Using the concept of data enclosing tunnel as an online feedback tool for simulator training

Laura Marcano[1,2] Anis Yazidi [3] Davide Manca [4] Tiina Komulainen [1]

[1] Department of Electronic Engineering, OsloMet – Oslo Metropolitan University, Norway
[2] Department of Electrical Engineering, University College of Southeast Norway (USN), Norway
[3] Department of Computer Science, OsloMet – Oslo Metropolitan University, Norway
[4] Department of Chemical Engineering, Politecnico di Milano, Italy
{Laura.Marcano,Anis.Yazidi,Tiina.Komulainen}@oslomet.no
davide.manca@polimi.it

Abstract

Feedback is one of the key factors that makes industrial simulator training an effective learning tool. Usually, the trainees receive feedback from the instructor, who guides them through the simulation tasks. However, nowadays the availability of expert instructors is scarce while the training demand is increasing. Therefore, there is a need for new simulator training practices that could allow the trainees to be more independent and decrease the need to rely so often on the instructor. This could be achieved by offering the trainees online automated feedback. This article presents a method for developing a tool meeting those requirements is presented. Simulation data were gathered representing different execution paths of the same scenario. Data were then analyzed and clustered using different clustering techniques. Interestingly, “good” and “bad” performances are shown to be separable using different techniques for clustering multivariate time series. Furthermore, we introduce the concept of enclosing data tunnel representing the trajectory of well-behaving execution paths in a reduced dimensional space. By conditioning the mal-behaving performances to be less than 20 % of the total simulation time inside the tunnel, an accuracy on 68 % was obtained. Being more flexible and allowing the mal-behaving performances to be inside the tunnel for a maximum of 35 % of the total simulation time, the accuracy of the enclosing tunnel was increased to 84 %.

Keywords: simulator training, online feedback, data clustering, enclosing tunnel, execution path

1 Introduction

A number of studies point out the importance of feedback during simulator training (Darken, 2009, Dozortsev, 2013, Håvold et al., 2015, Kluge et al., 2014, Kluge et al., 2009, Salas et al., 2012, Tichon and Diver, 2010). Feedback is a very effective learning mechanism that can be used to guide the trainees towards the development of a better performance. According to Salas et al. (2012) “Practice is most powerful when combined with timely, constructive, and diagnostic feedback”. Usually, trainees receive feedback from an expert instructor who guides them through the simulation scenarios. Commonly, expert instructors are experienced operators who have accumulated a great knowledge of the system through years. This dependency on expert instructors has raised concern in different industrial sectors given that many of the experienced operators have retired or will retire in the near future (Alamo and Ross, 2017, Dozortsev, 2013, Manca et al., 2012). Consequently, the availability of expert instructors is continuously decreasing while the demand for operator training continues increasing. Therefore, there is need for new (tutoring) methods and techniques that could help the current instructors to cope with the training demands by allowing the trainees to be more independent. One way trainees can be more independent is by offering them online automated feedback about their performance during the training scenarios.

The topic of automated feedback for simulator training has been mentioned in several studies, some of them indicate that the use of instructional tools embedded in the simulator can improve the efficiency of training (Bell et al., 2008, Malakis and Kontogiannis, 2012). Further, there are studies that present a method (Manca et al., 2014) or an already developed tool (Dozortsev, 2013) to give automated feedback.

In this work, a procedure for developing an automated feedback tool for a simulator training is presented. The procedure developed is based on the analysis of the data collected from different variations of the same training scenario. The analysis of the data allows defining a good performance reference. This work builds on previous ideas from our position paper (Marcano et al., 2017b).

In order to be able to provide online automated feedback, it is necessary to know the state of the system at all times. The state of the system can be determined based on some key variables that together give a suitable overview of the process. These variables can be compared to the defined good performance reference.

https://doi.org/10.3384/ecp18153132

Based on the results of the comparison the trainees can be informed whether they should reconsider the actions taken and try a different approach to solving the training task.

The case study we considered is a training scenario developed in K-Spice, a dynamic process simulation tool from Kongsberg Oil and Gas Technologies (Kongsberg, 2009). K-Spice resorts to a generic oil and gas production model. The goal of the studied training scenario is to increase the overall oil production flow with respect to the starting point.

In the next section, the procedure followed for developing the online feedback tool is described in detail, then the results obtained are presented, followed by the discussion, and finally some conclusions are drawn.

2 Methodology

The case study consists of a training scenario developed with the generic oil and gas production model integrated into the simulation tool K-Spice. Aim of the training scenario is to increase, in 30 min, the oil production flow with respect to the one given in the initial conditions of the simulation. The trainee must fulfill the goal without compromising the correct functioning of the process. In order to develop an online feedback tool for the case study, the following steps were followed.

2.1 Selection of variables

Figure 1 shows an overview of the generic oil and gas production model. The sections with the most relevant information of the process are the wells, the high-pressure separator (HP-separator), the export pump and the gas export compressor, the oil and gas export sections, and the high pressure flare (HP-flare). The variables studied were taken from these sections, being: 1) The total sum of outlet flows from the wells; 2) Inlet flow of the HP-separator; 3) Pump power consumption; 4) Compressor power consumption; 5) Oil export flow; 6) Gas export flow; and 7) HP-flare flow.

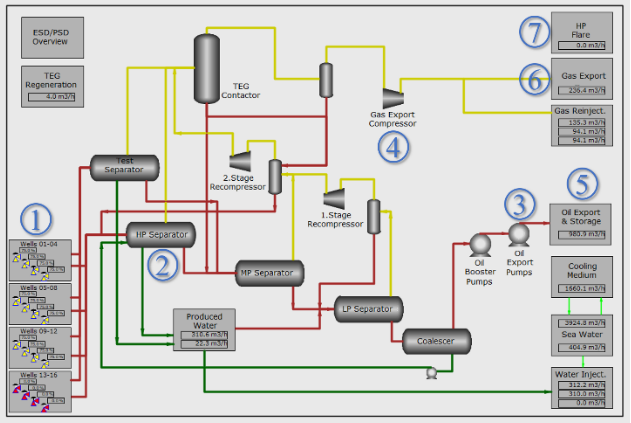


Figure 1. Overview of the generic oil and gas production process.

2.2 Data generation

In order to gather relevant data, a method to generate variations of the case study was developed. Each process variation was a random selection of five possible actions. The actions were defined based on the observations and results gathered from the simulator training sessions mentioned in Marcano et al. (2017a). During the development of the research (Marcano et al., 2017a), it was possible to extract knowledge about the students' understanding of the process and the probable actions they could execute. Based on this distilled knowledge, it was decided that a maximum of three actions could take place per variation of the case study, with a certain delay between them. The delay was set to 15 s, 45 s, 60 s, 120 s, and 180 s. These delays were chosen because during the simulator training sessions (Marcano et al. (2017a) we noticed that the trainees usually did not wait longer than 3 min to make changes in the simulation. The construction of one variation occurs as follows; first, a random action is chosen, and then, depending on the chosen first action there are some conditions that will determine the following random actions to choose among if any. The defined actions and the conditions triggering them are explained below.

1. *Opening the choke valve from a well.*

Opening a choke valve is the right execution path to follow when trying to increase the oil production in the process. Therefore, it is assumed that if the first decision of the trainee is to open a choke valve, then, if there are possible following actions, these will also be opening a choke valve. How much the choke is going to be opened is decided randomly between two options, being 85 % and 100 %. All choke valves in the simulation that are predetermined open, are open up to 75 %.

2. *Closing the choke valve of a well.*

Closing a choke valve is a wrong action to execute if the oil production needs to be increased. If the trainee is confused and closes a valve by mistake, then the next actions could, unfortunately, be to close even more valves. However, it could also happen that the student notices the mistake and tries to fix it by reopening the closed valve and opening an extra one. Then we randomly decide whether we will perform a sequence of 1 or 2 next actions. In case of choosing only one subsequent action, that second action could be either closing another valve or reopening the one closed. In case we choose two subsequent actions, then, the following two actions are to reopen the closed valve and open an extra one. How much the choke is going to be closed is decided randomly between two options, being 0 % and 65 %. For the opening, the same conditions explained in the first action are applied.

3. *Opening the pulse-controlled valve to the test separator.*

During the simulator training sessions carried out in Marcano et al. (2017a), it was noticed that few students opened a pulse-controlled valve thinking that it was a choke valve. This mistake was also noticed during the simulator training sessions performed later in 2017. There were just a few students who made the mistake but it seems to be common to happen. Consequently, it was decided to take it into account. However, given that opening a pulse-controlled valve is a rare mistake if this action takes place first, then, it will be the only action to be executed and there will be no subsequent actions.

4. *Opening the outlet control valve of the HP-separator.*

Opening the outlet valve of the HP-separator might occur due to the misconception that by increasing the outlet flow from the HP-separator the oil production would increase as well. The next step is to choose whether to proceed with a sequence of one subsequent action or two subsequent actions. If we randomly chose to follow with two actions, then, these were set to be the opening of choke valves. If only one action is following, then, this could be either opening a choke valve or a pulse-controlled valve.

5. *Increasing the pressure set point of the HP-separator.*

Increasing the pressure of the HP-separator leads the system to switch on the high-pressure flare. This action was defined to ensure the possibility of analyzing execution paths with a negative environmental impact. If only one more action follows this one, then, it could be either opening a choke valve or a pulse-controlled valve. If two actions follow, then, both will be to open a choke valve.

2.3 Data clustering

The data gathered in this work consists of multivariate time series. It is necessary to identify from the data what corresponds to well-behaving performances and what corresponds to mal-behaving performances. This way it is possible to make balanced groups for training and validation. In order to cluster the data, it is necessary to use a notion of similarity. This can be done by calculating the distance between every possible combination of pairs of execution paths. In this work, three methods for distance calculation were evaluated namely: Euclidean distance, dynamic time warping (DTW), and symbolic aggregate approximation (SAX). This was done in order to determine and select the most accurate method among such distances.

Euclidean distance

Given two time series X and Y of the same length N, (1) defines the Euclidean distance between them. Figure 2a shows an intuitive representation of the Euclidean distance (Lin et al., 2003).

$$D(X,Y) \equiv \sqrt{\sum_{i=1}^{N}(x_i - y_i)^2} \qquad (1)$$

Dynamic Time Warping (DTW)

The objective of DTW is to compare two (time-dependent) sequences $X := (x_1, x_2,...,x_N)$ of length $N \in \mathbb{N}$ and $Y := (y_1,y_2,...,y_M)$ of length $M \in \mathbb{N}$. These sequences may be discrete signals (time-series) or, more generally, feature sequences sampled at equidistant points in time (Müller, 2007). Being $\mathcal{F}$ a feature space, x_n, $y_m \in \mathcal{F}$ for $n \in [1{:}N]$ and $m \in [1{:}M]$. To compare two different features x, $y \in \mathcal{F}$, a local cost measure is needed, also referred to as local distance measure, which is defined to be a function $c{:}\,\mathcal{F} \times \mathcal{F} \rightarrow \mathbb{R}_{\geq 0}$ (Müller, 2007).

Typically, $c(x, y)$ is small (low cost) if x and y are similar to each other, and otherwise $c(x, y)$ is large (high cost). Evaluating the local cost measure for each pair of elements of the sequences X and Y , the cost matrix $C \in \mathbb{R}^{N\times M}$ defined by $C(n, m) := c(x_n, y_m)$ is obtained. Then the goal is to find an alignment between X and Y having minimal overall cost (Müller, 2007). Each matrix element (i, j) corresponds to the alignment between the points x_i and y_j. A warping path is created, which consists of a contiguous set of matrix elements that defines a mapping between X and Y (Keogh and Ratanamahatana, 2005). The signal with an original set of points X(original), Y(original) is transformed to X(warped), Y(original). However, even though DTW measures a distance-like quantity between two given sequences, it does not guarantee the triangle inequality to hold (Müller, 2007).

SAX (Symbolic Aggregate approXimation)

SAX allows a time series of arbitrary length N to be reduced to a string of arbitrary length w, ($w < N$, typically w << N). The alphabet size is also an arbitrary integer a, where $a > 2$. SAX uses an intermediate representation between the raw time series and the symbolic strings. First, the data is transformed into the Piecewise Aggregate Approximation (PAA) representation and then symbolize the PAA representation into a discrete string (Lin et al., 2003).

In dimensionality reduction via PAA a time series X of length N can be represented in a w-dimensional space by a vector $\bar{X} = \bar{x}_1, \ldots, \bar{x}_w$. The i^{th} element of $\bar{X}$ is calculated as follows (Lin et al., 2003):

$$\bar{x}_i = \frac{w}{N} \sum_{j=\frac{N}{w}(i-1)+1}^{\frac{N}{w}i} x_j \qquad (2)$$

Having transformed a time series database into PAA, a further transformation to obtain a discrete representation can be applied. SAX uses a discretization technique that produces symbols with equiprobability (Lin et al., 2003). If the original subsequences in the Euclidean distance are transformed into PAA representations, $\bar{X}$ and $\bar{Y}$, using (2), then a lower bounding approximation of the Euclidean distance between the original

https://doi.org/10.3384/ecp18153132

subsequences can be obtained (3), this is illustrated in Figure 2b (Lin et al., 2003).

$$DR(\bar{X},\bar{Y}) \equiv \sqrt{\frac{N}{w}}\sqrt{\sum_{i=1}^{w}(\bar{x}_i-\bar{y}_i)^2} \qquad (3)$$

If the data is further transformed into the symbolic representation, a MINDIST function that returns the minimum distance between the original time series of two words can be defined by (4), which is illustrated in Figure 2c (Lin et al., 2003).

$$MINDIST(\hat{X},\hat{Y}) \equiv \sqrt{\frac{N}{w}}\sqrt{\sum_{i=1}^{w}(dist(\hat{x}_i,\hat{y}_i))^2} \qquad (4)$$

The *dist*() function can be implemented using a table lookup as shown in Table 1 . Table 1 is for an alphabet of cardinality 4. The distance between two symbols can be read off by checking the corresponding row and column. For example, *dist*(a,c) = 0.67 (Lin et al., 2003).

Table 1. A lookup table used by the MINDIST function. This table was taken from Lin et al. (2003).

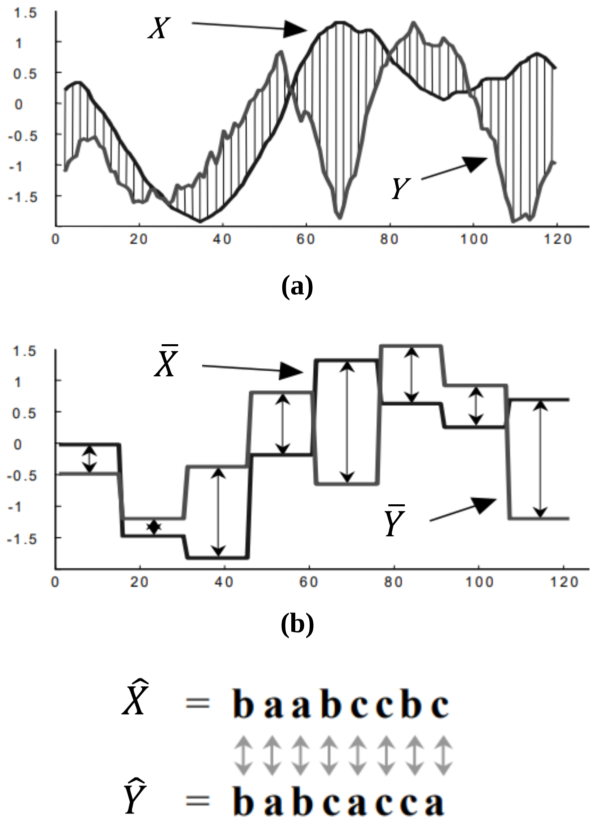
	a	b	c	d
a	0	0	0.67	1.34
b	0	0	0	0.67
c	0.67	0	0	0
d	1.34	0.67	0	0

Hierarchical clustering

Hierarchical clustering groups data over a variety of scales by creating a cluster tree. The tree is not a single set of clusters, but rather a multilevel hierarchy, where clusters at one level are joined as clusters at the next level. This allows deciding the level or scale of clustering that is most appropriate for the application required (MathWorks, 2018).

2.4 Samples selection

For this study, 75 out of 1145 possible variations were generated the way described in Section 2.2, this represents 6.5 % of all the possible combinations. The data were classified using the methods described in Section 2.3. Of the 75 variations, 2/3 of the data corresponding to the well-behaving execution paths, and 2/3 of the data corresponding to the mal-behaving execution paths were used for training i.e. 50 variations in total. The rest of the data was used for validation i.e. 25 variations (1/3 of the good performances, and 1/3 of the bad performances). For each time series, one sample was taken every 12 s during 30 min i.e. 150 data points, plus one additional point at time zero.



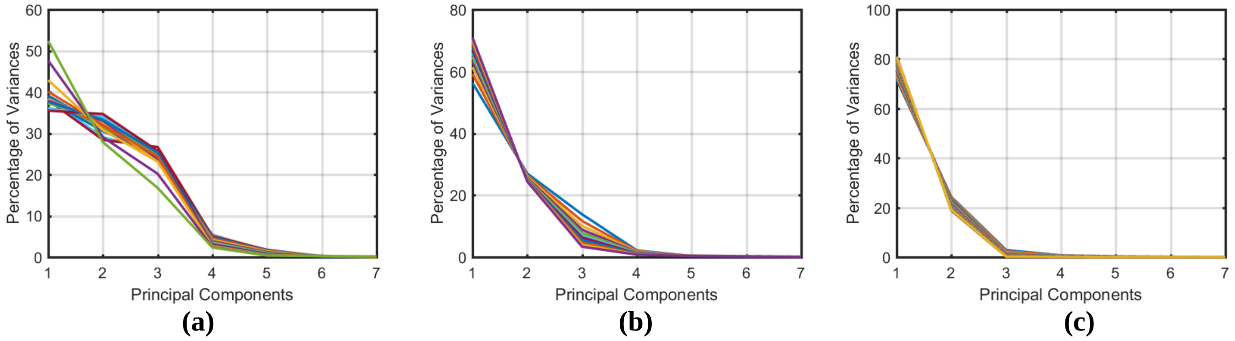
(c)

Figure 2. a) Euclidean distance between two time series. **b)** Distance measure defined for the PAA. **c)** SAX representations of two time series. This figure was taken from Lin et al. (2003).

2.5 Introducing the concept of enclosing tunnel

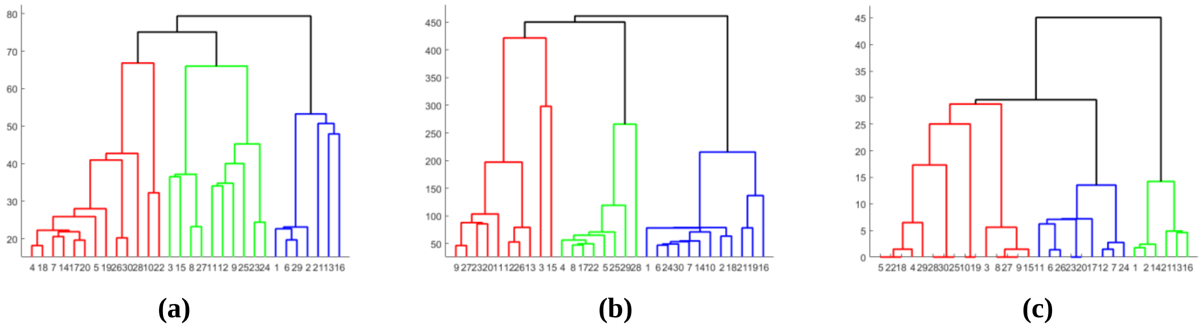
2.5.1 Data processing and dimensionality reduction

The training data was processed using PCA. In order to compare with each other all the variations in the training data, a PCA analysis was done for different time slots of the time series. For each time slot, the average of the data within the range was taken. The time moving average was calculated using the sliding window algorithm, with a window size of 35 samples. Which means that the first PCA calculated describes all the variations in the time span between the first and the 35th sample. The second PCA describes all the variations in the time span between the second and the 36th sample, and so forth until the entire time-range of 30 min (151 samples) is covered. After testing different sizes for the sliding window algorithm, a size of 35 samples was selected given that it was the one that resulted in the smoothest graphical representation of the data. The variance results show that the first and second principal components (PC1 and PC2, respectively) are the most representative, as shown in Figure 3. Figure 3a shows the first 19 sliding windows, it can be seen that the first two principal components describe approximately 70 to



(a) (b) (c)

Figure 3. Variances. a) Variances of the first 19 sliding windows. b) Variances from the 20th to the 30th sliding window. c) Variances from the 31st to the 117th sliding window.



(a) (b) (c)

Figure 4. Hierarchical clustering based on time series distances. a) Euclidean b) DTW c) SAX.

80 % of the data. Further, Figure 3b and Figure 3c show that the first two principal components describe approximately 90% of the data. Therefore, all analyses made are based on the first two principal components.

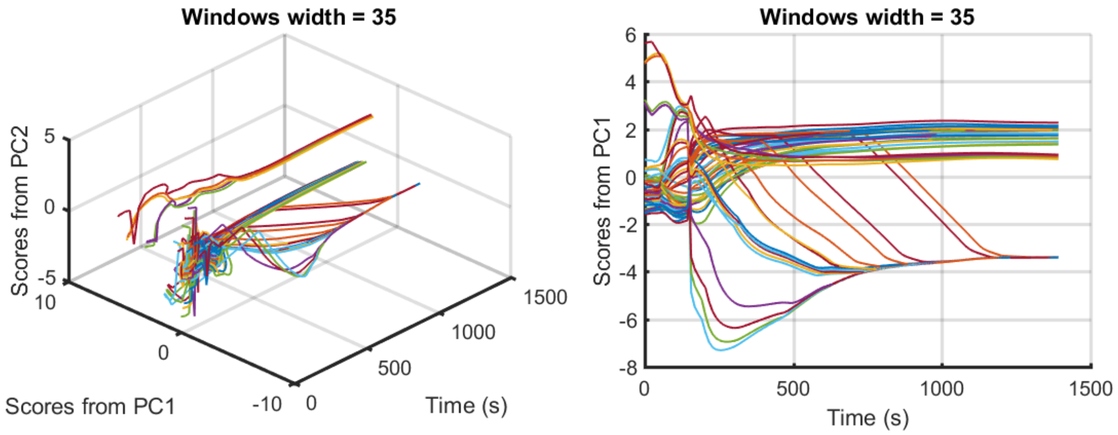
2.5.2 Data delimitation using an enclosing tunnel

The structure of the enclosing data tunnel is based on five main circles. The enclosing circles were calculated based on the data projected on the 2D plane formed by the data scores from PC1 vs the scores from PC2. In order to frame the data projected on this plane we resort to the minimal enclosing circle problem. The minimal enclosing circle problem consists of finding the circle of smallest radius that contains a given set of points in its interior or on its boundary. Jung's theorem states that every finite set of points with geometric span d has an enclosing circle with radius no greater than $d/\sqrt{3}$ (Weisstein, 2018). Each circle is located in a moment in time in which the well-behaving execution paths show a significant tendency of change. The data enclosing tunnel was constructed by creating a surface that connects each of the sections formed between the minimal circles.

3 Results

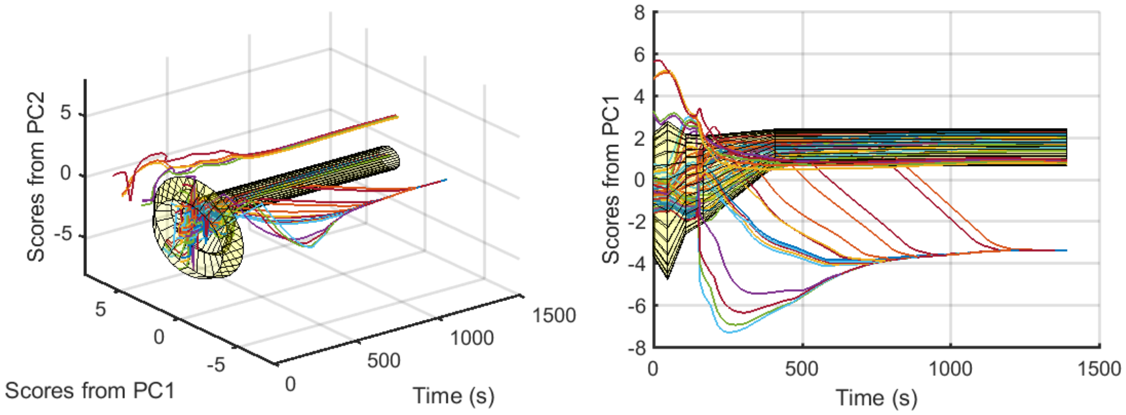
We started by clustering the raw data to separate the well-behaving and mal-behaving performances. The data clustering was carried out implementing the three methods described in the methodology. Once the distances were calculated with the three methods, they were clustered using hierarchical clustering. Figure 4 shows the three clustering trees obtained with each of the methods. In general, it can be seen that there are three main clusters formed by the data, given that three main groups (green, red and blue) were obtained with each method. However, the two main branches of the cluster tree obtained with the SAX method (Figure 4c) are more noticeable than the two main branches of the other two methods, Euclidean and DTW (Figure 4a, Figure 4b), which indicates that the clusters formed by the SAX method are defined more clearly. Numerically this is checked with the cophenetic correlation coefficient, which resulted to be 0.9347 for the clustering tree calculated with the Euclidean distances, 0.8769 for the clustering tree calculated with the method DTW, and 0.9392 for the clustering tree calculated with the method SAX. Therefore, the results obtained with SAX were the one used for classifying the data as good and bad performances.

Figure 6 shows the results of the data processing using PCA. Figure 6a depicts a 3D representation of the variation along time, of the scores obtained with the first two principal components. It can be noticed that there are three different patterns in the data. Figure 6b corresponds to an upper view of the previous one. It



(a) (b)

Figure 5. a) 3D view of the enclosing tunnel. **b)** 2D view of the enclosing tunnel.



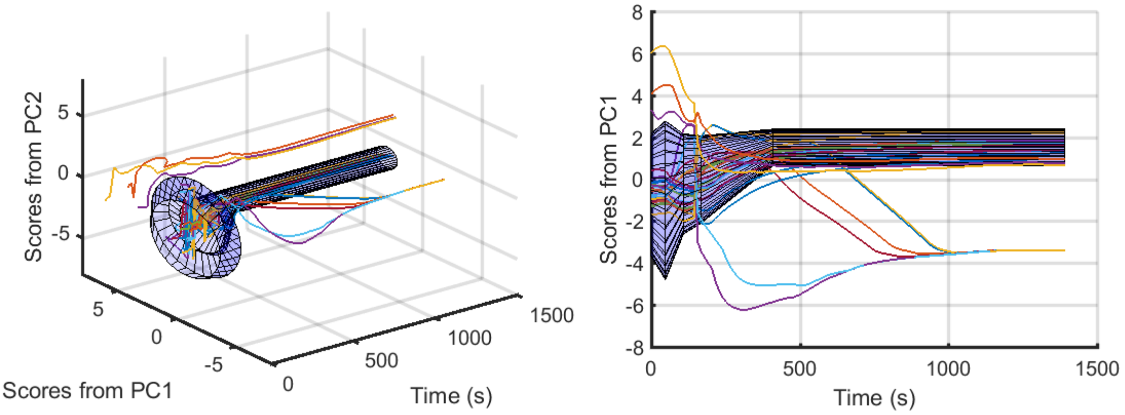
(a) (b)

Figure 6. a) Data scores from PC2 vs Data scores from PC1 vs Time. **b)** Data scores from PC1 vs Time.

shows the variation of scores obtained with the first principal component versus time, and three different tendencies of the data can also be appreciated.

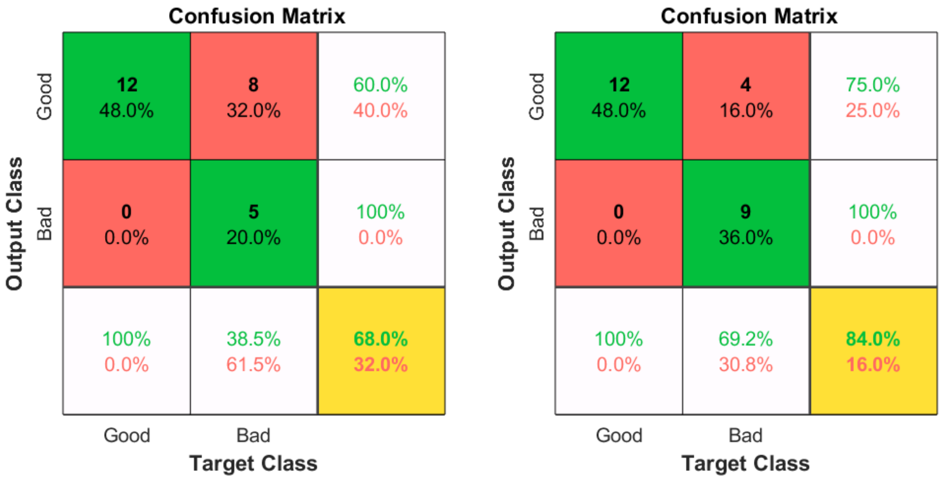
Figure 5 shows a 3D and 2D view of the tunnel enclosing the data that correspond to the good performances. The tunnel consists of five different circular sections that correspond to the minimal enclosing circle of the well-behaving execution paths in each section. All data that do not fall inside the tunnel corresponds to a bad performance. The trends that start being inside the tunnel but then go totally outside correspond to those actions where the outlet controlled valve of the HP-separator is opened. The trends that are above the tunnel correspond to those actions were the pressure set point of the HP-separator was increased.

In order to test the accuracy of the tunnel, the data left aside for validation was used. First, the validation data was projected on the PCA space calculated with the training data. Next, the processed validation data was plotted with the tunnel, as shown in Figure 7. Finally, the accuracy of the tunnel was determined by calculating the total amount of time that each good and bad performance spent inside the tunnel. In the case of the well-behaving execution paths, it was expected that they would be inside the tunnel at least 80 % of the total simulation time. While in the case of the mal-behaving execution paths, it is expected that they wouldn't be inside of the tunnel more than 20 % of the total simulation time. Based on these limits the accuracy calculated for the tunnel resulted to be 68 %, as shown in the confusion matrix presented in Figure 8a. The diagonal of the confusion matrix (green squares) represents the correct classifications. Figure 8a shows that 12 execution paths were correctly classified as “good”, and 5 execution paths were correctly classified as “bad”. On the other hand, the red squares show the incorrect classifications, and it can be seen that 8 execution paths were wrongly classified as “good”, these are false positives. In order to improve the accuracy of the tunnel, the tolerance of the bad performances inside of the tunnel was increased to 35 % of the total simulation time. This way the false positives were reduced from 8 to 4, given as a result an improved accuracy of 84 % as can be seen from Figure 8b.



(a) **(b)**

Figure 7. a) Validation data and 3D view of the enclosing tunnel. **b)** Validation data and 2D view of the enclosing tunnel.


Confusion Matrix
Output Class
Target Class
Confusion Matrix
Output Class
Target Class

Output Class	Good	Bad	
Good	**12** 48.0%	**8** 32.0%	60.0% 40.0%
Bad	**0** 0.0%	**5** 20.0%	100% 0.0%
	100% 0.0%	38.5% 61.5%	**68.0%** **32.0%**

Output Class	Good	Bad	
Good	**12** 48.0%	**4** 16.0%	75.0% 25.0%
Bad	**0** 0.0%	**9** 36.0%	100% 0.0%
	100% 0.0%	69.2% 30.8%	**84.0%** **16.0%**

Figure 8. a) Confusion matrix: mal-behaving execution paths inside of the tunnel for 20 % of the total simulation time. **b)** Confusion matrix: mal-behaving execution paths inside of the tunnel for 35 % of the total simulation time.

4 Discussion

In this work, the construction of the online feedback tool was based on the well-behaving execution paths. Consequently, it was necessary to find methods that could ease the laborious task of clustering data and identifying their typology. The results obtained with the data clustering techniques show that these are effective methods for finding similarities among data. Which is very useful when handling a large amount of data, such as those produced from simulator training scenarios.

The proposed enclosing tunnel could be used as an effective tool for generating online feedback. The data of a new trainee could be monitored, for instance, every two minutes. The first set of data should be projected on the PCA space, and later compared with the tunnel, if it is observed that the execution path is outside the tunnel a warning can be given to the trainee. If the execution path is inside the tunnel no warning should be generated. Later, in the next two minutes, the data of the last four minutes could be analyzed, and once again depending on the data trend it is decided if a warning should be given to the operator or not. This sequence should be repeated online over the total duration of the simulation scenario. Further, depending on the data behavior we could also determine the type of mistake made by the trainee, and more detailed feedback could be generated. This refers particularly to the cases in which the outlet valve of the HP-Separator is opened, and when the pressure set point of the HP-Separator is increased. These two cases present a very differentiated behavior around the tunnel, therefore it could be easy to identify them. However, the trends for the cases where the outlet control valve of the HP-Separator is opened may take several minutes before leaving the tunnel. These are the cases that were classified correctly by allowing them to be inside the tunnel 35 % of the total time.

In general, this method could be used for different training scenarios. This procedure shows that an enclosing tunnel, based on good performances, can be designed for any kind of scenario, thus online feedback can be offered to the operators, giving them more training independence.

5 Conclusion and future work

The data clustering methods implemented, Euclidean distance, DTW, and SAX showed to be effective for finding similarities among data. Of the three methods, SAX is shown to be the most effective of all with a cophenetic correlation coefficient of 0.9392. The clustering of the data helped to identify among the entire data set the well-behaving execution paths, which were used to design the online feedback tool for simulator training. The online feedback tool designed consists of an enclosing data tunnel. The tunnel developed has in principle an accuracy of 68 %. This value was calculated by allowing the mal-behaving execution paths to be inside the tunnel no more than 20 % of the total simulation time. However, with a more flexible tolerance (bad performances allowed to be inside the tunnel 35 % of the total simulation time) the total accuracy of the tunnel could be increased up to 84 %. It was demonstrated that it is possible to develop a method that can be used to generate automated online feedback, thus opening the possibility of more independent simulator training sessions.

Future work includes improving the accuracy of the tunnel without increasing the tolerance for mal-behaving execution paths. This could be done by increasing the amount of training data, so more differences can be noticed among the time series. Additionally, the method should also be improved so that it can detect if more specific requirements have been fulfilled by the operator. Furthermore, enclosing tunnels constructed for different training scenarios should be compared to each other in order to determine if a single generic tunnel could be designed to be used for different training scenarios.

References

Alamo, J. & Ross, M. 2017. Solve operator training in a challenging refining industry. *Hydrocarbon Processing,* 96**,** 67-69.

Bell, B. S., Kanar, A. M. & Kozlowski, S. W. J. 2008. Current issues and future directions in simulation-based training in North America. *The International Journal of Human Resource Management,* 19**,** 1416-1434. 10.1080/09585190802200173

Darken, R. P. 2009. Identifying the limits of training system effectiveness through taxonomies of human performance. *Theoretical Issues in Ergonomics Science,* 10**,** 231-243. 10.1080/14639220802151625

Dozortsev, V. M. 2013. Methods for computer-based operator training as a key element of training systems (Present-day trends). *Automation and Remote Control,* 74**,** 1191-1200. 10.1134/S0005117913070102

Håvold, J. I., Nistad, S., Skiri, A. & Ødegård, A. 2015. The human factor and simulator training for offshore anchor handling operators. *Safety Science,* 75**,** 136-145. http://dx.doi.org/10.1016/j.ssci.2015.02.001

Keogh, E. & Ratanamahatana, C. A. 2005. Exact indexing of dynamic time warping. *Knowledge and Information Systems,* 7**,** 358-386. 10.1007/s10115-004-0154-9

Kluge, A., Nazir, S. & Manca, D. 2014. Advanced Applications in Process Control and Training Needs of Field and Control Room Operators. *IIE Transactions on Occupational Ergonomics and Human Factors,* 2**,** 121-136. 10.1080/21577323.2014.920437

Kluge, A., Sauer, J., Schüler, K. & Burkolter, D. 2009. Designing training for process control simulators: a review of empirical findings and current practices. *Theoretical Issues in Ergonomics Science,* 10**,** 489-509. 10.1080/14639220902982192

Kongsberg. 2009. *K-Spice: A new and powerful dynamic process simulation tool* [Online]. Available: https://www.kongsberg.com/en/kongsberg-digital/news/2009/june/0625_kpice/ [Accessed April 30 2018].

Lin, J., Keogh, E., Lonardi, S. & Chiu, B. 2003. A symbolic representation of time series, with implications for streaming algorithms. *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery.* San Diego, California: ACM. 10.1145/882082.882086

Malakis, S. & Kontogiannis, T. 2012. Refresher Training for Air Traffic Controllers: Is It Adequate to Meet the Challenges of Emergencies and Abnormal Situations? *The International Journal of Aviation Psychology,* 22**,** 59-77. 10.1080/10508414.2012.635127

Manca, D., Nazir, S., Colombo, S. & Kluge, A. 2014. Procedure for Automated Assessment of Industrial Operators. *CHEMICAL ENGINEERING,* 36**,** 391-396.

Manca, D., Nazir, S., Lucernoni, F. & Colombo, S. 2012. Performance Indicators for the Assessment of Industrial Operators. *In:* IAN DAVID LOCKHART, B. & MICHAEL, F. (eds.) *Computer Aided Chemical Engineering.* Elsevier. http://dx.doi.org/10.1016/B978-0-444-59520-1.50143-3

Marcano, L., Komulainen, T. M. & Haugen, F. A. 2017a. Implementation of performance indicators for automatic assessment. *In:* ANTONIO ESPUÑA, M. G. & LUIS, P. (eds.) *Computer Aided Chemical Engineering.* Elsevier. https://doi.org/10.1016/B978-0-444-63965-3.50497-9

Marcano, L., Yazidi, A., Ferati, M. & Komulainen, T. 2017b. Towards Effective Automatic Feedback for Simulator Training. *Proceedings of the 58th Conference on Simulation and Modelling (SIMS 58) Reykjavik, Iceland, September 25th – 27th, 2017.* Linköping University Electronic Press, Linköpings universitet.

Mathworks. 2018. *Hierarchical Clustering* [Online]. Available: https://se.mathworks.com/help/stats/hierarchical-clustering.html [Accessed 25 April 2018].

Müller, M. 2007. Dynamic Time Warping. *Information Retrieval for Music and Motion.* Berlin, Heidelberg: Springer Berlin Heidelberg. 10.1007/978-3-540-74048-3_4

Salas, E., Tannenbaum, S. I., Kraiger, K. & Smith-Jentsch, K. A. 2012. The Science of Training and Development in Organizations: What Matters in Practice. *Psychological Science in the Public Interest,* 13**,** 74-101. 10.1177/1529100612436661

Tichon, J. & Diver, P. 2010. Plant operator simulation: benefits and drawbacks for a construction training organization. *Cognition, Technology & Work,* 12**,** 219-229. 10.1007/s10111-010-0140-9

Weisstein, E. W. 2018. *Minimal Enclosing Circle* [Online]. Available: http://mathworld.wolfram.com/MinimalEnclosingCircle.html [Accessed 24 April 2018].