OMFP: An Approach for Online Mass Flow Prediction in CFB Boilers

Indrė Žliobaitė[1], Jorn Bakker[1], and Mykola Pechenizkiy[1,2]

[1] Department of Computer Science, TU Eindhoven
P.O. Box 513, NL-5600 MB, Eindhoven, The Netherlands
{i.zliobaite,j.bakker,m.pechenizkiy}@tue.nl
[2] Dept. of MIT, U. Jyväskylä, P.O. Box 35, FIN-40014 Finland

Abstract. Fuel feeding and inhomogeneity of fuel typically cause process fluctuations in the circulating fluidized bed (CFB) boilers. If control systems fail to compensate the fluctuations, the whole plant will suffer from fluctuations that are reinforced by the closed-loop controls. Accurate estimates of fuel consumption among other factors are needed for control systems operation. In this paper we address a problem of online mass flow prediction. Particularly, we consider the problems of (1) constructing *the ground truth*, (2) handling noise and abrupt concept drift, and (3) learning an accurate predictor. Last but not least we emphasize the importance of having the domain knowledge concerning the considered case. We demonstrate the performance of OMPF using real data sets collected from the experimental CFB boiler.

1   Introduction

Online estimation of fuel consumption in mechanical devices is a challenging task due to noise, presence of outliers and non-stationarity of the signal. Mechanical devices typically are comprised of moving parts. The movements cause interference in the observed sensor data. The challenge is to filter out the *true* signal from the measured noise. In this study we develop a generic approach for online prediction of the true signal values from the sensor measurements under concept drift assumption. In particular, we address online mass flow estimation problem for a circulating fluidized bed (CFB) boiler.

Different amounts of fuel can be added to the boiler at irregular time intervals resulting in sudden drifts in a signal. Since the fuel is added mechanically (feeding), the start and the end time of this process is not necessarily (as in our case) available from the sensors as a direct measurement. Hence, in order to estimate accurately the amount of fuel in the container at each moment in time the algorithms should be able explicitly or implicitly handle these changes.

There is a lot of work on change detection and outlier detection, see e.g. a recent review [3]. However, the boiler problem exposes specific combination of change points and outliers at which existing change detection methods may fail. Statistical change detection methods, which are based on comparing pieces of raw data (e.g. [2]) do not take signal trends into account, which contain significant

part of discriminatory information in the boiler problem. The noise and outliers are not normally distributed making it hard to use statistical methods that assume a particular distribution of the data [1]. Learner based change detection (e.g. [5]) is not directly suitable for this problem due to the nature of the signal: noise, trends and specific outliers. Burning and feeding stages, which are observed in the fuel mass signal, are very different in nature and timing.

We design an online signal prediction method, which takes into account the properties of mass flow signal (noise, trends, specific outliers, switch between operational stages). The method is equipped with a tailored change detection, which is needed to drop out the old signal from the training sample of the predictor. In this study we take a data mining approach, we use no additional input data from the boiler except the noisy signal itself.

For evaluation of the performance of signal estimators labeled data is needed. There is no hard evaluation method for the actual amount of fuel present. It could be generated by the domain experts. It is difficult to extract the actual signal, since the data includes the effects of external influencers. In our approach we use an offline best fit method as internal validation for the estimators.

The rest of the paper is organized as follows. In Section 2 we overview the problem of a mass flow prediction in CFB boiler. In Section 3 we present our solution for online mass flow prediction. In Section 4 the experimental evaluation is presented and the results are discussed. We conclude and point out open problems in Section 5.

2   Problem Description and Related Work

To better understand and control the operation of CFB boiler it is important to know how much fuel mass is in the furnaces. Direct measurement is hardly possible in practice from the technological perspective. Therefore, this is done by estimating mass flow in the system that is equivalent to predicting the amount of fuel in the fuel feeding system at each point in time.

We start by briefly explaining how the input signal is generated, discuss the properties of the data and available solutions.

2.1   The Input Signal

The automatically available mass signal is a noisy estimate of fuel mass at each operation time point. The mass of the fuel inside the container is measured by a scale, sampled with a sample rate of 1 Hz.

The boiler is fed with fuel from the fuel container ('bunker') as depicted in Figure 1. The fuel inside the container is mixed using a mixing screw. There is a feeding screw at the outlet of the container, which transfers the fuel from the container to the boiler. During the burning stage the mass of fuel inside the container decreases (reflected by a decreasing amount of fuel in the data signal, as pointed by arrow (1)). As new fuel is added to the container (the burning process continues), the fuel feeding stage starts that is reflected by a rapid mass increase (arrow (2)).
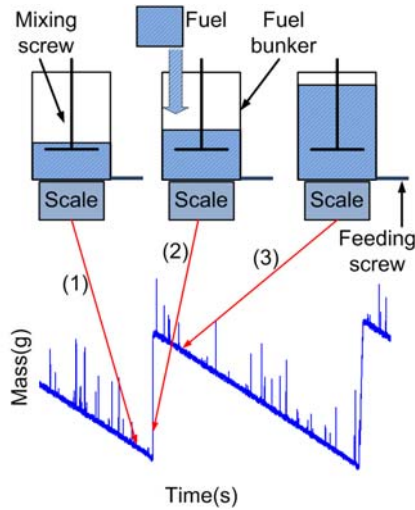
Fig. 1. The origin of the input signal

There are three main sources of changes in the signal.

First, fuel feeding is a manual and non standardized process, which is not necessarily smooth, it can have short interruptions (see Figure 2). Each operator can have different habits. Besides, the feeding speed depends on the type of fuel.

Second, the feeding screw rotation adds noise to the measured signal. Besides, fuel particle jamming often happens, slowing down the screw for some seconds and distorting the signal estimate. Therefore, the reported mass inside the container is not accurate, the signal contains extreme upward outliers in the original signal, that can be seen in Figure 3.

Third, there is a low amplitude rather periodic noise, which is caused by the mechanical rotation of the system parts. These amplitudes may become higher depending on the burning setup.

2.2   Data Properties

Due to the processes described above, the fuel mass signal has the following characteristics:

1. There are two types of change points: an abrupt change from burning to feeding and slower but still abrupt change from feeding to burning.
2. There are asymmetric outliers, oriented upwards. In online settings the outliers can be easily mixed with the changes from burning to feeding.
3. There is a symmetric high frequency signal noise.

Non stationarity of a signal can be regarded as a form of *concept drift* [15,8]. We focus on analyzing abrupt changes of a signal, which are caused by interchange of the boiler operation stages (burning and feeding).
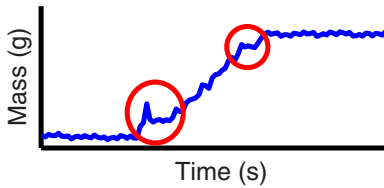
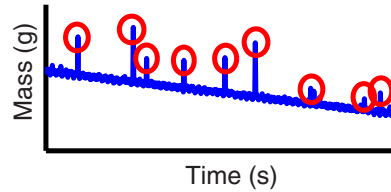**Fig. 2.** An example of a short burning within the feeding stage

Fig. 3. Upward outliers due to jamming of the screw

Algorithmic change detection is not trivial as it might seem from the visual inspection of the signal. The signal would be elevated if approximated directly due to the asymmetric nature of the outliers (no opposite negative outliers).

Besides, there are short burning periods within the feeding stages, due to possible pauses in a feed, which depend on human operator behavior. These interruptions can vary from 5 to 20 seconds are difficult to identify.

We assume that the mass flow signal has a nonzero second derivative. It implies that the speed of the mass change depends on the amount of fuel in the container. The more fuel is in the container, the higher is the acceleration, thus the more fuel gets into the screw. The weight of the fuel at higher levels of the tank compresses the fuel in the lower levels and in the screw, the fuel density is increased. Besides, compression and thus the burning speed depends on the type and quality of the fuel.

Our ultimate goal is to learn an accurate online prediction of the signal given that we can (1) catch and handle the changing behavior due to a process change, and (2) ignore the noisy patterns generated by anomalous behavior or the influence of moving parts.

2.3   Connection to the Related Work

Data mining approaches can be used to develop better understanding of the underlying processes in CFB boilers, or learning a model to optimize its efficiency [13]. Fundamental studies develop mathematical models for boiler operation [12,10,7,14], incorporating operational parameters in the models.

In this study we take a data driven approach for modelling the signal for online operation using only the historical data. A straightforward approach to non stationary time series prediction would be SARIMA model [4]. Seasonal periodicity is expected there, but in case of boiler mass flow prediction fuel feeding periods are not regular. The patterns might differ in every feeding round as well as the periods between two feedings. Our preliminary experiments confirmed that SARIMA indeed did not give satisfactory results.

In our previous work [1], performance of several change detection methods was compared in terms of detection accuracy and lag. In this study we develop a tailor made online method for the signal prediction and do a thorough quantitative evaluation.

Our present study differs from the previous works in the following way.

1. A change detection method tailored for trendy noisy drifting time series is proposed.
2. Experiments with two different fuel types are carried out.
3. Quantitative comparison of the alternative prediction methods is performed.

3   Online Mass Flow Prediction

In this section we present our solution to online mass flow prediction. We start with setting up a general framework, followed by depicting the base model, change and outlier detection mechanisms.

3.1   General Framework

Let's have the original signal $\mathbf{X} = (x_1, x_2, \ldots, x_t, \ldots, x_n)$. Having $\mathbf{X}$ as input we want to obtain the actual mass flow signal $\mathbf{Y}$ that can be achieved by learning a functional mapping of noisy sensor measurements to the true actual signal, so that $\mathbf{Y} = \mathcal{F}(\mathbf{X})$.

This problem has connection to the problem of *concept drift* [8,15] that refers to unforeseen changes over time in the phenomenon of interest. Once a change in the system stage happens (reasons are described in Section 2) the functional mapping $\mathcal{F}$ might become outdated. The learners capable of handling concept drift can be classified into proactive (explicitly detecting the change and dropping out the old training sample) or reactive (using forgetting heuristics at each time step to have the best adapted learner) [9]. The boiler data exhibits abrupt changes, thus we employ a proactive approach.

The intuition behind the model is the following: at each point in time t we fit a model $\mathcal{F}(x)$, using all or a subset of the historical data $\mathbf{X}$. If a change is detected, the old portion of the historical data is dropped out. A simplified estimation procedure is presented in Figure 4, the steps are explained in more detail in the following subsections.

3.2   Elimination of Outliers

The outliers are asymmetric, they do not have zero mean with respect to the signal. If not eliminated before fitting the model (step 1 in Figure 4), they can lead to significant distortion of the prediction, which as a result, will be elevated.

We know that the outliers are oriented upwards. For online detection of the outliers we check if the difference between the given point and moving average of the signal exceeds a threshold Tr_{out}. We replace the detected outliers with an average of the two nearest neighbors.

Note that in an online setting the nearest neighbors for calculating moving average are available only from the past, but not from the future, thus the detection accuracy is expected to be lower than it would be offline. It is obvious at the start of a feeding stage, when the distinction between the change and the outlier can be noted only after some time lag.
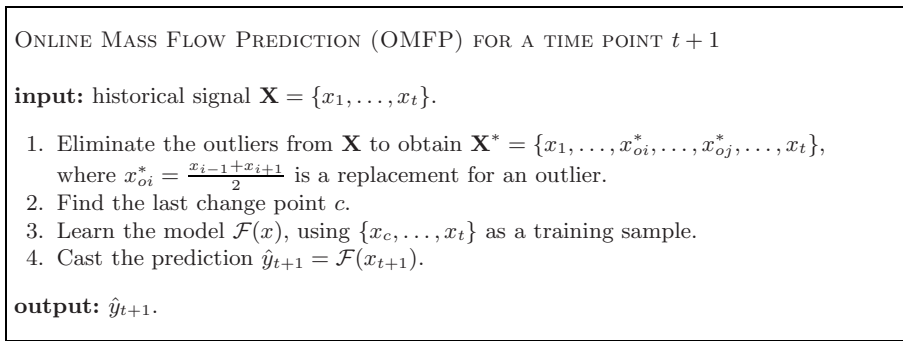

ONLINE MASS FLOW PREDICTION (OMFP) FOR A TIME POINT $t+1$

input: historical signal $\mathbf{X} = \{x_1, \ldots, x_t\}$.

1. Eliminate the outliers from $\mathbf{X}$ to obtain $\mathbf{X}^* = \{x_1, \ldots, x_{oi}^*, \ldots, x_{oj}^*, \ldots, x_t\}$, where $x_{oi}^* = \frac{x_{i-1}+x_{i+1}}{2}$ is a replacement for an outlier.
2. Find the last change point c.
3. Learn the model $\mathcal{F}(x)$, using $\{x_c, \ldots, x_t\}$ as a training sample.
4. Cast the prediction $\hat{y}_{t+1} = \mathcal{F}(x_{t+1})$.

output: $\hat{y}_{t+1}$.

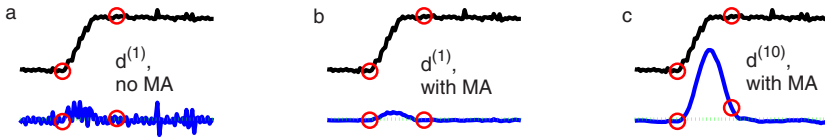**Fig. 4.** Online Mass Flow Prediction

Fig. 5. Change detection using L^{th} order signal differences $d^{(L)}$ and moving averages (MA). The upper (black) line represents the original signal and the lower (blue) is the differentiated signal. Dashed line (green) is the threshold for a change. Circles indicate *the ground truth* change.

3.3 Change Detection

The data exhibits trends, therefore change detection based on comparison of raw data subsets fails, when applied directly. We are interested in detecting the feeding stages, which are characterized by a steep increase in the signal value. An intuitive solution would be to take the first order differences of the signal $d_t^{(1)} = x_t - x_{t-1}$ and threshold these values. If $d_t^{(1)} > 0$ the system is at feeding stage, if $d_t^{(1)} < 0$ the system is at burning stage.

Unfortunately, due to signal noise, the stages are undistinguishable directly (see Figure 5a). We can try replacing the original signal with the moving average, before taking the first order differences, this already gives apparent feed regions, but that still is noisy (see Figure 5b).

We propose using L^{th} order differences $d_t^{(L)} = x_t - x_{t-L}$, applied on moving averaged signal for detection of stage changes. The more noisy the signal is, the larger lag is needed. In this case study we use $L = 10$ (see Figure 5c). Then we use a threshold Tr_{ch} to discriminate between feeding and burning stages. We use a high threshold $Tr_{ch} = 100$ to avoid false positives. The values were chosen based on preliminary experiments with the training set.

Tr_{ch} for changes is not to be mixed with Tr_{out} for outliers. The first is applied to a differentiated signal, while the second is applied to raw data.

Change detection might be equipped with a prior probability of switching the stages, based on the total amount of mass present in the container.

3.4   The Predictor

The functional mapping $\mathcal{F}$ (step 3 in Figure 4) is designed as follows.

In Section 2 we assumed that the mass flow signal has a nonzero second derivative. The true signal *in a single stage* can be modeled using the following equation:

$$y_t = \frac{a \cdot t^2}{2} + v_0 \cdot t + m_0 + A \cdot \sin(\omega_{feed} \cdot t + \alpha_{feed}) + B \cdot \sin(\omega_{mix} \cdot t + \alpha_{mix}) + e(t), \quad (1)$$

where y_t denotes the output of the scales at time t, a is acceleration of the mass change, v_0 stands for the speed of the mass change at time t_0, m_0 is the initial mass at time t_0; A and B, ω_{feed} and ω_{mix}, α_{feed} and α_{mix} are amplitude, frequency and phase of the fluctuations caused by feeding and mixing screws, respectively; $e(t)$ denotes the random peaked high amplitude noise caused by the jamming of the fuel particle at time t. We assume t_0 was the time of switch in the feeding/burining stages (change point X_c).

Since we are not interested in estimating the signal generated by the oscillations of the screw and the noise signal, we make a simplifying assumption that these parts can be treated as a signal noise. Thus we choose the following model:

$$\hat{y}_t = \frac{a \cdot t^2}{2} + v_0 \cdot t + m_0 + E(t), \quad (2)$$

where $E(t)$ is the aggregated noise component and the other terms are as in (1).

In our estimator we use a linear regression approach with respect to the second order polynomial given by (2). The model is inspired by the domain knowledge of the underlying process in the boiler, therefore seem more reasonable choice than alternative autoregressive models.

3.5   Learning the Predictor

To learn a regressor, the Vandermonde matrix [6] $\mathbf{V}$, which elements $v_{i,j}$ are the powers of independent variable x, can be used. In our case the independent variable is time $x_i = t_{i-1} - t_0$, $i = 1, \ldots, T$, where T denotes the number of the time steps. If the linear regression is applied for a polynomial of order n ($p^n(x) = p_n x^n + p_{n-1} x^{n-1} + \ldots + p_1 x + p_0$), $\mathbf{V}$ is computed from the observed time series of the independent variable as follows:

$$v_{i,j} = x_i^{n-j+1}, \ i = 1, \ldots, T, \ j = 1, \ldots, n+1, \quad (3)$$

where i and j run over all time samples and powers, respectively. Provided with $\mathbf{V}$ the problem of polynomial interpolation is solved by solving the system of linear equations $\mathbf{V}\mathbf{p} \cong \mathbf{y}$ with respect to $\mathbf{p}$ in the least square sense:

$$\hat{\mathbf{p}} = \text{argmin}_{\mathbf{p}} \sum_{i=1}^{T} (\sum_{j=1}^{n+1} V_{i,j} p_{n-j+1} - y_i)^2 \quad (4)$$

Here, $\mathbf{p} = [p_n \ p_{n-1} \ \dots \ p_1 \ p_0]^T$ denotes the vector of the coefficients of the polynomial, and $\mathbf{y} = [y(x_1) \ y(x_2) \ \dots y(x_T)]^T = [y_1 \ y_2 \ \dots y_T]^T$ is the time series of the dependent variable that is indication of the scales. Provided that the $n+1$ columns of the matrix $\mathbf{V}$ are linearly independent, this minimization problem has a unique solution given by solving the normal equation [11]:

$$(\mathbf{V}^T\mathbf{V})\widehat{\mathbf{p}} = \mathbf{V}^T\mathbf{y}. \tag{5}$$

This procedure is used to estimate the mass flow signal between change points. If the process switches from fuel feeding to fuel burning or the other way around, a new model is learnt on the new data.

3.6   Constructing *the Ground Truth*

The mass flow prediction is an unsupervised learning task in a way that the need for prediction arises from the fact that there is no method to measure *the ground truth*. However, to verify the validity of the model we still need a benchmark.

To obtain an approximation to *the ground truth* we use all the data set at once offline. We employ similar procedure as presented in Section 3.2. We identify the outliers by comparing the difference between the signal and the moving average against a threshold Tr_{out}. Then we take a moving average of the modified signal to obtain an approximation to *the ground truth*, which we associate as $\mathbf{Y}$.

Next we identify the change points from burning to feeding stage and vice versa (C_{feed} and C_{burn}). We employ different approach than in the online change detection. We use ADWIN method [2], which showed to be robust to false positives in semi-online settings [1]. We do not use it in online settings, because the lag needed to detect the change after it happened is too large.

Given a sequence of signals, ADWIN checks whether there are statistically significant differences between the means of each possible split of the sequence. If statistically significant difference is found, the oldest portion of the data backwards from the detected point is dropped and the splitting procedure is repeated recursively until there are no significant differences in any possible split of the sequence. More formally, suppose m_1 and m_2 are the means of the two subsequences as a result of a split. Then the criterion for a change detection is $|m_1 - m_2| > \epsilon_{cut}$, where

$$\epsilon_{cut} = \sqrt{\frac{1}{2m} \log \frac{4n}{\delta}}, \tag{6}$$

here m is the harmonic mean of the windows $m = \frac{1}{\frac{1}{n_1} + \frac{1}{n_2}}$, n is total size of the sequence, while n_1 and n_2 are sizes of the subsequences respectively. Note that $n = n_1 + n_2$. $\delta \in (0, 1)$ is a hyper-parameter of the model. In our experiments we used $\delta = 0.3$, $n = 200$ which were set during the preliminary experiments using the training data.

ADWIN identifies C_{feed} approximately. To get the exact change points we search for a maximum and minimum of the moving average in the neighborhood of the points identified by ADWIN. We validate *the estimated ground truth* by visual inspection of a domain expert.
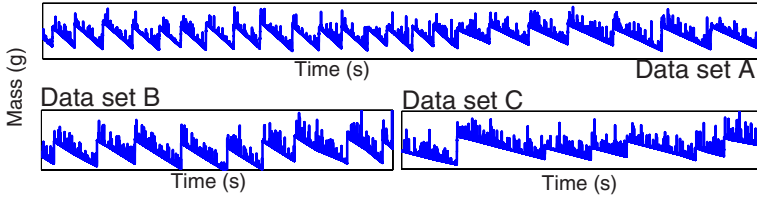
Fig. 6. The three complete data sets A, B, and C used in the experiments

4   Experimental Evaluation

4.1   Data sets

In this study we use three mass signal data sets A,B and C, which are plotted in Figure 6. The total length of A is different from B and C. A summary of the data sets is provided in Figure 7. Number of feeds means the number of feeding stages in the data set.

Data set A is used for training the model and selecting the model parameters. Data sets B and C are used as testing sets, the model trained on A with the same set of parameters is applied. Note that the level of noise and outliers in the data sets are different. B and C represent two fuel tanks, operating in parallel, therefore there are nearly twice as much of noise sources as in A.

Using training data set A we construct a representation of an average feeding stage pattern, which is depicted in Figure 8. This pattern is obtained by partitioning *the approximated ground truth* data into separate feeding sections. Then the partitions are matched by the change points from burning to feeding and averaged across.

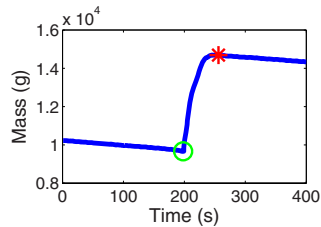| Name	Size	Number of feeds	Fuel
A	50 977	24	bio
B	25 197	9	bio
C	25 197	6	coal

Fig. 7. Data sets used

Fig. 8. An average feeding stage pattern

4.2   Experimental Setup

We conduct numerical experiments to test for prediction accuracy and for change detection accuracy. We chose moving average prediction as a 'naive' method to compare the performance.

In addition to next step $(t + 1)$ prediction experiments, we conduct a set of experiments allowing a delay D in predictions. For example, having $D = 5$ we would predict (filter) the signal x_t, but will have the historical data available up to time x_{t+4} inclusive. This gives a smoother moving average (nearest neighbors from both sides are available) as well as it allows more firm verification of outlier

Table 1. Mean average prediction accuracies. The best accuracies for each delay are **bold**; the best overall accuracy over a single experiment is underlined.

Delay	t+1	now	t-2	t-4	t-9	t+1	now	t-2	t-4	t-9	t+1	now	t-2	t-4	t-9
Data			A					B					C		
Overall performance															
OMFP	**34.1**	**29.4**	**27.8**	**_27.6_**	**29.0**	**23.8**	**20.9**	**16.6**	**_16.3_**	**31.4**	**12.9**	**13.0**	**10.3**	**_10.1_**	**16.3**
MA3	64.0	64.0	66.4			48.5	47.2	46.9			36.3	35.6	35.2		
MA5	63.1	51.9		39.9		49.7	45.3		41.7		35.9	33.9		32.5	
MA10	59.1	54.8			33.2	58.1	53.7			34.9	39.3	37.2			28.5
win50	53.0	45.0	44.4	44.4	44.4	40.3	34.3	32.0	32.0	32.0	19.7	16.7	15.2	15.2	**15.2**
all	1271	1269	1267	1265	1261	1313	1310	1308	1306	1301	1022	1021	1019	1019	1016
known	34.8	32.0	30.6	31.3	41.8	50.7	47.9	45.1	44.6	65.5	18.0	16.5	15.7	16.3	22.0
Feeding stages															
OMFP	463	325	229	231	321	1531	952	601	682	968	519	334	182	180	325
MA3	**308**	**181**	**115**			**733**	**510**	**434**			**260**	**163**	**119**		
MA5	438	294		**77**		1118	713		**359**		374	236		**105**	
MA10	781	640			**60**	2081	1714			**_171_**	714	578			**61**
win50	751	646	577	577	577	1867	1602	1248	1248	1248	860	735	645	645	645
all	1757	1753	1748	1744	1731	3259	3253	3248	3242	3225	2264	2259	2255	2250	2237
known	441	315	236	244	290	1493	924	561	594	752	464	306	249	269	296
Burning stages															
OMFP	30.0	28.7	28.5	**28.2**	28.1	22.1	19.7	**_16.9_**	17.2	34.9	10.7	11.4	**_10.5_**	11.3	19.1
MA3	60.8	62.4	65.8			48.1	46.9	46.8			35.6	35.0	34.8		
MA5	57.7	48.4		39.4		48.3	44.4		41.5		34.5	32.8		32.1	
MA10	48.3	45.9			32.8	54.6	50.6			**34.9**	36.0	34.3			28.3
win50	42.8	37.0	37.1	37.7	39.1	39.4	33.1	32.4	33.3	35.5	15.6	12.9	12.8	13.8	**16.3**
all	1264	1262	1261	1260	1257	1320	1317	1315	1314	1311	1015	1013	1013	1013	1013
known	**29.1**	**_28.0_**	28.1	29.3	40.7	50.6	47.7	45.7	46.0	69.1	16.2	15.0	15.5	16.9	25.1

and change detection. D is not to be mixed with L, which is a lag used by change detection method itself (Section 3.3).

We do the following verification: the stage (feeding or burning) is defined to be consistent if it lasts for not less than D time steps. Say at time t the system is at burning stage and at time $t + 1$ we detect the feeding stage. Having a delay $D = 5$ we are able to see the next four examples before casting the signal prediction for time $t + 1$. Thus we check if the feeding stage sustains at time $t + 2, \ldots, t + 5$. If positive, we fix the change point, if negative, we cancel the detected change and treat this as an outlier.

The domain experts suggested that maximum possible delay (D) in prediction could be 10 sec.

Once a change is detected, old portion of the data is dropped out of the training sample. We do not start using the 2nd order polynomial model until we pass 10 samples after the change. For the first 2 samples we use simple moving average rule: $x_{t+1} = x_t + s$, where s is a linear intercept term obtained using an average feeding stage pattern of the training data (A), which is presented in Figure 8. For burning stage $s_c = -2$ is used, for feeding stage $s_f = 81$. If from 2 to 10 historical data points are available after the change, we fit the 1st order

polynomial model, since the 2nd order approximation is too noisy with this few amount of points.

4.3 Prediction Accuracy

The mean absolute errors (MAE) with respect to our approximation to *the ground truth* (described in Section 3.6) are listed in Table 1. We present MAE for the whole data sets and then present MAE's for feeding and burning stages separately. Delay $t + 1$ means prediction of change one second ahead, 'now' means real time signal estimation and $t - 2$, $t - 4$ and $t - 9$ means estimation with respective delay of 2, 4 or 9 seconds.

For online prediction we set the following parameters. For outlier detection a moving average with a lag of 9 and a threshold $Tr_{out} = 400$ is used. For change detection a moving average with a lag of 8 and a threshold $Tr_{ch} = 100$ was used. the parameters were obtained from the preliminary experiments with the training data set A.

'MA3', 'MA5' and 'MA10' stand for simple prediction by moving averages, the number indicates how many instances are averaged. 'win50' uses the 2nd order prediction model presented in Section 3, but instead of change detection a simple

Table 2. Confusion matrixes of detecting changes to feeding (φ) and burning (κ) stages and outlier detection (o). P - positive, N - negative, T - true, F - false.

Training data set A									
	φ	P	N	κ	P	N	o	P	N
t+1	T	24	50946	T	12	50934	T	659	49784
	F	26	0	F	38	12	F	543	10
now	T	24	50946	T	12	50934	T	659	49784
	F	26	0	F	38	12	F	543	10
t-2	T	24	50967	T	10	50953	T	659	49783
	F	5	0	F	19	14	F	544	10
t-4	T	24	50969	T	10	50955	T	658	49783
	F	3	0	F	17	14	F	544	11
t-9	T	24	50972	T	8	50956	T	660	49782
	F	0	0	F	16	16	F	545	9

Testing data set B										Testing data set C									
	φ	P	N	κ	P	N	o	P	N	φ	P	N	κ	P	N	o	P	N
t+1	T	6	25162	T	2	25158	T	475	24597	T	6	25176	T	2	25172	T	362	24750
	F	26	3	F	30	7	F	104	21	F	15	0	F	19	4	F	75	10
now	T	6	25162	T	2	25158	T	475	24597	T	6	25176	T	2	25172	T	362	24750
	F	26	3	F	30	7	F	104	21	F	15	0	F	19	4	F	75	10
t-2	T	6	25165	T	2	25161	T	477	24597	T	6	25177	T	1	25172	T	364	24750
	F	23	3	F	27	7	F	104	19	F	14	0	F	19	5	F	75	8
t-4	T	6	25165	T	2	25161	T	477	24597	T	6	25177	T	1	25172	T	364	24750
	F	23	3	F	27	7	F	104	19	F	14	0	F	19	5	F	75	8
t-9	T	6	25183	T	2	25179	T	489	24594	T	6	25191	T	1	25186	T	372	24746
	F	5	3	F	9	7	F	107	7	F	0	0	F	5	5	F	79	0

moving window of the 50 last instances is used for the model training at each time step. 'all' uses the 2nd order prediction model with no change detection at all, it retrains the model at every time step. Finally we include a benchmark of the 2nd order model assuming known change points ('known'). We assume with this method that the change detection is 100% accurate.

MAE in 'overall performance' is rather close to MAE in 'burning stages' and very different from 'feeding stages'. This is because of uneven distribution of the stages in the data. 'Burning stages' comprise less than 2% of the data.

4.4   Change Detection Accuracy

We report the performance of the change detection in online settings in Table 2. For each method we present confusion matrixes of detecting sudden changes in feeding (φ) and burning (κ) stages and detecting of outliers (o). For φ and κ we allow 10 sec deviation. If a change is detected within the allowed region it is considered as identified correctly. We require the outlier detection to be precise.

We visualize change and outlier detection in Figure 9. The solid (blue) lines represent the true positives (TP) divided by the actual number of changes, the dashed (red) lines represent the number of false positives (FP) divided by the actual number of changes. The dotted black lines show the level of true changes (i. e. 24 change points for data set A, 9 for B, 6 for C).
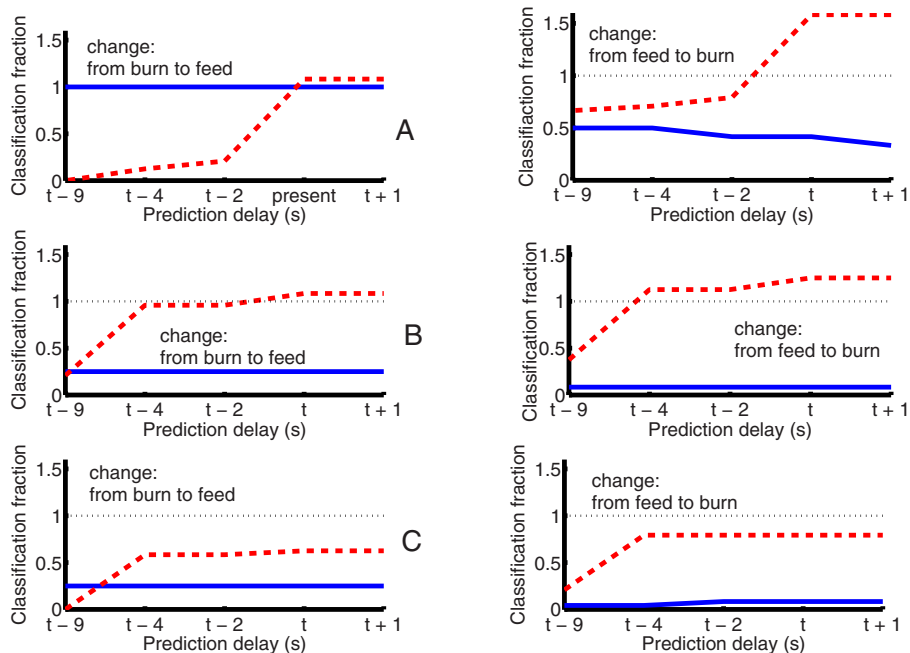

Fig. 9. Change detection accuracy as a function of the prediction delay for A (top row), B (middle row) and C (bottom row) data sets. Solid lines (blue) represent true positives, dashed lines (red) represent true negatives.

The number of FP is decreasing along with the increase in allowed prediction delay. A delay allows to inspect the following signal values after the detected change and if necessary cancel the alarm within the delay period.

The number of false negatives (FN) is relatively large. However, this does not mean that the changes from feed to burn were not detected at all. In this setting it means that they were not detected in time (within 10 sec interval).

4.5   Discussion

OMFP outperforms the competitive methods in terms of overall accuracy. However, for the feeding stage, simple moving average is the most accurate. Note that the approximation to *the ground truth* was constructed using moving averages, thus it could be expected that moving average performs well in this test setup.

OMFP method performance gets worse having a large delay in predictions. This is likely due to a fixed number of the nearest neighbors for moving average calculations, as we are using the same parameter settings for all the experiments.

Degradation of OMFP performance along with the increase in prediction delay also suggests, that there might be more accurate cutting points than just the change points themselves. Note that having a delay we allow canceling the detected changes.

In Figure 10 extraction from the prediction outputs is provided. In $t + 1$ prediction (a) the prediction follows previous points almost as a straight line. It is reasonable to expect, since in the fitted function 2nd order coefficient is mostly 0. Prediction $t - 2$ (b) is more curvy than $t - 9$ (c) likely due to more change points identified and therefore more cuts in history.
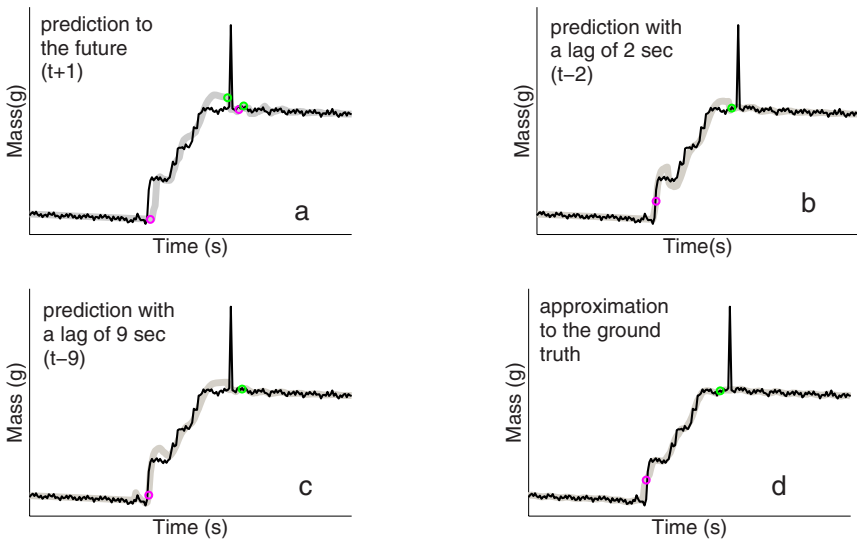

Fig. 10. The original signal (black) and OMFP estimation (grey)

Overall accuracy indicates that OMFP method is more favorable than moving average or alternative methods in the burning stage (see Table 1), while in the feeding stage optimal model selection is yet to be seeked. Separate handling of prediction in feeding and burning stages might be advantageous.

5   Conclusion

We developed and experimentally evaluated an online method for mass flow prediction during the boiler operation. We evaluated the performance of the method on three real data sets, including two distinct fuel types and two distinct operating stages (single vs multiple fuel).

One of the challenges in this task is coming up with approximation for constructing *the ground truth* for the signal, which we handle by a combination of moving average and responding for change and outlier points. We use this approximation to evaluate the performance of the online predictors.

Change detection is sufficiently accurate in transition from the burning to the feeding stage, where the incline in signal is rather sharp. However, the reverse detection still has room for improvement.

OMFP method clearly outperforms the competitive methods in terms of overall accuracy, while at the feeding stage simple moving average is a more accurate approach. The results suggest that separate handling of prediction in feeding and burning stages is needed.

The next steps of the research would be to employ the presented method in operational settings to see, what is the generalization on unseen cases. In addition, the effects of the rotation screw on the signal will be explored. Further, it would be interesting to come up with different models for different fuel types.

References

1. Bakker, J., Pechenizkiy, M., Zliobaite, I., Ivannikov, A., Karkkainen, T.: Handling outliers and concept drift in online mass flow prediction in cfb boilers. In: Proceedings of the 3rd Int. Workshop on Knowledge Discovery from Sensor Data (SensorKDD 2009), pp. 13–22 (2009)
2. Bifet, A., Gavalda, R.: Learning from time-changing data with adaptive windowing. In: Proc. of the 7th SIAM International Conference on Data Mining (SDM 2007), pp. 443–448 (2007)
3. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection - a survey. ACM Computing Surveys 41 (to appear, 2009)
4. Chatfield, C., Prothero, D.L.: Box-jenkins seasonal forecasting: Problems in a case-study. Journal of the Royal Statistical Society. Series A (General) 136(3), 295–336 (1973)

5. Gama, J., Castillo, G.: Learning with local drift detection. In: Li, X., Zaïane, O.R., Li, Z.-h. (eds.) ADMA 2006. LNCS (LNAI), vol. 4093, pp. 42–55. Springer, Heidelberg (2006)
6. Horn, R.A., Johnson, C.R.: Topics in matrix analysis. Cambridge University Press, Cambridge (1991)
7. Huilin, L., Guangboa, Z., Rushana, B., Yongjina, C., Gidaspowb, D.: A coal combustion model for circulating fluidized bed boilers. Fuel 79, 165–172 (2000)
8. Kubat, M., Widmer, G.: Adapting to drift in continuous domains. In: Proc. of the 8th European Conference on Machine Learning, pp. 307–310 (1995)
9. Kuncheva, L.I.: Classifier ensembles for detecting concept change in streaming data: Overview and perspectives. In: Proc. 2nd Workshop on Supervised and Unsupervised Ensemble Methods and their Applications (SUAMA ECAI 2008), pp. 5–10 (2008)
10. Kusiak, A., Burns, A.: Mining temporal data: A coal-fired boiler case study. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) KES 2005. Part III LNCS (LNAI), vol. 3683, pp. 953–958. Springer, Heidelberg (2005)
11. Lawson, Ch.L., Hanson, R.J.: Solving Least Squares Problems. Prentice-Hall, Englewood Cliffs (1974)
12. Park, C.K., Bas, P.: A model for prediction of transient response to the change of fuel feed rate to a circulating fluidized bed boiler furnace. Chemical Engineering Science 52, 3499–3509 (1997)
13. Pechenizkiy, M., Tourunen, A., Karkkainen, T., Ivannikov, A., Nevalainen, H.: Towards better understanding of circulating fluidized bed boilers: a data mining approach. In: Proc. ECML/PKDD Workshop on Practical Data Mining, pp. 80–83 (2006)
14. Saastamoinen, J.: Modelling of dynamics of combustion of biomass in fluidized beds. Thermal Science 8, 107 (2004)
15. Tsymbal, A.: The problem of concept drift: definitions and related work. Technical Report TCD-CS-2004-15, Department of Computer Science, Trinity College Dublin, Ireland (2004)