

Improving Blast Furnace Raceway Blockage Detection. Part 2: Signal Processing of Hot Blast Pressure Data

Stefan PUTTINGER^{1)*} and Hugo STOCKER²⁾

1) Department of Particulate Flow Modelling Johannes Kepler University, Altenbergerstrasse 69, Linz, 4040 Austria.

2) voestalpine Stahl Donawitz GmbH, Kerpelystrasse 199, Leoben, 8700 Austria.

(Received on July 26, 2018; accepted on October 15, 2018; J-STAGE Advance published date: December 19, 2018)

The present paper is part 2 of a three paper series discussing raceway blockages and various approaches for automated detection based on blast furnace (BF) plant data. Blockages of the raceway areas occur on a regular basis and in cases where the hot blast flow rate is reduced significantly, it is beneficial to shut down additional fuel systems like pulverized coal injection (PCI) on that specific tuyere. While part 1 gave an overview on the different appearances of raceway blockages and presented a common test-bench used for signal and image processing, part 2 focuses on the discussion of various signal processing algorithms and their applicability for raceway blockage detection. The algorithms are tested on data from a real BF with supplementary tuyere camera images to validate the results. Beyond this test dataset, a long-term test is performed by processing three month of BF data offline.

KEY WORDS: blast furnace; tuyere monitoring; raceway; blockage detection; signal processing.

1. Introduction

To achieve minimal coke rates in blast furnace operation it is crucial to obtain optimal burning of additional fuels like *e.g.* pulverized coal (PC). However, there are operating conditions where an optimal burning is not possible. The frequent case of a full or partial raceway blockage can lead to reduced hot blast throughput on the effected tuyere. If PC is injected in that tuyere the unburned coal might lead to locally reduced permeability of the burden.¹⁾ To avoid negative effects of high PCI rates, it is necessary to have reliable information of the current raceway condition. This increases also the requirements for the BF process control. A reliable raceway monitoring system must be able to detect blockages quickly to trigger the shutdown of PCI branches with short latency.

The different nature of raceway blockages (which shall also be denoted as events in the further discussion) has been discussed in part 1.²⁾ The frequency of occurrence may vary strongly between different furnaces and operating condition. Short blockages of only a few seconds duration may be observed several times per hour but are not relevant for BF operation. However, some blockages on BF1 at voestalpine Stahl Donawitz GmbH have been observed to last of more than 30 minutes in extreme cases. In these cases, the continuous injection of pulverized coal is not desirable and the PCI branch on the affected tuyere should be shut down to avoid accumulation of unburned coal particles.

In this second part of the paper we will address blockage

detection via hot blast flow rate data of the tuyeres. The flow rates can either be obtained via flow measurement devices like *e.g.* Venturi type flow meters or simply by measuring the pressure difference between the main hot blast bustle pipe and the tuyeres. **Figure 1** shows the testcase signal for which also the tuyere camera images are available (c.f. part 1). The signal contains some significant dips corresponding to reduced blast flow rates. Some of these dips are caused by the changing of the hot blast stoves and correlate with the signal of the absolute blast pressure. However, many other signal dips do not correlate with switching events of the stoves and it can be assumed that they are triggered by full or partial blockages of the tuyere.

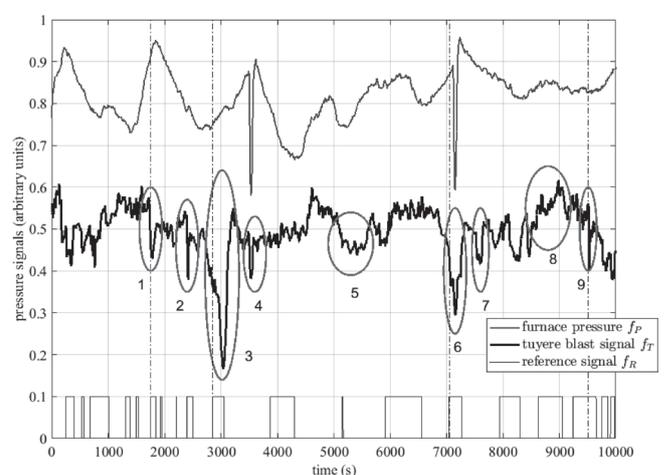


Fig. 1. Test-signal with some blockage events marked and the reference signal obtained by manually checking the tuyere camera images.

* Corresponding author: E-mail: stefan.puttinger@jku.at
DOI: <https://doi.org/10.2355/isijinternational.ISIJINT-2018-531>

A simple and common solution to shut down single branches of the PCI system is based on a constant threshold level which is compared with the hot blast flow rates on the individual tuyeres. As the average signal level can vary significantly between the tuyeres due to inhomogeneous permeability in different raceway areas this is an imperfect approach. Furthermore, the flow rate signals are prone to aging effects of the pressure sensors or signal drifts caused by e.g. electrical interferences. Hence, it is clear that using a constant threshold level is not an optimal solution.

In this paper we discuss various approaches for improved signal processing of the tuyere signals to obtain a more reliable blockage detection. Section 2 resumes the discussion of the test signal and the software testbench introduced in part 1. Section 3 presents the algorithms under test and their results and Sec. 4 discusses the findings of a long-run test.

2. Test Signals and Processing Environment

Examples of various blockage events were discussed in part 1 based on the hot blast signals and tuyere camera images of BF1 at voestalpine Stahl Donawitz GmbH. The dataset presented covers approximately 2 hours and 45 minutes of real blast furnace operation and will also be used for testing the algorithms in section 3. All signals obtained from the process control system have an update rate of $\Delta t = 2$ s and are therefore discrete signals $f(k)$ for timestamps $k = n\Delta t$. As a consequence, all algorithms in Sec. 3 are presented in their time discrete form.

Table 1 summarizes and classifies all intervals marked in Fig. 1. Major blockages are relevant for operation and must be detected (events #3 and #6). Events which are probably not relevant for BF operation are classified as minor blockages (events #1, #2 and #9). The signal dip of #4 is caused by the changing of the hot blast stove. During signal event

Table 1. Classification of blockage events marked in Fig. 1.

Event #	1	2	3	4	5	6	7	8	9
major blockage			X			X			
minor blockage	X	X							X
reduced permeability					X			X	
no blockage				X			X		

#7 no abnormal raceway behavior can be noticed in the tuyere images, thus not every signal dip does necessarily correlate with a blockage event. The signal intervals marked as #5 and #8 come along with a rather diffuse and darker raceway appearance but do not show complete blockages. It can be assumed that the surrounding zone of the raceway has a reduced permeability, which comes along with longer residence time of coal particles.

Although not all blockages marked in the reference signal are relevant for blast furnace operation, the event detection algorithms should be as sensitive as possible in the test phase. For a later online implementation of the blockage detection in the BF process control system, the sensitivities of the algorithms can be reduced as needed according to the frequency and significance of the events on a specific tuyere. However, only if most of the suspicious raceway conditions are detected correctly, one can build a sound statistics of blockage events which can later be used to find correlations with other data from blast furnace operation (e.g. burden charging, coke quality etc.).

Figure 2 recalls the structure of the software testbench for signal and image processing. The algorithms in section 3 are implemented as plugin functions and can easily be switched by a function pointer. For the details of the quality assessment by use of the manually defined reference signal f_R , the reader is referred to part 1.

Note that all analog and digital signals in the figures throughout this paper have been scaled and shifted adequately to keep the figures clearly arranged.

3. Discussion of Various Signal Processing Algorithms

In this section we will introduce the four different signal processing algorithms which have been used throughout this study. For easier referencing in the further discussion, the algorithms will be labelled 'A0' to 'A4'. The list of algorithms is, of course, not exhaustive and the selection was focused on widely used methods from other fields of signal processing. In the case of algorithms A1 to A3 the major goal was to obtain efficient algorithms which could easily be implemented in the existing process control system with a very low number of parameters that have to be adjusted. In the case of algorithm A4, which is based on

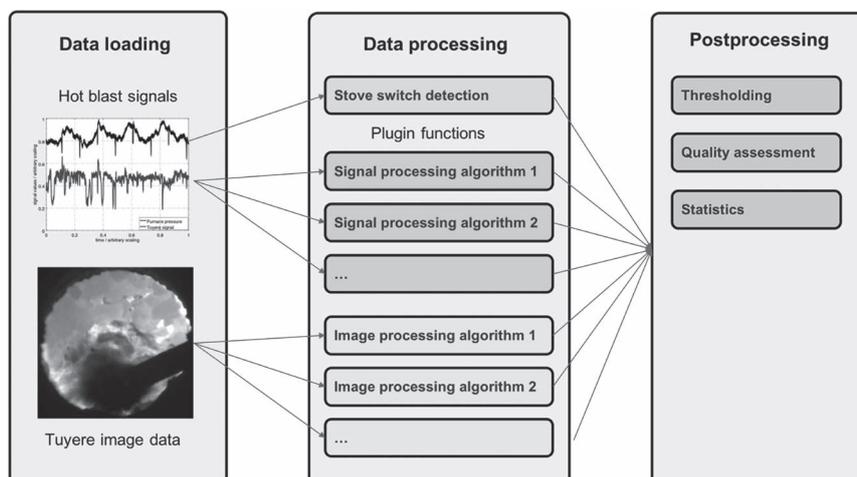


Fig. 2. Outline of the software testbench for the processing of blast signals and tuyere images.

wavelet transform, we wanted to test if a more sophisticated approach could perform significantly better than the rather simple algorithms A1 to A3 and therefore justify the much higher computational effort. In addition, a dummy algorithm A0 is implemented in the testbench which reproduces the simple thresholding of the hot blast signals as it is currently implemented in the process control system. For the test case of Fig. 1 only event #3 was recognized by this approach, thus no plot is given for the test-signal. However, it is interesting to compare the results of A0 with A1..A3 for the long-run test (c.f. section 4) and the long-term statistics of blockage events.

3.1. A1 - Signal Correlation

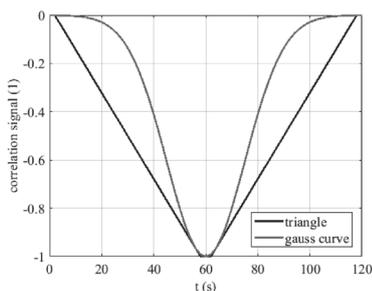
Most blockages produce a negative peak in the blast signal as shown in Fig. 1. Thus, one approach could be to compare the signals with a pattern signal and use the correlation result to detect blockages. To get rid of the varying absolute levels the first step is to calculate the time derivative of the pressure signal f'_T . The derivative is then correlated with the pattern signal f_M . This could be e.g. a negative ramp signal of a simple triangular shape or a bell-shaped Gauss curve as shown in Fig. 3(a). Before calculating the derivative of the time discrete signal f_T , the signal is filtered by a moving average according to Eq. (1) to remove some of the signal noise

$$\bar{f}_{T,k} = f_{T,k} \cdot w + (1-w) \cdot \bar{f}_{T,k-1}, \dots \dots \dots (1)$$

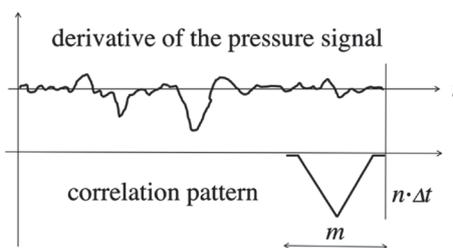
where w is a weight factor for new samples $f_{T,k}$. The backwards correlation of the signal derivative with the pattern signal f_M of length m can then be calculated according to

$$f_{S,k} = \sum_{n=k+1}^{k-m} [f'_T(n) - f'_T(n-1)] \cdot \frac{1}{\Delta t} \cdot f_M(k-n). \dots (2)$$

Figure 3(b) illustrates the backwards correlation for a certain timestamp $n\Delta t$. Equation (2) implies the need for a signal buffer which has the length of the pattern signal f_M which might be a disadvantage for the implementation in the process control system. In Fig. 4 the results for the test-signal are plotted. A triangle shaped ramp signal of 120 s length was used as the pattern signal. Ramp signals turned out to produce better results than gauss shaped pattern signals, probably because the changes in the tuyere signals have a more abrupt character rather than smooth changes. A length of 120 s is a good compromise to capture both, short and long events. The correlation result f_S in Fig. 4 follows quite accurately the signal derivative f'_T and thresholding with levels of 0.26 (on) and 0.40 (off) deliv-



a)



b)

Fig. 3. Principle idea of algorithm A1. a) two kinds of correlation patterns, b) sketch of short time backwards signal correlation for a pattern signal of length m .

ers the major blockage events in the test-signal. Especially the major events #3 and #6 are detected very early which is advantageous for the shutdown of PCI branches.

3.2. A2 - Signal Filtering

The filtering approach is based on the fact that slow changes in the permeability around the raceway are based on a different time scale than hot blast flow rate changes caused by blockages. Thus, filtering the signal by two different averaging window sizes can extract the rapid changes in the case of blockages from the long-term changes in the BF. The average filtering can be implemented in the same way as in Eq. (1) but using two different weight factors

$$\bar{f}_{T,S,k} = f_{T,k} \cdot w_s + (1-w_s) \cdot \bar{f}_{T,S,k-1}, \dots \dots \dots (3)$$

$$\bar{f}_{T,L,k} = f_{T,k} \cdot w_L + (1-w_L) \cdot \bar{f}_{T,L,k-1}. \dots \dots \dots (4)$$

The ratio of the short and long-term averaging

$$f_{S,k} = \frac{\bar{f}_{T,S,k}}{\bar{f}_{T,L,k}}, \dots \dots \dots (5)$$

then forms a signal which is also very insensitive to drifting signal levels as can be seen in Fig. 5. Unlike algorithm A1 whose correlation result is naturally normalized to the range

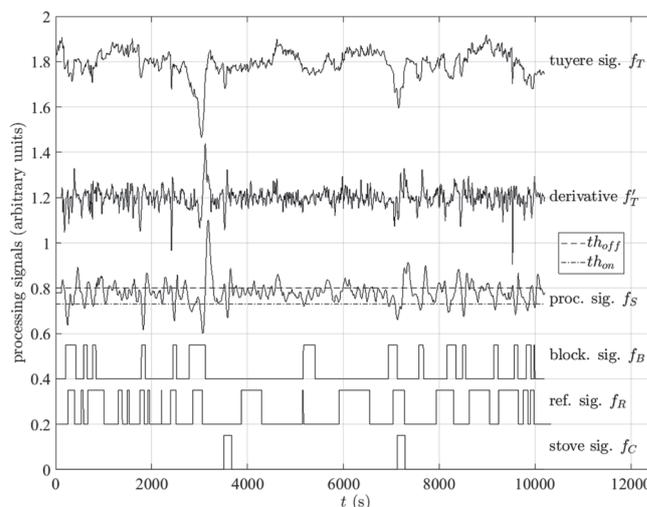


Fig. 4. Results of algorithm A1 (signal correlation). f_T is the blast signal of the tuyere, f'_T is the signal derivative, and f_S is the correlation result. The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B . Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.

of 0.1, the result of A2 is a simple ratio which can be larger than 1. To obtain a useful scaling around the base level 1 of the filter ratio the result f_S is clipped to the range 0.5 to 1.5 to cut off extreme ratios (e.g. the positive and negative peak of event #3 in Fig. 5 are clipped). The threshold levels to obtain the digital blockage signal f_B where $th_{on} = 0.44$ and $th_{off} = 0.48$, respectively. As with the signal correlation approach of A1 the two major blockage events #3 and #6 are detected at a very early state.

In contrast to algorithm A1, the implementation of A2 is only based on storing the current signal levels to be used as $\bar{f}_{T,S,k-1}$ and $\bar{f}_{T,L,k-1}$ for the calculations in the next time step. In general, the short-term averaging would not be needed and it would also be possible to compare the most recent sample with the long-term sliding average. However, the short-term filtering removes some of the signal noise and produces more reliable results. A method which is based on comparing the most recent (unfiltered) sample with a threshold level is described in the next section.

3.3. A3 - Adaptive Thresholding

This algorithm is based on a similar procedure used in radar technology called ‘constant false alarm rate’ (CFAR)³ which is, generally speaking, an adaptive thresholding approach. Figure 6 illustrates the basic idea. The average signal level of a sliding history window is calculated by neglecting a certain number of preceding samples n_{GS} from the most recent sample under test. By suppressing the signal values in this so-called guard samples, the most recent sample can be better isolated from the average signal level calculated from the training window of size n_{TS} . The original CFAR approach is based on symmetric signal windows around the sample under test, which makes it even more robust for signals of bad signal to noise ratio. However, a symmetric guard and training window around the sample under test imposes a time delay of $n_{GS} + n_{TS}$ in the result signal f_S to the most recent signal sample $f_T(k)$. In contrast to radar technology with high sampling rates, the sampling rate of a BF process control system is typically only in the

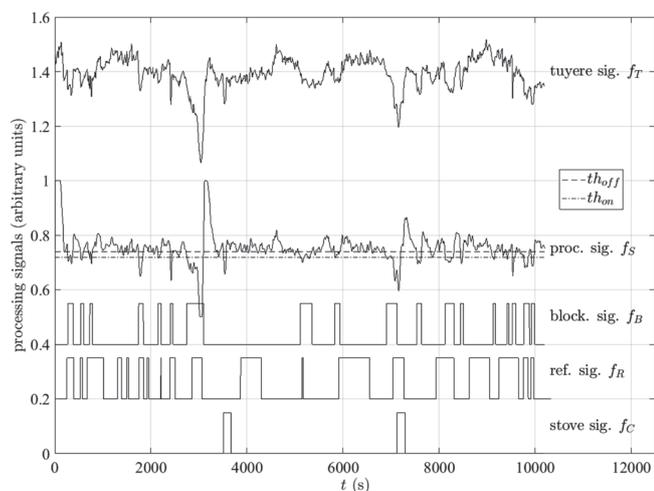


Fig. 5. Results of algorithm A2 (short and long-term filtering). f_T is the blast signal of the tuyere and f_S is the ratio of short and long-term averages. The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B . Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.

range of 1 Hz. Thus, this time delay is undesirably in our application and the present implementation skips one half of the guard and training window and uses the most recent sample $f_T(k)$ as the signal under test to retain a fast response time for blockage detection. The calculation steps are

$$\bar{f}_{T,k} = \sum_{n=k-n_{GS}-n_{TS}}^{k-n_{GS}} f_T(n), \dots\dots\dots (6)$$

for the average background level of the training window and

$$f_{S,k} = \frac{f_{T,k}}{\bar{f}_{T,k}}, \dots\dots\dots (7)$$

for the ratio of signal to background level. In fact, the final implementation is quite similar to algorithm A2, but the short-term average is replaced by the very last sample isolated by the guard window. In analogy to radar applications the resulting signal can be interpreted as a measure for the signal to noise ratio. As can be seen from Fig. 7, the results of A3 are almost identical with the results of A2 and most of the pressure signal dips are identified correctly.

3.4. A4 - Wavelet Transform

Wavelet transform is a powerful tool in signal analysis and has a wide spectrum of applications beyond signal theory. It is widely used in image processing,⁴ biomedical applications⁵ and also fluid mechanics.^{6,7} The fundamentals of wavelet transform can be found e.g. in the book of Kranz.⁸ Wavelet transform can overcome the shortcomings

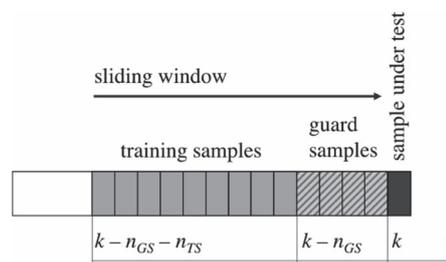


Fig. 6. Calculation principle of adaptive thresholding in algorithm A3 similar to the CFAR method.³

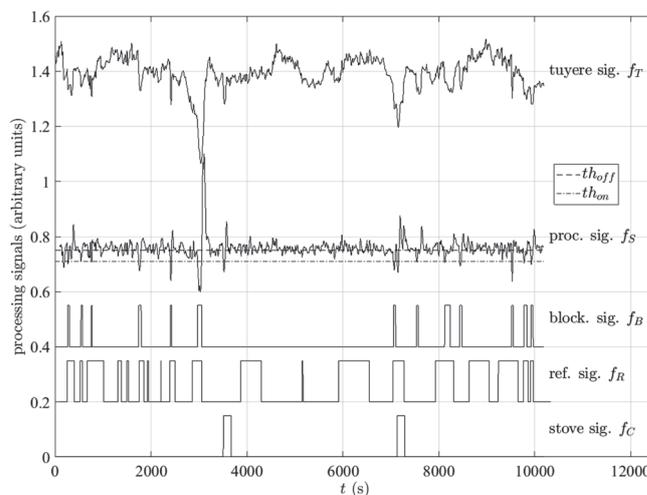


Fig. 7. Results of algorithm A3 (adaptive thresholding). f_T is the blast signal of the tuyere and f_S is the main result signal (can be seen as a signal to noise ratio). The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B . Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.

of simple signal correlation and classical Fourier transform. As discussed in section 3.1, the correlation with one specific pattern signal does not account for the different scales a signal peak might cover in the time domain. On the other hand, a sharp peak in the time domain will produce an infinite series of sine and cosine functions if transformed to the frequency domain via Fourier transform. Wavelet transform, in contrast, is based on dilation and translation of a reference signal (the so-called wavelet) according to

$$X(a, \tau) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \cdot \Psi\left(\frac{t-\tau}{a}\right) dt, \dots\dots\dots (8)$$

where a is the scaling factor and τ is the shift factor of the wavelet function Ψ . Equation (8) represents the continuous wavelet transform (CWT). Hence, a time-dependent signal $x(t)$ is transferred in a time and scale space instead of a time and frequency space. This makes wavelet transform perfectly suited to detect anomalies of varying duration in noisy signals.

For reasons of computational efficiency, the CWT is usually transformed in a discrete formulation (DWT) similar to the discrete Fourier transform (DFT). In the DWT the wavelet function is only shifted and scaled by powers of 2. The wavelet coefficients $c_{j,k}$ for a certain scaling j and shift parameter k can then be obtained by

$$c_{j,k}(t) = \int_{-\infty}^{\infty} x(t) \frac{1}{\sqrt{2^j}} \Psi\left(\frac{t-k2^j}{2^j}\right) dt. \dots\dots\dots (9)$$

Calculation of the DWT for multiple scales can be interpreted as a recursive high- and low-pass filtering. The DWT reduces the number of data points by a factor of 2 at each level of the filter process without loss of information. However, this has the disadvantage that the DWT is not translation invariant and makes data handling more difficult if *e.g.* the results of a certain scaling j shall be compared with another signal at the original sampling rate. The so-called stationary wavelet transform (SWT) solves this issue by upsampling the results of high- and low-pass filtering. So, the SWT results are actually redundant as they contain the same number of samples than the original signal.

We applied the SWT to our test data and compared the results of different wavelet functions and levels of scaling. **Figure 8** shows three examples of wavelets which are suited for SWT (some functions only allow CWT). Although the ‘Sym2’ and ‘Sym3’ wavelets seem more appropriate for the detection of signal peaks, a transformation of our test signal with the ‘Haar’ wavelet produces the best results.

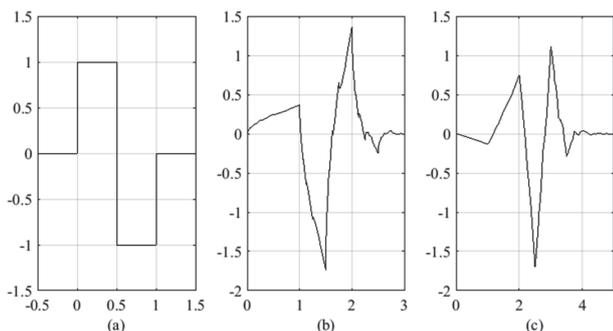


Fig. 8. Examples of wavelet functions. (a) the ‘Haar’ wavelet, (b) the ‘Sym2’ wavelet, and (c) the ‘Sym3’ wavelet.

The decomposition of the test signal f_T via SWT is plotted in **Fig. 9**. Signal A is the low frequency content of the first filter iteration and can be seen as the approximation of the original signal. The signals D_1 to D_5 are the high frequency results and represent the details of the signal f_T at different filter iterations.

The results for the test signal are shown in **Fig. 10** where the detail signal D_4 was used as the main result for block-age detection. The wavelet transform produces sharp peaks

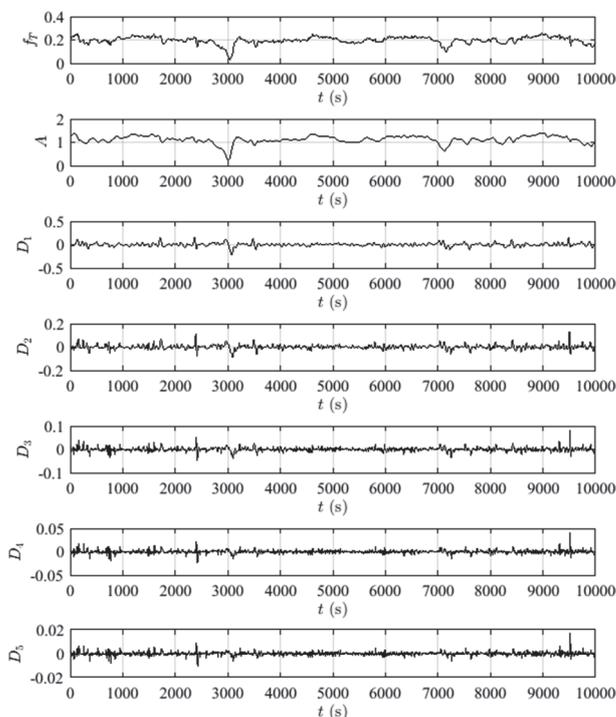


Fig. 9. Wavelet decomposition of the test signal with the ‘Haar’ wavelet. The top signal is the original tuyere signal f_T , A is the so-called approximation result (equals to low pass filtering result of the first iteration), and $D_1..D_5$ are the detail signals (high frequency content of recursive high and low pass filtering) for a stationary wavelet transform with 5 iterations.

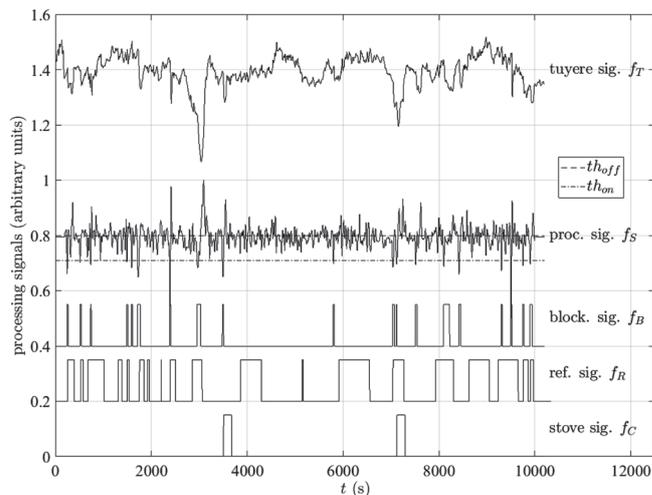


Fig. 10. Results of algorithm A4 (wavelet analysis). The topmost signal is the blast signal of the tuyere and f_S is the detail signal D_4 of the SWT. The horizontal lines indicate the threshold levels for switching on and off the blockage detection signal f_B . Supplementary the reference signal f_R and the state of hot blast stove switching f_C are plotted.

at each signal anomaly but, in contrast to algorithms A1 to A3, does not detect the major blockages #3 and #6 before they are visually apparent. The selection which iteration result fits best for further processing certainly is a trade-off between large-scale and small-scale fluctuations in the original signal. A more sophisticated approach could combine several results of different iteration levels. However, as the use of wavelet transform does not produce superior results compared to A1 to A3, which have a much lower computational effort, there was no further testing of signal processing based on wavelet transform.

4. Long Term Test and Runtime Efficiency

So far, the algorithms were only tested on a small dataset of 2 h 45 min for which tuyere images are available. To gain some long-term experience and runtime efficiency measures, the algorithms have been tested in a quasi-online implementation processing 1 500 hours of real plant data from voestalpine Stahl Donawitz GmbH. In this setup all 20 tuyeres are processed in an endless loop on a standard desktop PC continuously reading the latest pressure data values from a database. This setup is already very close to a real online implementation at the BF. In an online system the results cannot be normalized to a common signal level like the short-term test signal as the bandwidth of signal levels is not known a priori. Hence, also the threshold levels have been adapted to the original levels of the resulting signals f_S for each algorithm and the sensitivities have been set lower than in the offline test to produce less blockage signals which are not important from the operator's perspective. The used data covers a time span of 1 500 operating hours from January to March 2018. **Figures 11** and **12** show the results for one blast furnace and **Table 3** gives the summary of the most important numbers. In Fig. 11 the total number of blockage events calculated by algorithms A1, A2 and A3 are plotted for each tuyere along with the average number of blockages. In addition, the results of A0 is shown, which gives the number of events where the present BF control has shut down a PCI branch. These numbers are obviously much lower as the threshold levels are set very low. Thus Fig. 11(a) has a different scaling than (b) to (d). To get an idea if the total number of calculated blockages makes

sense, Table 3 also gives the relative number of blockage events per tuyere and operating hour and, for comparison, the corresponding value for the test-signal discussed in the previous sections. Considering the reduced threshold levels and observations from the blast furnace, the obtained numbers between 1 and 2 blockages per tuyere and hour give a realistic picture of the frequency of raceway blockages.

The distribution of blockage events over the tuyeres is quite similar for A2 and A3 but looks somewhat different for A1. For example, A1 delivers fewer results for tuyere 1 and more results for tuyeres 10, 12 and 16 compared to A2 and A3. A closer look on the pressure signals shows that while tuyere 1 delivers a quite low noise level, the latter three produce a quite noisy signal. Presumably A1 is more sensitive to signal fluctuations and while producing good results for low noise signals the performance deteriorates for noisy signals. A2 and A3 seem more robust to noise and do not show such large differences in the results.

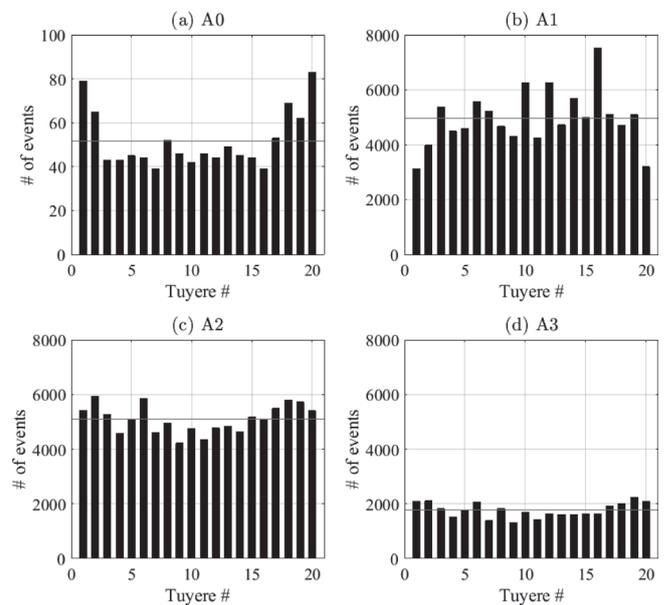


Fig. 11. Results of the long-run test over 1 500 BF operating hours. The total number of blockage events detected for the individual tuyeres and the average values are plotted for algorithms A0 to A3. Note that (a) has a different scaling than (b)–(d) due to the low number of shutdown events.

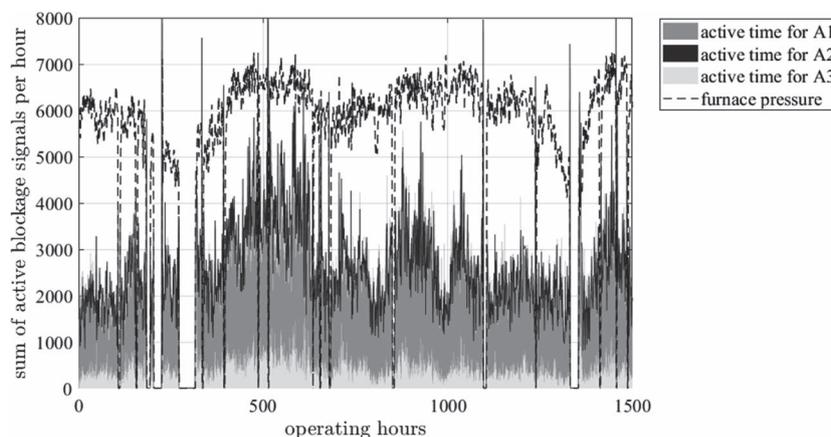


Fig. 12. Results of the long-run test over 1 500 BF operating hours. The figure shows the total number of active blockage signals (sum of all time stamps set to '1') on an hourly basis for the algorithms A1 to A3. In addition, the furnace pressure is plotted with an adequate scaling to visualize hot blast shutdown intervals.

Table 2. Summary of quality measures for the tested blockage detection algorithms after processing the testcase.

Algorithm	Events calculated ^{a)}	Events matched	Events missed	False positive events	Cum. time offset t_D (s)	$t_{s,ma}$ ^{b)} (%)	$t_{s,mi}$ ^{c)} (%)	$t_{s,fp}$ ^{d)} (%)	Threshold level 'on' ^{e)}	Threshold level 'off' ^{f)}	CPU time per sample (μ s)
A1	15	11	8	5	815	23.6	76.4	10.1	0.26	0.40	12.0
A2	18	14	5	3	486	29.0	71.0	8.80	0.44	0.48	0.59
A3	13	11	8	2	740	14.0	85.5	0.80	0.22	0.30	7.92
A4	19	12	7	5	374	10.9	89.1	2.70	0.42	0.60	29.1

^{a)} The reference signal contains 19 blockage events.

^{b)} Number of time stamps with active blockage signal relative to the manually identified blockage time.

^{c)} Relative number of missing active blockage signal based on the manually identified blockage time.

^{d)} False active blockage signal relative to the total number of time stamps in the test data.

^{e)} The blockage detection signal will be 'on' if the signal processing result is smaller than the given value.

^{f)} The blockage detection signal will be 'off' if the signal processing result is bigger than the given value.

Table 3. Summary of the results for the long run test over 1 500 operating hours.

Algorithm	total number of blockages	average number of blockages per tuyere	rel. number of blockages per tuyere per hour	runtime per sample for complete BF	rel. number of blockages per tuyere per hour for the test signal
A0	1 032	52	0.034	–	–
A1	99 121	4 956	3.25	4.04 ms	5.45
A2	101 923	5 096	3.34	4.14 ms	6.55
A3	35 367	1 768	1.16	3.97 ms	4.73

Figure 12 depicts the history of the total count of active time stamps in the blockage signals f_B of A1 to A3 in comparison to the overall hot blast pressure signal for the investigated time span. Each vertical signal bar represents the sum of all active signals on all tuyeres for the period of one hour. A2 produces a high number of event signals around hot blast shutdown intervals, thus A2 is obviously more prone to overall changes in signal levels than A1 and A3. To avoid this, the suppression of result calculation during hot blast shutdown must be extended with a phase-in and phase-out period. The results of A0 have been omitted in Fig. 12 as they would not be visible.

Table 3 also gives the averaged runtimes. While the numbers in **Table 2** give the pure calculation time per sample of pressure data, the numbers in Table 3 are the cumulated times for processing one set of pressure data for the complete blast furnace. As the processing times in Table 3 are almost equal for all three tested algorithms, it can be assumed that the majority of the time is spend on data handling and looping over the tuyeres instead of the signal processing itself. However, due to the simplicity of the algorithms and the low overall processing times of approximately 4 ms for the complete blast furnace, there are no limitations concerning the computational demand of an online implementation in the process control system.

5. Summary & Outlook

Running a blast furnace at high PCI rates demands improved process monitoring and control to ensure stable operation. The reliable shutdown of PCI branches during operating conditions where an efficient combustion of the injected coal is not possible is one factor to reduce possibly negative effects of high coal rates on blast furnace operation. This inherently brings the need for improved tuyere monitoring to assess the current state of the raceway and detect blockages at an early state.

In the present paper we discussed various signal processing methods to improve the currently used solution of a

primitive threshold comparison which does not account for drifting sensor signals or slowly changing burden permeability. The comparison of different processing strategies showed that basically all presented algorithms can provide a much better blockage detection than a simple threshold procedure. The different nature of blockage events was discussed on the basis of a test signal from a real BF with corresponding tuyere camera images.

A long-run test over 1 500 hours of real BF operation has demonstrated that the obtained number of blockages give a realistic picture of raceway behavior and that the computational demands are low enough for online implementation in the process control system.

In part 3 of this paper series, we will discuss various approaches for visual blockage detection based on tuyere camera images. Having a proper solution for both processing strategies, image processing as well as signal processing of hot blast data, provides a good basis for evaluating combined approaches for an optimal raceway blockage detection system.

Acknowledgements

The authors gratefully acknowledge the funding support of K1-MET GmbH, metallurgical competence center. The research program of the competence center K1-MET is supported by COMET, the Austrian program for competence centers.

REFERENCES

- 1) D. Sert and R. Godijn: Assessment of Industrial Operation at Low Coke Rate and Coal Injection in Excess of 200kg/Thm, EU Report 20107, European Commission, Brussels, (2002), 79.
- 2) S. Puttinger and H. Stocker: *ISIJ Int.*, **59** (2019), 466.
- 3) M. Richards: Fundamentals of Radar Signal Processing, McGraw-Hill, New York, (2005), 347.
- 4) P. Schröder: *Proc. IEEE*, **84** (1996), 615.
- 5) M. Unser and A. Aldroubi: *Proc. IEEE*, **84** (1996), 626.
- 6) M. Farge: *Annu. Rev. Fluid Mech.*, **24** (1992), 394.
- 7) M. Farge, N. Kevlahan, V. Perrier and E. Goirand: *Proc. IEEE*, **84** (1996), 639.
- 8) S. G. Kranz: A First Course on Wavelets, CRC Press, Boca Raton, (1996), 22.