Application of Computational Intelligence Techniques to Process Industry Problems

Petr Kadlec[1], Bogdan Gabrys[1]

[1] Computational Intelligence Research Group, Bournemouth University
Fern Barrow, Poole BH12 5BB, United Kingdom.
E-mail: {pkadlec, bgabrys}@bournemouth.ac.uk

Abstract. In the last two decades there has been a large progress in the computational intelligence research field. The fruits of the effort spent on the research in the discussed field are powerful techniques for pattern recognition, data mining, data modelling, etc. These techniques achieve high performance on traditional data sets like the UCI machine learning database. Unfortunately, this kind of data sources usually represent clean data without any problems like data outliers, missing values, feature co-linearity, etc. common to real-life industrial data. The presence of faulty data samples can have very harmful effects on the models, for example if presented during the training of the models, it can either cause sub-optimal performance of the trained model or in the worst case destroy the so far learnt knowledge of the model. For these reasons the application of present modelling techniques to industrial problems has developed into a research field on its own. Based on the discussion of the properties and issues of the data and the state-of-the-art modelling techniques in the process industry, in this paper a novel unified approach to the development of predictive models in the process industry is presented.

Keywords. Soft Sensor, Process Industry, Adaptive Modelling, Meta Learning

1 Introduction

Processing plants in the industry are heavily instrumented with a large variety of sensors. The original purpose of the instrumentation was for monitoring and controlling purposes but in the last two decades the data being measured and stored has found a new application in the form of *Soft Sensors* [13]. The term is a combination of the words “software”, because the models are usually computer programs, and “sensors”, because the models are delivering similar information as real or hard sensors. Soft Sensors extract useful information from the process data which in the first instance is recorded for process control purposes. On a very general level one can distinguish between two types of Soft Sensors. On one hand there are the so called *First Principle Models*. First principle models attempt to use the knowledge of the physical and chemical laws for building models of the processes. Unfortunately the processes are usually much too complex to be fully described by this kind of models. This fact was one of the reasons which triggered the interest of the process industry in the development of *Data-Driven* Soft Sensors, which are the topic of this work. Within the process industry the most common data-driven Soft Sensors are based on the Principle Component Analysis [22] and Partial Least Squares [1] from the field of multi-variate statistics and Artificial Neural Networks [6], Self Organizing Maps [26] and Neuro-Fuzzy approaches [21] from the soft computing research field.

There is a broad field of application possibilities for Soft Sensors which include:

- Prediction of values, which can be measured with high financial or temporal effort only. Typical example of such a value is the product purity (concentration), which is often related to the product quality. Soft Sensors provide in such a case on-line estimation of the target values based on the available process data.
- Another application domain is the back-up of measuring devices, where the Soft Sensors operate in parallel with the measuring device. Closely related to this application is another one, namely the replacing of measuring devices and thus reducing hardware requirements.
- Next area of application is process monitoring, in this case the Soft Sensor is trained to recognize process states which violate the limits of ”normal operation” of the process.
- Another area of Soft Sensor applications is their usage for validation, fault detection and diagnosis purposes.

Apart from advantages provided by the Soft Sensors there are some difficulties and drawbacks in their building and application. In order to take them in real-life operation, certain regulatory and process safety measures, like guaranteed precision and reliability, have to be fulfilled by the Soft Sensors. Another issue of the current Soft Sensor development practice is that there is still a lot of effort necessary for their development and maintenance. The main focus of the current research is to address the specific topic of adaptation to the changing environment or to internal changes of the process and, thus to achieve the ability to self-develop. This will, on one hand, reduce the development costs of Soft Sensors and, on the other hand, reduce the demand for their maintenance and life-cycle costs. This, in turn, will increase the productivity and bring additional economical benefits and higher quality products. The move in the direction of reduction of the incorporated a-priori information about the actual process facilitates the portability and deploying of Soft Sensors to different processes with as little effort as possible.

A lot of the issues of current Soft Sensor modelling originate in the nature of the data delivered from process industry. In larger plants it is quite common that parts of them are taken out of service because of maintenance or failure reasons. This fact causes that for a certain time there is no (useful) data provided by the affected sensors. There are several issues which have to be solved to be able to deal with missing values. The first one is the detection of missing values and their replacement if required. An alternative way of handling missing values is to provide the Soft Sensor with the capability of dealing with changing dimensionality of the input space. Another problem to deal with is data outliers. These are particular samples which, mainly because of measurement failures, are distinct from the other samples. These samples are also not easy to identify and a strategy for their replacing or skipping should be implemented by the Soft Sensor (for a review of outlier detection mechanisms see [17]). Drifting values are another problem for Soft Sensors. It is not easy to distinguish between drifts in the measurements (sensors) and drifts related to gradual process changes. The last issue discussed here is the co-linearity of the data. This originates in the redundancy of some of the process measurements, e.g. temperature sensors located relatively close to each other may represent highly correlated features. The problem of co-linearity may be handled using the Principle Component Analysis [22] or Partial Least Squares method [1].

Based on the theoretical consideration of the issues of the process industry data, there is a novel Soft Sensor architecture proposed in this work. This architecture reflects the need for the adaptation and evolution of the models. One of the main aspects of the architecture is the definition of data processing paths. A path consists of one or more pre-processing steps (e.g. feature selection algorithm, outlier detection, normalisation, etc.) and one computational learning method (e.g. multiple linear regression, ANN, etc.). Another key aspect of the architecture is the capability to use meta-learning approaches to model building and to it closely related model combination approaches.

The rest of the paper is organized as follows. The next section is giving an introduction to the Soft Sensor types as well as discussing current Soft Sensor development methodology and finally providing a list of application areas together with examples of published Soft Sensors. Based on the identified issues a novel Soft Sensor architecture focusing on the adaptation aspects of the Soft Sensor is introduced in Section 3. Finally, a summary is given in Section 4.

2 Soft Sensors

2.1 First Principle and Data-Driven Soft Sensors

As mentioned before, at a very general level one can distinguish between two types of Soft Sensors, namely First Principle Models and Data-Driven Models.

The original purpose of the First Principles Models (FPM) is for the planning and development of the process plants, therefore the models describe the knowledge about the processes in form of often complex mathematical equations. These equations describe the chemical and physical principles underlying the process. A typical example is applying mass-preservation principles, exothermal equation, energy preservation, reaction kinetics in form of reaction rate equations, etc. The drawback of the first principle models is that the models are usually systems of differential equations which are very complex for real processes. These systems of equations are not easy to solve and the results to solve them often results in numerical or stability problems. For these reasons the models focus only on the most important parts of the process and try to simplify the model by making various assumptions and simplifications as far as possible. The first principle models are usually used for planning and designing of new processes and plants. They usually focus on the description of the steady-state of the process. But due to the constraints mentioned above the use of the first principle models in the on-line monitoring and predictive analysis is very difficult or infeasible.

The focus of this work is therefore put on the Data-Driven Models (DDM) which have emerged as very attractive modelling approaches enhancing the toolbox of diagnostic, prognostic and decision support methods available for plant operators and embedded in automated control systems. These models are based on the data which is being recorded by the PIMS (Process Information Management Systems) of the processes. Dependent on the applied methods, one can further distinguish different types of DDM. One stream of data-driven Soft Sensors uses methods originating from statistics, for example regression methods, principle component analysis [22] or partial least squares methods [1]. Another stream of Soft Sensors is based on methods developed in the field of soft computing (for an overview of soft computing methods see e.g. [21] or [37]). Apart from the above mentioned two groups of data-driven methods there is also a number of approaches which combine both types of methods in hybrid approaches (e.g. NN-PCA [10], which is PCA in combination with ANN).

2.2 Soft Sensors Development Methodology

This section describes some practical steps and issues of the state-of-the-art of Soft Sensor development. It is based on extensive discussions with experienced Soft Sensor developers from the process industry and as such reflects the current adopted practices. The whole process of Soft Sensor development may be split into several steps.

First data inspection: During this initial step, the first inspection of the data is performed. The aim of this step is to gain an initial view of the data structure and identify any obvious problems which may be handled at this initial stage (e.g. locked variables having constant value). The next aim of this stage is to assess the likely requirements for the model complexity to be used. It is quite common that an experienced Soft Sensor developer can, already at this stage, make a reasonable decision whether to use a simple regression model, a rather more sophisticated but more powerful PCA regression model or a non-linear neural network to solve the problem. In some cases, the model family decision at this stage may not be correct, therefore the models and their performance should be always evaluated and compared to alternative models. It is also at this stage that the rough assessment of the number of outliers and missing data is carried out. This observation may then influence the selection of the strategy for handling such problems as previously mentioned.

Identification of stationary states: Here, the stationary parts of the data have to be selected. The further modelling will deal only with the stationary states of the process. This non-stationarity of the process occurs usually during the start-up or shut-down of the plant only and therefore usually does not need to be modelled. The identification of the stationary process states is not a trivial task and is usually performed by manual annotation of the data though it should be noted that there are some automated approaches for the selection of stationary process states.

Data pre-processing: The aim of this step is to modify the data in such a way, that it can be more effectively processed by the actual data-driven model. An example of a typical pre-processing step is the standardisation of the data to the mean value 0 and unit variance. In the case of the data which are produced in the process industry there are more pre-processing steps necessary. The usually involved steps are the handling of missing data, outliers detection and replacement, selection of relevant variables (i.e. feature selection), handling of drifting data and detection of delays between the particular variables. Most of the listed issues are at the moment handled manually or need at least a supervised inspection of the results. The data pre-processing is usually done in an iterative way, i.e. after the standardisation and missing values treatment which are usually performed only once, an outlier removal and feature selection are repeatedly applied until the model developer considers the data as being ready to be used by the actual model. At the moment the pre-processing of the data is the step which affords the most manual work and expert knowledge incorporation of the whole soft sensor modelling. The need for the discussions with plant operators and process experts is also stressed in the literature, e.g. [13].

Model selection: This is the next step toward the Soft Sensor. In this step the model type (if not done until now) and the parameters of the model have to be selected. Usually cross-validation or related approaches together with a significance test are used to accomplish this task. The usual approach is to start with a basic, i.e. low complexity, model and add complexity as long as the performance of the model improves significantly. The employed rule of thumb is to use a linear regression model for problems with less than 5 variables, for higher dimensional problems usually PCA regression is applied and for non-linear problems often Multi-Layer Perceptron neural network or other technique from the class of universal approximators are used. For the performance evaluation of the models after the learning stage it is crucial to use validation data which have not been seen by the model during the learning phase.

Soft Sensor maintenance: After developing and deploying the Soft Sensor, it has to be maintained on a regular basis. The maintenance is necessary due to the changing environment which causes the performance of the Soft Sensor to deteriorate and has to be compensated for by adapting or re-developing the model.

The typical flow of the Soft Sensor development methodology is summarised in Figure 1. The figure is partially motivated by the approach presented in [13], where a similar methodology for Soft Sensor development was presented. It has to be stressed that the above conforms to a number of common practice approaches for predictive model building which have been used in the computational intelligence community for a number of years and has now been firmly established in industrial and business contexts as highlighted above through the summary of the practices in the process industry.

2.3 Application Areas of Soft Sensors

The application of Soft Sensor can be found across different fields of process industry. Most of the applications can be found in the particular parts of process industry where the traditional modelling techniques like FPM fail to deliver the required precision. In chemical industry, there is a large number of processes where the final product quality may not be estimated using automated approaches. The traditional approach in this case is to evaluate the product quality by manual or semi-automated lab measurements.

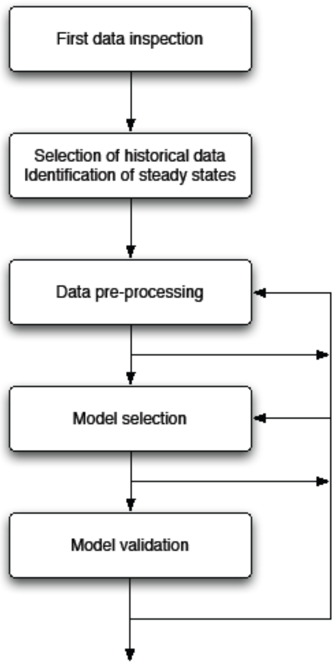

Figure 1: Methodology for Soft Sensor development

This represents a prominent application field for Soft Sensors, which are in more detail described later on in this section. Fermentation processes are another target group for Soft Sensors. Models of fermentation processes are very difficult to build because these processes are not easy to control and may differ significantly from one batch to the other even under constant condition. Similar problems occur in the polymerisation processes, these are very hard to control because they depend on a lot of external factors, which are out of the sphere of the traditional techniques. Therefore similarly to the previous case as well as in the paper process industry and various other fields of the process industry, Soft Sensors have established themselves as the method of choice.

Continuous Data Stream Prediction: The most common application of Soft Sensors is the prediction of values, which cannot be measured or else estimated on-line. This may be for technological reasons (e.g. there is no equipment available for the required measurement), economical reasons (e.g. the necessary equipment is too expensive), etc. From the computational learning point of view these problems are equivalent to supervised regression. Usually, there is historical data available. This data consists of the past plant measurements (e.g. temperatures, pressures, etc.) which form the input data space of the Soft Sensor. The typical modelling approach used for these problems is the application of artificial neural networks (ANN). ANN have found broad application in the computational learning, and thus also in the Soft Sensor modelling, after the invention of the back-propagation algorithm. They attracted the attention of scientists due to their generalisation power and abilities to solve non-linear problems. Apart from the previous fact, which probably led to the wide application areas of ANN, there are also some issues related to their application. Typically, the learnt knowledge is stored globally in the connection weights of the ANN which make it difficult to extract the knowledge in human readable form from the model. Besides this fact there is also a problem with the determination of the topology of the network. Usually it is selected using cross-validatory or heuristic approaches.

An application of ANN for the sugar quality estimation was published in [9]. The approached problem in this publication is the modelling of the massecuite electrical conductivity because this is an important value for the control loop controlling the sugar production process. The input features of the model were selected manually and limited to eight from the modeller perspective important process measurements. The results achieved by the ANN were good enough for taking the Soft Sensor into operation. In [23] artificial neural networks are compared to First Principle Models (FPM) and extended Kalman Filter (EKF) [43], which are other common approaches to Soft Sensor building. The disadvantages of FPM and eKF are the complexity of the development and amount of a-priori knowledge which has to be available for the model development. The before mentioned publication gives an overview of some ANN applications of bioprocess (i.e. fermentation) Soft Sensors. Thorough analysis of the application of ANN for Soft Sensor building has been presented in [35]. This work discusses a lot of practical issues of the application of neural networks for Soft Sensor modelling. A particular focus is set on the necessary pre-processing steps like, the handling of missing values, outliers, etc. Focusing the identified issues, there is also a modification of the error measure of the back-propagation algorithm (using of Manhattan distance instead of mean squared error) proposed.

In the last few years a very popular research topic in computational learning field is the hybridisation of different modelling approaches. An overview together with the definition of different hybridisation levels was presented in [2]. The advantage of successful hybrid systems is their ability to overcome the drawbacks of the particular component methods and make use of their advantages at the same time. The probably most commonly used hybrid approach is the Neuro-Fuzzy System (NFS). It combines the Fuzzy Information System (FIS) [21] with ANN. Neurofuzzy models combine the learning capabilities of ANN with the human-like reasoning of FIS. In this way the disadvantages of ANN, namely the problems with the interpretability of the learnt knowledge, and at the same time the issues of FIS, namely the missing of a straight-forward learning algorithm, are compensated. NFS inherits the connectionist structure from ANN.

The first example of Soft Sensors based on neuro-fuzzy approaches is an ANFIS-based [21] Soft Sensor being applied to rubber viscosity prediction [34]. Another ANFIS-based Soft Sensor was presented in [42], in this work the data is pre-processed using PCA transformation. The paper defines a methodology for the development of Soft Sensors using soft computing methods. The methodology is applied to prediction of polymeric-coated substrate anchorage. Neuro-fuzzy Soft Sensor based on rough set theory and optimized by a genetic algorithm is discussed in [32]. The application shown in this work is the prediction of freezing point of the light diesel fuel in a fluid catalytic cracking unit. In the framework of neuro-fuzzy techniques, there has been few publications dealing with the adaptivity and evolving capabilities of the neuro-fuzzy models. Examples of such models are the extended evolving fuzzy Takagi-Sugeno models exTS [4] and Dynamic Evolving Neural-Fuzzy Inference System DENFIS [25] or General Fuzy Min-Max GFMM model [15]. The advantage of these models is that they can, by modifying their knowledge, evolve together with the changing environment.

In [33] an exTS model has been applied to the prediction of the quality of crude oil distillation. The advantages of the proposed approach are numerous, one of them is the already discussed ability of the model to evolve together with the changing data by adapting and deploying new hidden units in the neuro-fuzzy network, another advantage common to all fuzzy based models is the interpretability of the learnt knowledge, which is represented in the form of human-readable fuzzy rules.

Apart from the combination of ANN and FIS, there is a large number of other models which are combination of two or more computational learning techniques. The work of Qin et al. [35] has been already mentioned. On of the contributions of this work is the definition of Neural-Network Partial Least Squares (NN-PLS) algorithm which is a hybrid system combining the PLS algorithm with ANN. This algorithm makes use of the capabilities of ANN to map the input variables onto the latent variables of the PLS. The discussed hybrid algorithm is also applied to a real-life problem, namely to a refinery process. Another application of NN-PLS to soft sensing was presented in [10] where the NN-PLS (and NN-PCA) algorithm is applied to the prediction of emissions of NO_x gas in exhaust streams. A hybrid system consisting of Particle Swarm Optimisation (see [7]) which is used for the training of an ANN was presented in [30]. In this work the PSO algorithm is combined with the Alopex algorithm [39] to avoid local minima to which the PSO is prone. The proposed algorithm is applied to the ethylene distillation column data set. Another hybrid approach to Soft Sensor modelling has been developed by Kordon et al. [27]. In this case, the hybridisation is done on a lower level. The involved methods perform pre-processing of the data for the succeeding modelling steps. The methodology for the inferential sensor building consists of three different steps. The first step is the analysis of the data by an analytical neural network [28]. The aim of this step is to perform feature selection on the input data to deal with time delays between the selected features. In the next step the data is processed using SVM [40]. During this step the outlier detection is done. In the third step the actual Soft Sensor is built. This is performed by applying Genetic Programming (GP) algorithm [29]. The GP algorithm selects a function from a pool of available functions and trains it to model the output variable using the pre-processed input data. The Soft Sensor is a set of analytical functions which maps the input space to the target variable space. The proposed approach was applied to several real-life problems, e.g. the interface level estimation in an organic process [24].

Process Monitoring: Another application area of Soft Sensors is process monitoring and related to it process and sensor fault detection. Process monitoring is usually an unsupervised learning task. The systems can be either trained to describe/ analyse the normal operating state or to recognize possible process faults. Commonly, process monitoring techniques are based on multivariate statistical techniques like PCA, more precisely on Hotelling's T_2 [18] and Q-statistics [19]. These measures have on one hand the advantage of involving all input features into consideration, whether the process remains within the acceptable limits, and on the other hand providing information about the contribution of the particular feature to a possible violation of the monitoring statistics [8]. The PCA algorithm [22] reduces the number of variables by building linear combinations of the input variables in such a way that these combinations cover the highest variance in the input space and are additionally orthogonal to each other. In case of the process industry data, there is a very useful feature because it is very often the case that there is a co-linear variable present in the data. Although the PCA is a powerful and very often applied algorithm it has several drawbacks. Probably the most important issue is the selection of optimal number of principal components.

This can be solved using cross validation techniques. Another problem is that the principal components describe very well the input space but do not say anything about the relation between the input data space and the output space which has to be modelled. A solution of the previous problem is the Partial Least Squares (PLS) algorithm [1]. This algorithm, instead of focusing on the covering of the input space variance, pays attention to the covariance matrix between the input space and the output space. The algorithm decomposes both spaces simultaneously with the constraint of explaining as much of the covariance between the input and output space as possible.

Li et al. is dealing with the application aspects of the PCA and related methods to the process industry problems in [31]. The focus is put on the development of a recursive PCA approach targeting adaptive process monitoring. Within this framework it has also been shown that the method can deal with outliers, missing values and delayed measurements. They presented an effective approach for the update of the correlation matrices and two algorithms for the update of the PCA base using the old PCA structure. Additionally a review of the most common techniques for the selection of the number of principle components, which is one of the drawbacks of the PCA, is presented and a new technique for recursive selection of the number of principle components is shown. For the purpose of adaptive process monitoring, it is necessary to update the confidence limits of the model with the new incoming data, therefore the authors define also a monitoring scheme, which detects and handles data outliers, missing values and process faults before updating the model. Finally the proposed monitoring scheme is applied to a rapid thermal annealing process monitoring.

Self-organizing maps (SOMS) [26] is artificial neural network type which is able to deal with unsupervised problems and can therefore be applied to process monitoring tasks. Provided a set of high-dimensional input samples it maps this features to a lower dimensional, usually 2-dimensional, space. The mapping is done with the constraint of keeping the topological properties of the data. The properties of the SOM make it useful for both the visualisation of multivariate data and for clustering. In terms of Soft Sensor modelling SOMs may be applied for process monitoring purposes. Process monitoring means the extraction of meaningful process states from the input data. A set of practical applications of process monitoring and quality prediction, etc, using SOMs was published in [3]. In this work SOMs have be found as useful tool for the monitoring of a continuous pulp digester. Before feeding the data into the SOM model they have been manually pre-processed using a-priori knowledge of the process. Another application presented in the work is the quality prediction of steel prediction based on the concentration of the input elements and some process parameters. The last application of SOMs presented in the work is the analysis of the data from paper and pulp industry.

Process and Sensor Fault Detection: It was already mentioned in the introduction section that process industry plants are usually equipped with a large number of various measuring and monitoring devices. Larger plants can house up to several hundreds of sensors. In such an environment, it happens quite often that either individual sensors or group of them fail and do not deliver any meaningful data. These data samples represent data outliers or missing values. As a vast majority of modelling techniques applied within the process industry as Soft Sensors are not able to handle this kind of data as a matter of their normal operation, there is a need to identify and replace sensor and process faults before the actual model building and model application.
Process and sensor faults are detected and handled using the PCA in [12]. The faults are detected in the PCA residual space. This has the advantage that one can on one hand identify the sensor or process faults effectively and on the other hand by projecting the fault state to the original space one can also find which particular sensors are responsible for the fault state. The proposed approach is again evaluated in form of case studies. In the case of this work it is the detection of faults in a boiler process.
In [36], a self-validating Soft Sensor is discussed. The input data is validated using a PCA-based approach for fault detection published in [11]. In the case of a detected failure, the sensor can be reconstructed using the correlation structure of the affected input measurement to the other input space variables, which is one of the valuable capabilities of the PCA. After this pre-processing step, which on one hand removes the co-linearity of the input data and on the other hand reconstructs fault sensor data, a Soft Sensor using traditional modelling techniques is built.

3 Adaptive Soft Sensor

To cope with the challenges listed in this work, especially those discussed in the introductory section, we propose a novel architecture for a self-adapting Soft Sensor. A broad overview of the architecture is shown in Figure 2.
A significant part of the proposed architecture are the two pools, namely the Pre-processing Methods Pool (PPMP), including all pre-processing methods, which is further split into actual pre-processing methods (e.g. filtering, normalisation), feature selection methods (e.g. correlation-based feature selection) and instance selection methods (receptive fields filtering). The second pool, Computational Learning Methods Pool (CLMP), consists of various computational learning methods (e.g. linear regression, multi-layer perceptron models, etc.). The particular instances of the methods in the pools are connected to paths by the Path/Pool Management module. Path may be for example built from the following elements: data standardisation, correlation-based feature selection and a multi-layer perceptron method (see Figure 3 for a path example).

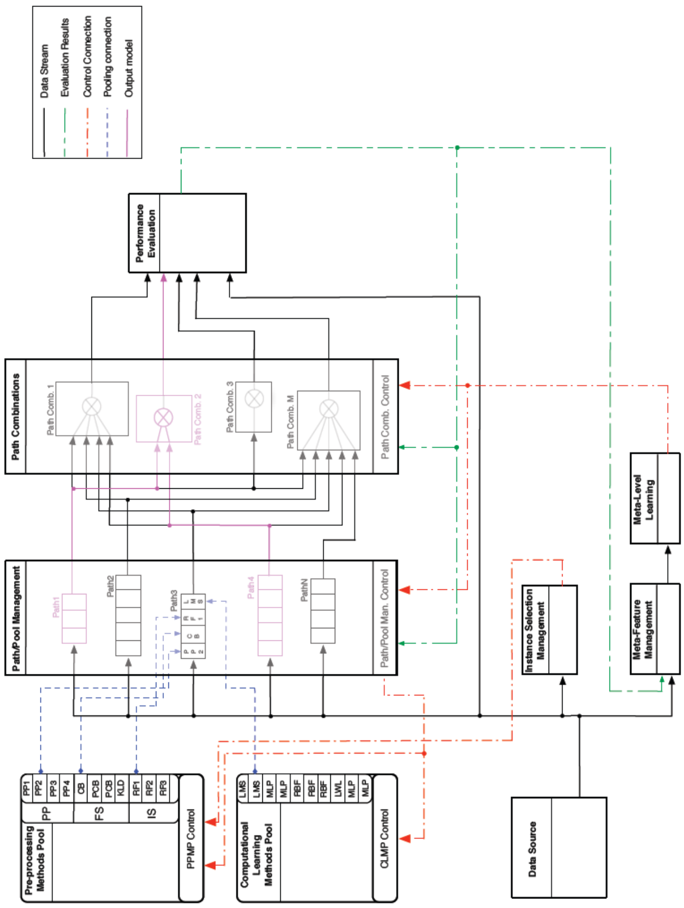

Figure 2: Proposed architecture of an self-adapting Soft Sensor

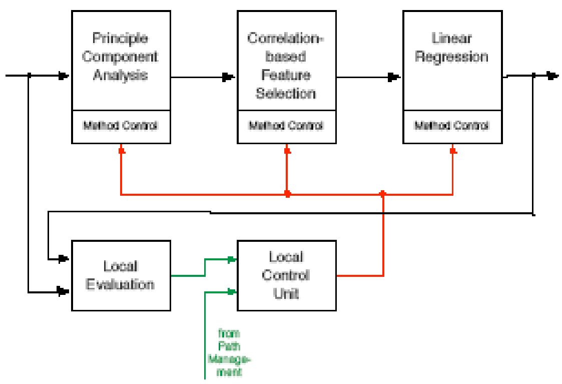

Figure 3: Transformation path example

At the path level, there is an additional mechanism for the local control necessary. Such a mechanism is shown in Figure 3. The presented path consists of three path elements. The first two elements are pre-processing steps, namely the PCA and feature selection, and the third one is a linear regression computational method. The prediction of the method is fed to the Local Evaluation, where together with the data input the evaluation of the path prediction is performed. The results of the prediction are passed to the Local Control Unit which based on the implemented control method controls the parameters of the path elements. Additional input to the Local Control Unit is the control information from the high-level decision making methods (i.e. Path/Pool management, Meta-Level Learning, etc.).

Another key aspect of the architecture is the Path Combination module. Typically the combination is carried out at the level of the computational learning methods (e.g. stacking of a group of MLP networks [13]). In the proposed architecture the combinations are performed at the path level which provides several advantages. One can do the combination while including different methods from PPMP (e.g. a combination of several paths consisting of MLP with different approaches to feature selection as pre-processing step). Another advantage is that it is possible to combine different methods from the CLMP, in this way it is possible to do combination across different types of computational learning methods (e.g. a combination of MLPs and RBF together with linear regression models is possible). The path combination module together with the instance selection methods from PPMP provides also the possibility to combine different local paths (local learning models) to a global path.

The architecture provides also the possibility of using meta-learning approaches [41], [16]. There are two modules in the architecture for this purpose. The first one, Meta-Feature Management, having information about the data together with the performance of the particular paths builds the meta-features. This module may e.g. extract the information about the performance of the different paths in the different parts of the input data space and pass this information further to the Meta-Level Learning module which can, using the provided information, control the Path/Pool Management and Path Combination modules. The Instance Selection Management module is responsible for the filtering of the instance and thus providing the possibility for building of local models, i.e. local experts, [5], [38], [14], [20]. The local approach to the model building is, apart from the pool and path concepts and meta-learning techniques, one of the key aspects of the proposed architecture.

3.1 Adaptivity of the architecture

One of the key functions of the proposed architecture for Soft Sensor development is its ability to adapt to the changing environment of the process industry data. A simplified diagram of the adaptation possibilities within the architecture are shown in Figure 4. There are three different levels of adaptation possible. The first one is at the local level. Local level means the level of the particular paths and their parameters in this context. In this case, the particular path are adapted using the knowledge about their performance (see feedback loop a) and some global parameters (see feedback loop d). The next level of adaptation is the level of the path combinations, here the particular combination are adapted in a similar way, like it is the case at the path level.

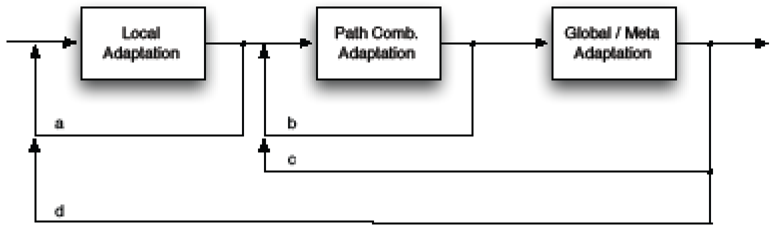

Figure 4: Adaptation loops within the architecture

The particular combinations are adapted independently to each other using their performance (feedback loop b) and again global/meta-level parameters (feedback loop c).

The last level is the global or meta-level adaptation. At this level decisions, which influence the dynamic behaviour of the whole architecture, are being made. The aims at this level may be more sophisticated than plain search for the best performance given an error measure. The goal of the meta-level learning may be for example trying to keep a large diversity of paths and not just to keep paths with the best performing type of method. Keeping a broad variety of methods in the pools may be of benefit if there is for example a sudden change in the process causing a change of the data which may further on cause a decrease of the performance of the best paths in the pool. Another aspect which may be handled at this level is the different adaptation speeds. There may be a set of paths focusing on short-term changes in the data and adapting to these changes. In contrast to that there may also be a set of paths or path combinations focusing on long-term performance and adapting rather slowly to the process changes.

Figure 5 shows the adaptation mechanisms in a more detailed way. It shows the details of two different paths together with their adaptation loops. The local adaptation consists of the feedback of the prediction which is compared to the correct target values in the Local Evaluation block. Given an error measure, there is the error between the prediction and the correct values measured within this block. The measured error is passed to the next part of the loop, namely to the Local Control Unit. Another input to this block is the information from the global level decision making parts of the architecture. This input may for example stimulate the speed of the adaptation or completely skip it. Another task of the Local Control Unit is the control of the parameters of the learning algorithm (e.g. the learning rate or momentum of the gradient descent learning approaches). The learning itself is a part of the particular method blocks and is indicated as Method Control in Figure 5.

The adaptation approach for the path combinations is similar to the one of the single paths. Again, the performance of the prediction of the combination, which is measured in the Path Combination Evaluation, is applied, together with the global level information, in the Path Combination Control Unit. This unit has access to the learning methods of the combination algorithm.

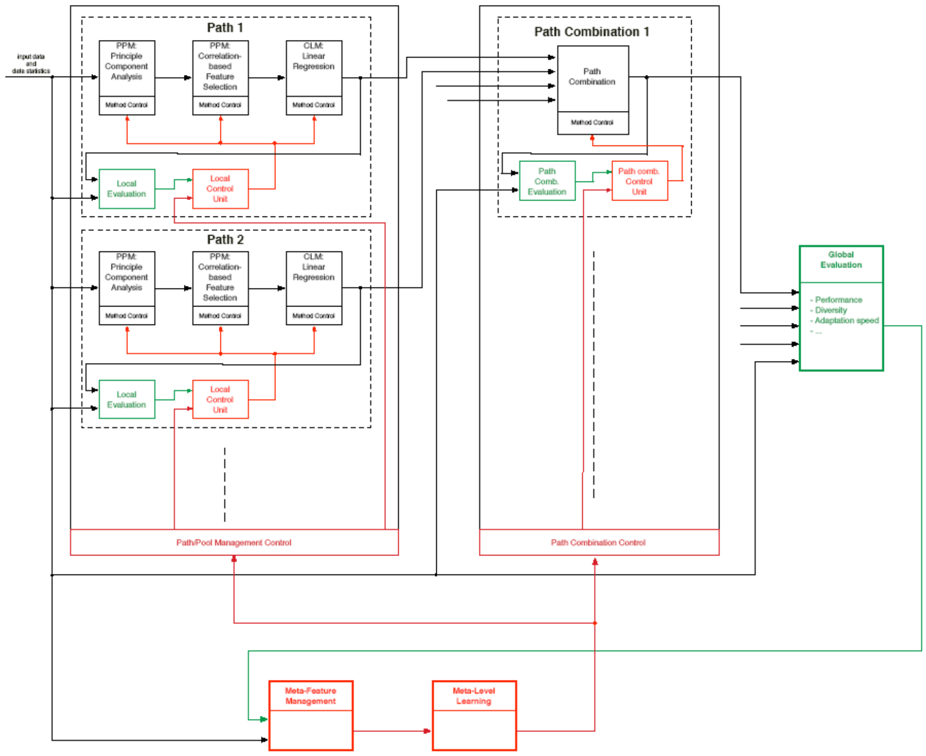

Figure 5: Adaptation mechansisms of the architecture in detail

The adaptation at the global level plays the most important role. It controls the global behaviour of the whole architecture. The decisions to deploy and remove path and path combinations are also done at this level. There are several parts of the architecture involved into this adaptation loop. Within the Global Evaluation part of the architecture, there are different criteria (i.e. evaluation functions) involved into the assessment of the path and path combination, such a criterion may be for example the diversity of the methods in pools. The results of the evaluation are passed to the Meta-Feature Management block. Here, together with some statistics of the data meta-level features are built. This kind of features may for example be the linearity of the data or the dynamics of the data, which may influence the need for the adaptation of the methods. It may also be a combination of certain characteristics of the data and of the performance of the path/path combinations. Based on the extracted meta-feature the Meta- Level Learning module takes appropriate actions and controls the Path/Pool Management and Path Combination Controls.

In a practical scenario the Path Combination Control could be a genetic algorithm controlling a set of individuals (i.e. path combinations). The parameters of the genetic algorithm (e.g. reproduction rate, number of individuals) would be in this scenario controlled by the Meta-Level Learning module, which makes the decision on the basis of the information delivered from Meta-feature Management.

4 Summary

Undoubtedly modern Soft Sensors have to be able to adapt to the changing environments as required by the process industry. Current practice of Soft Sensor development is too inefficient in terms of time spent on the development of the models as well as time needed for periodical re-tuning or re-training of the Soft Sensors. To make Soft Sensors a real alternative to the current process industry procedures, they should posses abilities like self-adaptibility, self-healing, etc. The proposed architecture is an attempt to move the Soft Sensor into the desired direction. The architecture provides the ability to manage a set of models. Within this model-pool there can be models from different modelling families, like regression models or neural networks, but also several instances of the same model type (e.g. several neural networks with different topologies) at the same time. In the same sense, there is a possibility to manage a set of model combinations. The goal of the management of the pools is to provide diversity of approaches to solve the given problem and to include a variety of different modelling philosophies with distinct strengths and weaknesses as well various approaches to adaptation into a single model. The hierarchical structure of the architecture allows the adaptation of the Soft Sensor at 3 different levels, namely at the level of the individual modelling methods, at the level of the model combinations and at global (meta-) level, where the behaviour of the whole Soft Sensor is controlled.

While there are many challenges remaining at each of these adaptation levels we believe that the added flexibility and robustness of the proposed architecture is necessary for the challenges already present in the process industry as well as a number of other application areas characterised by changing environments.

References

1. H. Abdi, (2003). Partial least squares (pls) regression. *Encyclopedia of Social Sciences, Research Methods*. Thousand Oaks (CA): Sage (2003).
2. A. Abraham, (2004). Intelligent systems: Architectures and perspectives. *Arxiv preprint cs.AI/0405009*.
3. E. S. A. Alhoniemi., (1999) Process monitoring and modeling using the self-organizing map. *Integrated Computer-Aided Engineering*, 6(1):3–14, 1999.
4. P. Angelov and Z. Xiaowei, (2006) Evolving fuzzy systems from data streams in real-time. *Evolving Fuzzy Systems, 2006 International Symposium on*, pages 29–35.
5. C. G. Atkeson, A. W. Moore, and S. Schaal, (1997) Locally weighted learning. *ArtificialIntelligence Review*, 11(1):11–73.
6. C. M. Bishop, (1995). *Neural Networks for Pattern Recognition.* Oxford University Press, USA.
7. E. Bonabeau, M. Dorigo, and G. Theraulaz, (1999). *Swarm Intelligence: From Natural to Artificial Systems.* Oxford University Press.
8. S. W. Choi, E. B. Martin, A. J. Morris, and I. B. Lee, (2006). Adaptive multivariate statistical process control for monitoring time-varying processes. *Industrial Engineering Chemistry Research*, 45:3108–3118.

9. D. Devogelaere, M. Rijckaert, O. G. Leon, and G. C. Lemus, (2002). Application of feedforward neural networks for soft sensors in the sugar industry. *In Neural Networks, 2002.* SBRN 2002. Proceedings. VII Brazilian Symposium on, pages 2–6.
10. D. Dong, T. J. McAvoy, and L. J. Chang, (1995). Emission monitoring using multivariate soft sensors. *In American Control Conference, 1995*. Proceedings of the, volume 1.
11. R. Dunia, J. Qin, T. F. Edgar, and T. J. McAvoy, (1996). Sensor fault identification and reconstruction using principal component analysis. *In Proceedings of the 13th Triennial World Congress*, pages 259–264, 1996.
12. R. Dunia and S. J. Qin, (1998). Joint diagnosis of process and sensor faults using principal component analysis. *Control Engineering Practice*, 6(4):457–469.
13. L. Fortuna, (2007). *Soft Sensors for Monitoring and Control of Industrial Processes.* Springer.
14. E. Frank, M. Hall, and B. Pfahringer, (2003). Locally weighted naive bayes. *In Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 249–256.
15. B.Gabrys, and A.Bargiela, (1999). Neural Networks Based Decision Support in Presence of Uncertainties. *Journal of Water Resources Planning and Management*, 125(5):272-280.
16. C. Giraud-Carrier, R. Vilalta, and P. Brazdil, (2004). Introduction to the special Issue on meta-learning. *Machine Learning*, 54(3):187–193.
17. V. Hodge and J. Austin, (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22(2):85–126.
18. H. Hotelling, (1931). The generalization of student’s ratio. *The Annals of Mathematical Statistics*, 2(3):360–378.
19. J. E. Jackson and G. S. Mudholkar, (1979). Control procedures for residuals associated with principal component analysis. *Technometrics*, 21(3):341–349.
20. R. Jacobs, (1991). Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87.
21. J. S. R. Jang, C. T. Sun, and E. Mizutani, (1997). Neuro-fuzzy and soft computing. Prentice Hall Upper Saddle River, NJ.
22. I. T. Jolliffe, (2002). *Principal Component Analysis.* Springer.
23. A. Jos de Assis and R. Maciel Filho, (2000). Soft sensors development for on-line bioreactor state estimation. *Computers and Chemical Engineering,* 24(2):1099–1103.
24. A. Kalos, A. Kordon, G. Smits, and S. Werkmeister, (2003). Hybrid model development methodology for industrial soft sensors. *In 2003 American Control Conference*, pages 5417–5422.
25. N. K. Kasabov and Q. Song, (2002). Denfis: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *Fuzzy Systems*, IEEE Transactions on, 10(2):144–154.
26. T. Kohonen, (1997). *Self-organizing maps.* Springer-Verlag New York, Inc. Secaucus, NJ, USA.
27. A. Kordon, G. Smits, E. Jordaan, E. Rightor, D. C. Co, and T. X. Freeport, (2002). Robust soft sensors based on integration of genetic programming, analytical neural networks, and support vector machines. *In Evolutionary Computation*, 2002. CEC’02. Proceedings of the 2002 Congress on, volume 1.

28. A. K. Kordon, (2004). Hybrid intelligent systems for industrial data analysis. *International Journal of Intelligent Systems*, 19(4):367–383.
29. J. R. Koza, (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Bradford Book, 1992.
30. S. J. Li, X. J. Zhang, and F. Qian, (2005). Soft sensing modeling via artificial neural network based on pso-alopex. *Machine Learning and Cybernetics*, 2005. Proceedings of 2005 International Conference on, 7.
31. W. Li, H. H. Yue, S. Valle-Cervantes, and S. J. Qin, (2000). Recursive pca for adaptive process monitoring. *Journal of Process Control*, 10(5):471–486.
32. J. X. Luo and H. H. Shao, (2006). Developing soft sensors using hybrid soft computing methodology: a neurofuzzy system based on rough set theory and genetic algorithms. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 10(1):54–60.
33. J. J. Macias, P. Angelov, and Z. Xiaowei, (2006). A method for predicting quality of the crude oil distillation. *Evolving Fuzzy Systems, 2006 International Symposium on* pages 214–220.
34. S. Merikoski, M. Laurikkala, and H. Koivisto, (2001). An adaptive neuro-fuzzy inference system as a soft sensor for viscosity in rubber mixing process. *Technical report, WSEAS NNA-FSFS-EC 2001*, Puerto de la Cruz, Tenerife, Spain, 11-15 February 2001, paper 446.
35. S. J. Qin, (1997). Neural networks for intelligent sensors and control—practical issues and some solutions. *Neural Systems for Control*, page 213234.
36. S. J. Qin, H. Yue, and R. Dunia, (1997). Self-validating inferential sensors with application to air emission monitoring. *Ind. Eng. Chem.* Res, 36:1675–1685.
37. S. J. Russel and P. Norvig, (2003). *Artificial intelligence.* Prentice-Hall, 2003.
38. S. Schaal and C. G. Atkeson, (1998). Constructive incremental learning from only local information. *Neural Computation*, 10(8):2047–2084.
39. E. Tzanakou, R. Michalak, and E. Harth, (1979). The alopex process: Visual receptive fields by response feedback. Biological Cybernetics, 35(3):161–174.
40. V. N. Vapnik, (1998). *Statistical learning theory*. Wiley New York, 1998.
41. R. Vilalta and Y. Drissi, (2002). A perspective view and survey of meta-learning. *Artificial Intelligence Review*, 18(2):77–95, 2002.
42. K. Warne, G. Prasad, S. Rezvani, and L. Maguire, (2004). Statistical and computational intelligence techniques for inferential model development: a comparative evaluation and a novel proposition for fusion. *Engineering Applications of Artificial Intelligence*,17(8):871–885.
43. G. Welch and G. Bishop, (2001). *An introduction to the kalman filter,* 2001.