Automatic Threshold Tracking of Sensor Data Using Expectation Maximization Algorithm

Arghad Arnaout
TDE GmbH
Leoben, Austria
arghad.arnaout@tde.at

Bilal Esmael
University of Leoben
Leoben, Austria
bilal@stud.unileoben.ac.at

Rudolf K. Fruhwirth
TDE GmbH
Leoben, Austria
rudolf.fruhwirth@tde.at

Gerhard Thonhauser
University of Leoben
Leoben, Austria
gerhard.thonhauser@unileoben.ac.at

***Abstract* - In this paper we present a novel method for automatic threshold handling and tracking of sensor data at drilling rigs. A hybrid system for automated drilling operation classification is extended by the Expectation Maximization algorithm in combination with the Bayes' theorem to find automatically threshold values required by a rule based system used in an automated drilling operations classification system. The streaming data from the rig site is gathered and analyzed, the main clusters in the sensor data are identified and monitored as in a real life case. The first part of the suggested method is based on the Expectation Maximization algorithm which is used to decompose Gaussian mixture models in the sensor data set. Bayes' theorem is used as a subsequent part to calculate optimal threshold values. The threshold values calculation concept is heavily depending on the likelihood probabilities of each data cluster. The work in this paper not only suggests a solution and analytical method for tracking this kind of thresholds in the sensor data but also verifies how to compute such reliable thresholds in real-time.**

Keywords: Expectation Maximization EM, Gaussian Mixture Model, Clustering, Bayes' Theorem, Thresholding

I. INTRODUCTION:

Automatic quality handling of sensor data is a big challenge in the drilling domain. Understanding the work flow – which is the drilling process in the current case – is the first step for analyzing the quality of the data. The forms and patterns of the sensor data should be known from the drilling data analyst. Expected and unexpected patterns in the data should also be isolated by the data analyst. Identifying the problems in quality of the received data from the rig site should be collected and gathered. These problems can be summarized under the main topics: threshold detection, missing values and outlier management, calibration, etc. [1]

In the drilling industry, automated drilling operations classification systems are going to become one of the required infrastructures for drilling projects [2]. The main concept behind such systems is based on rule based reasoning systems, so-called rule engines. Such rule engines are usually based on thresholds to detect different states of the drilling process at the rig. Normally such threshold values are configured by the drilling experts or analysts; they have a massive impact on the accuracy of such [2].

Unsupervised learning is an important approach in finding the structures of unlabeled data [7]. This approach from the data mining domain can be used intensively in analyzing the structures of sensor data. To be more precise, different techniques of data clustering can be applied to estimate correlations of the sensor data with the drilling operations. If the data can be separated into two or more natural clusters, the threshold values can – considering that those clusters explain different states of the drilling process – be estimated applying Bayes' theorem to the clusters. The automatic detection and tracking of the threshold values is considered as important task in avoiding lost time and fail classifications in the automated drilling operation classification systems.

II. DRILLING SENSOR DATA

Drilling is a process of making a hole in the ground in order to extract oil, gas or any other natural resources from the subsurface; usually performed by a rig. One of the most important parts of the drilling process is the drill-string. The drill-string is a chain of connected pipes usually having a length of 10 meters each. The bottom of the drill-string is made of special devices, denoted as bottom hole assembly (BHA), the last part of the BHA is the drill-bit.

Many sensors are mounted at the rig to record different (physical) measurements of the drilling process like block position, hookload, flow rates, pump and circulation pressures, hole & bit depth and torque, among others. Figure 1 shows a sketch of such sensor data over a period of about one hour.

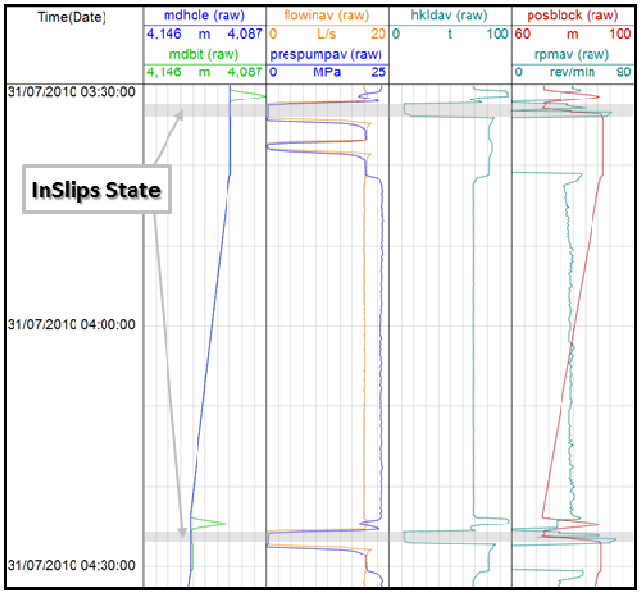

Figure 1: Drilling Sensor Data

The gray highlighted areas in Figure 1 refer to a special state of the drilling process; the drill-string is hanging in the rig floor fixed by slips, thus such a state is denoted as *InSlips* state. The non-highlighted areas refer to converse situations denoted as *OutOfSlips* state; this means that the drill-string is hanging at the hook of the rig [3] and therefore applies force to the hookload sensor. Such a hookload sensor usually measures the weight of the drill-string together with the weight of the hook; therefore the

hookload is not zero at the *InSlips* state. Two different patterns are formed by hookload measurements during the *InSlips* & *OutOfSlips* states [4]. At the *InSlips* state the hookload is low; the measured value indicates the weight of the hook only. At the *OutOfSlips* state the hookload is higher; the weight of the hook plus the weight of the drill-string hanging at the hook is measured.

The separation of the *InSlips* from the *OutOfSlips* states is one of the main steps of an automated drilling operations classification system [5]. Usually, the drilling experts set a threshold value manually for the hookload to separate the states. This manual configuration is the main source for false classification results and time losses caused by reprocessing of the data.

Figure 2 illustrates the hookload sensor data recorded for a period of two and half days, the regions for the main operations *TripIn*, *TripOut* and *Drilling* are highlighted on top of the figure.

The *TripIn* operation denotes that the drill-string is built-up and run into the borehole stand by stand. During that operation the drill-string increases in length, thus the hookload increases too; up to about 100 tons.

Once the drill-string reaches the bottom of the borehole the drilling process can be started; this is highlighted as *Drilling* operation.

Also the pipes are disconnected one by one when the string is pulled out of the borehole, and this what we call it *TripOut* operation.

III. EXPECTATION MAXIMIZATION

The Expectation-Maximization algorithm is an iterative optimization method for estimating some unknown parameters Θ, given a measurements data set **D** [6]. EM mainly looks for the maximum likelihood to evaluate the parameters of statistical models [7]. Figure 3 shows how the Expectation-Maximization algorithm works [9]. EM method is consisting of two main steps [8]:

Expectation E-Step: This step is responsible to estimate the probability $P(\Theta)$ of each data point belonging to each cluster in the measured data **D**.

Maximization M-Step: This step is responsible to estimate the parameters Θ_{new} of the probability distribution of each cluster for the next step. The difference between likelihood probabilities of the new estimated parameters Θ_{new} and the old parameters Θ_{old} is used to measure if we reach the maximum likelihood probability MLP.

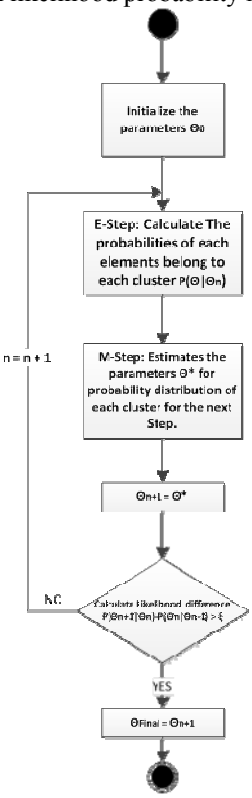

Figure 3: Expectation Maximization Algorithm

In Figure 4, the histogram of hookload data is shown. A Gaussian mixture model (GMM) is plotted over the histogram (red line). The Expectation Maximization algorithm can be used to decompose the GMM into Gaussian distributions. EM also estimates the parameters of these Gaussian distributions in hookload data. Each of those distributions corresponds with a data cluster. Comparing the histogram (Figure 4) and hookload data (Figure 2) we notice that there are three main data clusters

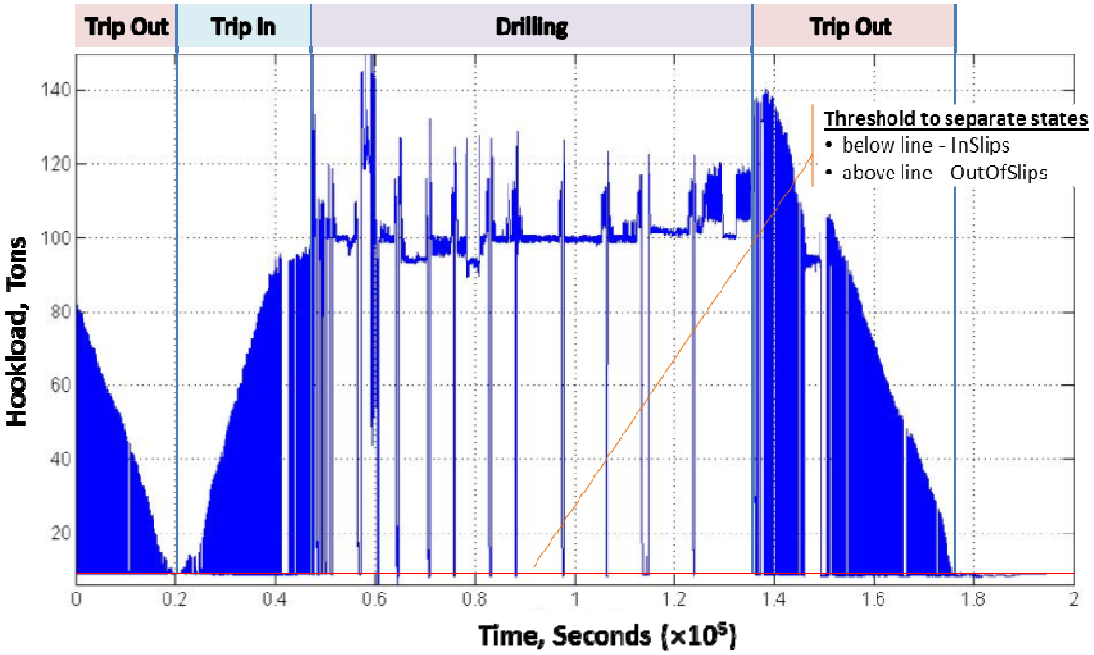

Figure 2: Hookload Sensor Data

in the histogram. Also we can easily find that the left cluster separates the *InSlips* state. While the middle cluster is formed during the *TripIn/Out* operations, the right cluster is obviously built throughout the drilling operation.

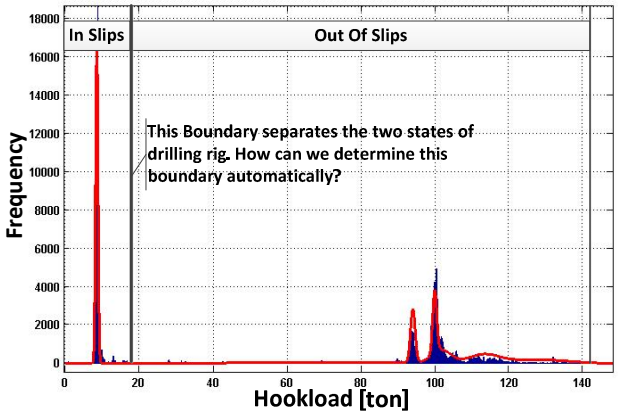

Figure 4: Histogram of Hookload Data.

IV. AUTOMATIC THRESHOLD DETECTION

In Figure 2, a theoretical threshold is plotted (red line). This threshold separates *InSlips* from *OutOfSlips* states. The same line is shown on the histogram of the data (Figure 4 – black line). The challenge is to isolate the *InSlips* Cluster, left of the line. The required threshold can be located between two data clusters (middle and left). This means that the required threshold is the threshold with the lower probabilities of two Gaussian distributions. In other words, the required threshold is the intersections point of two the probability density functions PDF of two Gaussian distributions (left and the middle). Figure 5 explains how we locate the required threshold as an intersections point of two Gaussian distributions.

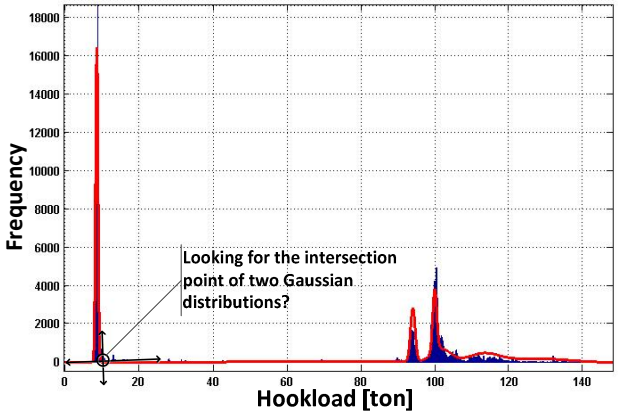

Figure 5: Threshold Locating

While monitoring the histogram of the data when it is streamed from the rig site, we find that the data clusters of *InSlips* state is formed very quickly, while the other clusters (*OutOfSlips* clusters) are formed slowly. We also can confirm that in any instances of time, the above description of the data histogram is applicable. Using this property we designed an algorithm for automated detecting of the threshold.

In this paragraph, we propose an algorithm for automatic calculation of the threshold. While in the next paragraphs we will discuss the results of applying the proposed algorithm on hookload data.

A. ***The Algorithm***

Initial phase:

- Initiate a buffer of data **B**. Choose a step **α** for increasing the buffer size.

Iterative phase:

1. Estimate the parameters $\Theta_{1,2,\dots,n}$ of Gaussian distributions using Expectation-Maximization Algorithm for the data buffer.
2. Sort the mean values vector of the estimated parameters, and pick up the Gaussian distribution parameters that correspond with two lower mean values (left and middle data clusters).
3. Calculate the intersection point between the two selected Gaussian Distributions (see detailed information about the calculation at the end of this paragraph).
4. Increase the buffer size by the step **α.**
5. Jump to 1.

B. ***Intersection of Two Probability Density Functions***

-*Given:* Two data clusters C_1 and C_2 assumed to be Gaussian distributed with $\Theta_1=\{\mu_1, \sigma_1\}$ and $\Theta_2=\{\mu_2, \sigma_2\}$.

-*Required:* The intersection point of the

The probability density $p(x|C_k)$ for the k^{th} cluster of a Gaussian distribution is given by

$$p(x|C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(x-\mu_k)^2}{2\sigma_k^2}} \quad (1)$$

Then the intersection point x is located where the probability of the cluster C_1 given x is equal to the probability of the cluster C_2 given x.

Using Bayes's theorem, the previous probability given by the following equations

$$P(C_1|\,x) = \frac{p(x\,|C_1)\,P(C_1)}{P(x)} \quad (2)$$

$$P(C_2|\,x) = \frac{p(x\,|C_2)\,P(C_2)}{P(x)} \quad (3)$$

where p(x) is given by:

$$P(x) = p(x\,|C_1)P(C_1) + p(x\,|C_2)P(C_2) \quad (4)$$

The prior probabilities $P(C_1)$ and $P(C_2)$ are given by

$$P(C_x) = \frac{\text{number of points belong to cluster } C_x}{\text{total number of points}} \quad (5)$$

The intersection point x or the threshold is calculated by solving the equation

$$P(C_1|\,x) = P(C_2|\,x) \quad (6)$$

V. RESULTS

Figure 6 shows the threshold which is calculated based on the suggested algorithm. The threshold (red line) as plotted on all the hookload data separates clearly between *InSlips* and *OutOfSlips* states.

These results fit very close with the suggested theoretical threshold. See Figure 2 for more information.

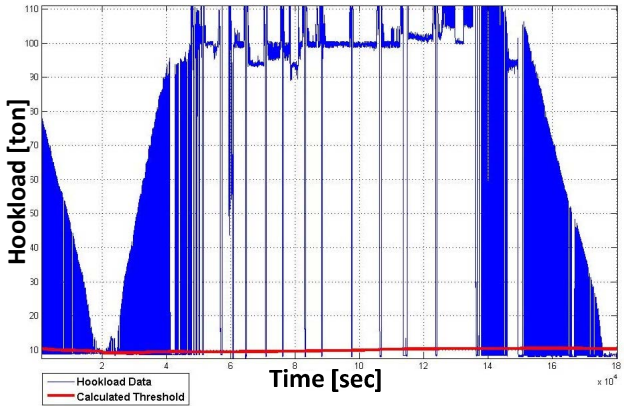

Figure 6: Hookload and calculated threshold

Figure 7 gives more detailed view on how the calculated threshold is accurately isolating the *InSlips* states during *TripOut* operations. We can also notice how the hookload (weight) is decreasing during tripping the drill-string out of borehole.

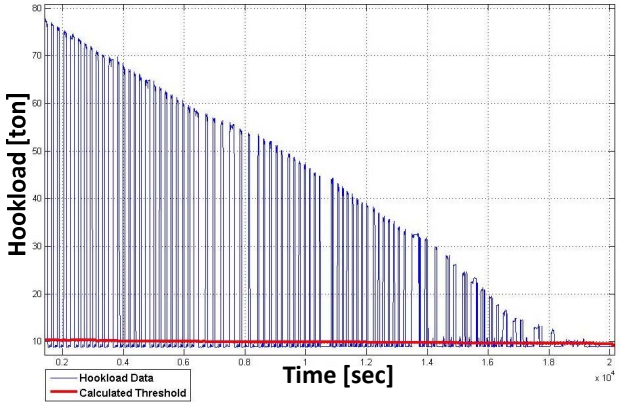

Figure 7: Hookload and calculated threshold during TripOut Operation.

Figure 8 sheds the light on different sections from the data during the drilling operation, and also we can easily notice how the calculated threshold separates *InSlips* state.

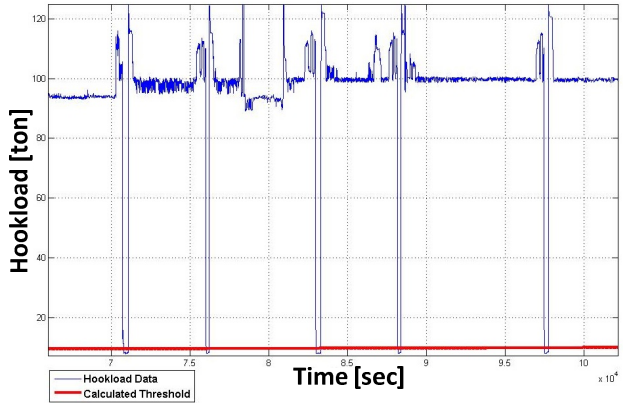

Figure 8: Hookload and calculated threshold during drilling operation

VI. SUMMARY

In this work, we introduced a method to use the unsupervised learning techniques in industrial applications. We also used the concepts from data analysis to design an automated method which can be easily implemented and integrated with the existing classification systems.

The algorithm is applied many real-life data from drilling rigs. The calculated threshold values accomplish high level of acceptance from the drilling experts.

VII. FUTURE WORK

The Automated Drilling Operations Classification Systems require configuration of many variables and threshold. The suggested method in this paper can be used to detect different thresholds at different sensor measurements.

Also the suggested algorithm can be extended on more than one dimension of data. This helps to find the clusters and threshold in multi-dimensional data.

VIII. CONTRIBUTION

This work can be considered as an important application of The Expectation-Maximization algorithm.

The reliability and flexibility are the significant features, which improve the functionality of the automated drilling operations classification systems.

IX. REFERENCES

[1] W. Mathis, G. Thonhauser, “How to Measure and Manage the Quality of (Rig) Sensor Data”, 11th International Conference on Petroleum Data Integration, Information and Data Management, Amsterdam, 2007

[2] G. Thonhauser, W. Mathis, “Automated Reporting Using Rig Sensor Data Enables Superior Drilling Project Management”, SPE Annual Technical Conference in San Antonio, Texas, U.S.A, 2006

[3] H. Rabia, “Oil Well Drilling Engineering, Principles and Practices”, University of Newcastle, 1985

[4] China National Logging Cooperation, “Mud Logging Technology and Services”, presentation, 2006

[5] G. Thonhauser, “Using Real-Time Data for Automated Drilling Performance Analysis”, OIL GAS European Magazine, 2004

[6] F. Dellaert, “The Expectation Maximization Algorithm”, Georgia Institute of Technology, USA, 2002

[7] E. Weinstein, “Expectation Maximization Algorithm and Applications”, Courant Institute of Mathematical Sciences, 2006

[8] A. P. Dempster; N. M. Laird; D. B. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm“, Journal of the Royal Statistical Society, 1977

[9] S.Y. Kung, M.W. Mak, S.H. Lin, “Biometric Authentication: A Machine Learning Approach”, Prentice Hall, 2004