Proceedings of the 9th International Symposium on
Dynamics and Control of Process Systems (DYCOPS 2010),
Leuven, Belgium, July 5-7, 2010
Mayuresh Kothare, Moses Tade, Alain Vande Wouwer, Ilse Smets (Eds.)

Multiple Model Approach for Inferential Instruments Design ⋆

Elom Domlan * **Biao Huang** ** **Fangwei Xu** *** **Aris Espejo** ****

* *Dept. of Chemical and Materials Engineering, University of Alberta Edmonton, AB, Canada, (e-mail: elom.domlan@ ualberta.ca).*
** *Dept. of Chemical and Materials Engineering, University of Alberta Edmonton, AB, Canada, (e-mail: biao.huang@ualberta.ca)*
*** *Syncrude Canada Ltd. Fort McMurray, AB, Canada (e-mail: xu.fangwei@syncrude.com)*
**** *Syncrude Canada Ltd. Fort McMurray, AB, Canada (e-mail: espejo.aris@syncrude.com)*

Abstract: In this paper, an application of a multiple model approach for the design of inferential instruments is reported. The multiple model of interest presents a decoupled structure in the sense that the sub-models do not share the same state variable. A two-stage identification procedure is developed for the model identification and a soft sensor application is later conducted with the decoupled multiple model structure. The soft sensor aims at predicting a quality variable for an industrial separation unit. The decoupled multiple model structure allows obtaining a dynamical model for the soft sensor despite the presence of practical constraints related to multi-rate sampling problem. Real-time implementation results are presented.

Keywords: soft sensor, multiple model, fuzzy model, multi-rate sampling

1. INTRODUCTION

The level of complexity of industrial processes has largely increased with the need for increasing plants productivity, profitability, and safety while conforming to regulations policies related to the limitation and reduction of environmental footprints. This represents a tremendous challenge for the operation of industrial processes as it generally requires a tight monitoring of the processes operating conditions along with the setup of production limits. A fundamental tool for the strict respect of these regulations policies and the fulfillment of production expectations is the existence of an overall reliable process instrumentation.

Classical process variables such as flow information, are generally easy to measure. However, in some particular situations, the environment in which the sensor is operating can make the measurement difficult or even practically impossible. Moreover, when dealing with the measurement of process variables related to product quality, the availability of real-time measurements can become problematic. On the one hand, the lack of an efficient sensing technology can refrain from having on-line measurements and, on the other hand, when the sensing technology is available, its price, the accuracy and frequency of the obtained readings constitute inconveniences that generally generate the need for alternative forms of instrumentation. One of the solutions generally adopted in the process industry is to take off-line samples on a certain frequency and proceed to off-line laboratory analysis for having a measurement of the product quality variable. Despite the accuracy gained with that approach, the resultant is a lack of on-line measurements with the introduction of delays in the measurements as the processing of the samples generally requires a significant amount of time.

Soft sensors (Patwardan et al., 2007) are an efficient alternative for handling the instrumentation problems mentioned above. They are mathematical models that are designed for various purposes such as serving as back-up for real instruments, having real-time or on-line measurements for infrequently sampled variables, or performing sensors validation.

The design of soft sensors finds its roots in process modeling. A first-principle modeling approach can be used for this task. However, this approach requires a complete and deep physical knowledge of the process, which can be time and money consuming without the guaranty of achieving the desired accuracy. Using the rich and huge historical dataset generally available from industrial processes, a data-driven approach (Kadlek et al., 2009) can also be considered for soft sensors design. The most commonly applied tools for data-driven soft sensors design relies on multivariate analysis tools such as principal component analysis (PCA) or projection of latent structures (PLS). Dynamic soft sensors are based on dynamic models, such as ARX (auto-regressive exogenous) models or Box-Jenkins models, that are used as the backbone of the soft sensors. These linear approaches quickly show their limitations when there exist non-linearities within the historical dataset. In this case non-linear approaches are better suited for soft sensors design.

The multiple model approach (Murray-Smith and Johansen, 1997) is a non-linear modeling tool for the rep-

⋆ This work was supported by NSERC and Syncrude Canada Ltd.

resentation of complex processes. The multiple model paradigm proceeds by a divide-and-conquer strategy. The operating space of the process to model is partitioned into multiple smaller operating regimes in which a sub-model with a simple structure, often a linear model structure, is used to locally describe the process dynamics. The sub-models are then interconnected to each other by the means of an interpolation function. Many non-linear models structures such as operating regime based models (Johansen and Foss, 1995), multiple models (Kiriakidis, 2007), piecewise linear models (Billings and Voon, 1987), mixture models (Titterington et al., 1985), or Takagi-Sugeno fuzzy models (Takagi and Sugeno, 1985; Yager and Filev, 1994; Babuška, 1998), follow the recipe of the multiple model approach. The multiple model approach is an efficient and simple framework for the identification and modeling of complex non-linear processes. Another advantage of the multiple model approach is that existing analysis and synthesis tools for linear systems can easily be adapted to this class of models at the cost of very little modification.

As mentioned earlier, the interconnection between the sub-models of a multiple model structure is done by an interpolation function that proceeds to a smooth blending of the local output of the sub-models in order to generate the global output of the model. Two interconnection structures can generally be distinguished for the sub-models. In the most frequent structure, the sub-models share a common state variables vector and only differ by their parameters and the local operating regime in which they operate. The typical example model of that type of interconnection is the Takagi-Sugeno fuzzy model (Takagi and Sugeno, 1985), which is well-known in the literature and has been reported in many successful process modeling application and soft sensor design problems. The second type of interconnection results in a decoupled multiple model structure in which each sub-model has its own state variables and evolves independently of the other sub-models. The identification of a decoupled multiple model structure is presented in this paper. A two-stage identification procedure is proposed. The identification algorithm alternates between the partitioning of the process operating regime and the estimation of the sub-models parameters. A non-linear optimization technique is used at each stage. The performance of the obtained model is illustrated through an industrial application for predicting a key quality variable for an industrial separation unit. The on-line results highlight the potential of the decoupled multiple model approach in modeling complex processes.

2. THE CLASSICAL MULTIPLE MODEL APPROACH

In the framework of the multiple model approach, the most used multiple model structure is the one in which the sub-models share a common state variable, resulting in a coupled multiple model architecture. The state space representation of that form of the multiple model is given by:

$$\begin{cases} x\left(k+1\right) = \sum_{i=1}^{s} \left(\mu_i\left(\xi\left(k\right)\right)\left(A_i x\left(k\right) + B_i u\left(k\right)\right)\right) \\ y\left(k\right) = \sum_{i=1}^{s} \left(\mu_i\left(\xi\left(k\right)\right) C_i x\left(k\right)\right) \end{cases} \tag{1}$$

In equation (1), the variable $x \in \mathbb{R}^n$ is the state vector. The input and output variables are respectively represented by $u(\cdot) \in \mathbb{R}^m$ and $y(\cdot) \in \mathbb{R}^l$. The interpolation or weighting functions are represented by the functions $\mu_i(\cdot)$, $i = 1, \ldots, s$. They act as a local validity measure for the sub-models regarding the current operating regime of the process and they are parametrized by the decision variables vector $\xi(\cdot) \in \mathbb{R}^p$. The decision variables can be composed of lagged inputs and outputs of the process or any auxiliary variable allowing the non-linearities of the process to be taken into account. They are generally process variables for which real-time measurements or estimation are available. Although numerous weighting functions can be selected, the weighting functions are often chosen as normalized Gaussian or sigmoid functions that verify the following constraints:

$$\sum_{i=1}^{M} \mu_i\left(\xi\left(k\right)\right) = 1, \forall k, \mu_i\left(\xi\left(k\right)\right) \geq 0, \forall k, \forall i \tag{2}$$

The classical multiple model of equation (1) can be linked to the Takagi-Sugeno fuzzy model(Takagi and Sugeno, 1985). An equivalent input/output form of the multiple model of equation (1) is given by:

$$\begin{cases} y(k) = \sum_{i=1}^{s} \mu_i\left(\xi\left(k\right)\right) y_i\left(k\right) \\ y_i(k) = \varphi^T(k)\theta_i, i = 1, \ldots, s \end{cases} \tag{3}$$

where $y_i(\cdot)$, $i \in \{1, \ldots, s\}$ is the output of the i-th sub-model which depends on a regression vector $\varphi(k)$ and on a local parameters vector θ_i, $i \in \{1, \ldots, s\}$ with:

$$\begin{aligned} \varphi^T(k) = \quad & [y(k-1) \ldots y(k-n_y) \\ & u(k-1) \ldots u(k-n_u) \quad 1] \end{aligned} \tag{4}$$

The orders of the sub-models are given by n_y and n_u. The input/output representation is more adapted for model identification purposes.

Once the decision and input variables are selected, the identification of a multiple model generally consists of two sub-problems. First, estimate the parameters of the weighting functions $\mu_i(\cdot)$, $i = 1, \ldots, s$, which is equivalent to the partitioning of the operating space of the process, and second, estimate the parameters vectors θ_i, $i = 1, \ldots, s$ of the sub-models. The selection of the premise and input variables generally relies on model-based exhaustive tests or available physical knowledge of the process. Different methods, such as usage of *a priori* knowledge on the process (Murray-Smith and Johansen, 1997), model structure search tree (Johansen and Foss, 1995) or clustering methods (Babuška, 1998), have been proposed in the literature for the task of partitioning the process operating regime. With the knowledge of the partitioning of the process operating regime, the local parameters of the sub-models are obtained through the minimization of an output error cost function. The sub-models parameters are finally given by(Babuška, 1998):

$$\Theta = \left(\Phi^T\Phi\right)^{-1}\Phi^T Y \tag{5}$$

where Y is composed of consecutive stacked values of the process output. The parameters vectors Φ and Θ are given by:

$$\Theta = \left[\begin{array}{cccc} \theta_1^T & \theta_2^T & \ldots & \theta_s^T \end{array}\right] \tag{6}$$

$$\Phi = [\,\psi_1\ \psi_2\ \ldots\ \psi_s\,] \tag{7}$$

$$\psi_i = \begin{bmatrix} \mu_i\left(\xi\left(1\right)\right)\varphi^T\left(1\right) \\ \mu_i\left(\xi\left(2\right)\right)\varphi^T\left(2\right) \\ \vdots \\ \mu_i\left(\xi\left(N\right)\right)\varphi^T\left(N\right) \end{bmatrix}, i = 1, \ldots, s \tag{8}$$

and N is the number of data points in the identification dataset.

Concerning the choice of the number of sub-models, the strategy generally adopted is to start the identification procedure with a large number of sub-models and then refine later the obtained model. The refinement procedure is based on the merging of "neighboring" sub-models using a similarity measure. An alternative method is to opt for a prior small number of sub-models and then progressively increase the number of sub-models based on a prediction error criterion.

3. THE DECOUPLED MULTIPLE MODEL APPROACH

The first reference to a decoupled multiple model structure can be traced back to (Filev, 1991). The state space representation of this form of multiple model is expressed as:

$$\begin{cases} x_i\left(k+1\right) = \sum_{i=1}^{s}\left(\mu_i\left(\xi\left(k\right)\right)\left(A_i x_i\left(k\right) + B_i u\left(k\right)\right)\right) \\ y_i\left(k\right) = C_i x_i\left(k\right) \\ y\left(k\right) = \sum_{i=1}^{s}\left(\mu_i\left(\xi\left(k\right)\right) y_i\left(k\right)\right) \end{cases} \tag{9}$$

The difference here with the classical multiple model of equation (1) is the apparition of the local state variables $x_i(\cdot)$ and the local sub-models outputs $y_i(\cdot)$. Indeed, each sub-model evolves in its own state space and the global output $y(\cdot)$ of the model is obtained by combining the local outputs $y_i(\cdot)$ of the sub-models. The input/output form of the decoupled multiple model structure is given by:

$$\begin{cases} y(k) = \sum_{i=1}^{s} \mu_i\left(\xi\left(k\right)\right) y_i\left(k\right) \\ y_i(k) = \varphi_i^T(k)\theta_i, i = 1, \ldots, s \end{cases} \tag{10}$$

with:

$$\varphi_i^T(k) = \begin{array}{l} [y_i(k-1)\ldots y_i(k-n_y) \\ \qquad u(k-1)\ldots u(k-n_u)\quad 1] \end{array} \tag{11}$$

Note that the decoupled multiple model structure allows more flexibility in the choice of the structure of the sub-models as they are not restricted to share a common structure.

In (Orjuela et al., 2006), an identification procedure for a decoupled multiple model has been proposed. The procedure provides an estimation of the sub-models parameters based on an *a priori* selected partitioning of the operating space. A two-stage algorithm is presented here for the simultaneous estimation of the weighting functions, which is equivalent to the partitioning of the operating space, and the sub-models parameters. It is assumed that the number of sub-models s, the decision variables, and the input variables are *a priori* selected. The weighing functions are also represented as normalized Gaussian functions:

$$\mu_i(\xi(k)) = \frac{\omega_i(\xi\left(k\right))}{\sum_{j=1}^{s}\omega_j(\xi\left(k\right))} \tag{12}$$

with

$$\omega_i(\xi(k)) = exp\left(-\frac{\left(\xi\left(k\right) - c_i\right)^2}{\sigma^2}\right) \tag{13}$$

The identification of the decoupled multiple model structure (10) requires the estimation of the parameters vector Θ and Ξ defined as: $\Theta = [\theta_1^T\ \theta_2^T \ldots \theta_s^T]$ and $\Xi = [c_1 \ldots c_s\ \sigma]$. The estimation of Θ and Ξ is achieved by minimizing the cost function (14):

$$J = \sum_{k=1}^{N}\left(y\left(k\right) - y_m\left(k\right)\right)^2 \tag{14}$$

where $y_m(\cdot)$ is the measured output and N is the number of available measurements.

It is obvious that the minimization of J is a non-linear optimization problem in regard to Θ and Ξ, which requires the usage of an iterative procedure. The Levenberg-Marquadt algorithm is adopted here and a two-stage hierarchical optimization is conducted. In each stage, the Levenberg-Marquadt algorithm is used to iteratively estimate whether the weighting functions parameters Ξ or the sub-models parameters Θ:

Algorithm *(Iterative identification procedure)*

1. *Initialization:* $l = 0$, $\Xi_l = \Xi_0$, $\Theta_l = \Theta_0$, $\eta_{\Xi_l} = \eta_{\Xi_0}$, $\lambda_{\Xi_l} = \lambda_{\Xi_0}$, $\eta_{\Theta_l} = \eta_{\Theta_0}$, $\lambda_{\Theta_l} = \lambda_{\Theta_0}$.
2. *Calculate* $J(l)$ *using equations* (10) *and* (14).
3. *Calculate* $\hat{\Theta}_{l+1}$ *using:* $\Theta_{l+1} = \Theta_l - \eta_{\Theta_l}(H_{\Theta_l}^{-1} + \lambda_{\Theta_l} I)^{-1} G_{\Theta_l}$
4. *Calculate* $J_\Theta(l+1)$ *from equations* (10) *and* (14).
5. *If* $J_\Theta(l+1) > J(l)$ *then update* η_{Θ_l} *and* λ_{Θ_l} *and goto step 3 ; else* $J(l+1) = J_\Theta(l+1)$.
6. *Calculate* Ξ_{l+1} *using:* $\Xi_{l+1} = \Xi_l - \eta_{\Xi_l}(H_{\Xi_l}^{-1} + \lambda_{\Xi_l} I)^{-1} G_{\Xi_l}$
7. *Calculate* $J_\Xi(l+1)$ *from equations* (10) *and* (14).
8. *If* $J_\Xi(l+1) > J(l)$ *then update* η_{Ξ_l} *and* λ_{Ξ_l} *and goto step 6 ; else* $J(l+1) = J_\Xi(l+1)$.
9. $l = l+1$*; repeat from step 2 until* $\|J(l+1) - J(l)\| < \varepsilon$ *or* $l > n_{iter}$

$J(l)$ is the value of the cost function (14) at the step l. The estimated values of the parameters vectors Θ and Ξ at the iteration l are respectively denoted Θ_l and Ξ_l. The cost functions $J_\Theta(l)$ and $J_\Xi(l)$ are respectively the values of the cost function (14) evaluated with $\Theta = \Theta_l$ and $\Xi = \Xi_l$. The variables H_{Θ_l} and G_{Θ_l} (respectively H_{Ξ_l} and G_{Ξ_l}) are respectively the Hessian and the gradient of the cost function (14) evaluated at Θ_l (respectively Ξ_l). They are calculated at each iteration using equations (10) and (14).

The scalars $\lambda_{(\cdot)}$ are non-negative scalars and $\eta_{(\cdot)}$ is related to the convergence speed of the algorithm.

From the initial guesses Ξ_0, Θ_0, η_{Ξ_0}, λ_{Ξ_0}, the parameters vector Θ is updated using the recursion in step 3. Θ is updated until $J_\Theta(\cdot)$ is lower than $J(\cdot)$. The second stage of the algorithm then starts with the update of the the parameters vector Ξ with the recursion in step 6. Again, the update is performed until $J_\Xi(\cdot)$ becomes lower than $J(\cdot)$. The procedure is stopped whenever the condition $\|J(l+1)-J(l)\| < \varepsilon$ is satisfied or the maximal number of iterations n_{iter} has been reached. Note that the presented algorithm is only valid if the the decision variables vector $\xi(\cdot)$ does not include lagged outputs of the process. The algorithm could however be extended to take into account that case. The modifications that have to be made in that configuration are related to the computation of the Hessian and the gradient of the cost function (14), which heavily depends in that case on the type of weighting functions being used.

The partitioning of the decision variables space into a grid partition provides a good initial guess for Ξ. Due to its iterative nature, the Levenberg-Marquadt algorithm inherits the local optima problem that is quite common with non-linear optimization algorithms. The identification algorithm has to be run in a Monte-Carlo-like scheme in order to reduce the side effects of the local optima problem. Concerning the selection and tuning of the scalars $\eta_{(\cdot)}$, $\lambda_{(\cdot)}$, some ideas are presented in (Fletcher, 1987). As a rule of thumb, $\eta_{(\cdot)}$ is increased and $\lambda_{(\cdot)}$ decreased when the error term $J_{(\cdot)}(\cdot)$ is decreasing and vice-versa.

4. APPLICATION

4.1 Process description

For the production of oil from oil sands facilities, the different components of the oil sands must be separated into different streams in order to retain only the stream that is almost exclusively composed of raw oil known as bitumen. The bitumen is then later further processed for obtaining crude oil as final product. The separation of the bitumen from the other components of the oil sands, which are generally water and solids, is accomplished through various separation units that relies on different variations of gravity separation principles (Dickey, 1961).

The inclined plate settling (IPS) unit is one of the separation units used for the primary objective of separating bitumen from the other components of the oil sands. This type of separation unit is very popular in minerals processes industry. Figure 1 shows a simplified representation of an IPS unit. The objective is to allow most of the bitumen present in the feed to leave the unit through the overflow product stream while allowing the other components of the feed (mainly water and solids) to settle down at the bottom of the vessel and leaves the unit through the underflow stream. Process aids are added to the feed as a facilitator of the separation process. One of the key variables that has to be monitored constantly is the quality variable related to the percentage of water (or water-content) present in the overflow product. This not only serves as quality indicator for the overflow product but also is essential for the good functioning of downstream processes.

In oil sands facilities exploitation, reliable instrumentation is one of the keys for an efficient and safe operation (Dougan and McDowell, 1997). Analyzers exist for the measurement of the water-content value but they require high maintenance for an optimal efficiency. Laboratory analyses are also performed on samples, taken at a rate of one to three samples per shift, in order to measure the percentage of water present in the overflow product stream. Although the measurements obtained through the laboratory analyses are the most reliable and accurate, they do not allow the real-time monitoring of the process. The reason is that the laboratory analyses need a long processing time and, therefore, they introduce delays in the measurement. Their availability is also dependant on the sampling frequency. As an alternative, soft sensors are considered for the frequent prediction of the water-content of the overflow product stream in real-time.

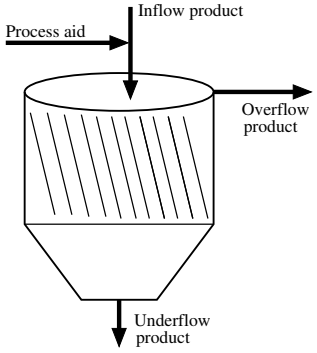

Fig. 1. Inclined Plates Settler (IPS) unit

4.2 Soft sensor design with a decoupled multiple model structure

For the design of the soft sensor, the usage of the classical multiple model structure presented in section 2 does not provide the means for designing a dynamical model. Indeed, as the sampling rate of the laboratory samples is irregular and relatively low (between two to six hours per sample) comparing to the sampling rate desired for the soft sensor, which is one minute, obtaining lagged values of the process output as described in the model of equation (3) is not feasible. Consequently, only a static model can be designed with the classical multiple model of equation (3). The past values of the output variables are not directly available during the design process. This motivates the choice of the decoupled multiple model structure.

The process variables that are considered are the process aid dosage, the feed flow rate, the underflow stream flow rate, and the level measurement for the product inside the IPS unit vessel. Due to the small number of process variables involved in the design, an exhaustive search approach has been conducted in order to define the structure of the sub-models. The level measurement and the process aid dosage are the components of the decision variables $\varepsilon(\cdot)$ and the input variables $u(\cdot)$ are the process aid dosage, the feed flow rate, the underflow

stream flow rate, and the level measurement. The regression vector $\varphi_i^T(\cdot)$ of each sub-model is given by $\varphi_i^T(k) = [y_i(k-1)\ \ u_P(k-1)\ \ u_F(k-1)\ \ u_U(k-1)\ \ u_L(k-1)]$, and the decision variables vector $\xi(\cdot)$ is given by $\xi(k) = [u_P(k-1)\ \ u_L(k-1)]$, where the variables $u_P(\cdot)$, $u_F(\cdot)$, $u_U(\cdot)$, and $u_L(\cdot)$ respectively denote the process aid dosage, the feed flow rate, the underflow stream flow rate, and the level measurement. The choice of the number of sub-models is done by comparing prediction perfomances for different values of the number of sub-models. The final model makes usage of nine sub-models. As stated before, the grid partition is a good starting point for choosing the initial guesses of the weighting function parameters, especially in this particular application where the dimension of the decision variables space is relatively low.

The identification dataset has been collected on a period of one year of operation of the IPS unit. A 3σ-edit rule has been used as data pre-processing technique for removing outliers in the identification dataset. The industrial data presented here has been normalized in order to protect proprietary information. Figure 2 shows a validation result with the decoupled multiple model. In the top graphic of figure 2, the solid line represents the trend of the laboratory data for the water-content values and the dashed line is the output of the soft sensor. The second graphic in figure 2 illustrates the alignment of the soft sensor output in regard to the laboratory data. The prediction accuracy achieved by the soft sensor on the validation dataset is quite good. Note that this is an off-line validation with laboratory data. The sampling frequency of the lab data during that period was around one to three samples per shift.

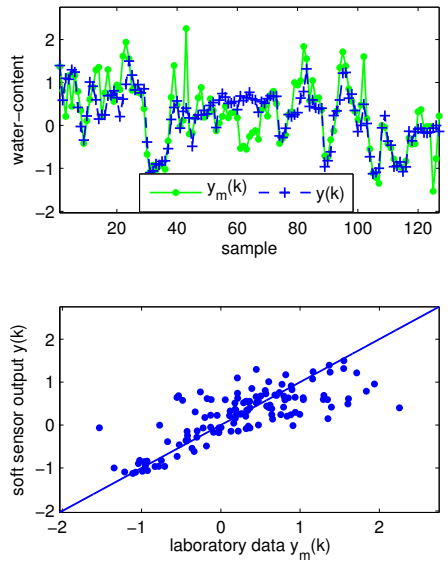

Fig. 2. off-line validation with the decoupled multiple model

In order to perform an on-line implementation of the soft sensor, an OPC (Object linking and embedding for Process Control) object in Matlab^{TM} has been used as communication channel between the soft sensor and a tag created on the DCS (Distributed Control System). By doing this, all the necessary computations for the soft sensor are kept in Matlab^{TM} and only the calculations results are sent back to the DCS through the OPC connection. Figure 3 shows the real-time trend of the process aid dosage, the feed flow rate, the underflow stream flow rate, and the level measurement during a period of two weeks. The sampling rate is one minute. Again, all the data presented is normalized for the sake of proprietary information. The

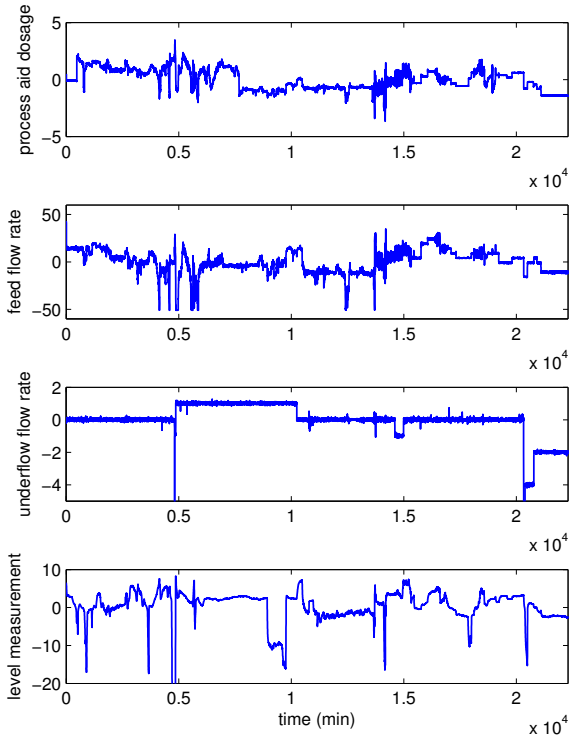

Fig. 3. input process variables

results obtained during this period with the soft sensor are presented on the top graphic of figure 4. The dot markers denote the timestamps at which laboratory data was available for the water-content measurement. The top graphic shows the soft sensor output during this period and the bottom graphic presents the readings provided by the on-line analyzer during the same period. The laboratory data was available on an average basis of one sample every four to six hours and with a delay of one hour between the time at which the sample is taken and the time at which the analysis result is available on-line. It can be noticed that the soft sensor exhibits a good prediction performance and shows a better alignment with the laboratory data than the on-line analyzer. It provides an improvement in terms of the accuracy of the on-line measurements of the water-content values.

Figure 5 displays a comparison of the water-content measurements provided by the soft sensor and the on-line analyzer with the laboratory data for the water-content values. Here again, a better alignment to the laboratory data is obtained with the soft sensor. For this dataset, the RMSE (Root Mean Squared Error) calculated for the soft sensor was 0.76 in contrast to 2.39 for the analyzer.

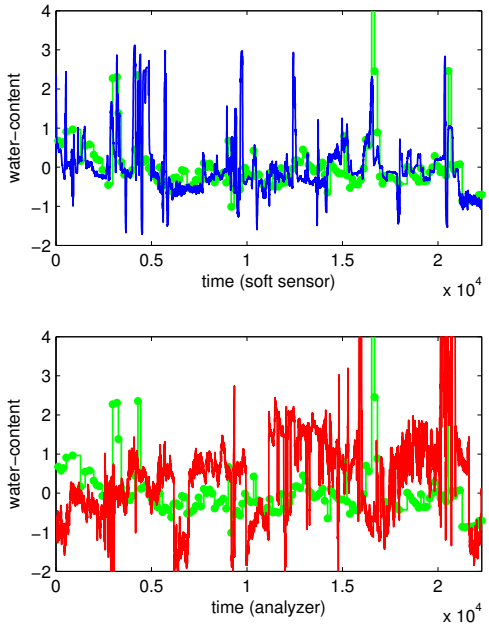

Fig. 4. real-time prediction (• lab data, — soft sensor output, — analyzer reading)

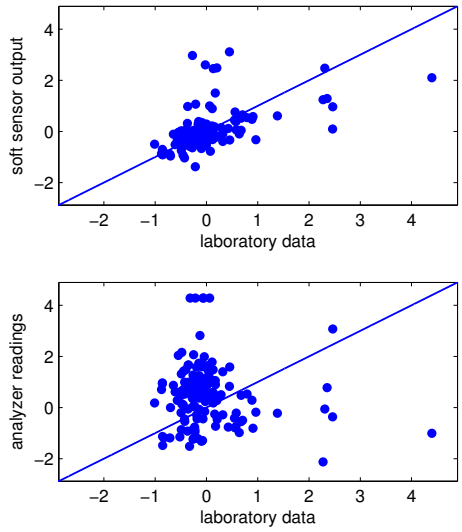

Fig. 5. lab data versus soft sensor output, lab data versus analyzer readings

In addition to the direct applications of the soft sensor as on-line predictor and back-up instrument, the information provided by the soft sensor can potentially be used for performing a tighter control on the process aid dosage. Future work will consider this control opportunity that can help to close the process aid dosage loop, which is currently in manual control.

5. CONCLUSION

A decoupled multiple model approach is presented for the design of an inferential instrument. The sub-models of the multiple model do not share a common state variable vector, allowing the design of a more general and flexible model structure. The counterpart of this flexibility is the need to use a non-linear iterative algorithm for the identification of the model parameters. An industrial application performed on a separation unit shows the effectiveness of the multiple model approach in modeling non-linear behaviors.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the financial and technical support of NSERC and Syncrude Canada Ltd.

REFERENCES

Babuška, R. (1998). *Fuzzy modeling for control*. Kluwer Academic Publishers, Boston.

Billings, S.A. and Voon, W.S.F. (1987). Piecewise linear identification of nonlinear systems. *International Journal of Control*, 46(1), 215–235.

Dickey, G.D. (1961). *Filtration*. Renhold, New York; Chapman and Hall, London, London.

Dougan, P. and McDowell, K. (1997). Sensor development in oil sand processing. In *Proceedings of the Dynamic Modeling Control Applications for Industry Workshop,*, 68–73. IEEE Industry Applications Society, Vancouver, Canada.

Filev, D.P. (1991). Fuzzy modeling of complex systems. *International Journal of Approximate Reasoning*, 5(3), 281–290.

Fletcher, R. (1987). *Pratical methods of optimization*. Wiley-Interscience, New York, NY, USA.

Johansen, T.A. and Foss, B.A. (1995). Identification of non-linear system structure and parameters using regime decomposition. *Automatica*, 31(2), 321–326.

Kadlek, P., Gabrys, A., and Strandt, S. (2009). Data-driven soft sensors in the process industry. *Computers and Chemical Engineering*, 33, 795–814.

Kiriakidis, K. (2007). Nonlinear model by interpolation between linear dynamics and its application in control. *Journal of Dynamic Systems, Measurement, and Control*, 129, 813–824.

Murray-Smith, R. and Johansen, T. (1997). *Multiple model approaches to modelling and control*. Taylor & Francis, London, United Kingdom.

Orjuela, R., Maquin, D., and Ragot, J. (2006). Nonlinear system identification using uncoupled state multiple model approach. In *Proceedings of the 4th Workshop on Advanced Control and Diagnosis*. Nancy, France.

Patwardan, S.C., Prakash, S.L., and Shah, S.L. (2007). Soft sensing and state estimation: review and recent trends. In *Proceedings of the IFAC Conference on Cost Effective Automation in Networked Product Development and Manufacturing*. Monterrey, Mexico.

Takagi, T. and Sugeno, M. (1985). Fuzzy identification of systems and its application to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15(1), 116–132.

Titterington, D., Smith, A.F.M., and Makov, U.E. (1985). *Statistical analysis of finite mixture distributions*. John Wiley & Sons, Chichester, UK.

Yager, R. and Filev, D.P. (1994). *Essentials of fuzzy model and control*. John Wiley, New York.