• U C •

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

DEPARTAMENTO DE
ENGENHARIA MECÂNICA

On-line Oil Condition Monitoring

Dissertation submitted to obtain a Master's Degree in Mechanical Engineering on the specialty of Mechanical Design

Author

José Miguel Marques Querido Salgueiro

Mentors

Prof. Dr. Amílcar Lopes Ramalho

Prof. Dr. Jožef Vižintin

Jury

President	**Professor Doctor Ricardo António Lopes Mendes** **Assistant Professor of the University of Coimbra**
Voting Members	**Professor Doctor Amílcar Lopes Ramalho** **Associate Professor of the University of Coimbra** **Professor Doctor Cristina dos Santos Louro** **Assistant Professor of the University of Coimbra** **Professor Doctor Jožef Vižintin** **Full Professor of the University of Ljubljana**

Institutional Collaboration

Faculty of Mechanical Engineering, University of Ljubljana

Center for Tribology and Technical Diagnosis, University of Ljubljana

Coimbra, September, 2011

Acknowledgements

The work presented here was only possible thanks to the collaboration and support of some people to whom I must give my gratitude:

To all my dearest family members who never forgot me and gave me all the love and support I needed along these years.

To my mentor in the Department of Mechanical Engineering, Faculty of Sciences and Technology, University of Coimbra, Prof. Dr. Amílcar Lopes Ramalho, whose unconditional support allowed for this partnership with the University of Ljubljana to be possible.

To my mentor in the Faculty of Mechanical Engineering, University of Ljubljana, Prof. Dr. Jožef Vižintin, for welcoming me so dearly since the first day. Without his vast experience and leadership, this project would have not been possible.

To my co-worker, B.Sc. Gabrijel Peršin, whose guidance and understanding was fundamental in the development of this work.

To all my colleagues and co-workers in CTD – Prof. Dr. Mitjan Kalin, Prof. Dr. Bojan Podgornik, Dr. Igor Velkavrh, Dr. Marko Sedlaček, B.Sc. Boris Kržan, B.Sc. Rok Simic, Mr. Franci Kopač, Mrs. Joži Sterle, B.Sc. Anton Urevc, B.Sc. Janez Kogovšek, B.Sc. Aljaž Pogacnik, B.Sc. Markus Kronberger, B.Sc. Vladimir Pejakovic, B.Sc. Jure Jerina and B.Sc. Marko Polajnar - for their friendship and support.

To our associated specialists in Institute Jožef Stefan – M.Sc. Pavle Boškoski and Prof. Dr. Đani Juričić - for their expertise and advice.

To Prof. Dr. Cristina Louro and Prof. Dr. Mirko Sokovič, Erasmus coordinators in the Department of Mechanical Engineering (Coimbra) and Faculty of Mechanical Engineering (Ljubljana), respectively, for their availability to deal with all the exchange-related aspects.

To all my friends in Portugal, for their comradeship and support along all these years in the University of Coimbra.

And to all my new friends in Rožna Dolina who shared with me the wonders of a great country and accept me as one of their own.

Resumo

O dia-a-dia das pessoas está dependente de máquinas que interpretam o mundo real e que devem fazê-lo com a máxima precisão. Dos sensores raramente se obtém valores fixos, isto devido a fenómenos externos, complexos e de difícil previsão. Aqui, as técnicas de processamento de sinal são usadas para "recriar" a realidade, permitindo uma análise posterior de maior confiança. No mundo da informação, onde tudo é optimizado através de processos digitais, a indústria não é excepção e a sua eficiência económica da maior importância.

Em ambiente industrial, a análise de lubrificantes é um procedimento indirecto de diagnóstico técnico que avalia a condição da máquina e os seus componentes, bem como o próprio estado do óleo. O método *offline* tem sido geralmente utilizado para a avaliação dos mesmos – as amostras são retiradas de um reservatório para análise em ambiente laboratorial. Uma avaliação em tempo real permitiria a implementação da manutenção condicionada nesta área, de tal forma que o planeamento e os custos das intervenções possam ser reduzidos através da detecção precose das falhas.

Hoje em dia, a mais fidedigna acção de monitorização depende do julgamento humano e até sem o conhecimento do que está para vir. Com este trabalho, pretende-se desenvolver um algoritmo de detecção de alterações em sinal capaz de avaliar quantitativamente e qualitativamente amostras adquiridas para monitorização em tempo real de vários parametros associados aos lubrificantes.

No futuro, para atingir uma monitorização autónoma, é necessário obter-se um método que seja leve, independente da grandeza da medição e tão isento de falhas quanto possível. Deste modo, um processo de análise sem supervisão humana constante pode ser implementado, permitindo um prognóstico do tempo disponível para a intervenção.

Foi desenvolvido em MatLab um algoritmo baseado na técnica de soma cumulativa linear e testado em sinais adquiridos de diferentes parâmetros de lubrificantes, obtidos a partir de uma unidade de monitorização independente desenvolvida no Centro de Tribologia e Diagnósticos Técnicos da Universidade de Liubliana.

Palavras-chave Detecção de mudanças, soma cumulativa (CUSUM), linearização, estado estável e transiente, tendência.

Abstract

Everyday people's way of life is dependent on machines that interpret the real world and must do it with the maximum accuracy. Sensor readings are not absolute and often fail to maintain level with a fixed value. This is due to complex external phenomena that cannot be predicted by easy means. Here, signal processing techniques play their part to "recreate" what is being read, allowing for more assured further analysis. In the world of information, where all is progressively optimized through digital procedures, industry is no exception and its economic efficiency of the highest importance.

In industrial environments, oil monitoring is a technical diagnosis indirect procedure to evaluate a machine's and its parts current condition, as well as the lubricant's condition itself. The offline method has been the generally implemented one for lubricant condition assessment - samples are taken from a reservoir and for analysis in laboratory. On-line evaluation would grant the possibility for predictive maintenance implementation in this field, in a way that scheduling and costs can be reduced through early detection of faults occurring.

Presently, the safest monitoring action depends on human judgment, and even then without knowledge on what is to come. The aim of this work here presented is the development of a change-detection algorithm capable of evaluating quantitatively and qualitatively sensor readings for on-line monitoring of various oil properties.

To achieve a multi-parameter oil diagnosis in the future it is necessary to reach a detection and evaluation method that is light, value-independent and as much fault-free as possible. With it, an automated persistent analysis process without constant human supervision can be implemented and prognostics made to determine how much time is available for repair interventions.

A linear CUSUM-based algorithm was developed in MatLab and tested in acquired data from different oil parameters retrieved from a standalone monitoring unit developed in the Center for Tribology and Technical Diagnosis, University of Ljubljana.

Keywords Change-detection, CUSUM, linearization, steady and non-steady state, trend.

Index

INDEX OF FIGURES

INDEX OF TABLES

NOTATION AND ACRONYMS

Notation

$A_{1,Q}$ – Decision quadrant's reference

$A_{1,cur}$ – Current trend output

$A_{1,ref}$ – Reference trend output

a_0 – Origin for linear regression function

$a_{0,cur}$ – Normalized current window LR origin

$a_{0,ref}$ - Normalized abnormal values window LR origin

a_1 – Slope for LR function

$a_{1,cur}$ – Normalized current window LR slope

$a_{1,ref}$ – Normalized abnormal values window LR slope

c – Coefficient of smoothed signal

d – Coefficient of detail signal (WT modulus)

E – CUSUM value

E_L – Likelihood estimator

e – Error value

f – Time function

$\tilde{f}$ – Approximated time function

H_0 – "No change" hypothesis

h – Step-change identification factor

i – General time index

j – Internal module index

K – Current window length

K_{abn} – Abnormal values window length

k_1, k_2 – Factors for slope weighting

L – Likelihood

m – Predicting abscissa index

N – General window length

n – Normalized value

P – Probability

Q – Quadrant array

Q_1, Q_2, Q_3 – Evolution state limits

R – Relative change

R_{MSD} – Ration of mean-square deviations

rms – Root-mean-square

rms_{cur} – Root-mean-square of the current window

S – Wavelet characteristic scale

s – Wavelet scale

Th_Q - Evolution quadrant's relative threshold

T_{ref} – Reference time

Th_0 - "Stable" state threshold

Th_1 – CUSUM "warning" condition threshold

Th_2 – CUSUM "critical" condition threshold

t – Time

U – Length of discretized scaling function

u – Scaling function index

V – Length of discretized wavelet function

v – Wavelet function index

W – First-order wavelet transform

WW – Second-order wavelet transform

Y – Set of acquired data

Y_f – Set of filtered data

$\bar{Y}$ – Average of acquired data

y – General ordinate/Acquired value

y^* – Predicted value

y^*_{ref} - Reference-predicted point

y^*_{cur} – Current-predicted point.

$\overline{y}$ – Average of values

x – General abscissa

Z – Slope evolution zone

Z_{cur} – Slope evolution zone for current window

Z_{ref} – Slope evolution zone for reference window

δ^2 – Mean-square deviation (2nd estimate for MSD-based algorithm)

δ_f^2 – Filtered mean-square deviation (2nd estimate for MSD-based algorithm)

$\lambda_1, \lambda_2, \lambda_3$ – Filter factors

μ - Average value

μ_1 – Contaminant Viscosity

ν^2 – Mean-square deviation (1st estimate for variance-based algorithm)

ν_f^2 – Filtered mean-square deviation (1st estimate for MSD-based algorithm)

μ_{ref} – Reference viscosity

σ^2 – Variance

φ – Scaling function

ψ – Wavelet function

Acronyms

ANN – Artificial Neural Network

CDA – Change-Detection Algorithm

CDP – Change-Detection Problem

CTD – Center for Tribology and Technical Diagnosis

CUSUM – Cumulative Sum

ELL – Expected Log-Likelihood

LCC – Life Cycle Cost

LR – Linear Regression

Lr – Likelihood Ratio

MSD – Mean-Square Deviation

OL – Observation Likelihood

PPM – Post-Processing Module

pdf – probability density function

SMU – Standalone Monitoring Device

SOM – Self-Organizing Map

WT – Wavelet Transform

1. INTRODUCTION

In modern day industry, equipment life-cycle cost (LCC) is thoroughly planned to ensure maximum profit in production. LCC considers all costs inherent to an asset since its acquisition to the time of its disposal and, in between, the costs of stoppage time (Riggs 1982). This way, the optimization concept cannot be without a chosen maintenance strategy among three main (Zhao et al. 2005):

1. Corrective – the maintenance intervention is only performed upon failure;
2. Preventive – the maintenance intervention in performed on regular time spans, generally determined by statistical data;
3. Predictive – interventions are decided based on operating condition.

The focus of the work here presented lies in predictive maintenance. It can have high initial costs in comparison to the previous two, namely because of monitoring expenses (personnel training, equipment acquisition, etc.). On the other hand, it can reduce significantly the number undesired and unpredicted setbacks during the machine's operating time which lead to a reduction in later maintenance expenses.

So, when choosing predictive maintenance strategy, diagnosis techniques play a decisive part in its efficiency. According to Robin (2006), the operating lubricant analysis can be used to assess its own health or the general mechanical condition of the machine components, by looking at the wear particles count. Other relevant oil parameters include viscosity, temperature, dielectricity and moisture. The target parameter can be monitored permanently (on-line) or at regular time intervals (off-line).

Traditionally, oil analysis is done off-line by the lubricant's seller using randomly collected samples which might not be representative of the machine's actual condition. In this sense, on-line oil monitoring sets a state-of-the-art technique for predictive maintenance, which the Center for Tribology and Technical Diagnosis (CTD) is thriving to achieve through its already built standalone oil monitoring device unit (SMU).

The focus of this thesis is then set on on-line diagnosis and software processing of data. It is intended for the processing software to reach the maximum autonomy possible in issuing warnings signals, eliminating the need for constant monitoring by an operator. It

should be capable to output the rates of change for all targeted parameters and accurately detect their points of trend alteration. The challenge here lies on data analysis and its complex processing, mainly due to the different nature of acquired signals (cumulative and stochastic), producing the minimum number of false warnings. As it will also become clearer along this report, the small variations in the surrounding environment or in the machine's operating conditions will have impact on the ideal readings in the sensor and on the data analysis process.

In short, a true state-of-the-art monitoring and assessment process will allow for maintenance managers to schedule only the necessary interventions with appropriate intervention, therefore cutting off repair costs. In this work, a data analysis procedure is presented as the first step to achieve this.

1.1. Objectives

In the train of thought of what was described before, the following objectives were established for this thesis:

1. To develop a software for condition monitoring and diagnosis of lubricants, including:
    a. Detect oil condition variations through appropriate detection algorithm - to be developed on MatLab® (The MathWorks™) programming environment with obtained data from experiments;
    b. Indicate both signal trend and evolution state based on set thresholds.
2. Test the software after its completion by performing contamination experiments on the *in situ* test machine - this also includes the establishment of an appropriate protocol for the following contaminators:
    a. Water;
    b. Fuel;
    c. Different lubricant;
    d. Cooling fluid;
    e. Solid ferrous and non-ferrous particles.
3. Final validation of the software.

2. LITERATURE REVIEW

In the context of automated on-line supervision in industrial environment, signal assessment represents the greatest of challenges. Even to a human observer, decisions have to be taken without knowledge of what is to come (Kalai and Vempala 2004). This non-deterministic process has been propelling the need to search and develop faster and more solid change-detection algorithms in the field of Technical Diagnosis.

In essence, change-detection problems (CDP) are divided in two types – Bayes and minimax (Isom 2009). They differ according to the information that is available on the change-time distribution. The first, associated with English mathematician Thomas Bayes (1701 - 1761), takes a policy of minimizing false alarms (also called "cost") by accepting future uncertainty as a stochastic distribution with reflection in prior data (Onatski 2000).

Minimax criterion, on the other hand, does not look at new values as another statistical function but simply unknown, therefore minimizing the expected change detection delay. This decision-making policy is met be applying the cumulative sum (CUSUM) procedure, which will be discussed in 2.4.

So far, the problem presented is where to mark the end of a steady state that is, the beginning of a transient state in the acquired data, and how well can the trend be evaluated.

2.1. Wavelet-based Algorithm

Jiang et al. (2002) suggest a method to identify steady states in stochastic signals based on the use of wavelets. Wavelets are cyclic oscillations with an impulse-like response. Commonly used in data processing, these have very particular properties depending on their defining parameters (impulse frequency, impulse duration, etc.).

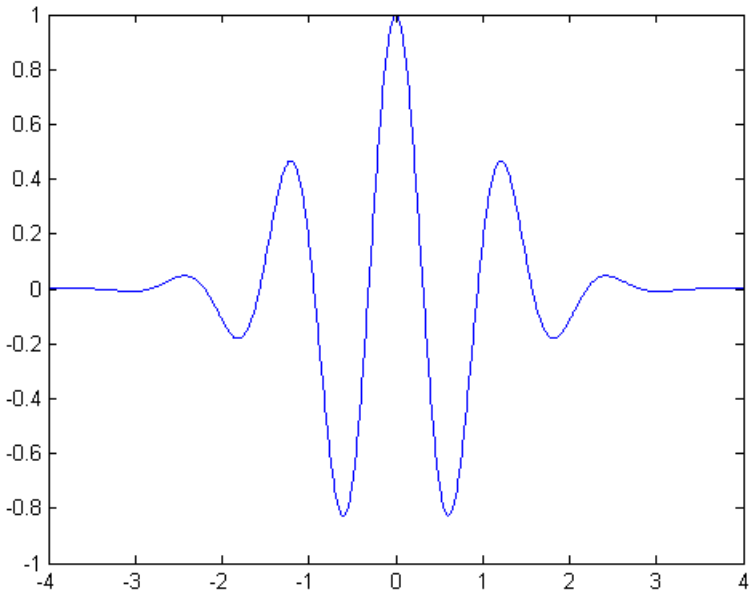

Figure 1. Example of a wavelet (Morlet type)**.**

This method starts by denoising/filtering incoming data using a wavelet transform (WT) to approximate the analyzed data window:

$$\tilde{f}(t) = \sum_{u \in U} c_u \varphi_u + \sum_{v \in V} d_v \psi_v \tag{1}$$

If the wavelet function $\psi(x)$ (Figure 1) is the first-order derivate of a scaling function $\varphi(x)$:

$$\psi(x) = \frac{d\varphi}{dx} \tag{2}$$

Then the wavelet transform (W) of a general function $f(t)$ at scale s is given by the convolution:

$$W_s f(t) = f * \varphi_s(t) \tag{3}$$

With this in mind, denoise operation starts by thresholding the WT modulus (d). It identifies the so-called "abnormalities" – a peak value of short duration. This concept must not be confused with "abnormal values", described in Charbonnier et al. (2004) as being acquired values that ceased to follow a previously established prediction model. Also, it should not be confused with step-changes, which are on the focus of this work and imply a change in the signal's actual average.

This process of peak search and elimination uses wavelet transformations just until a scale of 2. Afterwards, it is raised until a pre-determined characteristic scale (S) and steady state search starts, thresholding first and second-order (WW_s) wavelet transforms:

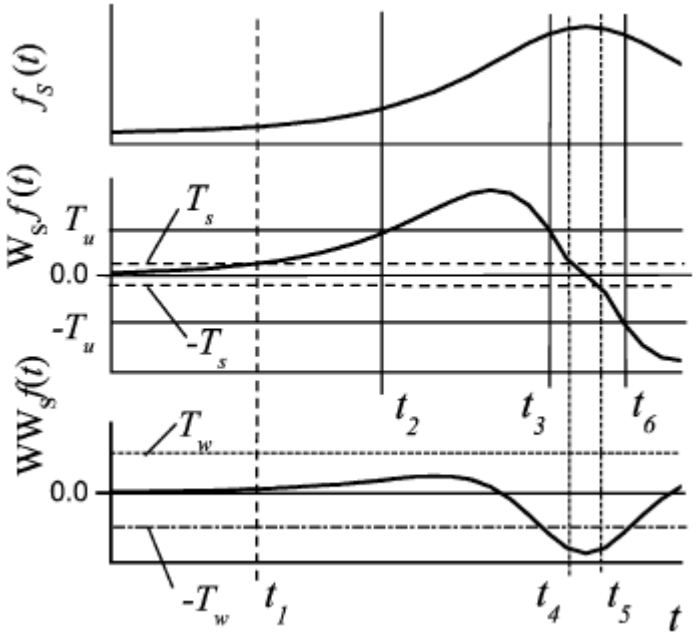

Figure 2. Wavelet transformation results. (Jiang et al. 2002).

According to Figure 2, the position for each W_s and WW_s in relation to their respective thresholds will define their "degree of stability". The incoming signal finds itself in a "more transient" state if both surpass their limits.

Because of the symmetry inside the wavelet window (Figure 1), this method naturally operates with some time delay. Jiang et al. (2002) overcame this by applying a symmetric extent technique – mirroring past data around the latest time instant t.

2.2. ANN-based Algorithm

Tambouratzis and Antonopoulos-Domis (2004) suggested a non-linear statistical data modeling method for on-line trend identification in nuclear power plants neutron density supervision. It is based on an artificial neural network (ANN) to establish non-linear patterns which will test the signal's evolution.

ANNs are adaptive systems that change their structure according to the information acquired that is, during the so-called "learning phase". The connection between the input and output is done through an interconnected group of neurons or nodes (Figure 3).

According to the authors, the self-organizing map (SOM) is a standard type of ANN for "unsupervised learning", meaning that no knowledge about the density

probability distribution (pdf) of future data is available or assumed. The SOM network is particularly useful to reduce data storage – by creating a "map" (a discretized representation of the input space), high-dimensional data is reduced to a bi-dimensional array that conserves the information about past topological relationships. In this array, information about the connection weights are kept in a way that, during the decision process, a "winner" node is chosen if the inter-nodal weights that originated it are the most similar to the feature values of the input test pattern.

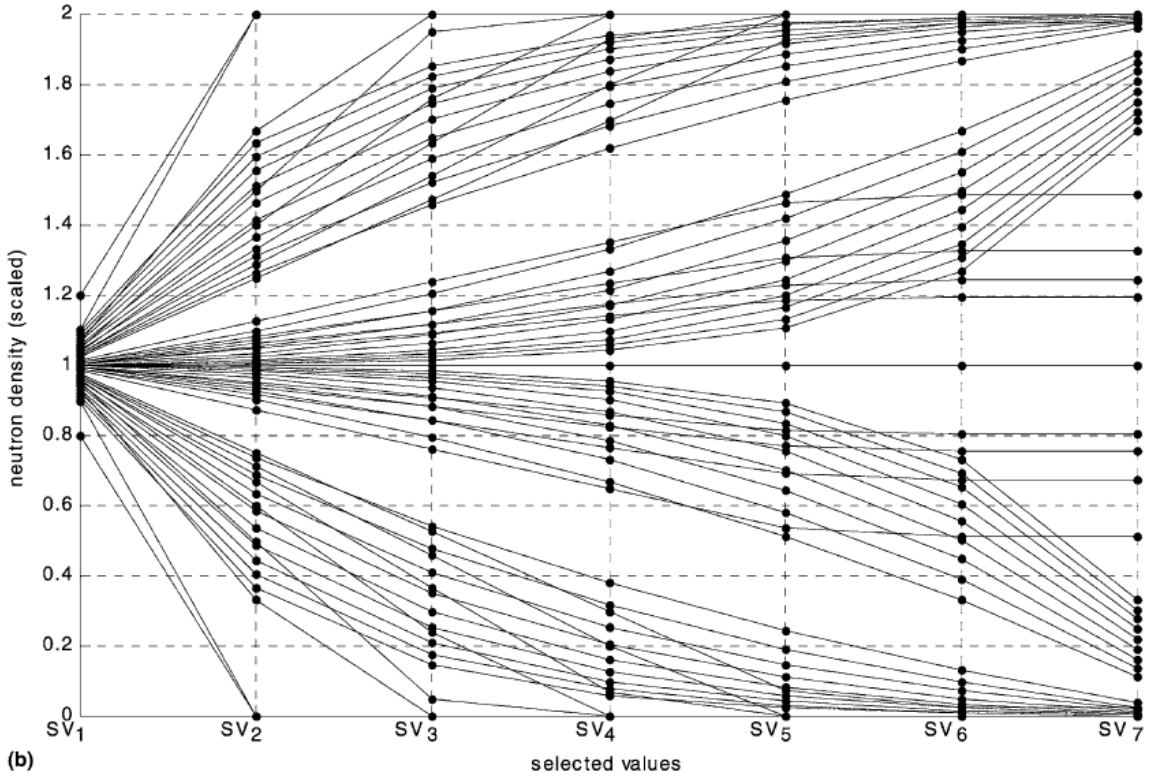

Figure 3. SOM's training patterns (Tambouratzis and Antonopoulos-Domis 2004).

As shown in Figure 3, 61 training patterns where established and the signal must be scaled accordingly to meet pre-defined limits. This scaling (or normalization) is done by subtracting the first signal value (here marked as SV_1) from the following. This method takes seven signal values equally spaced to reach a decision, involving six inter-nodal weights to reach a semi-qualitative evaluation of the acquired signal. In the end, the trend is classified into one of these temporal shapes.

2.3. MSD-based Algorithm

Traditionally, steady-state identification relies on the T-test (from a T-student distribution - Figure 4) of a given data window and in the comparison of its variance (σ^2) with an expected noise contribution reflected on the mean-square deviation (MSD - v^2), also called variance estimator:

$$\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}(Y_i - \bar{Y})^2 \tag{4}$$

$$v^2 = \frac{1}{N-1}\sum_{i=1}^{N}(Y_i - Y_{f,i-1})^2 \tag{5}$$

These estimators in (4) and (5) differ in the use of the mean value of data inside the window ($\bar{Y}$) and the previously filtered value ($Y_{f,i-1}$).

Cao and Rhinehart (1995) present a method for on-line steady-state identification based in the statistical F-test of a stochastic signal rather than T-test. As it can be seen in Figure 4 and Figure 5, F-distributions are pdf distributions that take into consideration the null hypothesis.

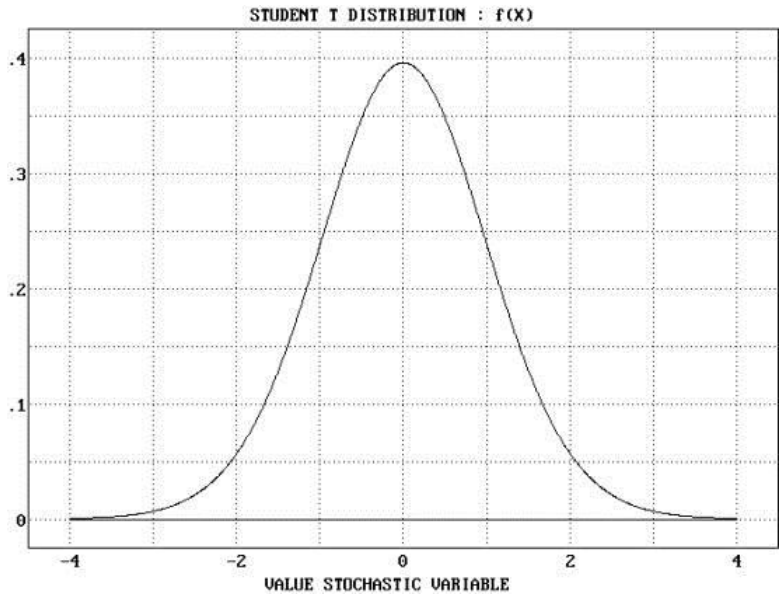

Figure 4. T-Student distribution probability density function (Borghers and Wessa 2011).

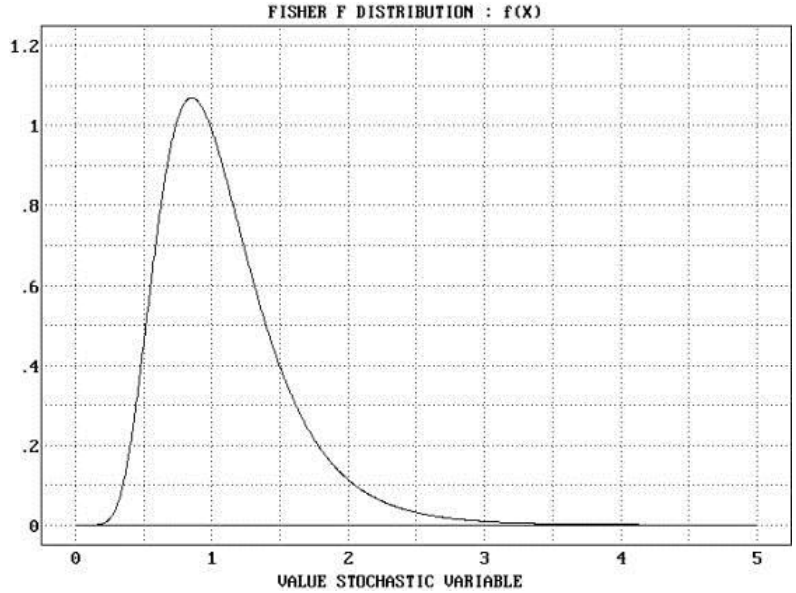

Figure 5. F-Distribution probability density function (Borghers and Wessa 2011).

The method depends on three weight parameters (λ_1, λ_2 and λ_3), referred in documentation as filter factors, and relies on thresholding a ratio (R_{MSD}) of 2 estimates of variance (v^2 and δ^2) defined by:

$$R_{MSD} = \frac{(2-\lambda_1)\cdot v_{f,i}^2}{\delta_{f,i}^2} \tag{6}$$

Where the filtered MSDs are calculated according to:

$$v_{f,i}^2 = \lambda_2 \cdot \left(Y_i - Y_{f,i-1}\right)^2 + (1-\lambda_2)\cdot v_{f,i-1}^2 \tag{7}$$

$$\delta_{f,i}^2 = \lambda_3 \cdot (Y_i - Y_{i-1})^2 + (1-\lambda_3)\cdot \delta_{f,i-1}^2 \tag{8}$$

The filtered data window is also done by weighting the contributions of past values:

$$Y_{f,i} = \lambda_1 \cdot Y_i + (1-\lambda_1)\cdot Y_{f,i-1} \tag{9}$$

The results achieved on temperature and pH readings, are interesting but it became clear that the choosing of λ_1, λ_2 and λ_3 can have some undesired effect. If these factors are too small, noise influences are reduced, making the pdf of both steady and non-steady states to fall apart. On the other hand, the fast identification of state change can be compromised due to the great influence of past data.

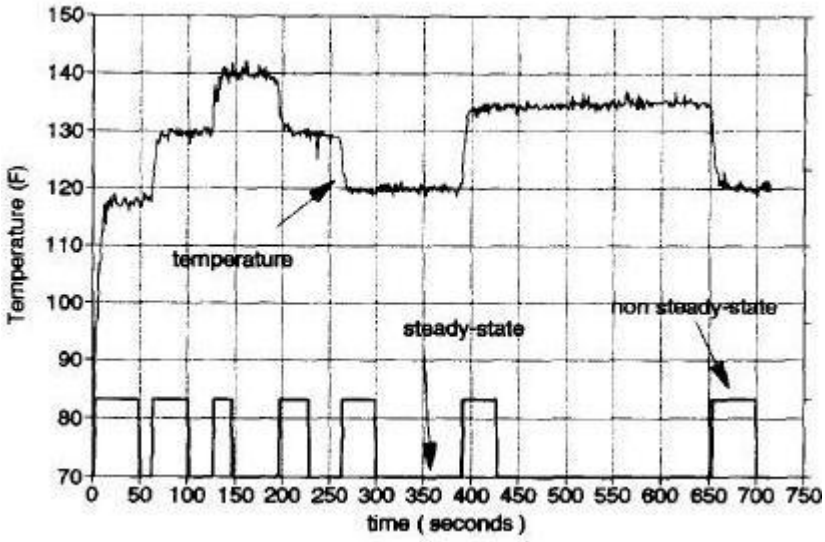

Figure 6. Demonstration of the steady-state identifier (λ_1=0.2, λ_2= λ_3=0.1) (Cao and Rhinehart 1995).

The results presented (Figure 6) show mainly step-changes with following close-to-zero trends. In the focus of this work, constant increase/decrease is considered to be a steady-state, so it is necessary to know how well the algorithm works when confronted to this situation, but the weighting of past data information is an idea that will be later used for qualitative evaluation, described in section 3.7.2.

2.4. Linear CUSUM-based Algorithm

CUSUM is a sequential analysis technique used to detect changes in a given time series. The calculation process involves the consecutive sum of error values, according to equation (15). The primary assumptions is that the acquired data's statistical distribution is of Gaussian-type with an average (μ) of 0 and variance (σ^2) equal to 1 (Basseville and Nikiforov 1993). T-distributions (Figure 4) are closely shaped to Gaussian but with heavier tails that is, with more probability of outliers occurrence.

This is the approach that sets the basis for the trend extraction algorithm presented by Charbonnier et al. (2004), which was tested in industrial and biological monitoring processes, and also for the one presented in this work.

This method does not require pre-filtering. The justification for it is that filtering would lead to a masking effect, also known as "aliasing", on important high-frequency features such as step-changes.

The method involves the establishment of successive linear models and consequent extrapolation of values, as shown in Figure 7. This process will become clearer on the algorithm's explanation in chapter 3.

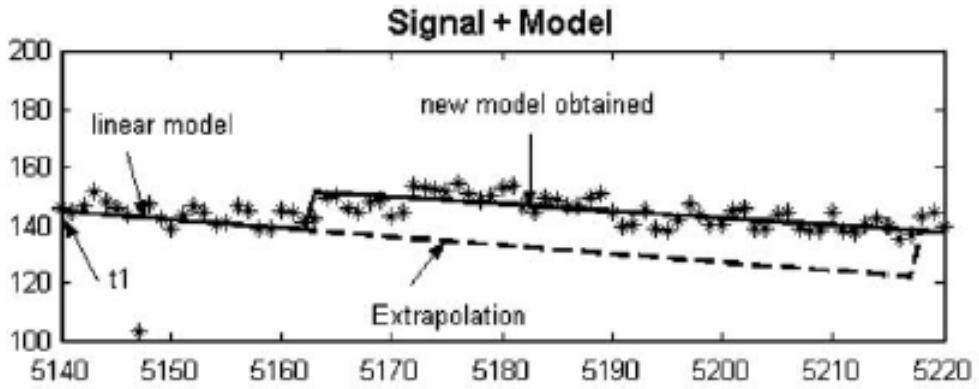

Figure 7..Linear model extrapolation (Charbonnier et al. 2004).

The decision process for qualitative evaluation, on its turn, is similar to the one presented here in section 3.7 and is based in the comparison of segments of data and their slopes. The difference though, lies in the usage of more information to reach a qualitative assessment not so prone to false alarms, as it will be discussed later.

2.5. Non-Linear CUSUM-based Algorithm

Vaswani (2005) presents a more complex method by applying two different likelihood functions – Expected (negative) Log Likelihood (ELL) and Observation Likelihood (OL) – which are suitable to slow and fast changes, respectively. In a discrete set of data Y, likelihood can be defined as the probability of y_t given the already output set $Y_{1:t}$:

$$\mathrm{L}(y_t|Y_{1:t}) = P_{y_t}(Y = Y_{1:t}) \tag{10}$$

ELL confronts this by estimating the negative logarithm of the previous likelihood of the state (at time t) under the "no change" hypothesis (H_0):

$$ELL(Y_{1:t}) = \mathrm{E}_L[-\log\ \mathrm{p}_t(\mathrm{y}_t)|Y_{1:t}, H_0] \tag{11}$$

OL is defined as being the negative log-likelihood of the current observation conditioned on past observations under the "no change" hypothesis:

$$OL = -\log P(Y_t|Y_{1:t-1}, H_0) \tag{12}$$

The likelihood ratio (Lr) will then compare consecutive values of ELL or OL and substitute the error value in equation (15). The identification process of changes is common in all CUSUM-based algorithms and will be detailed later in 3.6. Vaswani (2005) tested this idea on simulated results to observe the method's response towards fast and slow changes.

Other variations of the CUSUM technique are available in Basseville and Nikiforov (1993) with proper theoretical justification.

3. CHANGE-DETECTION ALGORITHM

The change-detection algorithm (CDA) represents the pivotal part of this work. Within this software, all acquired data is to be properly processed and from it the important change features retrieved, that is the signal's trend and its evolution status.

The CDA's process relies on the linearization of important segments of data which are representative of a system's condition change, as shown in Figure 8.

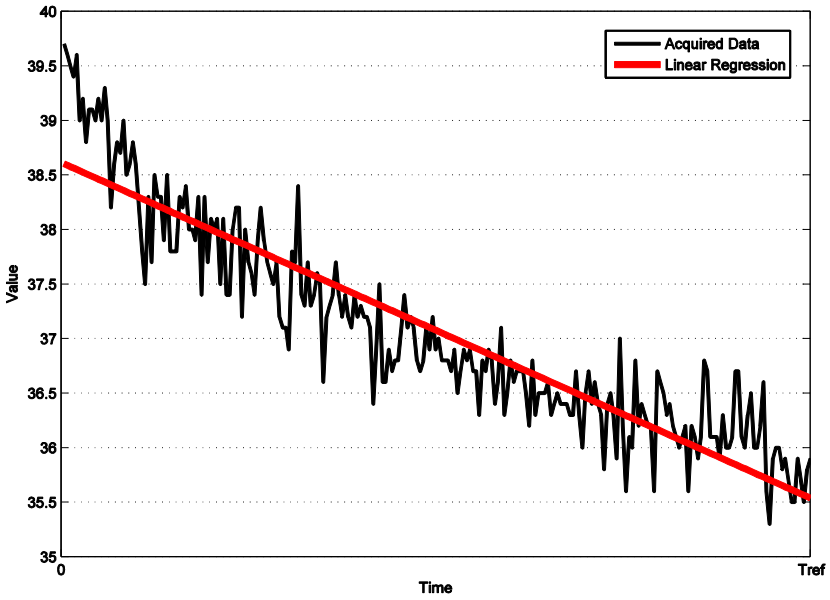

Figure 8. Linearization of a data window.

The red line in Figure 8 plots the points predicted by the LR parameters retrieved from the window, according to the following equation:

$$y = a_0 + a_1 \cdot x \tag{13}$$

With a_0 being the ordinate at the axis' origin and a_1 its slope.

The method to retrieve these parameters is fully explained in section 3.4.3.

3.1. The CUSUM Computation

The CUSUM computation, introduced in section 2.4, can be applied this way:

Given the general index i at the latest instant t, if we consider the difference between the acquired value (y_i) and a predicted one (y_i^*), the error (e_i) is defined by:

$$e_i = y_i - y_i^* \quad (14)$$

The cumulative sum (E_i) is then computed by summing consecutive error values:

$$E_i = e_i + E_{i-1} \quad (15)$$

In this work, the mathematical principles applied for change-detection are the same as in Charbonnier et al. (2004), but the method for error computation has some differences which will be thoroughly detailed in 3.5.

3.2. General View

The on-line change-detection algorithm developed can be perceived as a cycle of operations which include five main modules (Figure 9):

- Step-Changes Identification;
- Quantitative evaluation of the current data window (Window Analysis Module);
  - Current trend output ($A_{1,cur}$);
- New point prediction (Predicting Module);
- Quantitative evaluation of the abnormal values' window (Post-CUSUM Module);
  - Reference trend output ($A_{1,ref}$);
- Qualitative evaluation (Post-Processing Module);
  - Evolution "State".

Figure 9 represents the general flowchart of the algorithm. Each line color indicates the origin of the variable that is being routed in between modules:

- **Green** – Data array writing process;
- **Orange** – Step Identification process;
- **Blue** – Window Analysis module;
- **Cyan** – Predicting module;
- **Red** – Post-CUSUM module;
- **Yellow** – Post-Processing module.

Notice that the ending tips of some flow lines are different. The square tip indicates that the variable is being used as a trigger for a specific process or entire module.

This aspect is of the uttermost importance when dealing with information that is processed in different time periods.

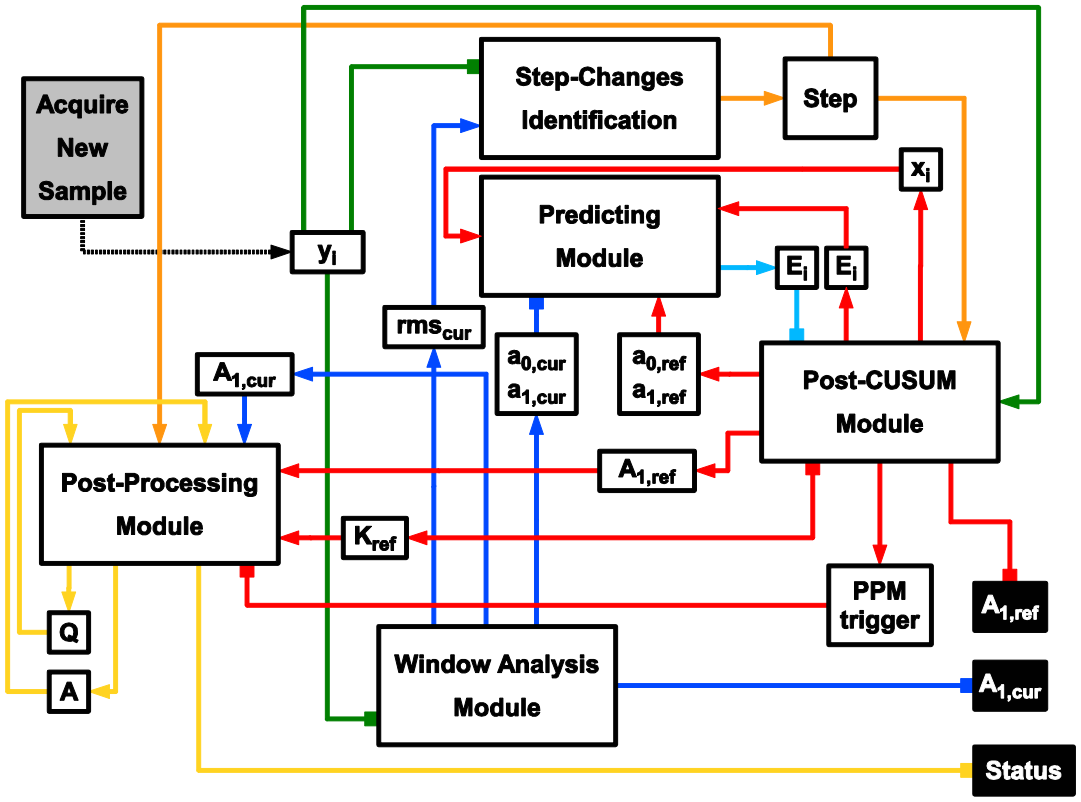

Figure 9. CDA's general view.

3.3. Step-Changes Identification

During the planning of the CDA it was established that one of the important features that must be identified are the step-changes. These kind events are important because they might happen due to a sensor reading fail or an actual change in the mean of the read parameter (Figure 10).

These kinds of events were clearly seen during the relative moisture readings in the water contamination experiment (4.4.2.1) and indicate an abrupt change on the system.

Consider this figure:

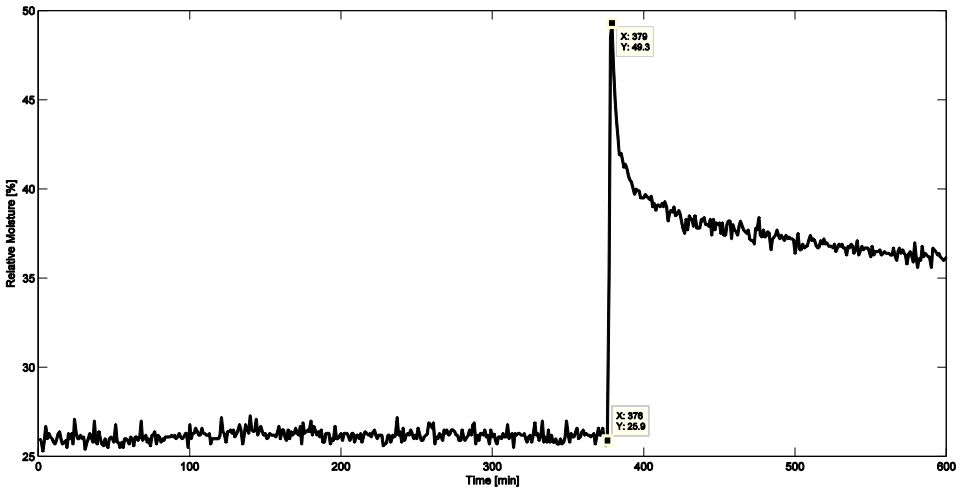

Figure 10. Step-change occurrence after contamination with water.

The step-change in Figure 10 occurred in a span of three samples, where the read parameter increased its value by 90% of its previous mean.

The high-frequency fluctuations make it difficult for a correct identification of step-changes. A simple way to do it would be to fix a threshold for the relative change occurred between neighboring points. If so, the calculation for the relative change (R) would be:

$$\mathrm{R} = \frac{y_i - y_{i-1}}{y_i} \tag{16}$$

But this method presents two main problems:

- If the absolute value of y_i is too close to zero, then R tends to infinity which means that false step-change warnings would be sent permanently.
- The algorithm should be independent from the signal-to-noise ratio, so the typical relative change can have very different values depending on the source of the read data, making it impossible to fix a threshold.

The magnitude of variation of a signal around its mean can be calculated considering the definition of the root-mean-square (rms):

$$rms = \sqrt{\frac{\sum_j^N {y_j}^2}{N}} \tag{17}$$

Applying this equation to a data window with a length of N samples, identification of a step change becomes:

$$Step = \begin{cases} 1, |y_i - y_{i-1}| > h \times rms \\ 0, |y_i - y_{i-1}| \leq h \times rms \end{cases} \quad (18)$$

The value of h can be set depending on the sensitivity CDA should have towards finding these step-changes. In the version of this work, h was set to 20 after several run tests performed on obtained data.

As it can be seen in Figure 9 (**Orange** flow lines), the algorithm will send the information of this occurrence to different modules, changing its general behavior.

3.4. Window Analysis Module

Here is where the algorithm begins to differ from the approach suggested by Charbonnier et al. (2004). For this group, three sub-modules were established, as shown in Figure 11:

- Current window array writing;
- Window normalization;
- Window linearization (Retrieve LR parameters).

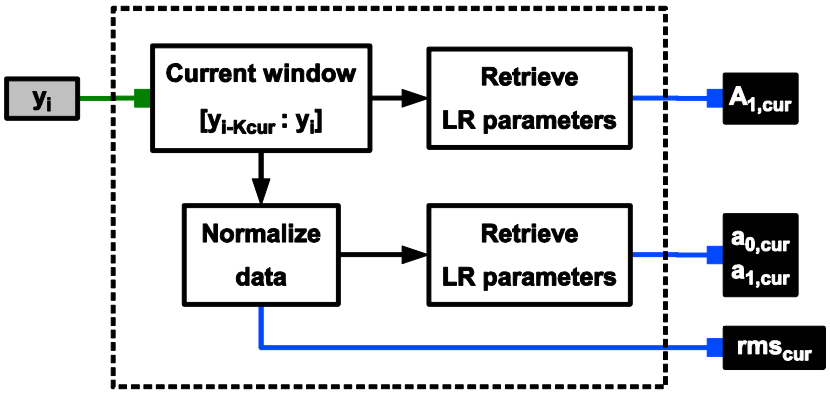

Figure 11. Window Analysis Module.

3.4.1. The Current Window

This data window is a fixed-length time series, much as the one in Figure 8. It has a span of K samples, correspondent to a pre-chosen reference time – T_{ref}.

This reference time is one of the most important input parameters for the CDA. If we see the window linearization process that follows as a low-pass filter for incoming data, attenuating high-frequency events then T_{ref} is its cutoff period.

The cutoff period would be the inverse of the cutoff frequency, which is defined as being the boundary in a system's frequency response at which energy levels start to be attenuated. This means that short-time fluctuations in the acquired data are damped which happens to be a standard procedure in signal processing methods.

According to Figure 11, this current window is rewritten each time a new sample arrives.

3.4.2. Window normalization

On the beginning of the algorithm's development, one idea that was settled was that it should be independent of the readings amplitude. Being so, normalization of the data window became a regular process. This way, the quantitative evaluation of trend changes is merely relative and value-independent.

In this sub-module both the average – $\overline{y}$, defined in (19) – and the root-mean-square of the current window, here called rms_{cur} (17), are calculated.

$$\overline{y} = \frac{\sum_j^K y_j}{K} \tag{19}$$

The window is normalized when all its points are normalized according to the following equation:

$$n_j = \frac{y_j - \overline{y}}{rms_{cur}}, j \in [1:K] \tag{20}$$

Being n_j the normalized value correspondent to the j[th] sample inside.

3.4.3. Window linearization

The LR model is set according to (13). The determination of a_0 and a_1 is done through the least-squares method. Considering the j[th] sample (y_j) and its correspondent abscissa x_j, the LR parameters are retrieved following the next equations:

$$a_1 = \frac{\sum_j^K x_j \cdot y_j - \frac{1}{K}\sum_j^K x_j \cdot \sum_j^K y_j}{\sum_{\boldsymbol{j}}^{\boldsymbol{K}} {x_j}^2 - \frac{1}{K}\left(\sum_j^K x_j\right)^2}, j \in [1:K] \tag{21}$$

$$a_0 = \bar{y} + a_1 \overline{x} \tag{22}$$

Notice that, in Figure 11, there are two slopes being output. $A_{1,cur}$ is the immediate trend of the original signal, updated at the same rate of sample writing and displayed to the user. The variable $a_{1,cur}$ is the slope of the normalized window which is used only for internal calculation the same way as $a_{0,cur}$.

3.5. Predicting Module

In this module the process of predicting a new point is quite straightforward. In the approach by Charbonnier et al. (2004), the error is calculated between the real acquired value and a predicted one (section 3.1). Here, the error is calculated using two predicted points, which means two sets of LR parameters from distinct data windows.

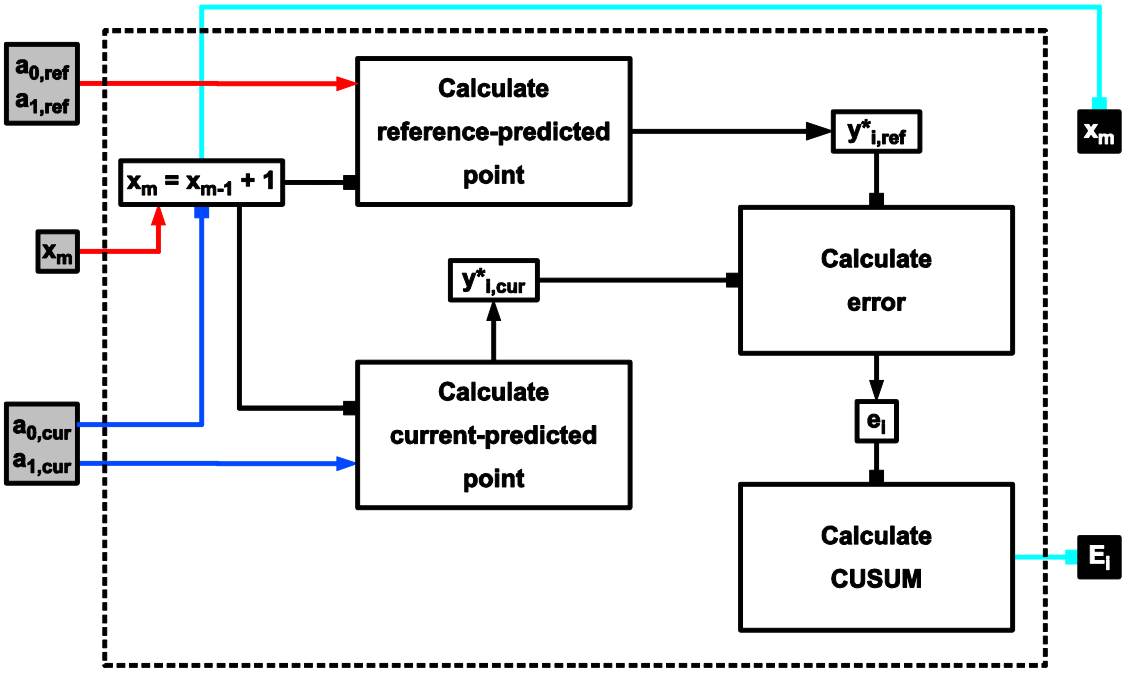

Figure 12. Predicting Module.

The calculation of predicted points uses a shared internal abscissa (x_m) to obtain the following:

- $y^*_{i,ref}$ – reference-predicted point.
- $y^*_{i,cur}$ – current-predicted point.

The index ref should be not confused with the one used in the CDA's input parameter T_{ref}. The variables $a_{0,ref}$ and $a_{1,ref}$ come from a reference LR model obtained

in the Post-CUSUM module and will be explained in 3.6. They are calculated the same way as $a_{0,cur}$ and $a_{1,cur}$, using equations (21) and (22).

The newly predicted points are determined by adapting the LR parameters mentioned to the linear equation (13).

The error and CUSUM are computed from equations (14) and (15), respectively.

In Figure 13, the difference between both predicted points is shown:

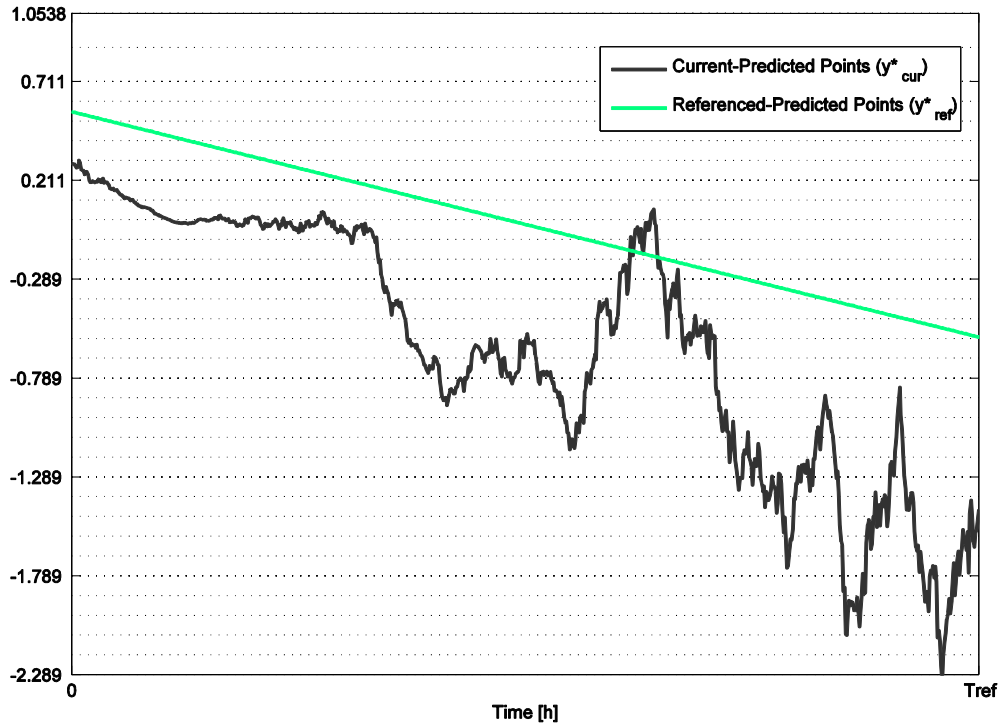

Figure 13. Reference and current-predicted points.

The difference between both plotted functions is justified by the rate of the LR parameter's update. Since $a_{0,cur}$ and $a_{1,cur}$ are computed every time a new sample is acquired, the current-predicted points' evolution is more "unstable" than the reference-predicted, whose parameters are obtained after longer periods, as described in the next section.

3.6. Post-CUSUM Module

In Figure 13 there are stretches where the current-predicted points deviate considerably from the reference-predicted ones. According to (15), this will have a direct impact on CUSUM's calculation, making it grow.

The standard CUSUM algorithm's approach needs to set two thresholds – Th_1 and Th_2 (Figure 14). This way, for each raw value y_i there is a correspondent CUSUM condition towards the established LR model:

- $0 \leq E_i < Th_1$ – Acceptable.
- $Th_1 \leq E_i < Th_2$ – Warning.
- $Th_2 \leq E_i$ – Critical.

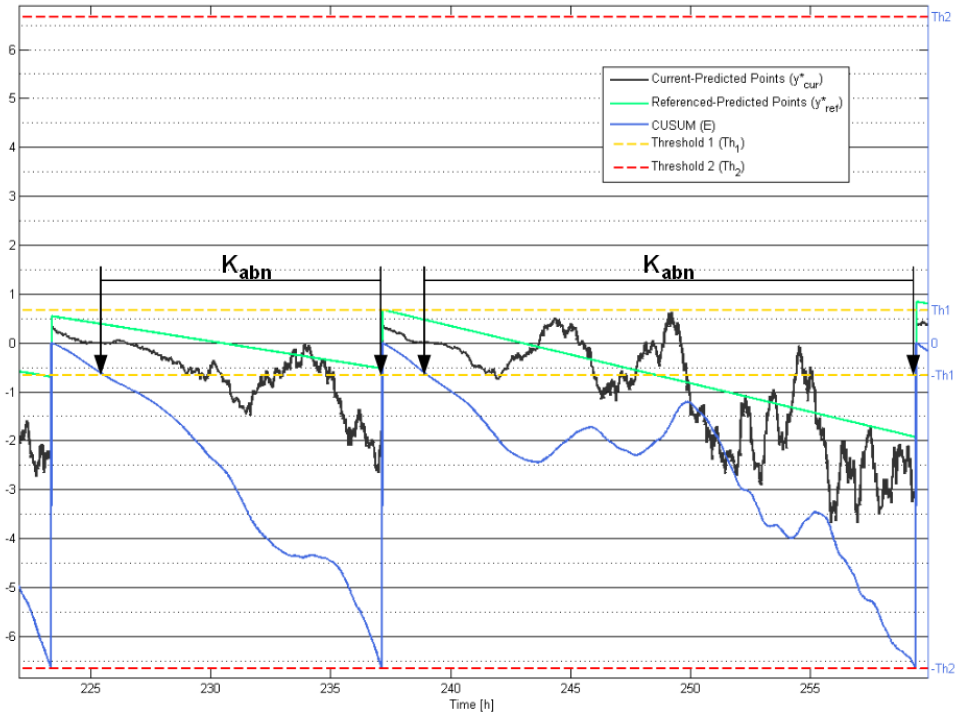

Figure 14. CUSUM evolution.

According to the Figure 14, when E_i reaches the critical state it is reset so that a new LR model can be established. The model is set according to the procedure detailed in section 3.4, with a difference – the length of the data window is variable (K_{abn}).

Following Charbonnier et al. (2004) method, the values used for establishing the new model are the ones that pair with CUSUM when it evolves between "warning" and "critical" conditions – therefore the plotted abnormal values in Figure 14 which are coincident with E_i going through this area.

With this in mind, the Post-CUSUM Module's flowchart is presented in Figure 15:

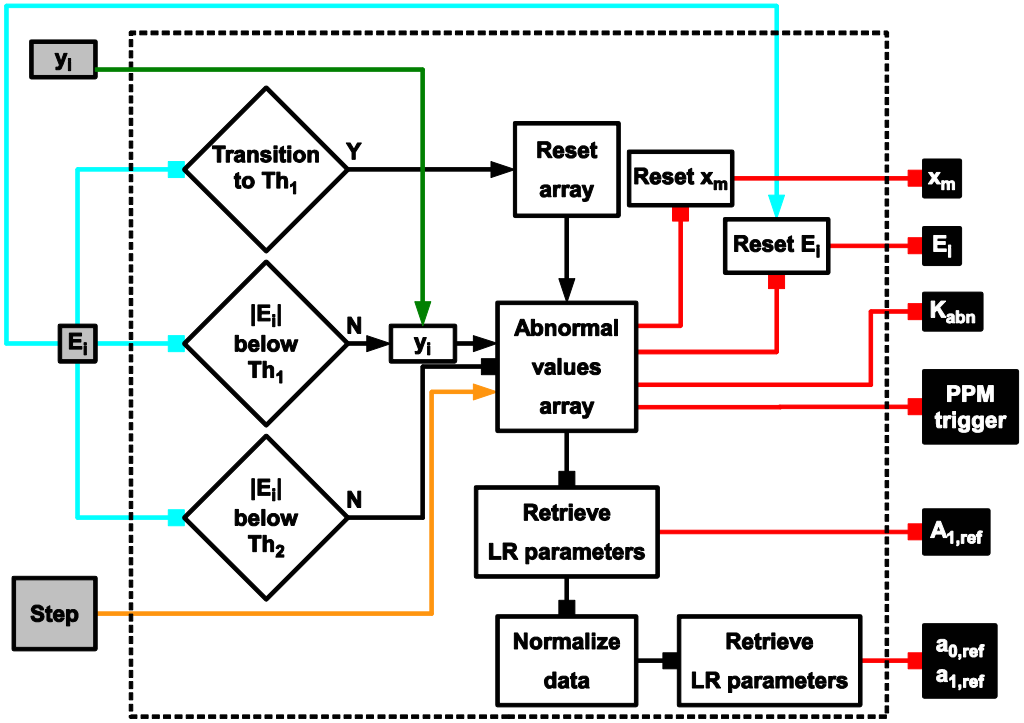

Figure 15. Post-CUSUM Module.

Once E_i surpasses its second threshold the abnormal values array is "released" for normalization and linearization.

Notice that, in the upper decision block, the abnormal values array is reset upon transition to "warning" condition. This is an internal procedure only to ensure that former abnormal values that never reached Th_2 are not incorporated in the new window.

Again, $A_{1,\mathrm{ref}}$ represents the window's real slope to be displayed, $a_{1,\mathrm{ref}}$ and $a_{0,\mathrm{ref}}$ the LR parameters from the normalized window. These two will be used to establish a new model inside the Predicting Module, as referred.

The occurrence of this change in CUSUM's condition also triggers another set of events, mainly two:

- The resetting of the Predicting Module's abscissa (x_m);
- The setting of Post-Process access ("PPM trigger").

At this point all quantitative evaluation of the signal has been performed. Further considerations about this are done later in section 3.8.

3.7. Post-Processing Module

This is the module responsible for the qualitative evaluation of the signal. After its process, the algorithm should be able to output an evolution state differed in eight levels, as presented in the results annexed – "Step Down", "Changing (Decreasing)", "No Change (Decreasing)", "Stable", "Stabilizing", "No Change (Increasing)", "Changing (Increasing)" and "Step Up".

To simplify this analysis, the signal's trend sign will be ignored for now since, for the most part of its operation, the module works with absolute slope values. Therefore, we can define five main evolution states:

1. Step;
2. Changing;
3. No Change;
4. Stabilizing;
5. Stable.

3.7.1. Decision Quadrant

The update of the signal's evolution state can be done using a quadrant as shown in the figures below. In this quarter of a circle, a maximum of 4 evolution states can be defined but, depending on the previous state, one or more state zones can be suppressed.

As mentioned before, at this point only the absolute value of the incoming reference slope, retrieved from the previous module, is being looked at. Considering the example of a high slope signal that will stabilize after some time, all possibilities can be explored.

The next figure relates to the quadrant's phase that is applied after both "Changing" and "No Changing" state.

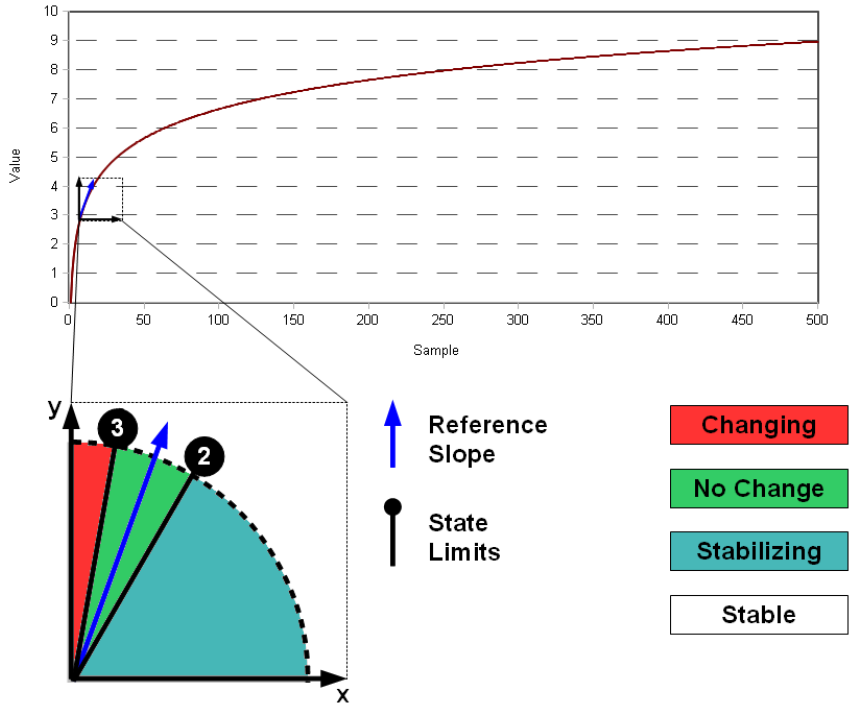

Figure 16. "Changing"/"No Change" state quadrant.

At first instance, the quadrant's reference ($A_{1,Q}$ - Figure 19) is set and all state limits defined, according to the diagram in Figure 16. The "No Change" zone lies between limits 2 (Q_2) and 3 (Q_3), which are defined through a relative threshold (Th_Q):

$$\begin{cases} Q_2 = |A_{1,Q}| \cdot Th_Q \\ Q_3 = |A_{1,Q}| \cdot (2 - Th_Q) \end{cases}, 0 < Th_Q < 1 \quad (23)$$

According to (23), Th_Q defines the limits for the "Stabilizing" state, also meaning that the next processed slope value can be considered in an "unchanged" if inside an interval of $A_{1,Q} \pm A_{1,Q} \cdot (1 - Th_Q)$.

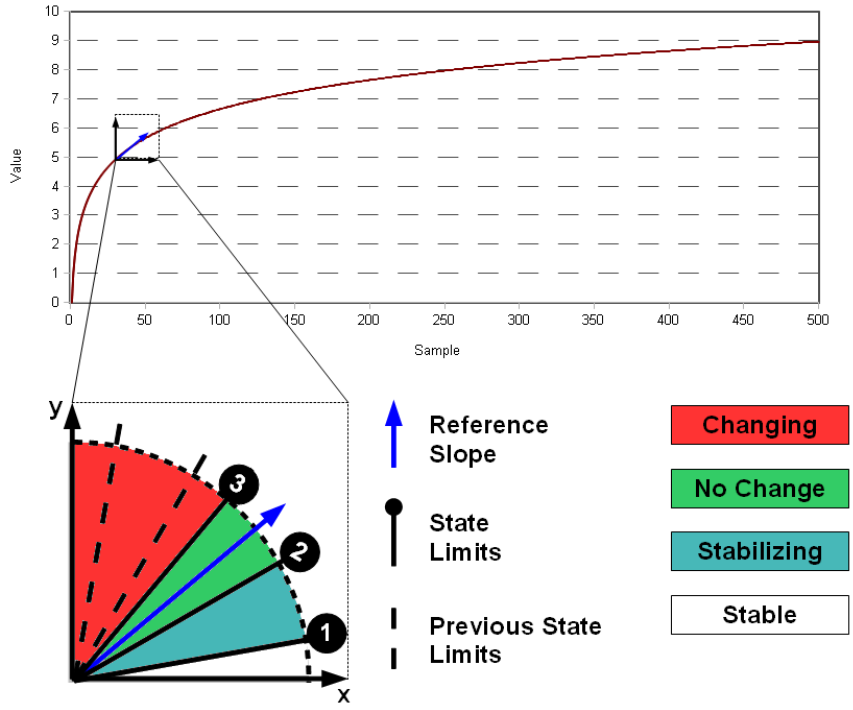

Figure 17. "Stabilizing" state quadrant.

From the state shown in Figure 16, a "Changing" warning will be issued if the slope leaves the zone delimited by Q_2 and Q_3. In this example, it will deviate to the previous "Stabilizing" area.

As is seen in Figure 17, the previous quadrant's reference changed and the state limits redefined according to (23). Because now the signal state is "Stabilizing", the "Stable" zone becomes available to access. Q_1 is simply defined as being equal to the stable threshold (Th_0) and this is the only parameter in CDA that is value-dependent.

Notice that if the trend continues to drop eventually Q_2 will fall below Q_1. In this case the "Stabilizing" zone will be narrowed, having Q_1 as its lower limit.

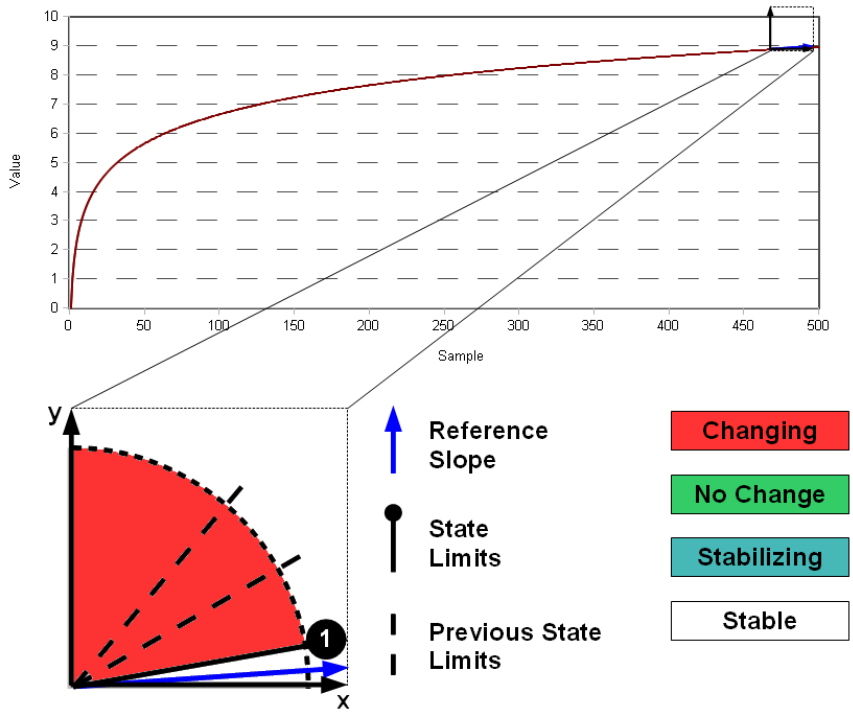

Figure 18. "Stable" state quadrant.

When the signal reaches a "Stable" state, its only options are to remain "Stable" or return to a "Changing" status, therefore all other zones in the quadrant are suppressed, as shown in Figure 18.

These state limits define the quadrant array (Q – Figure 19) that retains the information about their position every time the Post-Processing Module is called to into process.

Notice that no case for step-change has been mentioned so far. The fact is that step-changes occur faster than regular reference slope analysis, which means that a change to "Step" state overcomes any other and it redefines the same quadrant display as in "Changing"/"No Change" state (Figure 16).

As referred before, the quadrant only works with absolute slope values but its analysis is only valid as long as the general data trend (negative/positive) remains the same. If two consecutive reference slopes are different in sign, then the decision quadrant is reset, that is the signal is brought back to a "Changing" state (Figure 16).

In the following section the effective determination of $A_{1,Q}$ is fully explained.

3.7.2. Quadrant's Reference

One could think that the quadrant's reference could be the same as the reference slope ($A_{1,ref}$) determined previously in the Post-CUSUM Module. Theoretically that would be true, but run tests have proved that another criterion has to be established in order to avoid incorrect state evaluations.

The main problem concerns the fact that sometimes current and reference trends are too different. Knowing that one is based on a fixed-length window and the other on a variable and unpredictable one, this event mostly happens due to the frequent irregular shape of acquired signals, so the issues stands in deciding which one to choose.

Figure 19 shows the flowchart of the Post-Processing Module:

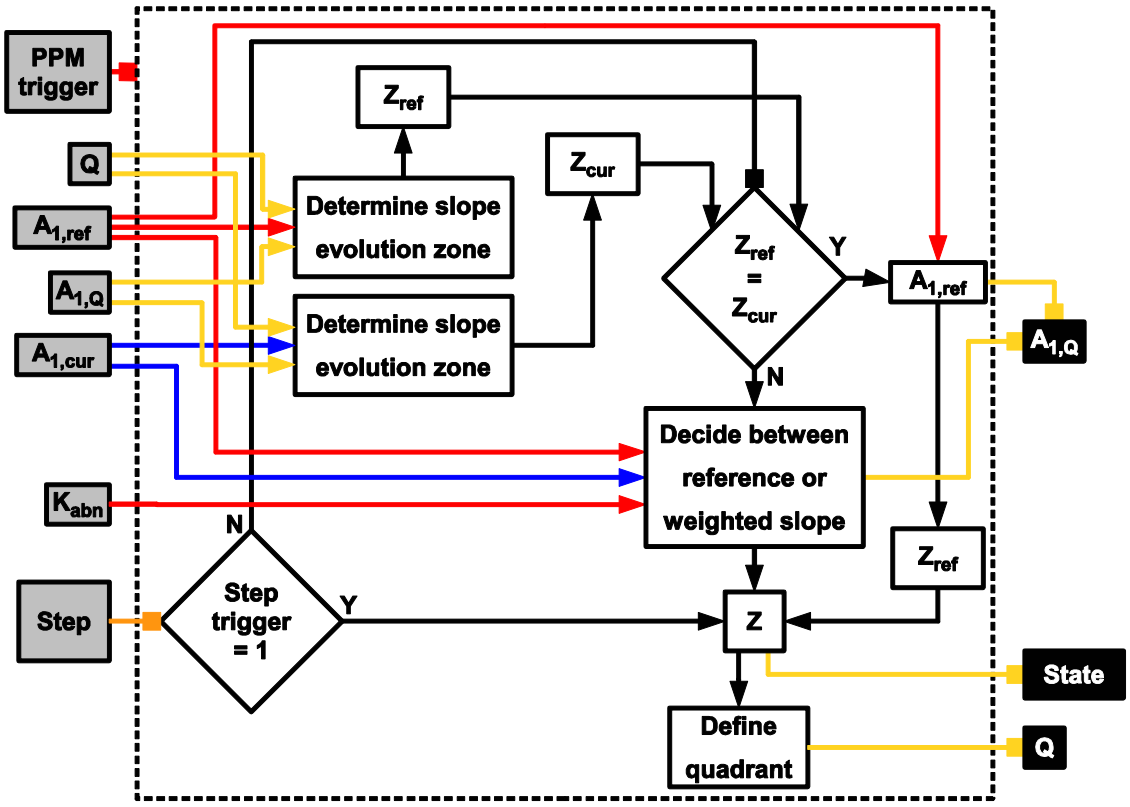

Figure 19. Post-Processing Module.

Inside this module, a decision block compares the position of both slopes in the quadrant (Z_{ref} and Z_{cur}). If their position match, then the reference slope ($A_{1,ref}$) is set as the new $A_{1,Q}$. Otherwise, the quadrant's reference must be a weighted value between $A_{1,ref}$ and $A_{1,cur}$:

$$A_{1,Q} = k_1 \cdot A_{1,ref} + k_2 \cdot A_{1,cur} \quad (24)$$

The values of the k_1 and k_2 weights should then reflect the amount of points – K and K_{abn} – used for the calculation of their respective slopes:

$$\begin{cases} k_2 = \dfrac{K}{K + K_{abn}} \\ k_1 = 1 - k_2 \end{cases} \tag{25}$$

With this method the proper contribution of each data window (reference and current) is taken into account. This way, if either $A_{1,ref}$ and $A_{1,cur}$ were retrieved from a small number of samples in comparison to the other, its effect, considered to be a punctual abnormality in trend, will be minimized in the final state evaluation.

In case of a recent step-change occurrence, this process has to be skipped and $A_{1,ref}$ chosen as the quadrant's reference. This happens because the current window still contains the step-change points which increase the magnitude of the current slope after linearization, therefore leading to an erroneous $A_{1,Q}$ value if determined by (24).

3.8. Input/Output Parameters

Having into consideration what was exposed previously, the following tables set some considerations about the input and output parameters of the Change-Detection Algorithm:

Table 1. CDA's input/output parameters.

Parameter	Meaning	Remarks
Input		
$\boldsymbol{T_{ref}}$	Reference time length (For the current window).	This value sets the time span of the current window. Except for step-changes, no target events should be shorter than this.
$\boldsymbol{Th_Q}$	Evolution quadrant's relative threshold	According to section 3.7.1, this parameter is used to define the "unchanged" area where the signal's trend can dwell.
$\boldsymbol{Th_0}$	"Stable" state threshold	It is an absolute value and the only value-dependent parameter. that defines the trend limit from below which the signal is considered to be stable. It should be set to avoid unnecessary warnings small signal fluctuations.
$\boldsymbol{Th_1}$	CUSUM's "warning" condition threshold.	These parameters set the regions of CUSUM's condition development (Figure 14). The larger these are, the longer the cumulative sum needs to mark a new reference, meaning that evaluation will be performed less often.
$\boldsymbol{Th_2}$	CUSUM's "critical" condition threshold.	
Output		
$\boldsymbol{A_{1,cur}}$	Current Trend	This is the slope retrieved from the current window. It can be displayed as "the trend in the last T_{ref} time period"
$\boldsymbol{A_{1,ref}}$	Reference Trend	This is the slope retrieved from the analysis of the abnormal values' It is always reflective of the true change occurred in the system.
State	Signal's state	This is the variable responsible for the issuing of warnings. Warnings are sent upon step-change occurrence, "changing" and also "stabilizing" states.

4. EXPERIMENTAL RESULTS

In order to diagnose correctly oil changes during machine work, it is necessary to relate monitored parameters with a specific known occurrence. The algorithm described before will be tested with the data retrieved from some experiments that were conducted in a laboratorial environment, using (for every case) a reference oil with the following description:

- Olma d.d. Olmaredol VG-68 – reference viscosity (μ_{ref}) = 68 mm^2/s at 40 ºC.

4.1. Experimental Setup Description

Figure 20 is an actual image of the experimental assembly:

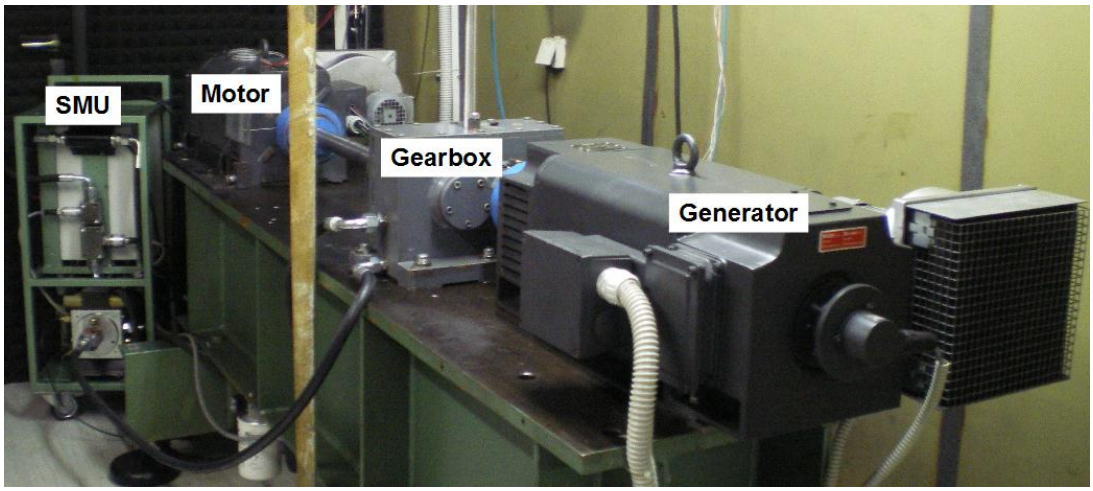

Figure 20. The experimental assembly.

On this experimental setup, torque is transmitted from the Synchronous Electric **Motor** to a **Brake-Generator**. The **Gearbox** connected to the **SMU** (described in 4.1.1), will be duly sealed after 2 liters of unchanged oil are poured inside. The **Sensors' Block** (Figure 21) will be outputting an analog signal to the **DAQ Card** (Figure 21) which will convert it to a digital one, ready to be registered by the already implemented LabView interface.

Regarding the components of the experimental setup, the following table shows their main characteristics:

Table 2. Experimental setup components characteristics.

Component		Main Characteristics	
Synchronous Electric Motor		Rated Power	12.7 kW
		Rated/Maximum Speed	1470/1700 rpm
Brake-Generator		Rated Power	20.2 kW
		Maximum Torque	110 N.m
DAQ Card	**NI USB 6215 (Isolated Multifunction I/O)**	16 Analog Input Channels (250 kS/s) (with 16-bit analog-to-digital converter)	

4.1.1. SMU's Description

The Standalone Monitoring Unit is responsible for providing data for the analyzed parameters. Figure 21 is a schematic drawing of the experimental setup:

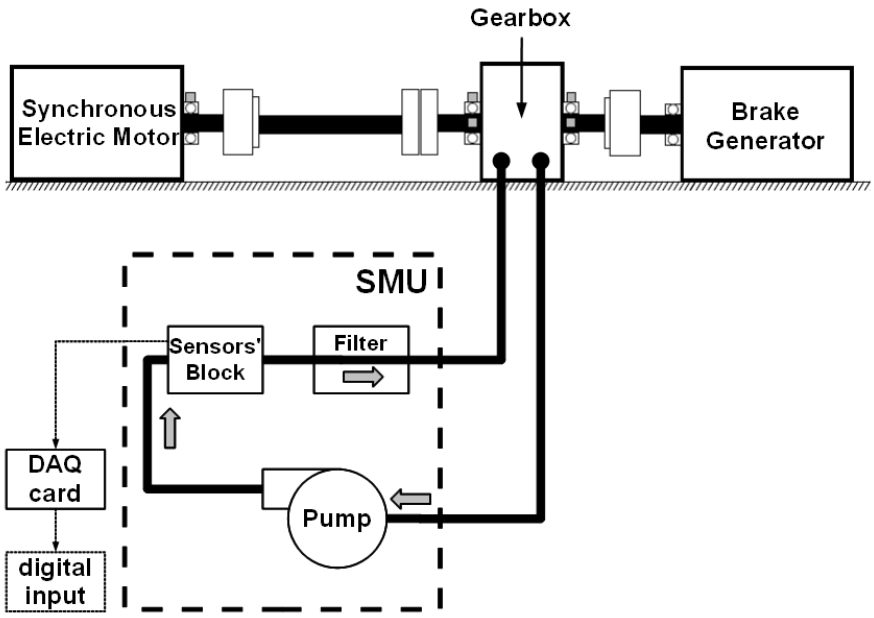

Figure 21. Schematic assembly of the experimental setup.

Designed and developed at CTD, the SMU takes oil from the gearbox and pumps it towards the sensor's block, through the filter and back to the oil container. For on-line assessment of the oil condition, the sensor's block incorporates the following three sensors:

- **HYDACLab® AquaSensor AS 1000** (temperature compensated) – for measuring water content relative to saturation point and oil temperature:
  1. Relative moisture content (0 to 100%);
  2. Temperature (-25 to 100 ºC).
- **HYDACLab® Fluid Condition Sensor** (temperature compensated) – for measuring multiple parameters, including:
  1. Relative change in viscosity (-30 to 30%);
  2. Relative change in dielectric constant (-30 to 30%);
  3. Relative moisture content (0 to 100%);
  4. Temperature (-25 to 100ºC).
- **KITTIWAKE GmbH Metallic Particle Sensor** – to measure and classify (by size) wear metal particles concentration. Size categories include:
  1. Ferrous bins:
     a. 40 to 60 µm;
     b. 61 to 100 µm;
     c. 101 to 200 µm;
     d. 201 to 300 µm;
     e. Larger than 300 µm.
  2. Non-ferrous bins:
     a. 135 to 150 µm;
     b. 151 to 250 µm;
     c. 251 to 350 µm;
     d. 351 to 450 µm;
     e. Larger than 450 µm.

According to the described sensors, two main groups of experiments were performed. These are detailed in the following sections 4.2 and 0.

4.2. Pitting Experiments

The pitting phenomenon is one type of mechanical fault with most interest in the field of technical diagnosis. Due to its frequent occurrence in industrial environments, it is on the interest of maintenance departments to detect the start of this event as soon as possible.

In this sense, two pitting experiments were conducted at CTD with different load conditions – at constant (Figure 22) and variable load (Figure 23). The load values in Figure 22 and Figure 23 come as a percentage of the generator's maximum torque, shown in Table 2.

.

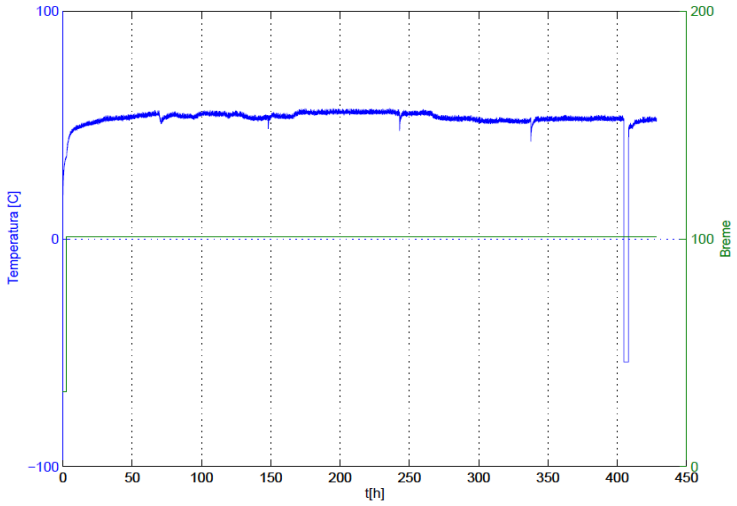

Figure 22. Pitting experiment from 04/02/2011: Temperature and load conditions (Axis: Temperature [ºC], Load [%]) (Peršin, G. 2011a).

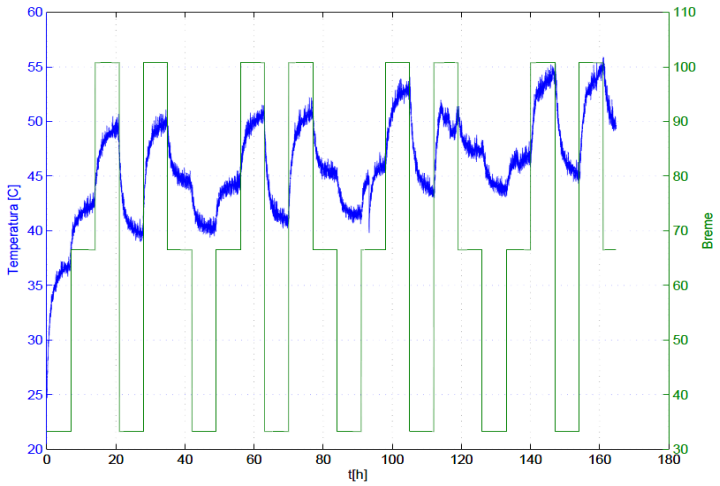

Figure 23. Pitting experiment from 04/02/2011: Temperature and load conditions (Axis: Temperature [ºC], Load [%]) (Peršin, G. 2011b).

Comparing both figures, it can be seen that load variation has a direct impact on oil temperature. Concerning the metal particles count associated with pitting, oil temperature has no impact, but load variation does and it is visible on the CDA's run results.

These load conditions were established and uploaded into the generator's control unit and the motor put to run via remote connection. The motor's speed was set to 1296 rpm in both cases.

The objective was to drive the mounted gears into a fault state and register the important changes in the read signals, that include the ones discussed in 4.1.1.

For this there were used two sets of spur gears, both pairs with a transmission ratio of 1.5 (24 to 16 teeth). The gears are from steel according to designation DIN 42CrMo4.

A change in the nitration process influenced the outer layer thickness. This was determined visually after the tests, through microphotography.

Table 3. Measured outer layer thickness.

Experiment Date	Outer Layer Thickness
04/02/2011	20 µm
03/03/2011	2 – 3 µm

4.3. Contamination Experiments

Oil contamination by alien agents can be most important in cases where equipment cleanliness greatly affects its reliability and life cycle extent. Also, it can be indicative of oil aging status, in case of increased viscosity, or an existent leak in surrounding modules (i.e. coolant/fuel contamination, etc.). Knowing that the presence of these agents mostly affects the lubricant's viscosity, the contamination's response profile for each should be determined.

Water contamination test was divided in three sessions, each starting with contamination of the running oil with different amounts of water – 1ml, 2ml and 5ml.

The 8th of August experiment (section 0) is a repetition of the one performed on the 24th of March where all 3 sessions where complied. Unfortunately, undetected

crash-downs in the acquisition LabView modules lead to the loss of great stretches of data, causing visible discontinuities in registered values. In this sense, it was decided that the previous experiment should be disregarded and a new one performed.

The driving motor was put to work at a speed of 1000 rpm and the torque set to 33% of its maximum torque (Table 2). Contaminants were introduced through an access conduit welded to the covering top of the oil container, allowing for this process to take place without stopping the machine's running.

The obtained results are discussed in detail on section 4.4.2.

4.4. Results

Concerning what was described in sections 4.2 and 0, the results here presented were taken from the KITTIWAKE Metallic Particle Sensor, for metallic particle count (Pitting Experiments), and from the HYDACLab AquaSensor AS 1000 for relative moisture reading (Contamination Experiments).

Although tests with other contaminants were performed, faults in the HYDACLab Fluid Condition Sensor did not allow for viscosity variations to be observed, a setback that will be discussed in section 4.4.2.2 and chapter **Error! Reference source ot found.**.

4.4.1. Pitting Experiments

The pitting phenomenon can be observed in the trends of particle count that is, the velocity of newly released particles to the operating fluid – especially in large-sized ones (above 200 µm). Wear, on the other hand, is mostly expressive in particle sizes below 200 µm and its evolution is more logarithmic-shaped rather than exponential. The two phenomena are visible on both sets of signals obtained.

The input parameters for particle count signals are shown in Table 4.

Table 4. Input parameters for pitting experiments.

Parameter	Value
T_{ref}	8 hours
Th_Q	40 %
Th_0	5 particles/hour
Th_1	50
Th_2	500

4.4.1.1. Wear Confirmed

The test that started on the 4th of February (Figure 24) shows a high rate of initial wear, followed by stabilization as shown by the progress of both output trends.

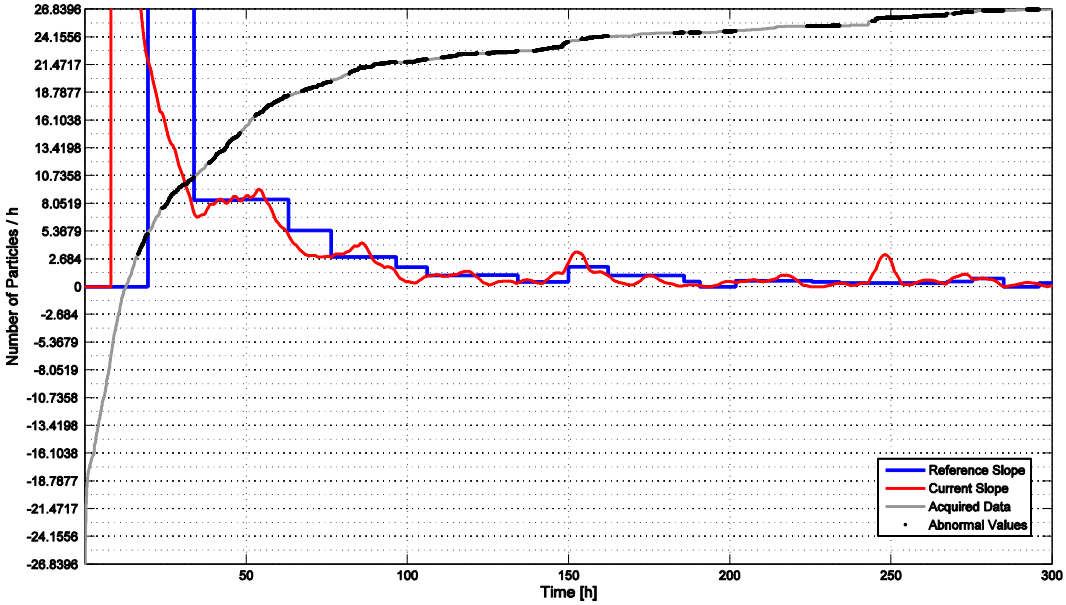

Figure 24. Quantitative evaluation: Number of ferrous particles (101 to 200 µm) (Pitting experiment from 04/02/2011).

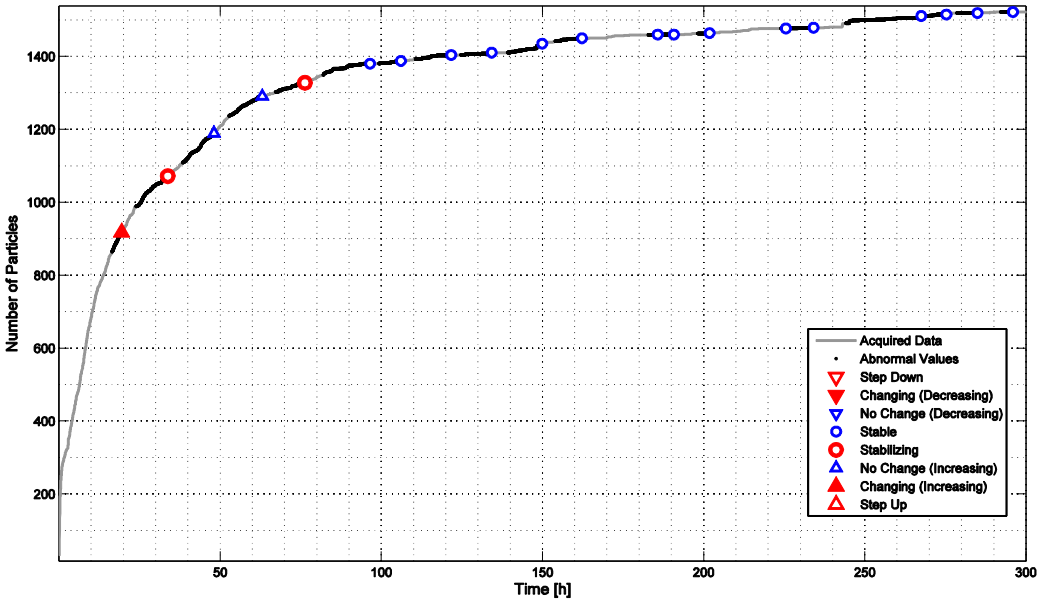

Figure 25. Qualitative evaluation: Number of ferrous particles (101 to 200 µm) (Pitting experiment from 04/02/2011).

Similar results are shown in appendixes from A.1 to A.4.

In line with the CDA's description and its intended progress, a "Changing" state was marked after 20 hours (Figure 25) from the test start and final stabilization reached after 96 hours. This corresponds to an initial rate of more than 30 particles generated per hour before reaching stagnant level below 2. Also, two transitions to "Stabilizing" state were obtained in between, the first marking a small change in trend and the second spotting the end of the initial wear phase.

In this experiment, pitting did not occur, mainly due to the large thickness of the outer layer (20 µm - Table 3). In the next section, the test discussed uses gears with a layer thickness ten times smaller, and this will influence greatly the general particle count.

4.4.1.2. Pitting Confirmed

In the 3rd of March experiment, according to the reports produced at CTD, pitting was confirmed after 72 hours of the test start and the algorithm should pinpoint this with a "Changing" warning.

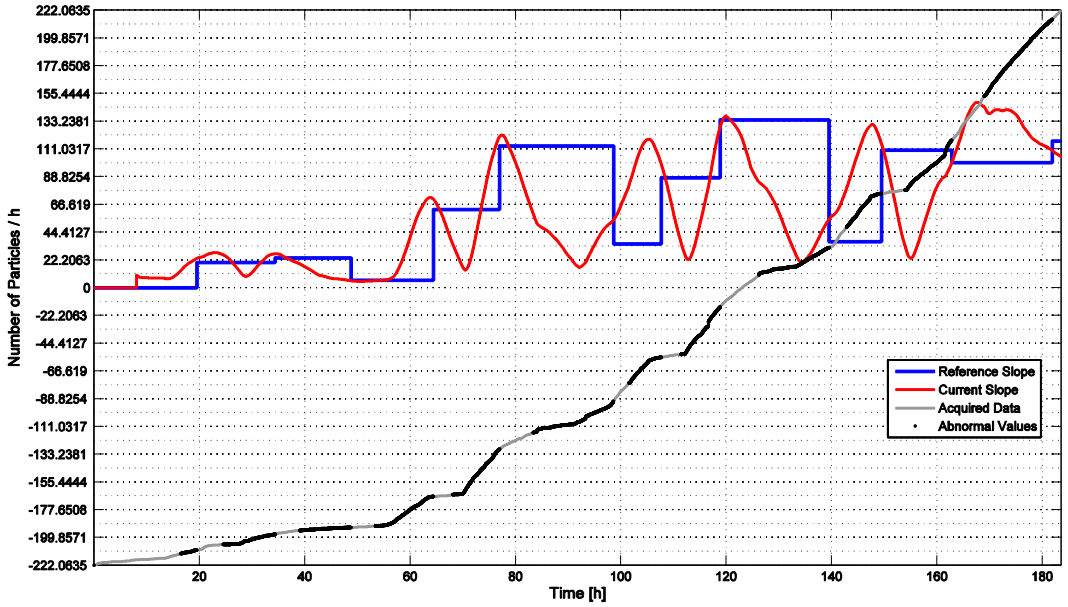

Figure 26. Quantitative evaluation: Number of ferrous particles (101 to 200 µm) (Pitting experiment from 03/03/2011).

The current trend/slope is changing in a cyclic mode due to the load variation (Figure 23), showing peaks which are coincident with the highest stretches of load, as shown in Figure 26. This reflects later on the reference slope output that clearly changed in average after 78 hours. The values registered that followed remained with similar peak values – from 110 to 130 particles/hour, approximately – and keeping the same varying profile as before.

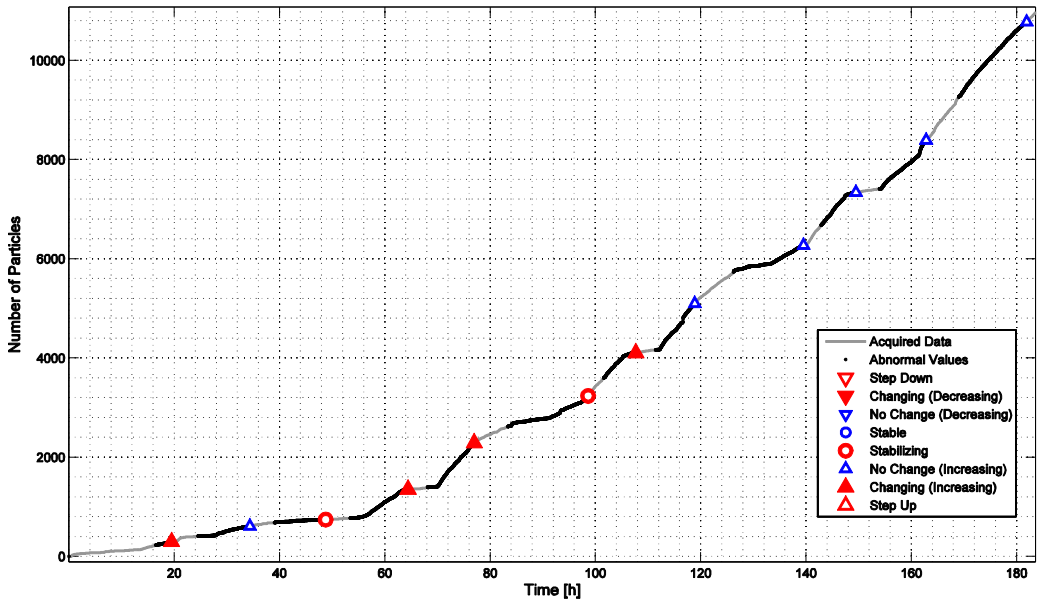

Figure 27. Qualitative evaluation: Number of ferrous particles (101 to 200 µm) (Pitting experiment from 03/03/2011).

The trend's high fluctuations make it difficult to find the pitting start point. In any case, between 60 and 80 hours, two "Changing (Increasing)" warnings were issued (Figure 27), confirming that a real changing in the system's condition was occurring. The same phenomenon can be seen in similar results in appendixes B.1 to B.4, the latest concerning non-ferrous particles.

4.4.2. Contamination Experiments

The results presented in this section have very interesting features from step-changes to high slope fluctuations. Moreover, the signals obtained are not cumulative-type like the previous but stochastic, which reflects on the noisy readings obtained.

Because the reference time is intentionally smaller than the one input on the pitting experiments, state issuing will be more frequent.

For the signals analyzed, the CDA's input parameters are:

Table 5. Input parameters for contamination experiments.

	Relative Moisture	**Temperature**
Parameter	**Value**	
$\boldsymbol{T_{ref}}$	4 hours	4 hours
$\boldsymbol{Th_Q}$	75 %	75 %
$\boldsymbol{Th_0}$	0.01 %/hour	0.15 ºC/hour
$\boldsymbol{Th_1}$	50	50
$\boldsymbol{Th_2}$	500	500

Notice that the CUSUM thresholds values (Th_1 and Th_2 - Table 5) are similar for input parameters used in pitting experiments (Table 4), proving that value-independency has been achieved.

For these contamination parameters, only the "Stable" state threshold is different between both analyses. That is clearly the obvious change in parameters because it is the only one which is value-dependent.

Furthermore, the reference time chosen the same for both comes from the assumption that changes in their condition occur within the same time range and faster in comparison to particles count.

4.4.2.1. Relative Moisture

The contamination of the operating lubricant with water produced immediate "spikes" in the signal's readings, identified right away with a "Step Up" warning issue and followed by a low-paced decrease. This happens because of the long period that even the smallest amount of water (1 ml) needs to disperse completely inside the oil container.

If ideal conditions had been matched, the readings should have stabilized into a new value, but variations in the room's temperature did not allow for this feature to be registered, an aspect which will be discussed further. Even so, in the first 100 hours after the contamination mostly "Stabilizing" states were output, preceded with a "Changing (Decreasing)" warning after the step-change (Figure 29).

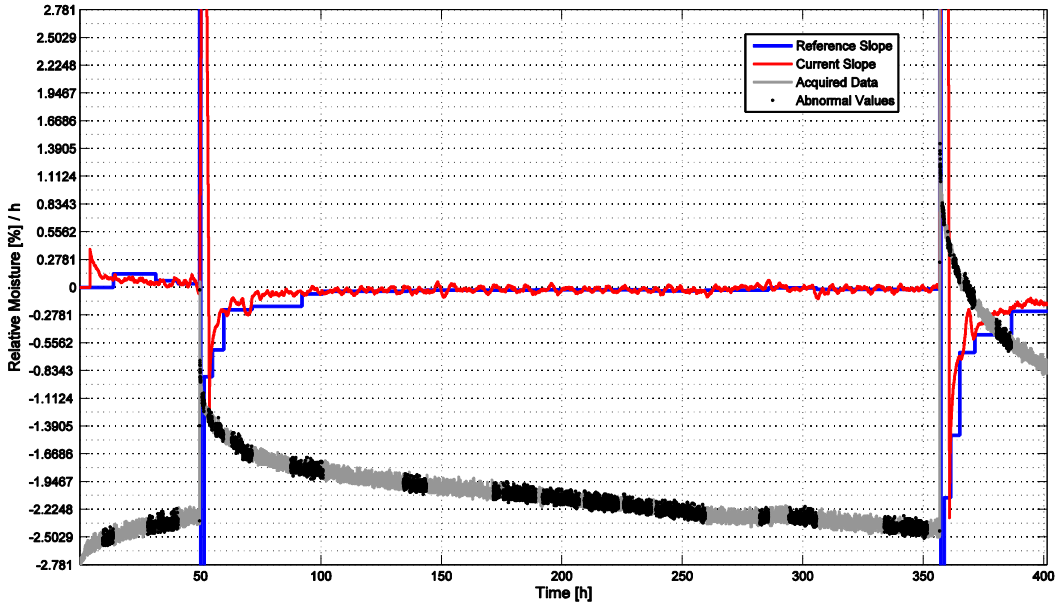

Figure 28. Quantitative evaluation: Relative moisture (Water contamination experiment from 01/08/2011).

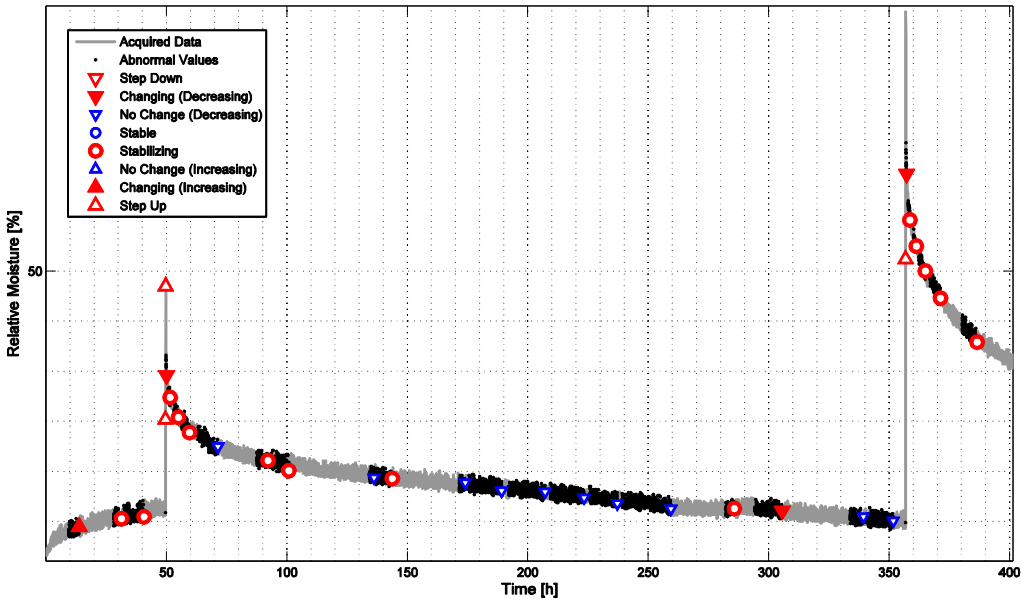

Figure 29. Qualitative evaluation: Relative moisture (Water contamination experiment from 01/08/2011).

After 170 hours of running experiment a permanent "No Change" state was achieved, being only disturbed briefly at the 285th hour before the signal continued its constant descent. Although the rate of change is very low (0.2 %/hour - Figure 28), it is clear on this graph that the signal is still decreasing, therefore this state should be maintained.

4.4.2.2. Temperature

As it was referred on the beginning of the previous section, temperature fluctuations inside the room had an impact in relative moisture readings. These can be seen now in the following Figure 30 and Figure 31:

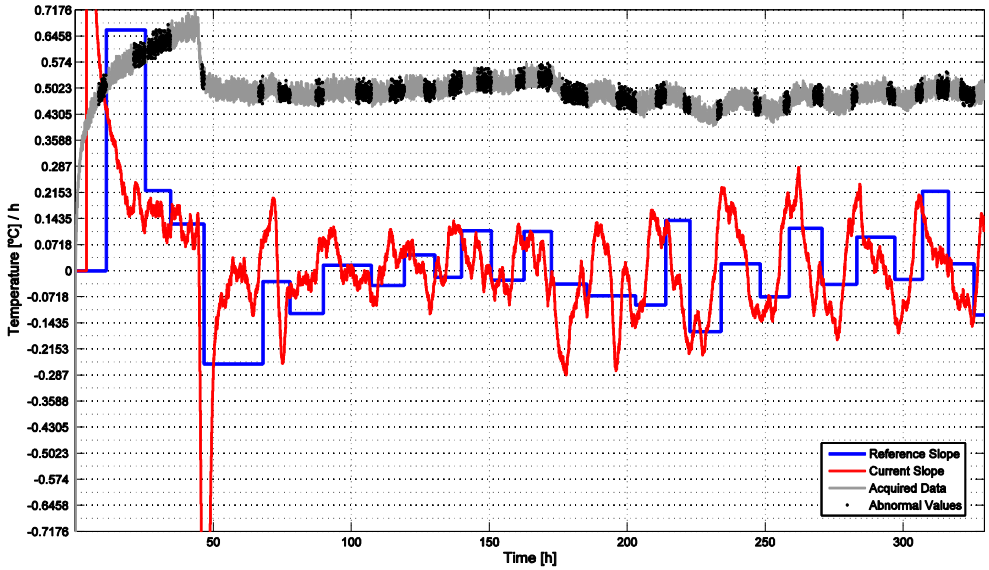

Figure 30. Quantitative evaluation: Temperature (Water contamination experiment from 01/08/2011).

After the initial temperature rising and stabilization a high decrease was registered (Figure 30). At this time, it was decided that the increasing heat could jeopardize the safety of the running machine, therefore measures were taken to ensure proper cooling flow in the room. Later, new problems concerning readings stability were brought by the natural day temperature variations.

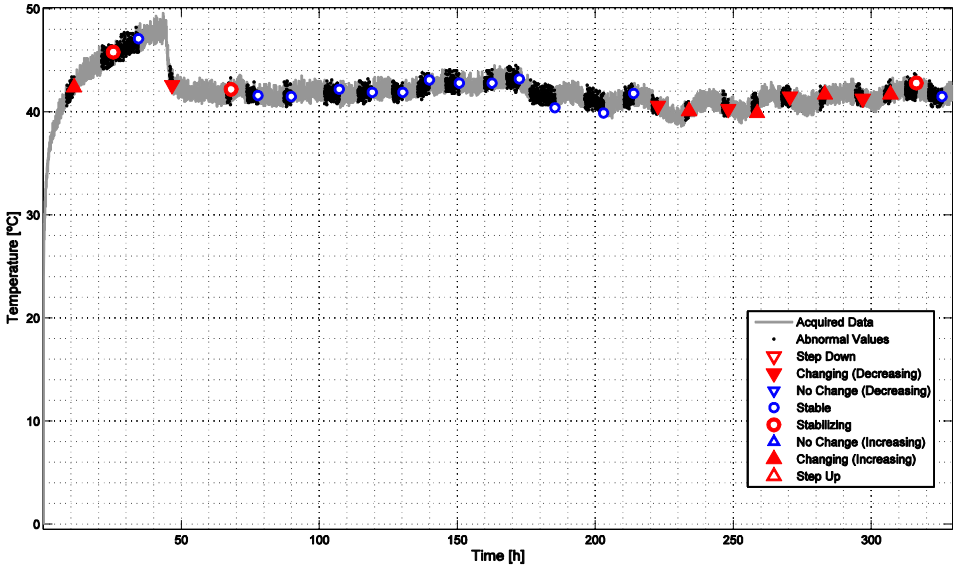

Figure 31. Qualitative evaluation: Temperature (Water contamination experiment from 01/08/2011).

Figure 31 shows how unstable trend evolution can be when dealing with these daily temperature variations. A "Stable" set of states was correctly issued, but 220 hours after the experiment's start, cycle amplitudes became larger causing alternate "Changing" states to be output.

In Appendix C, the temperature readings and results from the 4th of April experiment – contamination with different oil - are presented. This test lasted for little more than 41 hours, moment when the test was put to stop.

The contaminator, Olma d.d. Gear Oil SAE 80W-90 (μ_1 = 90 mm^2/s at 40 ºC), was not producing any variation in viscosity readings and for this reason, even after an attempt to recalibrate the sensor, is was concluded that it was unable to deliver the necessary parameters.

In any case, the data retrieved was kept for testing the algorithm's run progress and to compare with the results from the 8th of August experiment.

5. CONCLUSIONS

In line with the proposed objectives detailed in chapter 1, a change-detection algorithm was developed, capable of outputting trend and evolution state from obtained signals on-line. In its development process, value-independency was achieved which made it possible to reduce the number of input parameters, simplifying future implementation in a multi-parameter diagnosis unit.

The algorithm was tested with retrieved data from different lubricant parameters with very distinguished behavior. Despite the setbacks already referred, enough data was obtained to validate its consistency.

It performed particularly well when confronted with cumulative-type signals in metal particles count, identifying all stages in initial wear evolution (4^{th} of February experiment) and spotting in definite the occurrence of pitting 8 hours after the moment it was confirmed by observation of the progress of gear-mesh frequency amplitude (3^{th} of March experiment).

Run tests were performed on vibrational features but only to assess the consistency of the algorithm. In order to evaluate CDA's performance in pitting detection, it would be necessary to achieve this fault with constant load conditions, which was not possible. The high variations in load, especially in gear-mesh frequency amplitude which is indicative of pitting phenomenon, did not allow for the signal to have a continuous evolution. In this sense, it would be wise develop a programmed module to shift between historical data arrays every time the load conditions change. If these changes are not scheduled that is, consequence of a malfunction, then the algorithm should naturally issue the appropriate warning.

New tests regarding oil contamination parameters should be performed in a properly controlled environment in order to register the signal's evolution in full. On the other hand, it must be kept in mind that in industrial environment the influence of external factors is always present, something that the diagnosis process should be prepared to deal with.

Concerning the faults occurred in the experiments of 24th of March and 4th of April, the most decisive was the loss of viscosity readings. Without this parameter, the presence of fuel, wrong lubricant or cooling fluid in the reference oil could not be observed. Consequently, the initially pretended variety of contamination response profiles wasn't obtained.

Here, at Center for Tribology and Technical Diagnosis, we feel that the developed algorithm is solid and ready to be integrated in automated signal diagnosis. The fact that it uses simple mathematical operations makes it light, as intended to be, and therefore suitable to be implemented in a multi-parameter evaluation process. Although trend is an important feature, its interpretation is dependent on experience and needs tests to establish appropriate critical thresholds. Still, the combination of states between different parameters can be indicative of a specific fault, from which the final assessment would be refined with confirmation through the rates of evolution.

6. BIBLIOGRAPHY

Basseville M. and Nikiforov (1993), I. V., "Detection of Abrupt Changes: Theory and Application", Prentice-Hall.

Borghers, E. and Wessa, P. (2011), Statistics - Econometrics – Forecasting (Statistical Distributions), Office for Research Development and Education. Accessed in March 5, 2011, in: http://www.xycoon.com/.

Cao, S. and Rhinehart R. (1995), "An efficient method for on-line identification of steady state", Journal of Process Control, 5 (1995), 367-374.

Charbonnier, S., Garcia-Beltan, C., Cadet, C. and Gentil, S. (2004), "Trends extraction and analysis for complex system monitoring and decision support", Engineering Applications of Artificial Intelligence, 18 (2005), 21-36.

Isom, J. D. (2009), "Exact solution of Bayes and minimax change-detection problems", Dissertation submitted for the degree of Doctor of Philosophy in Chemical Engineering, University of Illinois, Urbana, Illinois.

Jiang, T., Chen, B., He, X. and Stuart, P. (2002), "Application of steady-state detection method based on wavelet transform", Computers and Chemical Engineering, 27 (2003), 569-578.

Kalai, A. and Vempala, S. (2004), "Efficient algorithms for online decision problems", Journal of Computer and System Sciences, 71 (2005), 291-307.

Onatski, A. (2000), "Minimax Analysis of monetary policy under model uncertainty", Economics Department of Harvard University.

Peršin, G. (2011a), "Testiranje zobnikov za jamičenje (pitting) pri 100% obremenitvi", Technical report, CTD, Faculty of Mechanical Engineering, University of Ljubljana, February 2011.

Peršin, G. (2011b), "Gear pitting at load torque variation 33%, 66% and 100%", Technical report, CTD, Faculty of Mechanical Engineering, University of Ljubljana, March 2011.

Peršin, G., Kržan, B. and Vižintin, J. (2010), "The potential of signal processing for the purpose of machine oil diagnosis", The Seventh Conference on Condition Monitoring and Machinery Failure Prevention Technologies, Ettington Chase, Stratford-upon-Avon, England, 22 – 24 June 2010.

Riggs, James L. (1982), "Engineering Economics", 2nd Ed., McGraw-Hill.

Robin, L. (2006), "Slick tricks in oil analysis". Accessed in March 2, 2011, in: http://www.plantservices.com/articles/2006/212.html.

Tambouratzis, T. and Antonopoulos-Domis, M. (2004), "On-line signal trend identification", Annals of Nuclear Energy, 31 (2004), 1541–1553.

Vaswani, N. (2005), "The modified CUSUM algorithm for slow and drastic change detection in general HMMs with unknown change parameters", Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '05), 4, iv/704.

Zhao, Z., Wang, F., Jia, M. and Wang, S. (2005), "Predictive maintenance policy based on process data", Mechanical Systems and Signal Processing, 21 (2007), 208–233.

APPENDIX A.1

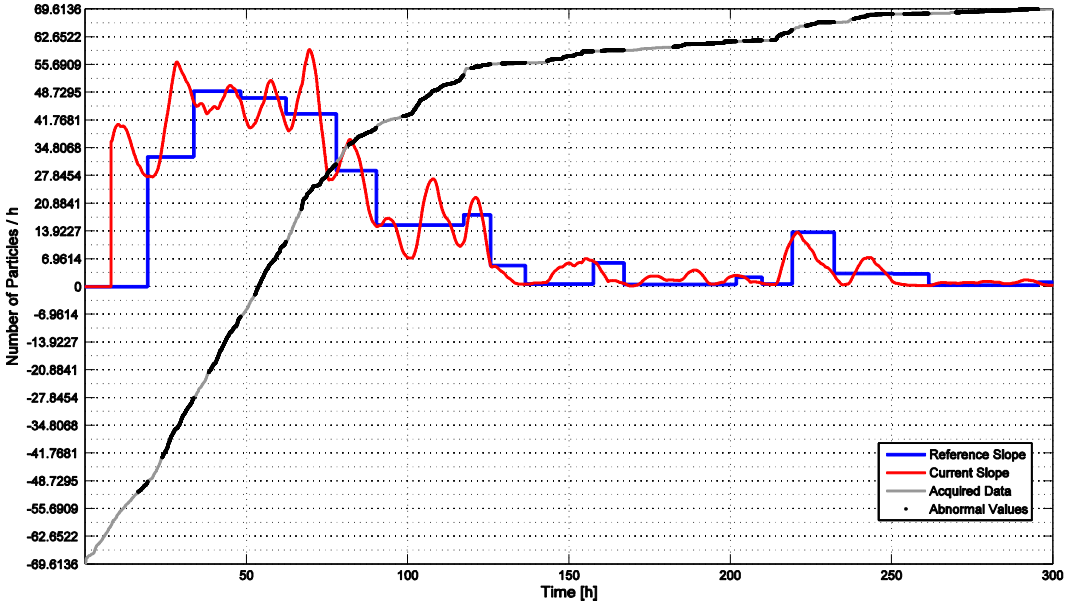

Figure 32. Quantitative evaluation: Number of ferrous particles (40 to 60 µm) (Pitting experiment from 04/02/2011).

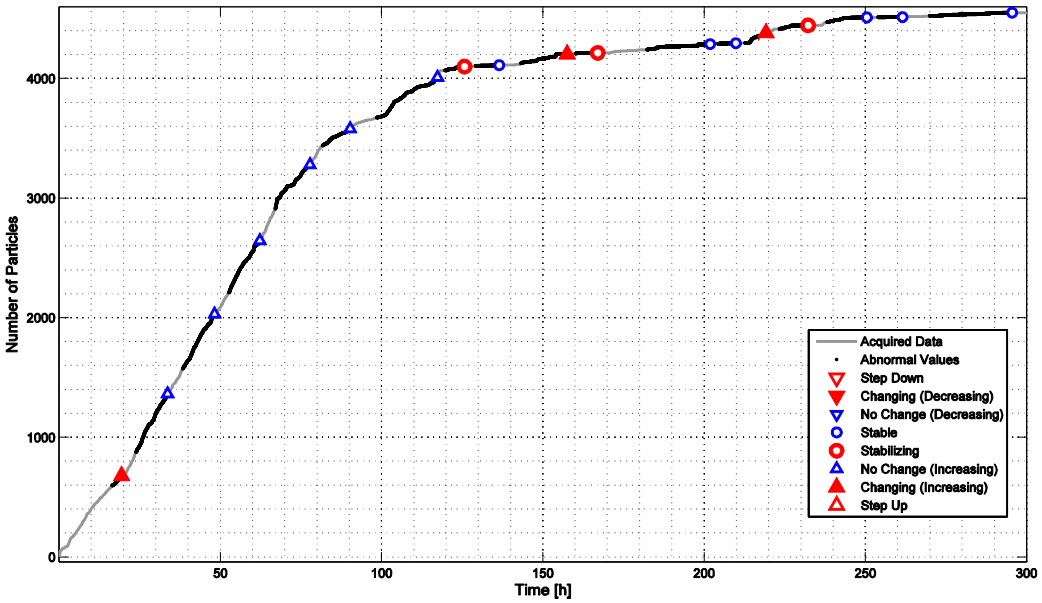

Figure 33. Qualitative evaluation: Number of ferrous particles (40 to 60 µm) (Pitting experiment from 04/02/2011).

APPENDIX A.2

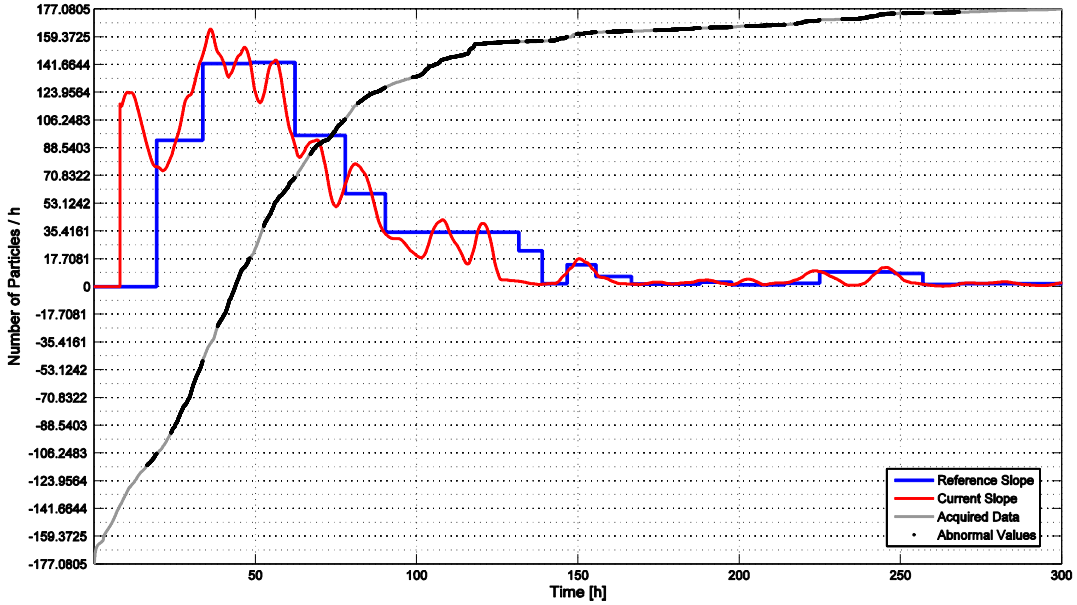

Figure 34. Quantitative evaluation: Number of ferrous particles (61 to 100 µm) (Pitting experiment from 04/02/2011).

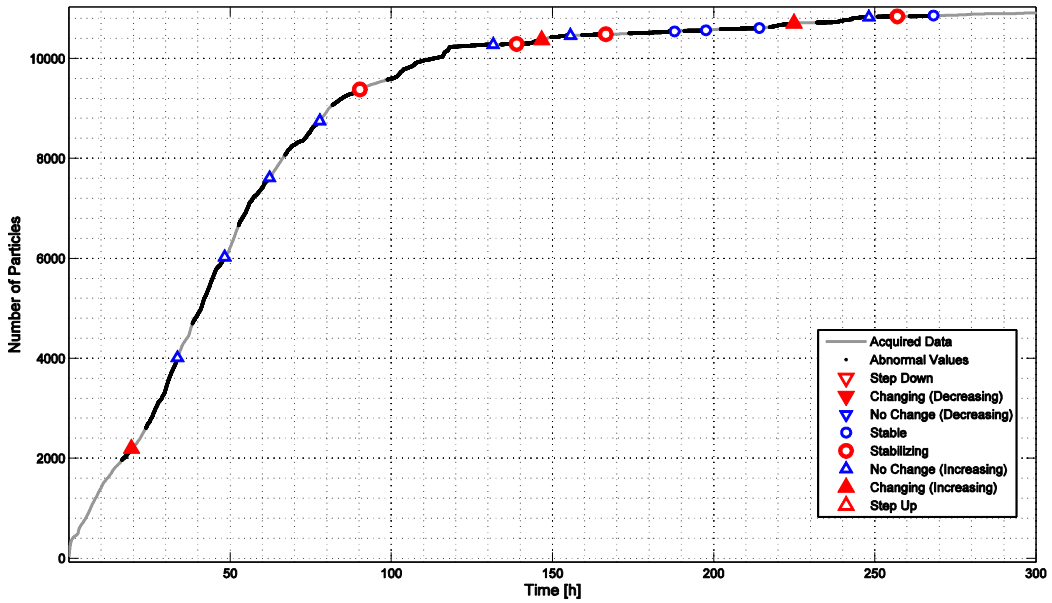

Figure 35. Qualitative evaluation: Number of ferrous particles (61 to 100 µm) (Pitting experiment from 04/02/2011).

APPENDIX A.3

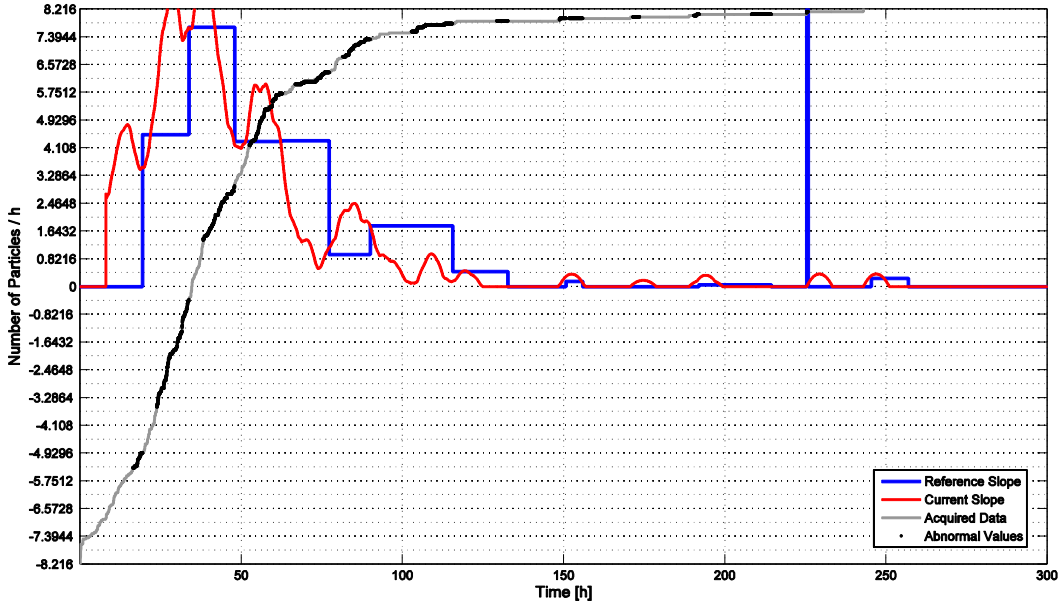

Figure 36. Quantitative evaluation: Number of non-ferrous particles (135 to 150 µm) (Pitting experiment from 04/02/2011).

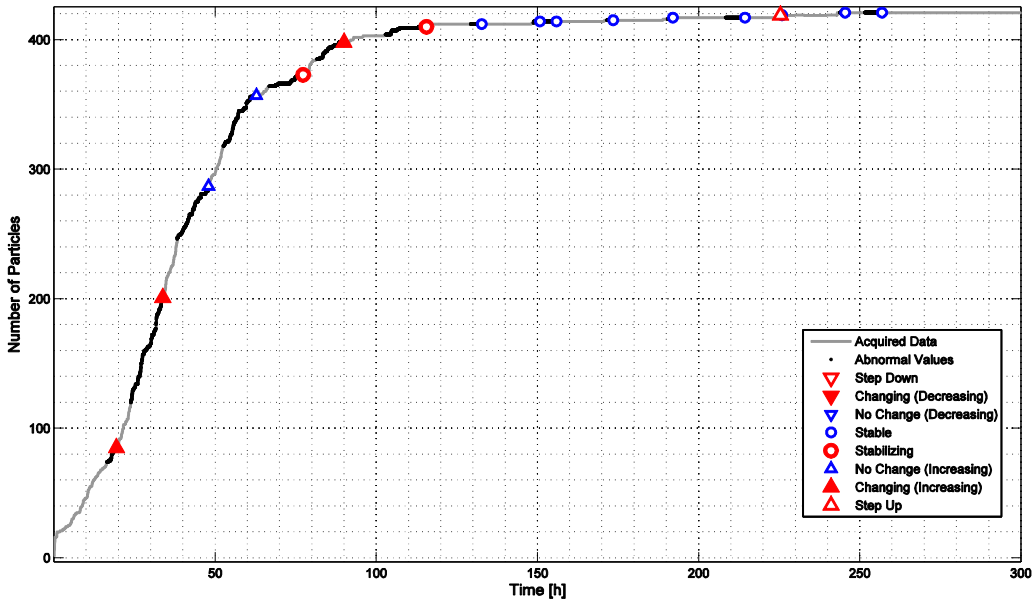

Figure 37. Qualitative evaluation: Number of non-ferrous particles (135 to 150 µm) (Pitting experiment from 04/02/2011).

APPENDIX A.4

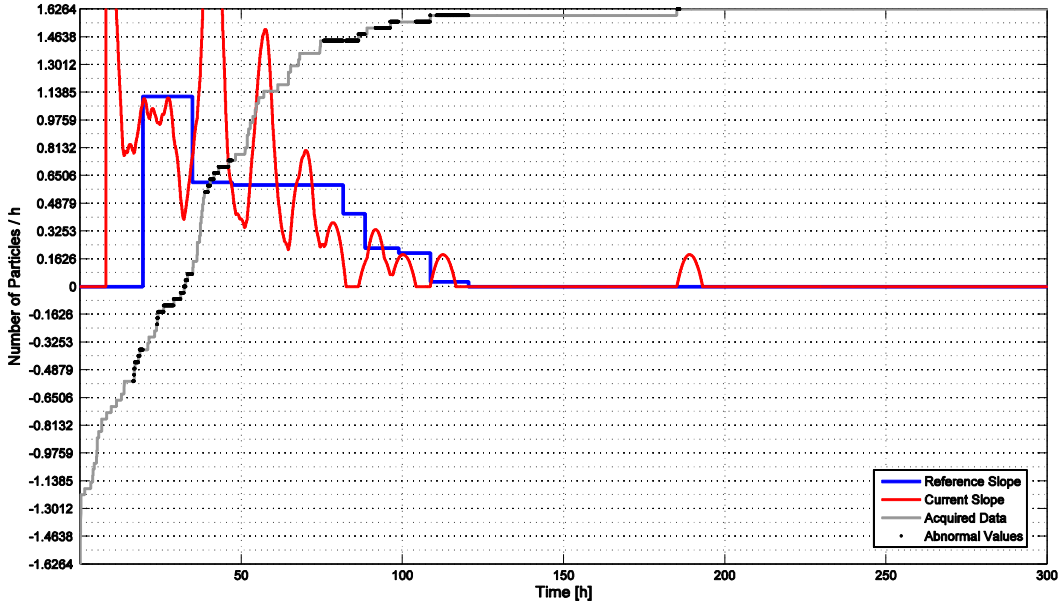

Figure 38. Quantitative evaluation: Number of non-ferrous particles (151to 250 µm) (Pitting experiment from 04/02/2011).

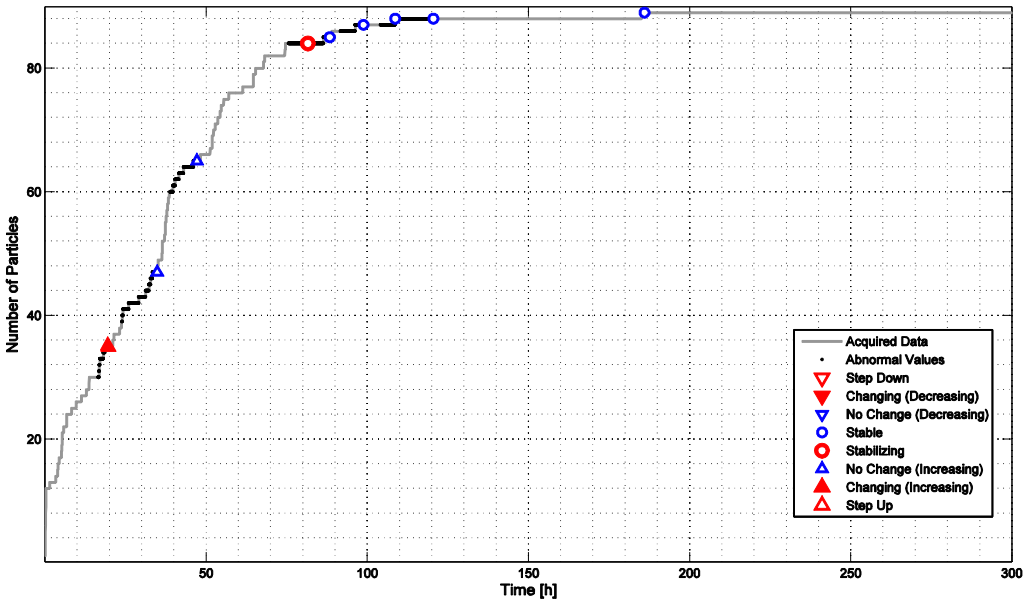

Figure 39. Qualitative evaluation: Number of non-ferrous particles (151 to 250 µm) (Pitting experiment from 04/02/2011).

APPENDIX B.1

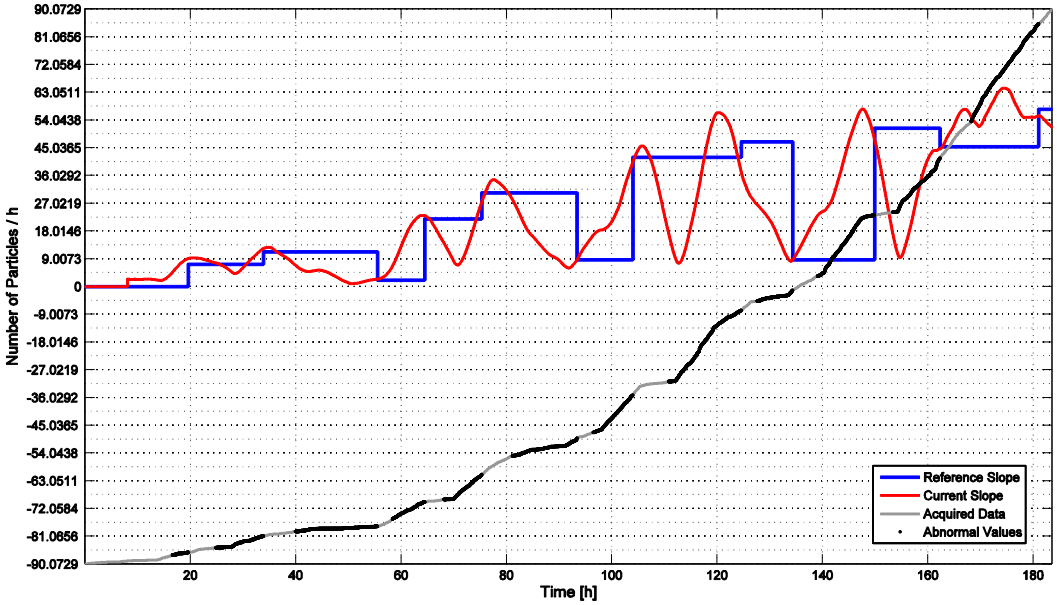

Figure 40. Quantitative evaluation: Number of ferrous particles (40 to 60 µm) (Pitting experiment from 03/03/2011).

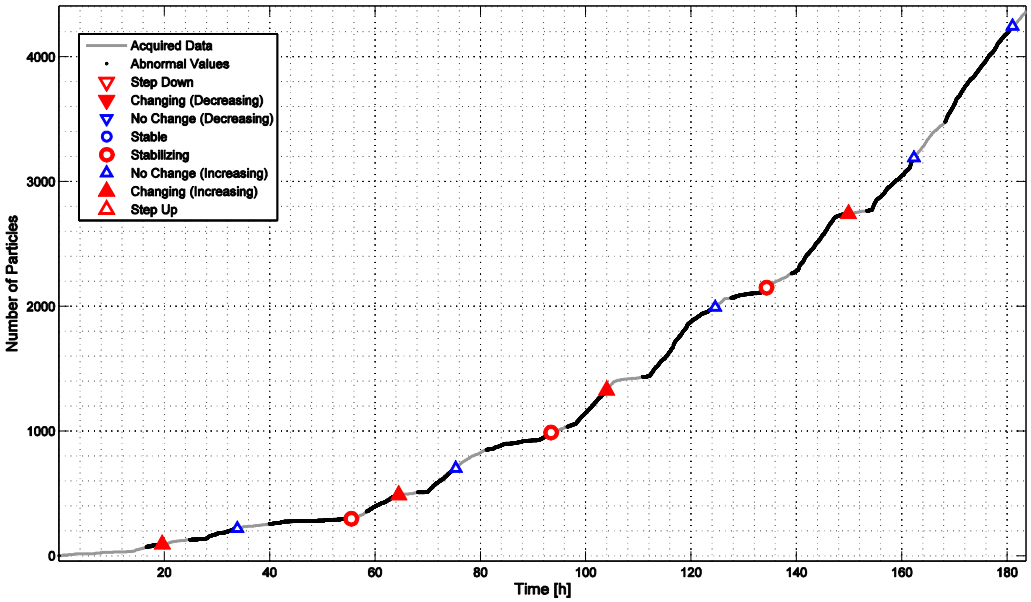

Figure 41. Qualitative evaluation: Number of ferrous particles (40 to 60 µm) (Pitting experiment from 03/03/2011).

APPENDIX B.2

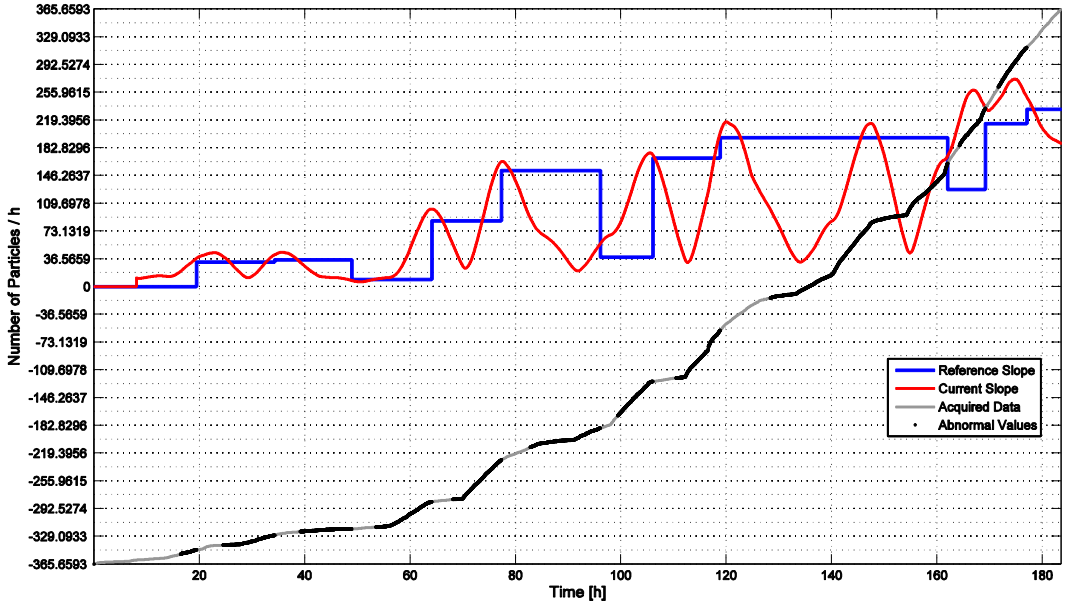

Figure 42. Quantitative evaluation: Number of ferrous particles (61 to 100 µm) (Pitting experiment from 03/03/2011).

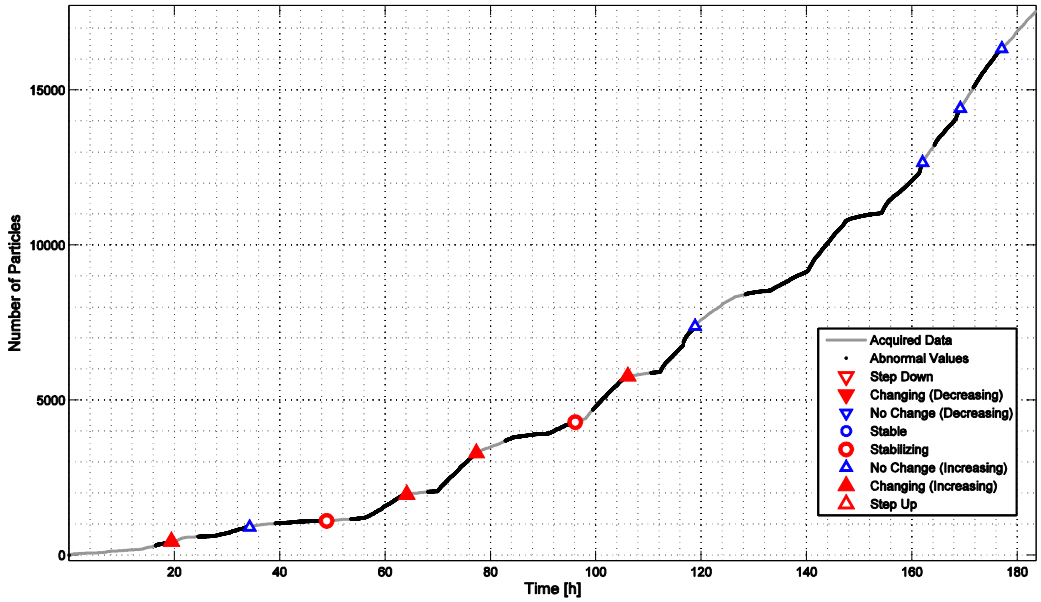

Figure 43. Qualitative evaluation: Number of ferrous particles (61 to 100 µm) (Pitting experiment from 03/03/2011).

APPENDIX B.3

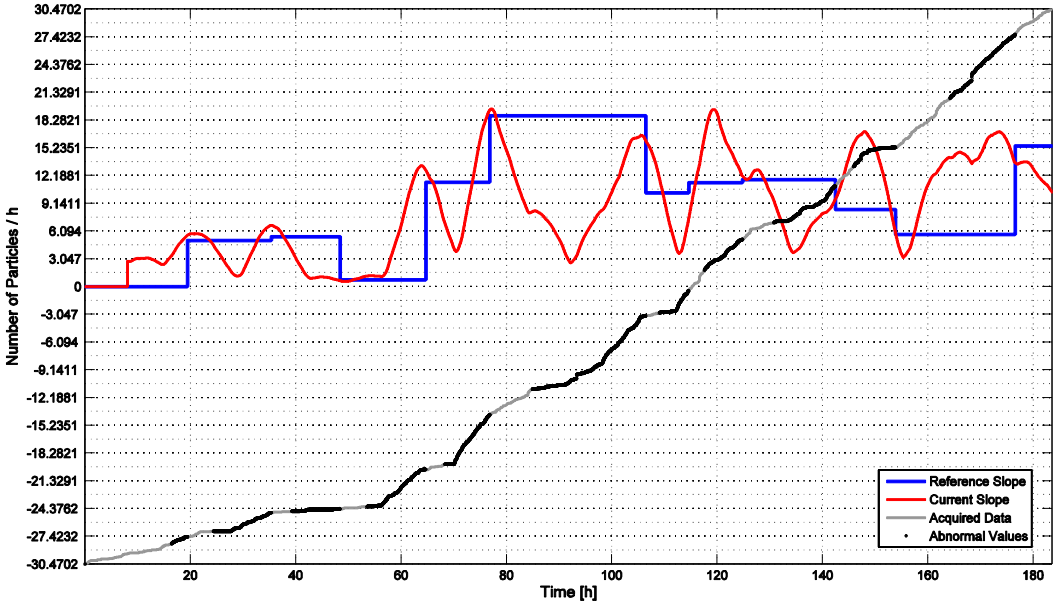

Figure 44. Quantitative evaluation: Number of ferrous particles (201 to 300 µm) (Pitting experiment from 03/03/2011).

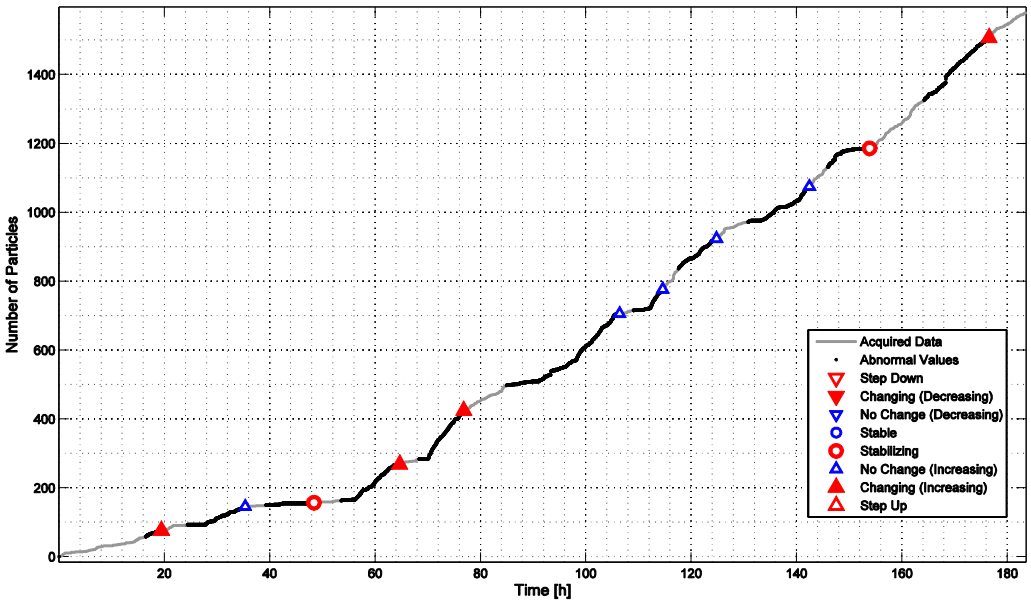

Figure 45. Qualitative evaluation: Number of ferrous particles (201 to 300 µm) (Pitting experiment from 03/03/2011).

APPENDIX B.4

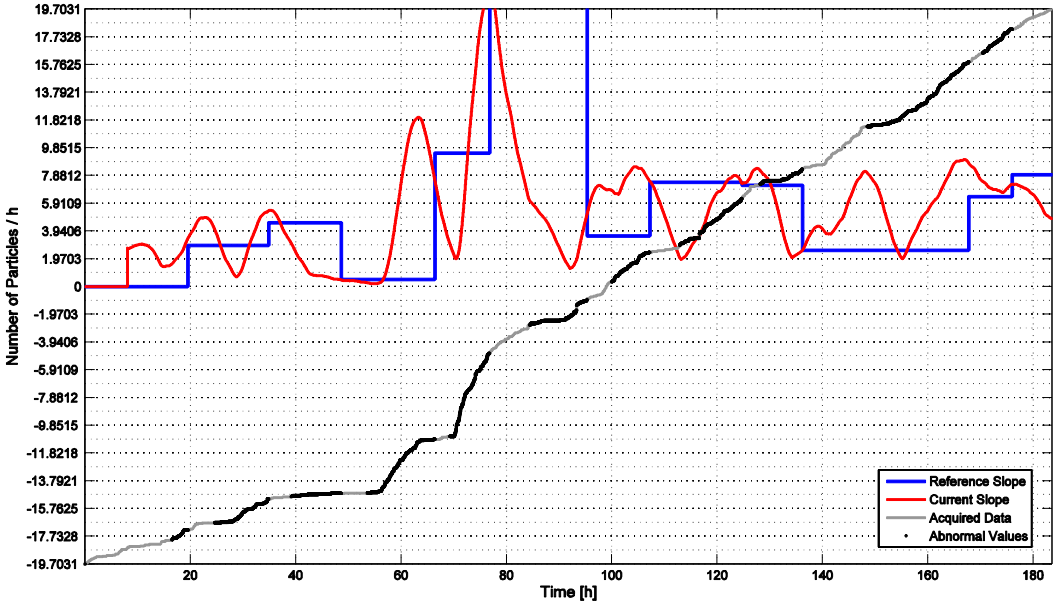

Figure 46. Quantitative evaluation: Number of ferrous particles (larger than 300 µm) (Pitting experiment from 03/03/2011).

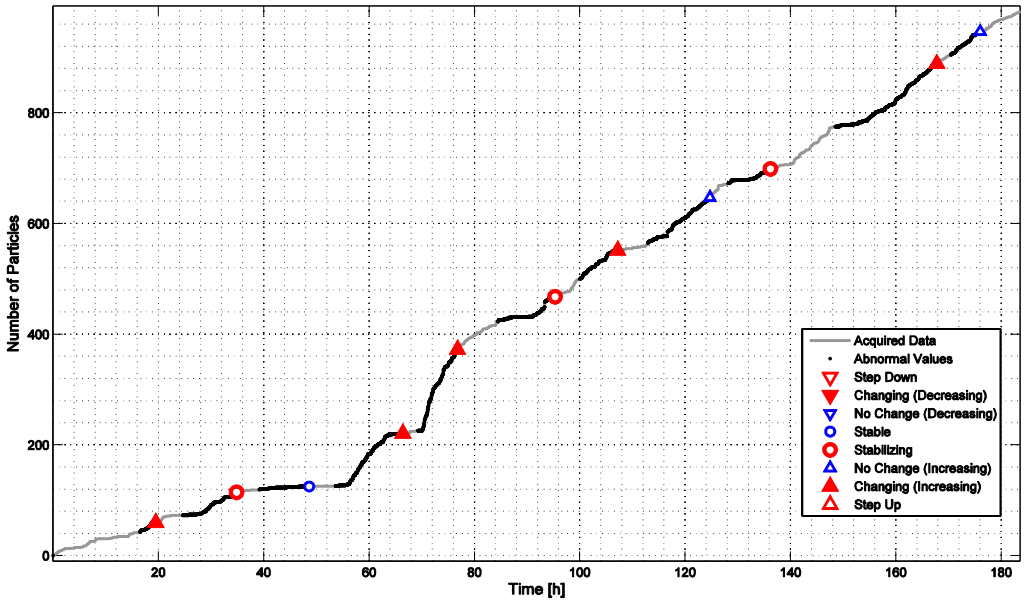

Figure 47. Qualitative evaluation: Number of ferrous particles (larger than 300 µm) (Pitting experiment from 03/03/2011).

APPENDIX B.5

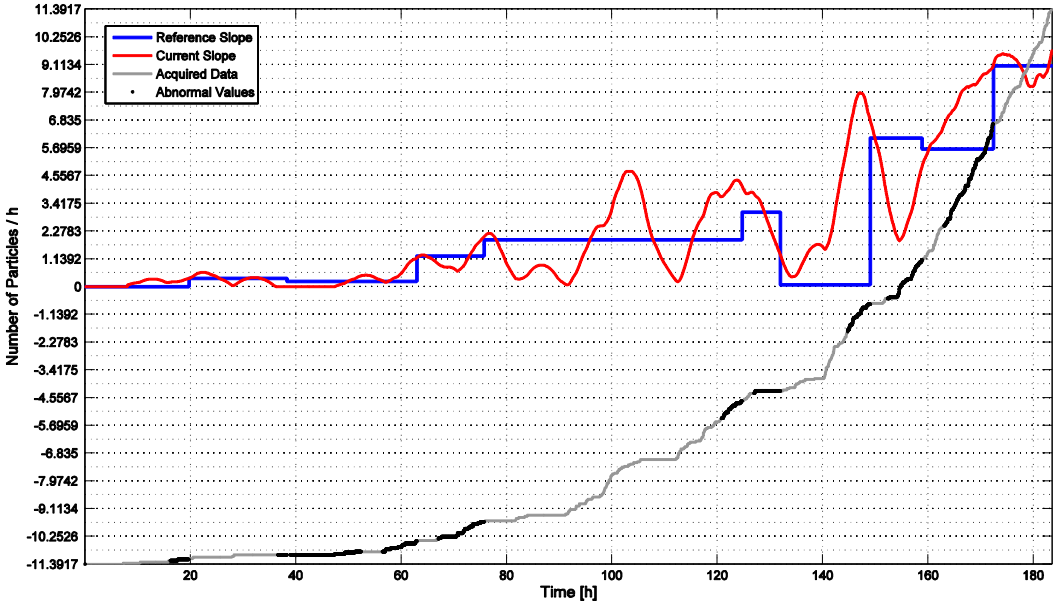

Figure 48. Quantitative evaluation: Number of non-ferrous particles (135 to 150 µm) (Pitting experiment from 03/03/2011).

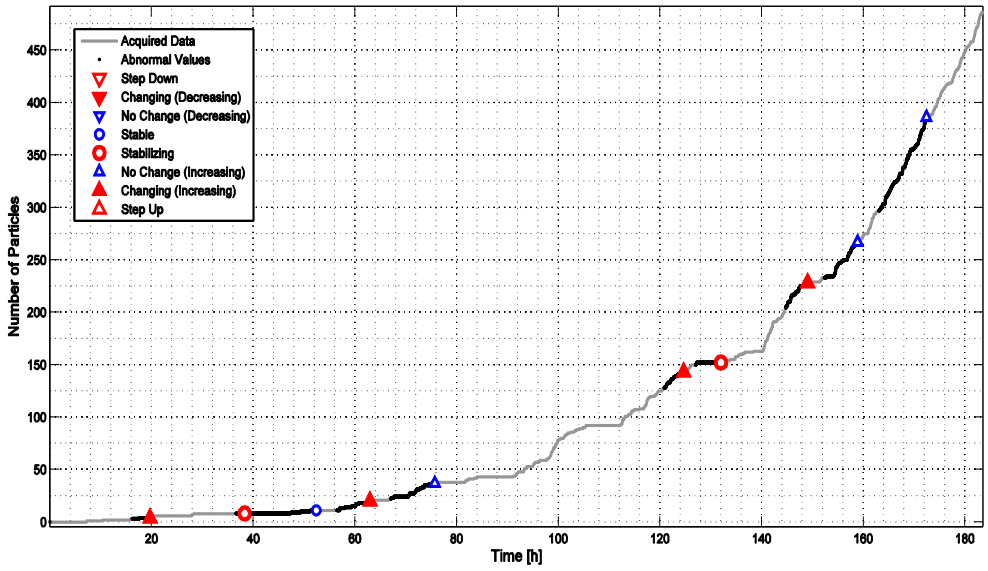

Figure 49. Qualitative evaluation: Number of non-ferrous particles (135 to 150 µm) (Pitting experiment from 03/03/2011).

APPENDIX B.6

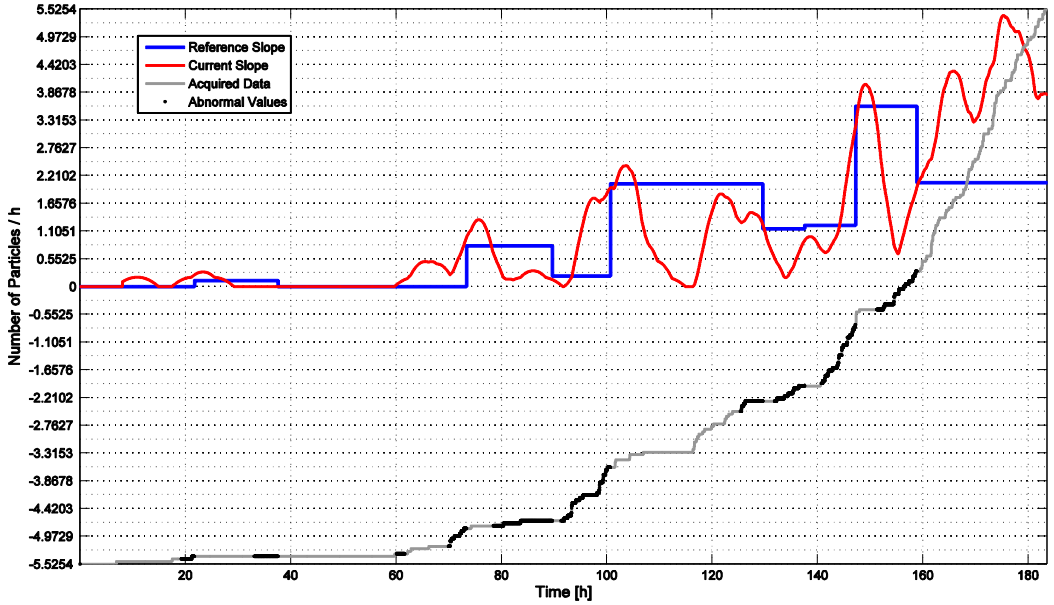

Figure 50. Quantitative evaluation: Number of non-ferrous particles (151 to 250 µm) (Pitting experiment from 03/03/2011).

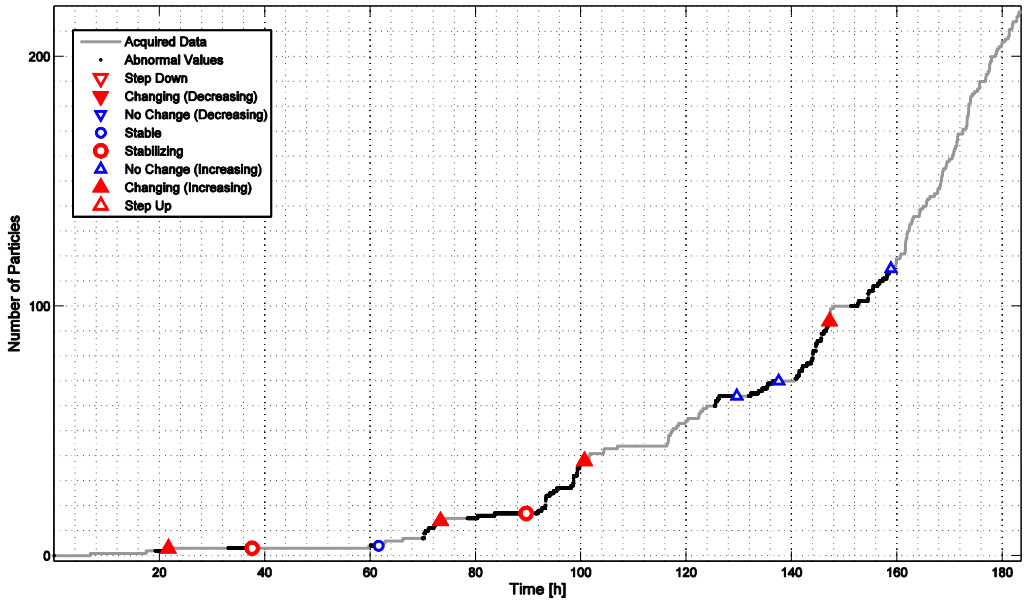

Figure 51. Qualitative evaluation: Number of non-ferrous particles (151 to 250 µm) (Pitting experiment from 03/03/2011).

APPENDIX C

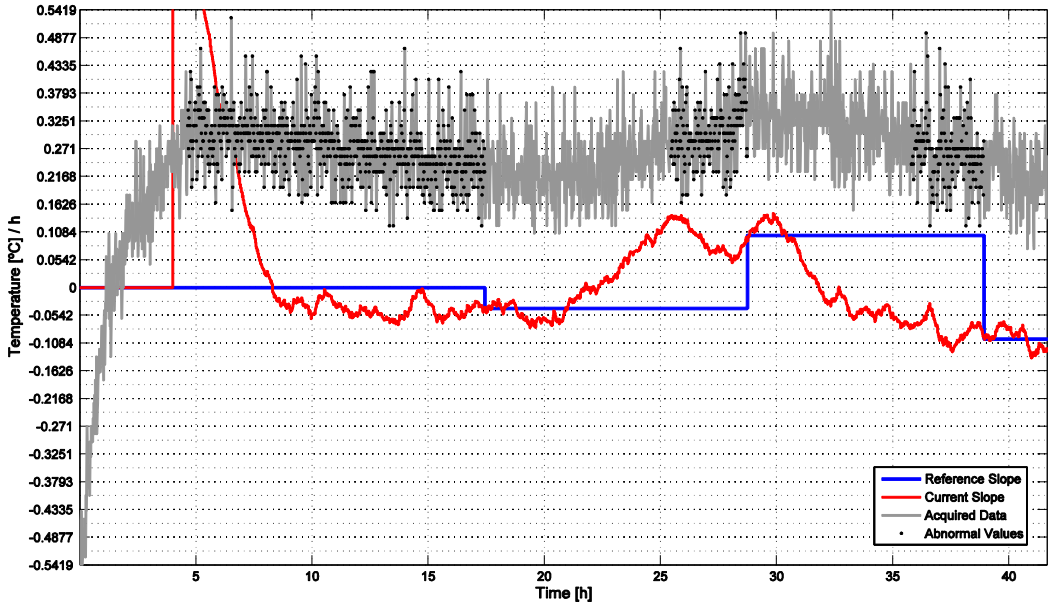

Figure 52. Quantitative evaluation: Temperature (Different oil contamination experiment from 04/04/2011).

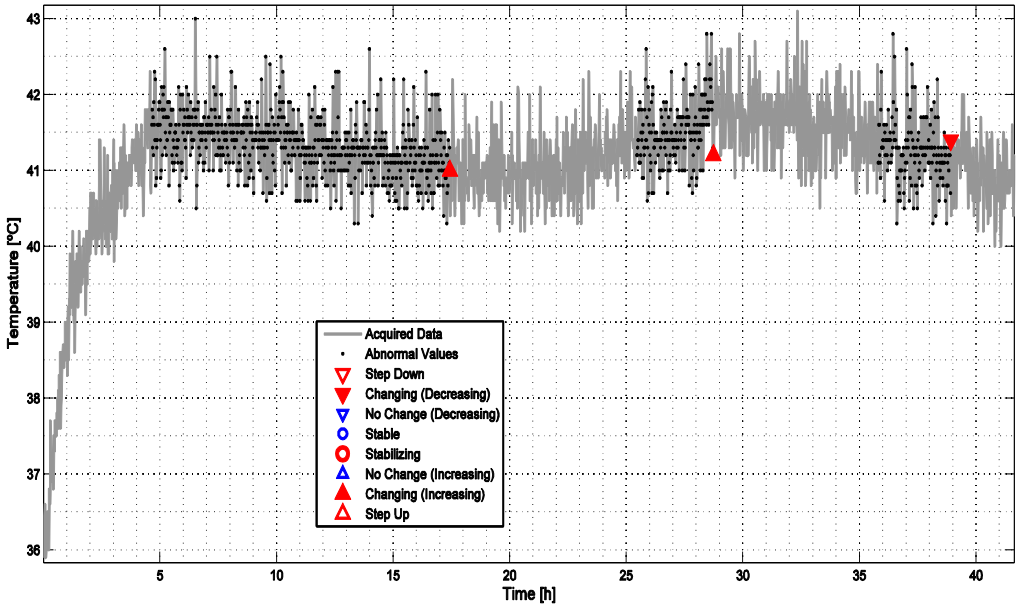

Figure 53. Qualitative evaluation: Temperature (Different oil contamination experiment from 04/04/2011).