input data, using a suitable updating method for the specific instance. Dependence of updating method selection criteria on the output variable (product quality variable) behavior renders the updating scheme able to deal with the irregular grade changing operations along with the regularly occurring events. We are hopeful about the usefulness of the proposed schemes in similar industrial operations.

NOMENCLATURE

a : index of factors (a=1, 2, …, A)
A : number of factors in PLS model
X : matrix of process input data, size (N*K)
Y : matrix of response variable, size (N*M)
t : score vector for X
u : score vector for Y
p : loading vector for X
q : loading vector for Y
E : residual matrix for X
F : residual matrix for Y
b : inner model coefficient
r : residual vector for inner model of PLS
C_{pls} : PLS regression coefficient matrix
W^* : matrix of transformed PLS weights
w^* : column vector of W*
B : matrix of regression coefficients, size (K*M)
Q : weight matrix for Y, size (M*A)
W : weight matrix for X, size (K*A)
w : column vector of W
k : index of w (k=1, 2, …, a)
K : no. of variables in X
bias : difference between current and previous lab measurement of response variable
Y_{pred} : prediction value of response variable
Y_{mod} : modified value of response variable by bias term
d : threshold constant
d_1 : lower bound
d_2 : upper bound
Range: difference of d_2 and d_1

REFERENCES

1. P. Geladi and B. R. Kowalksi, *Anal. Chim. Acta.*, **185**, 1 (1986).
2. S. Wold, N. Kettaneh-Wold and B. Skagerberg, *Chemom. Int. Lab. Sys.*, **7**, 53 (1989).
3. K. Helland, H. E. Berntsen and O. S. Borgen, *Chemom. Int. Lab. Sys.*, **14**, 129 (1992).
4. B. S. Dayal and J. F. MacGregor, *J. Process Control*, **7**, 169 (1997).
5. S. J. Qin, *Comput. Chem. Eng.*, **22**, 503 (1998).
6. S. Mu, Y. Zeng, R. Liu, P. Wu, H. Su and J. Chu, *Journal of Process Control*, **16**, 557 (2006).
7. R. U. Sharmin, U. Sundararaj, S. Shah, L. V. Griend and Y. J. Sun, *Chemical Engineering Science*, **61**, 6372 (2006).