MODEL LEARNING ALGORITHMS FOR ANOMALY DETECTION IN CERN CONTROL SYSTEMS

F. Tilaro, B. Bradu, M. Gonzalez-Berges, F. Varela, CERN, Geneva, Switzerland
M. Roshchin, Siemens AG, Corporate Technology, Munich, Germany

Abstract

The CERN automation infrastructure consists of over 600 heterogeneous industrial control systems with around 45 million deployed sensors, actuators and control objects. Therefore, it is evident that the monitoring of such huge system represents a challenging and complex task.

This paper describes three different mathematical approaches that have been designed and developed to detect anomalies in any of the CERN control systems. Specifically, one of these algorithms is purely based on expert knowledge; the other two mine the historical generated data to create a simple model of the system; this model is then used to detect faulty sensors measurements.

The presented methods can be categorized as dynamic unsupervised anomaly detection; "dynamic" since the behaviour of the system and the evolution of its attributes are observed and changing in time. They are "unsupervised" because we are trying to predict faulty events without examples in the data history. So, the described strategies involve monitoring the evolution of sensors values over time in the historical data. Indeed, consistent deviations from the historical evolutions can be seen as warning signs of a possible future anomaly; these warning signs have been used to trigger a generic anomaly alarm for the specific incoherent sensors, requiring further checks by system experts and operators. The paper also presents some results, obtained by the application of this analysis to the CERN Cryogenics systems.

Finally, the paper briefly describes the deployment of Spark and Hadoop platform into the CERN industrial environment to deal with huge datasets and to spread the computational load of the analysis across multiple hosts.

INTRODUCTION

The performance of the CERN Large Hadron Collider (LHC) relies on the operation of a multitude of heterogeneous industrial control systems. More than 600 industrial Supervisory Control and Data Acquisition (SCADA) systems have been deployed for the supervision and monitoring of CERN accelerators chain, detectors and technical infrastructure. Currently the stored data volume produced by the different industrial control systems exceeds the 100 terabytes per year; nevertheless, the volume of the controls data is actually much higher since multiple filters (deadbands both in time and value) are applied both at the SCADA systems and at the control level (PLCs and Front-End Computers) in order to reduce the data flow.

The generated controls data, after a proper analysis, constitutes a source of useful information about the current state of the processes, their performance, stability and overall behaviour. Obviously, an extensive analysis of this massive data flow requires specialized frameworks to handle big datasets and cannot be achieved by operators through manual operations.

The detection of anomalies and disturbances in an industrial process represents a key factor in the quality of the overall system [1, 2]. Nevertheless, an anomaly, if not properly handled, could lead to a system failure, therefore causing a downtime of the entire system. In our scenarios, the anomaly detection is strictly connected to the ability to identify sensors' measurements which do not conform to the expected patterns; this explains the use of machine learning techniques to extract patterns from both historical and online data. The correct detection of such types of unusual behaviours allows system experts to take actions in order to avoid, correct and react to the situations associated with them.

The temporal aspect plays an important role in the analysis of control data; in this domain time series data represents the main object of analysis in order to detect regular patterns against the sensors' measurements as described in [3]. In recent years a growing attention has been paid to online knowledge discovery and data mining (KDD) techniques [4] for multivariate time series data. In the systems analysed, the change point or anomaly detection from data streams was an unsupervised learning task, which aimed at deciding whether the new generated sensors' measurements showed a different trend from the historical reference.

In this paper, we address the problem of change point detection for streams of multivariate time-series data. Specifically, this paper describes three different algorithms for online detection of faulty measurements that have been developed and integrated into the CERN control system as a continuous monitoring task of the machine operation. Once these analyses have been deployed, the system experts are notified on specific issues or possible anomalous conditions through the generation of alarms. To achieve the aforementioned task the proposed solutions are based on unsupervised techniques due to the lack of labelled training data.

The last sections of the paper present a comparison of the three developed algorithms and some anomalies detected through the analysis of the CERN cryogenics control system.

ISBN 978-3-95450-193-9

FAULTY SENSORS DETECTION IN THE CRYOGENICS SYSTEM

The presented analyses mainly focused on the LHC cryogenics system. It plays a fundamental role to keep the magnets and the RF cavities under superconducting state, which allows an electric current to pass almost without resistance and avoid the joule heating effect. Actually, 27 km of superconducting magnets have to operate in superfluid helium below the temperature of 1.9 K. The cryogenics helium distribution line consists of 8 sectors with a length of about 3.3 km each. As a result of that, the cryogenics control system is highly distributed and made of heterogeneous devices. However, since each sector is operationally autonomous, a similar behaviour is expected across the entire system. A large number of industrial sensors, electronic conditioning units and actuators (mainly heaters and valves) are deployed in the cryogenics control system. Moreover, those components located in the LHC tunnel have to properly operate in a hostile radiation environment in order to deliver reliable measurements. The cryogenics process control logic consists of more than 4700 control loops executed in 80 PLCs which are connected to a multitude of I/O channels; the specific numbers are shown in the table below as documented in [5]:

Table 1: LHC Cryogenics I/O and CCL

	Tunnel	Production	Total
Analog Inputs	12136	9200	21336
Analog Outputs	4856	2152	7008
Digital Inputs	4536	13820	18356
Digital Outputs	1568	2644	4212
Control Close Loop	3680	1024	4704

The above given description underlines both the criticality and the complexity of the cryogenics system whose entire cooling process consists of three different stages and takes almost one month to complete. The signals' heterogeneity is another important aspect which increases the complexity of the problem. The analysis exploits the well-defined control system structure to discover possible systematic relationships among different signals; these relationships can be classified in 2 main categories:

- Physical relationships: two or more sensors deployed in the same sector or even in the same cell should not differ much in value (like to temperature sensors in contiguous cells).
- Logical relationship: the control loops regulate some output signals as a function of some input signals.

Both classes of these relationships need to be discovered within the controls data in order to make a model of the system. The latter can be then used to detect change points that are distance from the reference state. It is worth noting that the cryogenics sensors' values are highly dynamic: different machine runs can show completely different trends depending on the operational mode and energy of the HC. For this reason, the existing alignment techniques [6, 7] cannot be used to calculate the "anomaly score" (that is the distance from the reference state) over multiple machine runs (or time intervals). On the contrary, the machine learning process must continuously keep the model in line with the dynamic of the control system. As it will be described in the following paragraphs, this has been achieved through an incremental learning methodology that uses the streams of input sensors' data to extend the existing model.

ANALYTICAL SOLUTIONS

The following paragraphs will present three different algorithms that have been developed and integrated into the control system for online detection of faulty sensors. The three algorithms make use of different techniques to extract patterns from the historical data; these patterns are then compared against the input stream of measurements for online anomaly detection.

Anomaly Detection by Sensors Correlation and Conditional Nearest Neighbours

The main idea of this algorithm is to group the large list of sensors and actuators into different clusters based on the previously described logical and physical relationships. In order to find these relationships, the historical data are mined and a correlation matrix is computed. As shown in the equation below the Pearson correlation coefficient is used:

$$C = \begin{bmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{bmatrix},\ a_{ij} = \frac{c_{ij}}{\sigma_i \sigma_j},\ where:$$

$c_{ij}, \sigma_i, \sigma_j$ *are respectively the covariance and the standard deviations of the two signals.*

If the standard deviation of the single signals is close to zero, then the Kronecker's delta function is used instead:

$$a_{ij} = \delta_{ij} = \begin{cases} 0, & x_i \neq x_j \\ 1, & x_i = x_j \end{cases},$$

where x_i, x_j *are the values of the two signals*

This kind of formulation is able to properly deal with different sensors parametrization, calibrations and offsets. However, as one can see from the formulations above, the use of the Pearson correlation coefficient implies a linear dependency among the sensors; nevertheless, this limitation could be overcome by extracting some linear features from the initial raw values or by projecting them in to a linear domain. In the study conducted for the cryogenics system, the linearity limitation did not constitute an issue since multiple linear relationships have been discovered among the input signals.

In the literature, many studies [8, 9] represent streams of time series data into weighted graphs where each node corresponds to a specific signal and each edge shows the similarity between a pair of signals. The proposed method adopts a similar approach by defining a K-Nearest-Neighbour graph [10] based on the computed correlation coefficients; precisely the distance between signals, represented by the graph edges, is calculated as follows:

ISBN 978-3-95450-193-9

$$d_{ij} = -\ln|a_{ij}|$$

Therefore, the distance between highly correlated signals will be close to zero and for uncorrelated signals will tend to infinity. 'K' represents the dimension of the graph and theoretically should be chosen as the minimum size of the highly correlated clusters. The accuracy of KNN method can be severely degraded with high-dimension clusters because of the difference between the nearest and farthest neighbours (increase of the inner-cluster distance). Furthermore, increasing the size of the graph leads to an increment of the computational complexity of the KNN graphs. In our scenario, after some initial test, 'k', the number of correlated signals in each graph, was set to 3. During the analysis of the cryogenics system most of the unimportant fluctuations in the measurements under nominal operation did not affect highly correlated pairs of signals.

In order to evaluate each significant change in the signals (change point analysis), the following dissimilarity function was defined:

$$E(d_i) = \sum_{j=1}^{k} d_{ij} * p(j|i), \text{ where:}$$
$$p(\text{j|i}) = \text{probability of j to be in the KNNi } graph$$

The so-defined expected dissimilarity of each signal is a function of the previously described distance between the nodes in the KNN graph and the probability of the sensors to be in the same cluster (KNN_i) during different time windows.

In our scenario, the conditional entropy Hi quantifies the amount of information needed to describe the value of a signal s_j, given that the value of another signal s_i is known. The conditional entropy is equal to zero if the value of the sensors s_j is completely determined by the value of s_i:

$$Hi = -\sum_{j \in iUNNi} p(j|i) \ln p(j|i)$$

Consequently, the probability distribution can be calculated as a minimization problem introducing the Lagrange's multipliers for the conditional entropy between the sensors:

$$\boldsymbol{\nabla} E(d_i) - \lambda \nabla Hi = 0 \rightarrow \frac{\boldsymbol{\partial}}{\boldsymbol{\partial i}} \sum d_{ij} p(j|i) - \boldsymbol{\lambda} \frac{\boldsymbol{\partial}}{\boldsymbol{\partial i}} Hi = 0$$
$$\rightarrow d_{ij} - \boldsymbol{\lambda} \frac{\boldsymbol{\partial}}{\boldsymbol{\partial i}} [-\sum p(j|i) \ln(p(j|i)] = 0$$
$$d_{ij} + \boldsymbol{\lambda} \ln\big(p(j|i)\big) + 1 = 0 \rightarrow \mathbf{ln}(p(j|i)) = -\frac{d_{ij}+1}{\lambda} \rightarrow$$
$$p(j|i) = e^{-\frac{d_{ij}+1}{\lambda}} \rightarrow p(j|i) = e^{-d_{ij}}$$

In order to have a meaningful probability value within the range [0, 1] the probability normalization condition has been imposed:

$$p(i|i) + \sum_{j \in NNi} p(j|i) = 1$$

Therefore, the probability related to each pair of signals can be calculated as:

$$p(j|i) = \frac{e^{-d_{ij}}}{1 + \sum_j e^{-d_{ij}}}$$

The faulty sensor detection was achieved by detecting any change exceeding a predefined threshold in the dissimilarity function for each KNN_i graph. Due to the high dynamic of the system, the learning process continuously updates the KNN-based model, which is used as a reference during the online fault detection analysis.

Anomaly Detection by Stochastic Clustering of Sensors Measurements

This algorithm aims at detecting faulty sensors by clustering them into different partitions based on their historical measurements. The clustering is based on K-Means [11] method by splitting the historical data sets into time windows; by limiting the size of the signals analysed the computational load and the execution time are reduced.

Due to the presence of different sensors calibrations it was necessary to standardize the data. Therefore, for each time window, the mean of each signal is subtracted from each measurement and then divided by the standard deviation to remove any scaling effect.

One of the main disadvantages of K-Means is having to provide as a parameter the number of clusters. To overcome this issue and optimize the choice of this parameter the Davies-Bouldin index [12] was used. This index tries to minimize the intra-cluster distance and at the same time maximize the inter-cluster distance. This allows to isolate in a single cluster only those sensors/actuators with similar behaviour and, at the same time, very different from other signals clusters. In order to find the optimal cluster number, the algorithm is running multiple times increasing it in each execution. The execution ends when the difference between the standard deviations of the intra-cluster distances is less than a specific threshold. The latter is calculated for two contiguous execution as follows:

$$d_{k,k-1} = \frac{\sum_{g=1}^{G_k} \sum_{x_{it}^k \epsilon C_j^k} \left\| x_{it}^k - c_g^k \right\|^2}{\sum_{g=1}^{G_{k-1}} \sum_{x_{it}^{k-1} \epsilon C_j^{k-1}} \left\| x_{it}^{k-1} - c_g^{k-1} \right\|^2}, where:$$

- $d_{k,k-1}$ *is the ratio between the sum of intra − cluster distances with* k *and* $k-1$ *clusters*
- c_g^k *is the centroid of the cluster* g *with* k *clusters*
- x_{it}^k *is the value of the signal at the execution with* k *number of cluster*

Moreover, the time window length was selected trying to maximize both the number of clustered signals and the probability to find each cluster in multiple time windows. To achieve it, a quality index of the analysis has been defined. Let *NSC* be the number of signals clustered and *N* the total number of signals. Then, for a time window with the chosen parameter *p* (number of seconds), the quality *Q* of a time window is defined as:

$$Q_p = \frac{NSC_p}{N} * \mu_{\max\left(P_p(g_j)\right), \forall j}, where$$
$$\mu_{\max\left(P_p(g_j)\right)} \text{ is the maximal average of probability}$$

ISBN 978-3-95450-193-9

$$for\ the\ cluster\ g_j$$

Let Q be the quality of a time window with parameter p (number of seconds), *MPCS* the minimal percentage of clustered signals, *MPC* the minimal probability of each cluster and *MIQ* the minimal increment of quality index for the time window optimization; then the thresholds are defined as follows:

$$Tq_{p,p-1} = \begin{cases} Q_p > MPCS * \mathrm{MPC} \\ Q_p - Q_{p-1} < MIQ \end{cases}$$

The result of this phase is a probability model of the most frequent (across all the time windows) signals clusters. Since there is a binary possibility to find a cluster in each time window, then the binomial model can be applied. Let $P(g_j)$ be the probability of the cluster g_j, n the number of time windows, k_j the number of success for the cluster g_j and X_j the binomial random variable, then the binomial distribution model can be defined as:

$$P(X_j = k) = \binom{n}{k} * P(g_j)^k * (1 - P(g_j))^{n-k}$$

If the expected value of the binomial distribution of any cluster is changing more than a defined threshold then an alarm is generated.

Anomaly Detection Based on Experts' Knowledge

This algorithm has been entirely based on experts' knowledge; the latter has been translated into mathematical conditional equations in order to detect if an actuator's behaviour deviates from the actuators of the same type (mainly valves). Specifically, the system experts defined a list of static "actuators groups/clusters" within the cryogenics distribution line that should behave almost identically. If one or several actuators show a different operational pattern than the rest of the group, it could represent a potential fault. Most of actuators in a group can have different absolute values but their derivatives should be similar. This is why after an initial first order filtering of the signals, the derivatives of the signals are computed. The so obtained derivatives are then grouped into global statistical indexes (i.e. the average of the signal group derivatives). An alarm is generated if the difference between the single signals indexes and the global indexes exceeds specific thresholds (also defined by the system experts).

PARALLELIZATION OF THE ANALYSES

A massive amount of computational resources is necessary to analyse the data produced by the CERN control systems. Any naive attempt of using standalone scripts or even single node applications results ineffective due the limited amount of resources, like I/O bandwidth, memory consumption and CPU load. Consequently, a cloud computing approach would match the computational requirements to run the analysis. This is why the described algorithms have been implemented as Spark [13] jobs and executed against the CERN Hadoop [14] cluster. Particular attention has been paid to parallelize the execution of such algorithms, showing the positive benefits of scaling the analysis across multiple nodes. Moreover, the lightweight portability of the spark jobs solves any deployment issues related to different execution environments.

All three presented methods combine measurements of multiple signals to detect anomalies in the control system. Therefore, the multivariate nature of the analyses (MVA) influenced the strategy of spreading the signals' datasets across different Spark nodes. More precisely, each cluster or group of signals has been loaded within the same node; this avoids any shuffling and communication overload among the Spark nodes. Moreover, a time window approach has been adopted to avoid keeping in memory massive datasets and to reduce the computational load; on the contrary, depending on the time window length, only a portion of the dataset was kept in the nodes' memory for the analysis.

Spark provides data scientist with high-level APIs in Scala, Java, Python and R development languages. Unfortunately, the Cloudera distribution currently installed at CERN does not support R. Python was chosen mainly because of the large number of modules and packages that are readily available for signal processing. Specifically, NumPy [15] and SciPy [16] have been combined with Spark Python API to deal with multidimensional arrays and implement algebraic operations.

ALGORITHMS COMPARISON AND VALIDATION

Due to the unsupervised nature of the analysis and the lack of an anomaly database, the three aforementioned algorithms have been tested and compared through the use of synthetic data. Specifically, the sensor faults have been classified as following:

- Spike: a single faulty measurement which is abnormally high/low if compared with the range of values next to it (i.e. a temporary glitch).
- Step: a temporary degradation of the signal which results in a step function.
- Noise: the signal deviates from its normal pattern for multiple measurements.
- Flipping: a special case of the noise fault, where the signal is passing from a high value (higher than the proximity values) to a low one (lower than the proximity values).
- Offset: offset that alters in a constant manner the sensors values (e. s. wrong calibration).

For each of these anomaly types multiple faulty measurements have been generated (synthetic data) with different frequency and amplitude. Therefore, the three algorithms have been compared by computing the confusion matrix containing: positive predictive value (PPV), negative predictive value (NPV), sensitivity (Sens), specificity (Spec) and accuracy (Acc). The PPV and NPV values reflect respectively the precision at detecting true positives

ISBN 978-3-95450-193-9

TP and true negatives TN. The Sens and Spec indexes represent the ratio of actual TP/TN which are correctly identified. Finally, the accuracy value shows the portion of correct predictions. They are calculated as functions of the number of true/false positives (TP/FP), the number of true/false negatives (TN/FN):

$$PPV = \frac{\sum TP}{\sum TP + \sum FP}, NPV = \frac{\sum TN}{\sum FN + \sum TN},$$

$$Sens = \frac{\sum TP}{\sum TP + \sum FN}, Spec = \frac{\sum TN}{\sum FP + \sum TN},$$

$$Acc = \frac{\sum TP + \sum TN}{\sum TP + \sum FP + \sum FN + \sum TN}$$

Figure 1 represents these different indexes for the three algorithms previously described. As one can notice from Figure 1 the three algorithms are able to detect different anomaly types without changing the input parameters. The stochastic and NN (SNN) and the Expert's knowledge (EK) algorithms have a lower PPV for different reasons. The SNN algorithm generates a higher number of FP because a change in one of the cluster affects the calculus of the centroids of the rest of clusters; the EK algorithm is really specialized at detecting the faults defined by the experts, but not general enough to detect new types of faults; this latter point explains why it also achieves the lowest sensitivity value, which represents the anomaly ratio correctly identified. On the other end, the EK method reaches the highest specificity value since it generates less FP.

From the performed evaluation, the correlation and NN algorithm demonstrate to be the more robust and at the same time general method.

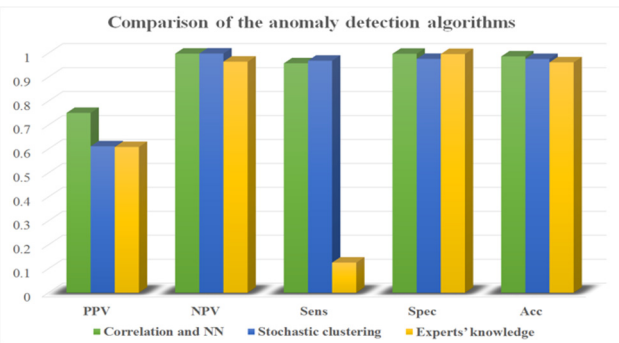

Figure 1: Comparison of the algorithms.

FAULT DETECTION ANALYSIS FOR THE LHC CRYOGENICS SYSTEM

As mentioned in the previous chapters the main subject of the presented analyses was the LHC cryogenics system. The analysis has been applied for anomaly detection in the cryogenics 1.8 K cooling loops, the beam screen cooling loops and current leads.

The following pictures show different faulty patterns that have been detected by the analytical algorithms. Specifically, Figure 2 displays a noisy movement of one actuator, while the others are mostly constant along the full time window. During the learning phase the actuators were clustered together due to their historical behaviour. The anomaly was detected because one of the signals (the one in black) starts moving in an uncoordinated way with respect to the others.

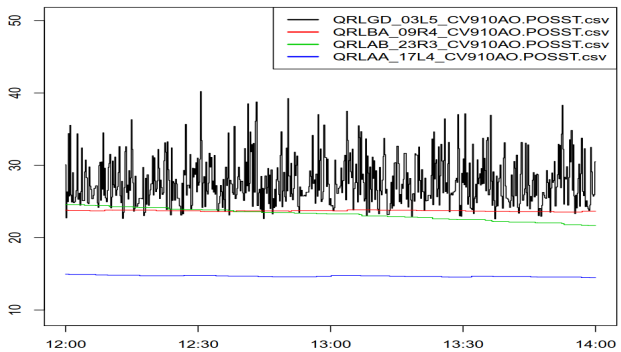

Figure 2: Flipping fault detection in a cluster of cryogenics valves.

Figure 3 shows another type of signal fault with the shape of a step function; the signal in black changed its offset for a limited period of time before returning to its nominal state. Similar behaviours have been frequently discovered during the actions of an operator overriding the process control output. Even in this case, the necessary human manual action can be interpreted as a control process anomaly.

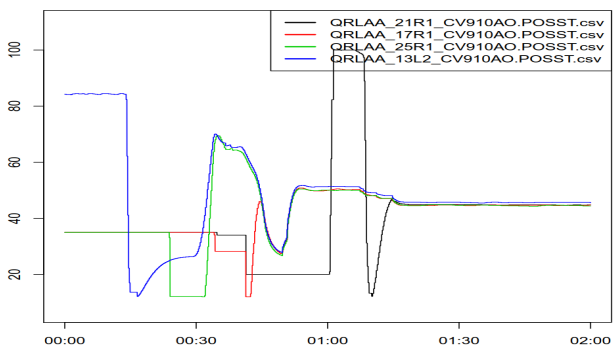

Figure 3: Step fault detection of one cryogenics actuator.

In specific, isolated cases some process instabilities have been discovered. Figure 4 clearly depicts this situation where the valve shows an unstable oscillatory behaviour to control the temperature process. Obviously if this faulty oscillation had been present in all signals then it would not have been detected by the clustering algorithms, but only by the experts' knowledge formulation. However, in our scenario the probability of a faulty situation involving multiple signals at the same time was almost close to zero.

ISBN 978-3-95450-193-9

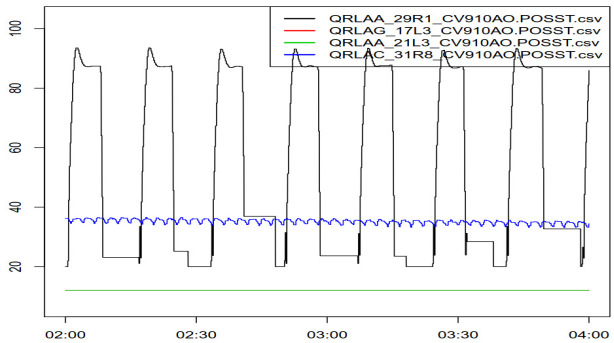

Figure 4: Multiple signal oscillation detection of a cryogenics valve.

In Figure 5, one of the cryogenics valves shows a faulty range of movement if compared to the other actuators. This excessive movement of the valve can be seen as an inefficiency of the control process, or most probably due to an abnormal heat load in the cell, or even a wrong mechanical valve setting. The algorithms detect the operational anomaly but they are not able to identify the root cause; a direct intervention from the system expert is necessary to fully understand the nature of the problem.

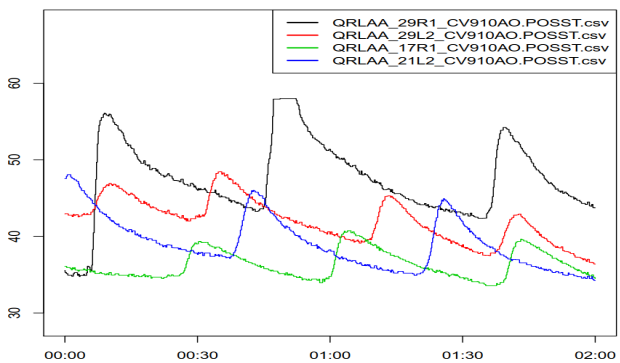

Figure 5: Faulty signal amplitude detection of a cryogenics valve.

In the last example, one of the actuator changes its offset. Figure 6 underlines the ability of the analytical algorithm to detect any change in the parametrization of sensors/actuators.

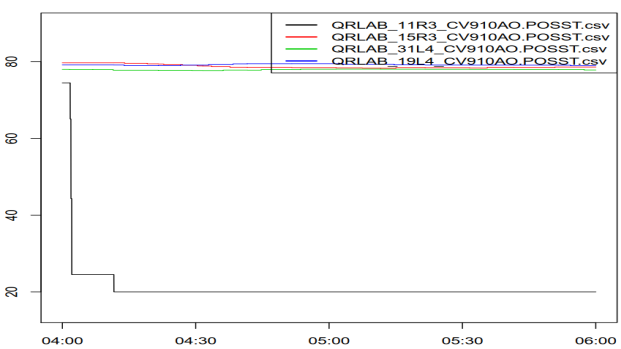

Figure 6: Signal offset detection of a clustered valve.

The above examples demonstrate the algorithms' ability to detect different types of anomalies that may affect the control systems. Moreover any change or upgrade in the system will be automatically be handled by the continuous learning phase without any direct intervention from engineers. Currently the analyses have been implemented as Spark jobs; they are daily executed to monitor several thousands of sensors and actuators, which otherwise could not been manually checked. The detected anomalies are published in a CERN internal website that engineers and operators can consult. Furthermore a specialized reporting tool has been designed and implemented in order to show the analytical results directly into the SCADA (Supervisory Control and Data Acquisition) applications at CERN [17]. The detection of these anomalies has allowed engineers to improve the tuning of thousands of regulation loops and reduce undesirable mechanical movements (e.g. wearing of valves due to abnormal opening/closing).

CONCLUSION AND FUTURE WORK

The paper describes three different algorithms that have been developed and integrated into the LHC control system for online detection of faulty sensors. Different data mining techniques have been used to extract patterns from the historical data; these patterns are then used for online anomaly detection. The three algorithms have been compared to understand their advantages and disadvantages. From this comparison, it is evident that the use of machine learning techniques makes the anomaly detection more generic and suitable for dynamic systems, since they are able to identify class of errors initially not foreseen by the experts. Finally, the usefulness of the presented methods has been demonstrated with real control data produced by the LHC cryogenics system. Nevertheless, the analytical solutions are so generic that they can be applied for anomaly detection in any other CERN domains.

As possible future work, these anomaly detection algorithms could be combined with a root-cause analysis. Currently, once an anomaly is discovered, it requires the manual investigation of an operator or system expert to identify their root cause. This last process could be partially automatized or supported by other analytical processes.

ACKNOWLEDGEMENT

The presented work has been achieved within the openlab collaboration between CERN and Siemens. A particular thanks to E. Blanco Vinuela, P. Gayet and F. Alves for their helpful contribution.

REFERENCES

[1] S. J. Qin, "Control performance monitoring – a review and assessment", *Computers & Chemical Engineering*, 23, 1998.

[2] L. Desborough, R. Miller, "Increasing customer value of industrial control performance monitoring - Honeywell's Experience", Chemical Process Control VI, in *Proc. of the Sixth International Conference on Chemical Process Control*, Tucson, Arizona, January 7-12, 2001 pp. 169-189.

[3] E. Keogh, S. Lonardi, B.C. Chiu, "Finding surprising patterns in a time series database in linear time and space", in

ISBN 978-3-95450-193-9

Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Edmonton, AB, Canada, 23–26 July 2002, pp. 550–556.

[4] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, "The KDD process for extracting useful knowledge from volumes of data", *Magazine Communications of the ACM*, 1996, pp. 27-34.

[5] E. Blanco, P. Gayet, "LHC Cryogenics Control System: Integration of the Industrial Controls (UNICOS) and Front-End Software Architecture (FESA) Applications", in *Proc. ICALEPCS'07*, Knoxville, TN, USA, October 2007, paper WOAA03.

[6] D. Berndt, J. Clifford, "Using dynamic time warping to find patterns in time series", in *Proc. AAAI-94 Workshop on Knowledge Discovery in Databases*, Seattle, Washington, USA, 1994.

[7] J. Listgarten, R. M. Neal, S. T. Roweis, A. Emili, "Multiple alignment of continuous time series", in *Proc. Advances in Neural Information Processing Systems*, 17, 2005, pp.817-824.

[8] J. Sun, H. Qu, D. Chakrabarti, and C. Faloutsos, "Neighbourhood formation and anomaly detection in bipartite graphs", Proceeding ICDM '05, in *Proceedings of the Fifth IEEE International Conference on Data Mining*, 2005, pp. 418-425.

[9] J. Sun, Y. Hie, H. Zhang, and C. Faloutsos, "Less is more: compact matrix decomposition for large sparse graphs", in *Proceedings of the SIAM International Conference on Data Mining*, 2007.

[10] R. J. Samworth, "Optimal weighted nearest neighbour classifiers", *The Annals of Statistics,* Volume 40, number 5, 2012, pp. 2733-2763.

[11] A. K. Jain, R. C. Dubes, "Algorithms for clustering data", *Englewood Cliffs*, 2012, pp. 55-222.

[12] U. Maulik, S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, pp. 1650-1654.

[13] Spark apache project, `https://spark.apache.org`

[14] Hadoop, `http://hadoop.apache.org/.`

[15] NumPy, `http://www.numpy.org`

[16] SciPy, `http://www.scipy.org`

[17] P. J. Seweryn, M. Gonzalez-Berges, J. B. Schoefield, F. Tilaro, "Data Analytics Reporting Tool for CERN SCADA Systems", presented at ICALEPCS'17, Barcelona, Spain, October 2017, paper TUPHA035, this conference.

ISBN 978-3-95450-193-9