Journal of Manufacturing Systems 63 (2022) 304–316

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Data-based model maintenance in the era of industry 4.0: A methodology

Paul-Arthur Dreyfus [a,*], Antoine Pélissier [a], Foivos Psarommatis [b], Dimitris Kiritsis [a]

[a] ICT for sustainable manufacturing, EPFL, Lausanne, Switzerland
[b] SIRIUS, University of Oslo, Gaustadalléen 23 B, N-0373 Oslo, Norway

ARTICLE INFO

Keywords:
Drift handling
Concept drift
Zero defect manufacturing
Machine learning
Data-based model maintenance
Industry 4.0

ABSTRACT

Despite the high number of investments for data-based models in the expansion of Industry 4.0, too little effort has been made to ensure the maintenance of those models. In a data-streaming environment, data-based models are subject to concept drifts. A concept drift is a change in data distribution which will, at some point, decrease the accuracy of the model. To address this problem, various frameworks are presented in the literature, but there is no optimal methodology for implementing them. This paper presents a methodology to implement a problem-oriented complete solution to ensure the maintenance of an industrial data-based model. The final drift-handling solution is composed of a sampling decision system and an update system. The methodology begins with a concept-drift identification phase. Solutions are then pre-selected based on the identified concept drifts. Next, an optimization problem is designed to select the solution that optimizes the costs and respects the constraints. To better link the concept drift characteristics and the drift-handling solutions, a causal concept-drift classification system is proposed. The industrial implementation of such a solution is discussed and several questions are raised. This paper presents an original and detailed methodology that shows encouraging results to address the model-maintenance challenge; however, concept drift identification, and links between concept-drift characteristics and drift detection, require further research.

1. Introduction

With the emergence of Industry 4.0, more and more processes are monitored digitally, thus continuously generating tremendous quantities of data. Data accessibility enables the implementation of impactful data-driven technologies, which can lead to higher levels of sustainability [1,2]. For example, Zero Defect Manufacturing (ZDM) is one key area of Industry 4.0 where data-driven technologies are utilized to improve product and process quality [1,3]. More specifically, virtual metrology and predictive maintenance are two data-driven concepts within ZDM that are heavily dependent on data, as their performance relies on the accuracy and adaptability of the corresponding models. Furthermore, another topic that is gaining significant attention and appreciation from the scientific and industrial communities, requiring data and adaptability, is digital twins (DT) [4,5]. An important factor that strengthens even more the need for more accurate data-driven solutions is the fact that data-driven methodologies perform significantly better than traditional analytical solutions; therefore, data-driven technologies constitute a viable alternative [6–8]. Currently, most of the developed models are implemented on a data stream often considered stationary [9]. However, this is usually not the case, which means the models become obsolete over time. The cause of this phenomenon is often referred as *concept drift* (CD), which is defined as *unpredictable changes in the data stream distribution over time* [10]. CD is becoming a foundational aspect in the well-known ZDM paradigm. In other words, considerable effort has been invested in the *development* of the industrial data-based models, but little attention has been paid to the *maintenance* of those models.

In the industrial environment, models estimate quantities which are usually expensive or difficult to measure. Defining when to measure them and how to use the data to update the model are foundational issues for data-based model maintenance. Without those considerations, the implemented model is bound to fail. This failure could even be harmful, depending on the critical nature of the industry being supported. The estimation model must adapt to its environment. In this context, the use of model-based approaches is detrimental as all sources of the CD need to be known and modeled in advance, which is not realistic in practice. A data-based model can use the data to update itself, and this the focus of this paper. Frameworks for maintaining data-based models exist in the state-of-the-art form; however, a methodology for optimal implementation is currently lacking in the literature.

This paper proposes a new methodology for designing and

* Correspondence to: EPFL SCI-STI-DK, ME A1 391 (Bâtiment ME) Station 9, CH-1015 Lausanne, Switzerland.

https://doi.org/10.1016/j.jmsy.2022.03.015
Received 12 November 2021; Received in revised form 16 January 2022; Accepted 31 March 2022
Available online 13 April 2022
0278-6125/

implementing the most suitable solution for addressing CD in a defined environment. The characteristics of the involved CDs are used to explore the different solutions through the steps of the methodology. An optimization problem is designed to select the most suitable solution among the various possible algorithms. To support the methodology, a new causal approach for CD type is proposed, thus introducing a new type of CD called upstream CD.

2. Literature review

The literature reveals proposed solutions that are designed to address CD issues. CD handling has been studied from diverse perspectives in numerous fields, and some examples are presented in this section. This will provide an overview of similar research and prepare for the proposed methodology to be introduced.

First, CD handling frameworks have been proposed in various fields. A framework is proposed in the novelty detection field [11] that tackles many questions related to the different mechanisms for handling CD. The framework divides the process into two steps: the offline phase when the model is designed and the online phase when the model is running. For the online phase, questions about the use of external feedback or forgetting mechanisms are addressed. Other topics, such as the treatment of outliers or recurring contexts, are also considered when proposing an adequate solution. In the virtual metrology field, a framework is proposed with a direct industrial approach [12]. Virtual metrology involves estimating the quality of a product using production process data to avoid costly physical measurements. The framework is composed of numerous elements including, among others, data pre-processing, the sampling decision system sampling decision system (SDS), model updating, as well as the model connection to the manufacturing execution system. Each step is tackled from a practical and industrial point of view. Updating the system and SDS are discussed in detail.

Second, numerous specific solutions for drift handling have been implemented in several fields. For instance, in the domain of active learning for data stream, one paper [13] proposes a solution that includes the measuring cost constraint in the solution. This is an attractive solution, given its industry-oriented approach. If the allocated budget is exceeded while a sampling is required, the measurement is not performed and there is no update. Different sampling strategies are presented such as random strategy or variable uncertainty strategy; however, the paper does not focus on the model update. Another paper proposes a unique, complete solution in the field of semi-supervised learning called SAND [14]; it is based on a semi-supervised adaptive novel class detection and classification over Data Stream. This solution uses an ensemble classifier composed of k-NN type models to classify the new incoming data. Outlier detection is applied to each new instance to identify the emergence of a novel class using novelty detection technique. A change detection technique is applied to the classifier confidence estimates to actively request samples for updating the classifiers. This solution makes it possible to reduce the measurements while their SDS is based on the classifier confidence estimates.

As described, frameworks do exist to structure the CD-handling solutions as well as solutions that have been implemented on specific applications. However, no methodology has been proposed to implement them. The CDs are never identified and characterized to support solution optimization. In the present research, a context-oriented methodology is proposed based on the link between the solution performance and the drift characteristics, thus enabling solutions to be more robust and generalizable. Many studies have stressed the need for generalized solutions that address the CD [15–17]; consequently, the proposed methodology considers the full CD handling framework from the SDS to the updating system (US) as well as the industrial cost and constraint to select the optimal solution.

In the following section, the methodology to implement the maintenance of a data-based model solution is described. Moreover, a simulation that illustrates the proposed methodology is presented. Following this, a general discussion on the importance of such a methodology for the industry is presented. Limitations of the research are also discussed, and some further directions for research are proposed.

3. Methodology for maintenance of a data-based model

This paper proposes a solution to ensure the maintenance of an industrial data-based model. The operational framework to consider for the proposed solution is presented in Fig. 1, which gives an overview of the different involved systems.

Process data, which can be machine parameters or sensor data, are fed as inputs of the framework. They are used as inputs in the estimation model as well as in the CD detector [18,19]. The model returns the estimation of an unavailable physical variable and can be written as follows:

$$Y = f(X)$$

where X is the measured inputs variables, Y the targeted output variable, and f the estimated physical model.

The CD detector returns an alarm if there is a CD in the data stream. The outputs of the two previous blocks as well as time can be used in the SDS. The SDS defines when to measure new samples [20]. It triggers the sampling. Then the studied process which does the measuring. When new examples are available, they are used in the US to build an updated training set. This training set is given to the model as well as to the CD detector to keep them updated. The CD detector and time are dash-line blocks because they might not be used, depending on the chosen sampling strategy. Indeed, the SDS will need time if a passive strategy is chosen, and it could need a CD detector for an active strategy. These strategies will be discussed later in detail. There is a dash-line arrow between the model and SDS because this relationship is application dependent. For instance, the model estimate in virtual metrology is taken into account by the SDS.

The model maintenance solution regroups the SDS and US. There are many possible combinations and selecting the optimal one can be complex. To help with this task, a methodology is proposed for industrial implementation. The choice of the estimation model is independent of the methodology, which might or might not already be implemented. Its influence on the solution selection is not discussed in this paper. The proposed methodology is presented in Fig. 2.

The different steps of the methodology are briefly presented are follows:

1. **Concept drift identification:** Identifies the characteristics of the involved CD. Those characteristics can afterwards be leveraged to guide the selection of a CD-handling solution. This is subdivided in three steps:

- o **Knowledge-based concept drift identification:** Uses the domain experts' prior knowledge to identify the CD characteristics and CD sources. This makes it possible to extract relevant information for the data acquisition, such as the required measurement sampling frequency or duration.
- o **Data acquisition:** Performs measurements used in the data-based CD identification and in the off-line solution evaluation.
- o **Data-based concept drift identification:** Extracts information from the measured data to identify characteristics of the involved CDs such as the recurrence or geometrical properties. This makes it possible to complete the CD description to enable the CD handling solution pre-selection.

These three steps can be repeated (dash line on Fig. 2) until the obtained information is sufficient to move to the second stage of the methodology.

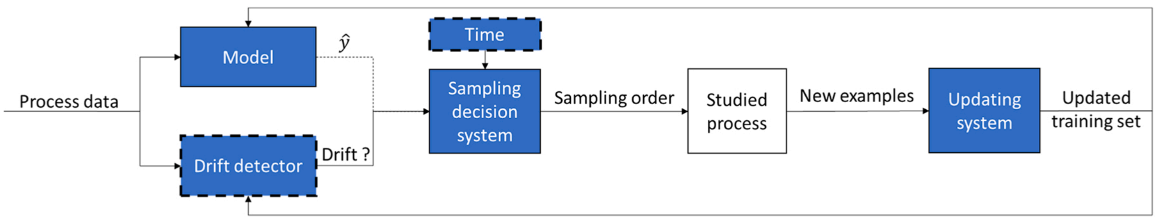

Fig. 1. Operational framework.

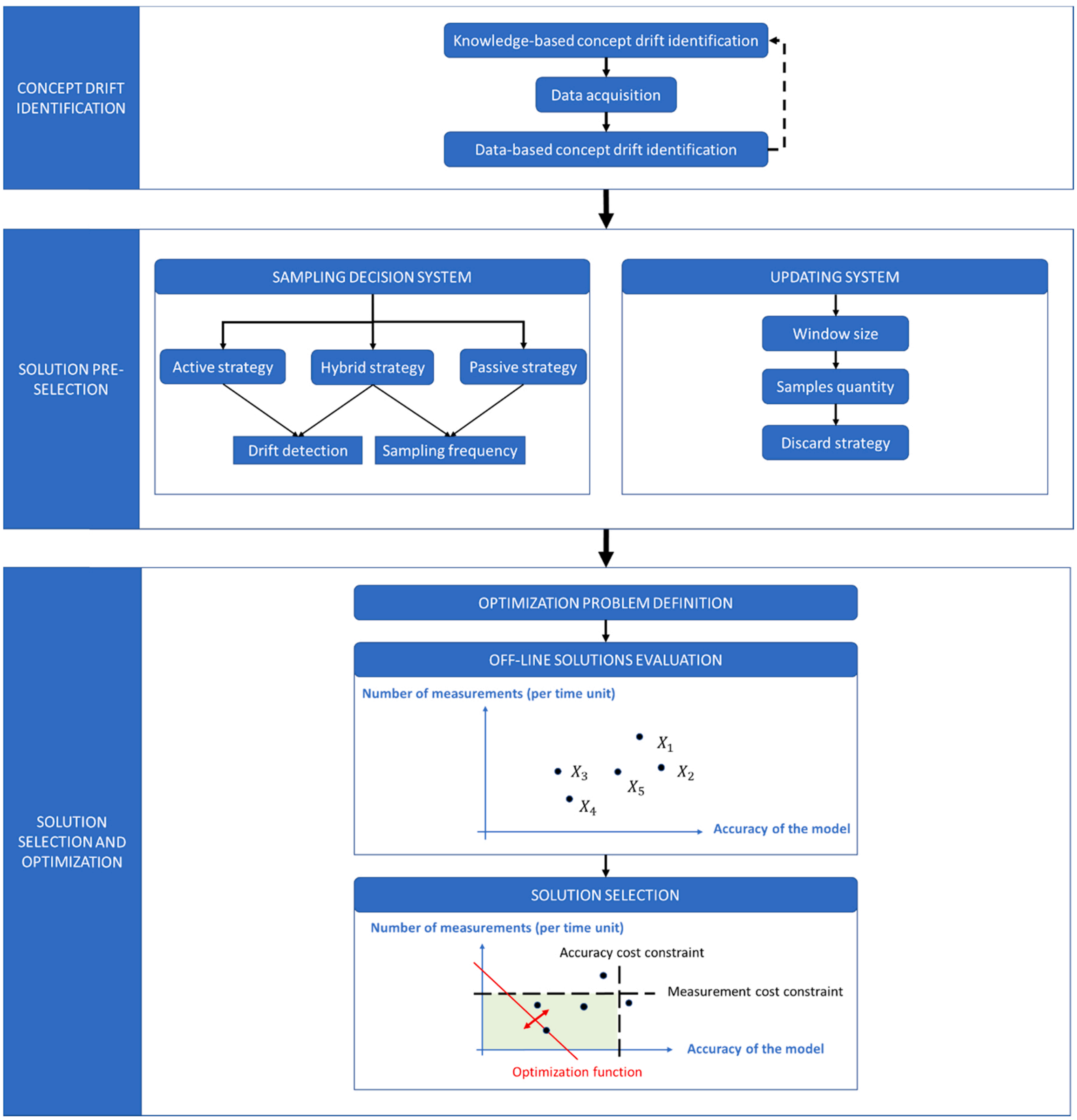

Fig. 2. Methodology.

2. **Solution pre-selection:** Perform a reduction of the solutions space, by keeping only the solution that is known to be performant on the identified CD. The solutions are composed of a SDS as well as a US.

o **Sampling decision system (SDS)**: Defines when to measure new samples. A good solution would make it possible to optimize the measurements cost by maintaining good model accuracy. **Active**, **hybrid** and **passive** strategies are the three different alternatives for SDS, as defined below:

- **Passive strategy** is a triggering measurement based on a pre-defined sampling frequency.
- **Active strategy** is based on CD detection to trigger a measurement when a CD is detected.

- **Hybrid strategy** is the combination of active and passive strategies.

o **Updating system (US):** Defines how to update the model when the new training data are available. This affects the model adaptation speed as well as the model performance described by the stability plasticity dilemma. Three criteria must be carefully tuned: the size of the moving window, the number of new examples to use, and the strategy to discard the less informative samples from the current update training set.

The links relating the CD characteristics to the possible solutions indicate how those systems are pre-selected. Then the most performant solution needs to be selected.

3. **Solution selection and optimization:** Evaluates the pre-selected solutions to select the best one using an optimization problem.

o **Optimization problem definition:** Defines the optimization problem and its parameters based on both the measurement and model accuracy constraints and cost.
o **Off-line solutions evaluation:** Evaluates the pre-selected solutions. Each solution is implemented on a testing dataset to compute their relative accuracy and the number of measurements needed to reach it. This makes it possible to map the solution space on an optimization graph (black dots on the graph).
o **Solution selection:** Selects the optimal solution that minimize the objective function (red slope on the graph) and satisfies the constraints (black dash line).

The different elements of the proposed methodology are discussed in more detail in the following sub-sections.

3.1. Concept drift identification

In order to design the optimal model maintenance strategy, the CD characteristics must be known. In the current research, an original CD definition and characterization is used. CDs are defined as events that induce a significant drop in the accuracy of the estimated model. This definition of definition makes it possible to focus on the application of CD-handling methodology, thus preserving the model's accuracy. CDs are classified on the basis of three degrees of freedoms: their types, recurrency, and geometry.

CD type describes the visibility and the relevance of the CD and is composed of two new degrees of freedom:

- **CD visibility:**

o **Visible CD** are visible on the X.
o **Hidden CD** are not visible on X but only on Y.

- **Causal position of the CD**:

o **Upstream CD** does not change the relation between X and Y.
o **Inside CD** changes the relation between X and Y.

CD recurrence gives information about its criticality. Indeed, CDs with high recurrence may be more problematic than CDs that appear only one time. In the current research, CD recurrence refers to the occurrence frequency characterized by the time interval between two repetitions. This should not be confused with CD drift duration, which can be shorter or longer than the time interval between two repetitions, leading to a CD overlap. The geometric properties are characteristics that provide information about the shape of a CD. They regroup for instance the magnitude, duration, and maximum slope of the drift. They are often qualitatively described as sudden or gradual. A more detailed description of those CD characteristics is given in the Annex.

It is foundational to the proposed methodology to have a good understanding of the CD so as to implement the adapted CD-handling approach. CD identification can be separated into three steps: knowledge-based CD identification, data acquisition, and data-based CD identification. For each of these steps, the identifiable CD characteristics and related methods are presented. Their relevance to the methodology is also discussed.

3.2. Knowledge-based concept drift identification

Knowledge of the studied environment makes it possible to estimate intuitively the sources as well as the properties of potential CDs. Experts in the application field are required to identify as much information as possible regarding the involved CDs. The sources of the CD should first be identified; following this, the characteristics of the CD can be studied.

Trying to connect the different sources of CDs makes it possible to establish the causal relationships between variables. Accordingly, it is possible to identify the type of CD related to each source. A new tool has been designed with this purpose in mind, and it can be found in the Annex. This tool makes it possible to extract the CD type for each potential identified source. It must be pointed out that the CD type depends not only on the environment but also on the measured variables X. In that way, knowing the causal relation between measured and latent variables could lead to enhancing the estimated model set up.

In terms of geometry, it is difficult to obtain a precise description. However, based on a qualitative CD description magnitude, the duration or even the slope of the concept drift can be roughly estimated using the expert's knowledge.

In the same vein, recurrency can be roughly estimated in some cases based on the case study prior knowledge. For instance, in a manufacturing environment, if the tool wear is identified as a source of CD, then experts can roughly estimate the frequency of the necessary tool change. Therefore, concerning the geometry, no specific tool, but only prior knowledge, is required at this stage to estimate the recurrency of a CD.

In the next stage, data is acquired to enable the drift identification and the solution characterization. This stage makes it possible to design the data acquisition parameters:

- **The measured variables X** may induce a more hidden CD or CDs that are harder to detect.
- **The measurement sampling frequency** will be based on the Nyquist law make some CDs appear as noise.
- **The measurement sampling duration** may make the dataset not representative of the studied system as events could be missed or too slow to be impactful yet.
- **The number of examples** may limit the utilization of particular machine-learning algorithms.

3.3. Data acquisition

CDs cannot all be identified in off-line mode using the expert's knowledge. Measurements must be made to visualize and characterize CDs. The knowledge-based CD identification stage becomes fundamental to orient the selection of the data acquisition parameters previously defined. This makes it possible to reduce costs related to the measurement, which are difficult to estimate at this stage of the methodology. Data acquisition provides for building a dataset to later identify CD characteristics and evaluate the pre-selected maintenance solutions based on the proposed framework. The performance of the final solution will depend to a great extent on the quality of the dataset acquired, which itself depends on numerous parameters defined in the previous step. A user would in some cases want to leverage already available data so as to skip the data acquisition step. However, a close inspection of data acquisition parameters need to be done to validate the usability of the dataset.

3.4. Data-based concept drift identification

This step makes it possible to identify the characteristics of the involved CD by using the dataset built during the acquisition phase. However, the CD source cannot be directly identified, unless alarms are available in the dataset to warn of an incoming CD—for instance, a maintenance on a machine. The CD source can be identified by cross checking the characteristics identified in the data-based and knowledge-based identification phase. Knowing the CD source also makes it possible to reduce some CD effects by improving the studied process.

The type of drift is easily detected as X and Y are measured in the data acquisition step. If X and Y are changing, this indicates a visible upstream CD. If X is changing but not Y, this indicates a visible inside CD. Moreover, if Y is changing but not X, this indicates a hidden inside CD. Drift type can be easily used to correlate information from both knowledge-based and data-based phases.

The data-based model makes it possible to identify the occurrence frequency of a CD, providing for a clear definition of the potential source that had not been identified in the knowledge-based method. By contrast, if the sources are identified and known, the frequencies can be obtained by comparing the time between similar patterns.

CD geometry is easily identified using visual tools that allow CD magnitude to be represented over time [21]. This makes it possible to define quantitatively the CD geometry. This method could be used for different kinds of geometric characteristics, thus specifying or completing the information obtained in the knowledge-based method.

The overall identification process can be repeated if the CD identification seems incomplete after the data-based CD identification. This makes it possible to target new or more precise characterizations and adapt the dataset for better information extraction. Once the involved CDs are fully identified, it is possible to move to the next step—the solution pre-selection.

3.5. Solution pre-selection

At this stage of the methodology, the characteristics of the CD, involved in the working environment, are supposed to be known. Based on this knowledge, adequate solutions composed of a SDS, and an US must be pre-selected. They are discussed in this section.

3.6. Sampling decision system

To ensure the model's long-term sustainability, the training dataset must be regularly updated with fresh data. The SDS is the component that handles the measurement strategy, there are three types of strategies a SDS can manage.

Passive strategies, also called "time-based" strategies, sample measurement without any explicit detection. The measurements are made at a fixed frequency, which allows them to handle both hidden and visible CDs. A passive strategy does not require the implementation of a data-based algorithm; the only parameter to tune is the timer sampling frequency. Numerous timers can be defined, depending on the number of drifts. This is the simplest strategy, but it is not the most optimal one in terms of the number of measurement and CD handling.

The *active strategy* or "event-based strategy" decides to sample measurements based on a CD detector. Unlike the passive strategy, this approach can only deal with visible CDs. However, using real-time data can optimize CD rejection as well as the number of measurements needed. As it is always unlikely to have only visible CDs, a timer taken from the passive strategy can be added to the active strategy to act as a safeguard. In this case, its sampling frequency can be optimized, depending on the drift detection performance [12,22,23].

The *hybrid strategy* is the most complete approach as it combines the active and passive ones. When both visible and hidden CDs are present, the hybrid strategy should be used. Timers should be engineered for every hidden CD, and a drift detector should be implemented to deal with visible CDs.

3.7. Sampling frequency

The sampling frequency is defined by the geometry of the involved CD as well as the occurrence frequency. If the CD is gradual, the sampling frequency will be set based on the ratio of the CD slope and the acceptable drop in the accuracy of the model. If the CD is brutal, its slope will be infinite, and the sampling frequency should therefore be defined based on its occurrence frequency. The frequency can also be determined experimentally by performing tests with the training dataset [20]. It must be emphasized that too high a frequency would induce a high measuring cost. It could also affect the model's accuracy if the chosen samples are not providing relevant information to the update. Too low a frequency would miss too much CD, and the model would become obsolete over time.

3.8. Drift detection

Drift detection algorithms are data-based algorithms. In this paper they are classified into three categories: statistical tests, clustering, and "in-built"-based methods. Statistical test methods compare two data distributions to spot any significant change that would result in a CD. Distance functions are used to compare and quantify historical data distribution with the new data distribution [24]. Clustering methods, which are the most popular family of drift detection, examine the change in data density. Clusters are used to identify concepts. Several different clusters can coexist at the same time [11]. "In-built" methods are drift detectors integrated into the estimation algorithm. Most of the time, they will estimate the uncertainty of every inference and threshold it. Currently, there is no way to choose a category of algorithms that is dependent on the involved CD characteristics. Indeed, the literature does not contain research justifying the chosen drift detector based on the identified characteristics of a CD, and this will be a major gap to study in the future. Nevertheless, it is possible to discuss guidelines to tune the drift detector based on CD characteristics. Indeed, all the approaches have one or multiple hyper parameters to set the sensitivity of the drift detector. In the following paragraphs, this sensitivity is related to CD characteristics.

Concerning the CD geometry and particularly the CD magnitude, a higher sensitivity will enable the detection of smaller magnitude CDs; by contrast, a lower sensitivity will limit false positives. This is the same for the CD slope: the higher the slope, the less sensitive the detector must be. For instance, in statistical test methods, the robustness of the algorithms can be tuned by the choice of the hypothesis test. Too low a low threshold would lead to poor detector performance over small CDs. The process is the same with the clustering methods, where the sensitivity is tuned by changing the density threshold or the distance between clusters.

Tuning the sensitivity is related to the management of outliers, which can be a major source of false positives. As previously discussed, if the drifts have high amplitude and slopes, sensitivity will naturally reject outliers. However, other approaches could be used for outlier rejection. When the drift detector is triggered, the case could be added in a buffer. Depending on the amount of successive or similar elements in the buffer, the example could be classified as CD or as an outlier. The decision can be based either on heuristics or statistics [25,26]. This adds another layer of protection against outliers and makes it possible to increase the maximum sensitivity that can be chosen.

The CD detection concerns only visible CDs, which are either upstream CDs or inside CDs. Upstream drifts start to have an impact when the model inputs X goes out of the training set in the extrapolation zone of the estimation model. By contrast, inside drifts will directly lower the model accuracy by changing the function between X and Y. Most of the time, upstream CDs will require sensitivity that is lower than inside CD. This effect will depend on the extent of the training set and on the model's extrapolation capability.

As previously explained, there is no existing rule to choose a suitable algorithm based on the CD degrees of freedom. Currently, the best solution is to choose different algorithms, tune them according to the CD geometry information, and test them on a testing dataset to be able to choose the most suitable one for the application.

3.9. Updating system

When the role of the SDS is to decide when to measure, the role of the US is to define how to update. Once the fresh measurements are acquired, the US can start the model adaptation. By modifying the learned concept, the US minimizes the effect of the CDs on the model's accuracy. As with the SDS, the longevity of the model will depend on which US is chosen and its tuning. Indeed, there are many factors that can influence the updating mechanism, which can even reduce the estimator MAE in the worst case. The selection and tuning choice depend on the CD's characteristic. In this section, the different degrees of freedom defining the US are discussed.

The slope of a CD is the first CD's characteristic which influences the updating strategy. From a qualitative point of view, the slope is defined as sudden or gradual. An important dilemma—one that highlights the difference between sudden and gradual CDs—is called the stability-plasticity dilemma. This dilemma results from the tradeoff between being stable and handling noise and outliers, on the one hand, or being plastic and adapting more quickly to CDs, on the other. In general, a US should invest more in the noise and outlier impact mitigation of the MAE. However, the more sudden the CDs are, the more plastic the US should be. Indeed, the stability becomes a flaw if the adaptation time is slower than the concept evolution. This phenomenon is represented in the most used approach for a US—the moving window. The window size is a parameter that illustrates the tradeoff between stability and plasticity. Small ones are suited for detecting abrupt CDs, whereas large ones are better at detecting gradual CDs [27]. The type of the sliding window is the first characteristic necessary to design a moving window. The simpler form is a fixed-size window—a method where the model is periodically updated using a window containing a fixed number of instances, where each new instance replaces another one in the window. (The strategy of deciding which point to discard relates to forgetting capabilities, and this is discussed later.) The size needs to be designed iteratively, as previously described, and will be adapted to one fixed ranged of CDs [28]. When dealing with different geometries of CDs, the optimal size of the moving window may depend on time. Accordingly, the use of a variable window size, where the size changes depending on the error of estimation [29,30], can be considered. The higher the error, the smaller the window (and vice versa). The proposed framework assumes that the error of estimation is available only on measurements. Thus, it is possible to optimize the window size based on the error from the new measurements before performing the update. Important parameters to take into consideration when tuning the window size are outliers and noise. Indeed, both influence the performance. A small window will be more affected by noise and outliers, and the model accuracy will decrease, while large windows will dump their effects.

The second important CD characteristic for updating approach selection is CD magnitude, which describes the severity of the CD. During updates, the new examples will be added to the previous ones before retraining. However, if the training set is too large, the new information might be drowned. In the literature, this dilemma corresponds to the class-imbalanced problem, which also applies to regression. Therefore, the higher the CD magnitude, the greater the modification of the concept and the higher the number of required examples to restabilize the MAE. Each time the SDS triggers an alarm, the number of measurements performed can be higher than one. This is one important parameter that can mitigate the class-imbalance problem if it is well defined. However, with a moving window, a fresh data added is more of the time another data deleted. Too many measured points can be expensive and counterproductive. A solution to address this issue is called "instance weighting," a method whereby the model is updated by applying weights to each example to give more importance to some of them [27]. Different techniques exist for weighting the instances. Some assume that the most recent data is the most informative and thus give them more weight than the old ones; however, this assumption is not appropriate in every context. For instance, it does not hold in presence of recurrent concepts.

The third important CD characteristic to consider while tuning the US forgetting capability is its type. The moving window comes with a forgetting capability, which is mandatory as industrial implementation mostly comes in a data streams form. Data streams assume an infinite number of iterations; it would not be feasible to remove elements to assure a finite data storage and inference time. The forgetting capability should remove the less informative example. In the case of an inside CD, the relation between X and Y changes, making all the old examples less informative. The forgetting capability should therefore remove the oldest example. In the case of an upstream CD, the relation between X and Y does not change, which does not necessarily make the old examples less informative. Thus, the density can be representative of the example's informativeness. In cases where there is a diverse type of drift, the most conservative approach should be selected, which is the one based on the example's age.

The proposed approach makes it possible to select and tune the adequate algorithms to address the involved CD. Therefore, an ensemble of solutions, composed of an SDS as well as a US, can be extracted from this stage of the methodology. This is a first step in reducing the number of solutions. The next step will evaluate the pre-selected solution to eliminate the ones that do not respect the environmental constraints and to select the best solution that minimizes the costs.

3.10. Optimization and solution selection

The last step aims at choosing the most optimal solution among the pre-selected ones. First, an optimization problem is defined where the objective function and the different constraints are discussed. Then, an evaluation phase, where possible solutions are compared, is performed. Finally, as a result of the optimization problem, the best solution is selected.

3.11. Optimization problem definition

The optimal solution can now be selected because of the cost and constraints related to the number of measurements and the accuracy of the model. The optimization problem can be defined as such:

$$\textit{minimize } C^T X$$

$$\textit{subject to}$$

$$X \leq Constraints = \begin{pmatrix} Accuracy\ constraint \\ Measurement\ constraint \end{pmatrix}$$

$$X = \begin{pmatrix} Accuracy \\ Nbr\ of\ measures \end{pmatrix} \in SolutionsC, = \begin{pmatrix} Accuracy\ cost \\ Measurement\ cost \end{pmatrix}$$

Where,X is the vector of variables defined as the metrics of the tested solutions, which have been pre-selected in the previous stage of the methodology. It is composed of:

- o **Accuracy** of the model when using the algorithms chosen for the solution X.
- o **Number of measures required** by the relative solution X to maintain the model updated with the corresponding accuracy.

C is the cost vector characterized by two metrics:

- o **Measurement cost** is the cost of measuring one unit for the company. This cost gathers resource costs such as the workforce used or the renting cost for the measure machine and potential shortfall. This cost varies from one industry to another and from one process to another.
- o **Accuracy cost** defines the cost of risking a bad estimation. The range of acceptable error over the estimations must be defined. As for the measuring costs, this accuracy cost is dependent on the industry. For instance, in predictive maintenance applications, this cost could be represented by machine compensation cost and induced shortfall.

C^TX is the objective function which must be minimized. It relates to the cost induced by the solution. The purpose is to find a solution that causes a minimal cost. C^TX can be seen as a ratio between acceptable model performance variation and acceptable measurement budget. Acceptable model performance variation is defined by the product/process specification particular to the industry. Most of the time, this is regulated by customer expectations. The acceptable measurement budget is defined by the company, and it is relative to the attributed budget. Indeed, in the simplest case it would be a linear function.

The constraints are induced by the environments. There are two types of constraints: accuracy constraint and measurement constraint. The first type comes from the problem requirement definition. For instance, for VM, one could use the adaptation of the ISO norm to define this tolerance based on the industry [9]. The second one can come from the measurement material or human resource limitations.

Each parameter of the optimization problem must be defined. X is given by the solution pre-selection stage. The cost vector, the objective function, and constraints are given by the problem environment. Once the problem is defined, the evaluation of the pre-selected solution can be performed.

3.12. *Off-line solutions evaluation*

Pre-selected solutions need to be evaluated using the dataset built during the data acquisition step. The data acquisition step is subdivided into three subsets: (a) a training set for training the different algorithms of the solution, (b) a validation set to evaluate the performance of the solutions, and (c) a test set to evaluate the final performance of the selected solution. Therefore, for each pre-selected solution the following process is performed:

- o Solution implementation in the framework.
- o Training of the algorithms with the training dataset.
- o Evaluation of the accuracy of the estimation model on the validation set.
- o Evaluation of the number of measures required to reach such a degree of accuracy on the validation set.
- o Mapping of the solution to be able to visually compare the different solutions. A graphic representation of the solution mapping is given in Fig. 3.

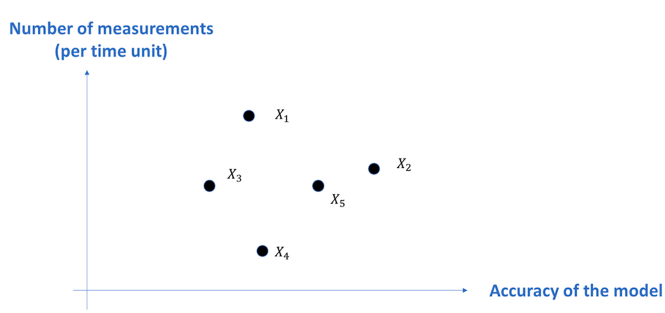

Fig. 3. Representation of the pre-selected solutions mapping. Black dots represent different solutions.

Once all pre-selected solutions are evaluated and mapped in a graph, the problem optimization can be implemented to select the best solution.

3.13. *Solution selection*

To select the most suitable solution, the optimization problem is implemented. By minimizing the optimization function, considering the constraint, the adequate solution is obtained. The following graph is produced where the optimal solution can be graphically identified.

The two optimization constraints—the measurement constraint and the accuracy constraint—set the limits of the valid solution space (the green area on Fig. 4). Solutions outside the valid space do not respect the problem constraints and therefore can be eliminated. The optimization function is represented by a ratio (the red slope on Fig. 4)—in this case linear—between the acceptable model performance variation and acceptable measurement budget, which makes it possible to select the best solution. This ratio corresponds to a pareto efficiency situation where the pareto front is used as an optimization function to define the optimal solution called the pareto efficient. In this example, a linear function is considered for the pareto front, and it can be seen that the solution X_4 is the most suitable one. The selected solution can be tested on the test set to ensure its generalizability.

The proposed methodology makes it possible to select an adequate solution to address the identified CD for a given environment. The methodology is illustrated through a simulation in the next section.

4. Methodology illustration

The following case study represents, on a simulated industrial example, the previously explained methodology.

This example pictures an enterprise wanting to use virtual metrology (VM) on the manufacture of a product. The VM algorithm should estimate the *cutting width* (a_e) realized by a machine tool during the milling of a certain region of a part on a known operation. Its inputs, provided by sensors placed on the machine, are the *cutting power* (P_c) and the *feed speed* (v_f). This simulation mainly aims at illustrating the methodology and not at comparing the CD detection algorithm or virtual metrology algorithm.

The company is interacting with a simulation developed with the following equations.

P_c (kW): Actual Cutting Power	$P_c = \dfrac{a_p \;\; a_e \;\; v_f \;\; k_c}{6010^6 \quad \eta}$
a_p (mm): Depth of Cut	$a_p = 5$
a_e (mm): cutting width	$a_e = \mathcal{N}(2, 0.5)$
v_f (mm/min): Feed speed	$v_f = \mathcal{N}\,(15000, 1000)$
k_c (N/mm2): Specific Cutting Force	$k_c = k_{c1} \;\; h_m^{-m_c}\left(1-\dfrac{\gamma_0}{100}\right)$
k_{c1} (N/mm2): variable	$k_{c1} = 2200$
h_m (mm): chip thickness	$h_m = 1.75$
γ_0 (°): rake angle	$\gamma_0 = 0$
$\eta \in [0,1]$: efficiency	$\eta = 0.8$

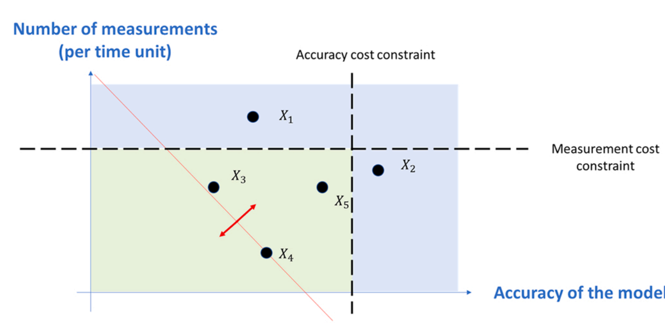

Fig. 4. Optimization graph. Black dash lines represent the problem constraints. The green area is the valid solution space. The blue area is the invalid solution space. The red line represents the optimization function.

Three CDs occur in the simulation. The first CD (d_1) linearly changes the efficiency of the machine due to the wear of the tool, which is reset to zero when maintenance occurs. A second CD (d_2) suddenly changes the value of k_{c1}, due to the evolution of the raw material lot used for manufacturing the part. The third CD (d_3) linearly increases the mean of the Gaussian function that describes the cutting width (a_e), due to the wear of the mechanical stop, which is corrected at each maintenance.

The virtual metrology algorithm chosen for this experiment is a multilayer perceptron with two inputs, P_c and v_f, one output a_e, 2 hidden layers of 7 neurons each, and RELU activation function.

4.1. Concept drift identification

Before beginning the case study, the train set of the VM algorithm was created measuring 4000 consecutively produced parts with their cutting power and feed speed data. The measured values were consecutively sampled to diminish the chance that a CD manifests during the VM training. For the validation set, another dataset, with a lower sampling frequency, has been measured. The methodology to develop a maintenance plan for the VM algorithm has been motivated by the decline in accuracy of the VM model.

4.2. Knowledge-based concept drift identification

Experts identified tool wear as a possible source of gradual CD followed by a sudden CD when the tool is replaced. The tool lifetime is generally 30,000 manufactured parts long. For this reason, the dataset must be taken during at least this number of parts; moreover, it must be taken with a relatively high frequency to be reactive to the sudden CD happening after maintenance is performed and to other possible CDs with high frequency. Therefore, it has been decided to measure 3000 parts, one every 10.

Fig. 5 shows the causal graph of this experiment. The supposed equation has been identified by experts, as recorded in the literature. The arrow represents Eq. 1; in other words, it represents the physical model linking the cutting width, the feed speed, the CD over the motor efficiency, and the cutting power. It is an anti-causal problem (Y is causing X), which means there is no hidden CD; moreover, d_1 is a source of a visible inside CD, which will affect the efficiency.

$$P_c = \frac{a_p \quad a_e \quad v_f \quad k_c}{60 \quad 10^6 \quad \eta} \mathrm{kW} \tag{1}$$

4.3. Data acquisition

After 7000 measurements with different frequency, the dataset is separated into a train and validation set. Fig. 6 presents the plots of the validation set.

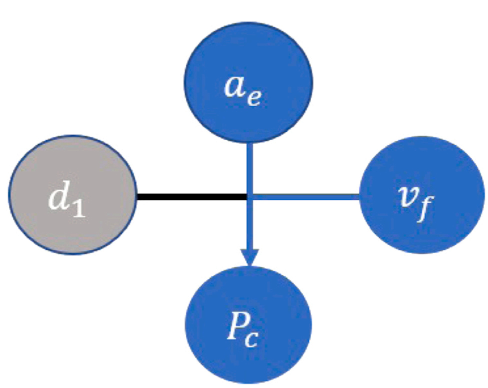

Fig. 5. Preliminary directed graph.

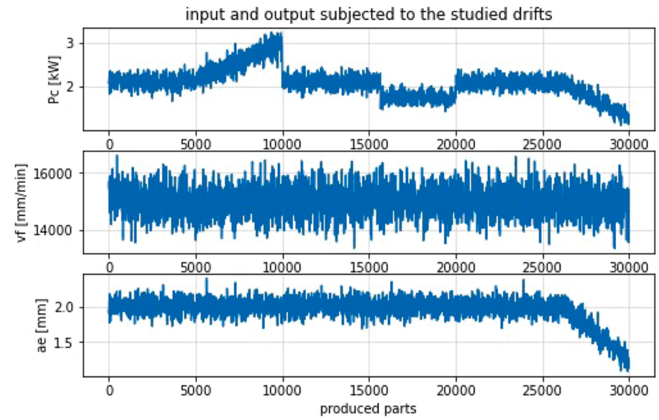

Fig. 6. CD on the validation set.

4.4. Data-based concept drift identification

The data presented in Fig. 6 confirms the impact of the CD due to wearing of the tool (d_1), visible from the 5000th -produced part until the 10000th. Fig. 6 also shows two unpredicted CDs: one, (d_2), that suddenly changes the value of the power (at points 16,000 and 20,000), and the second, (d_3), that is linear and affects simultaneously the cutting power and the cutting width (starting around the 27,000th manufactured part). This last one seems to be an upstream CD. The signal is somewhat noisy, which highlights the potential presence of outliers.

The different CDs of the evaluation set are presented in Fig. 7 on a X-Y plot. First, d_3(in green) follows the same shape as without CD (in blue), but over an unexplored region. For this reason, it can be validated as an upstream CD. Both d_1 and d_2 CDs modify the concept between the cutting power and the cutting width. They are only identifiable on the cutting power signal. They are, therefore, classified as visible inside CDs.

4.5. Knowledge-based concept drift identification (second iteration)

After investigation, it has been noticed that the d_2 corresponds to one specific lot of raw parts. It corresponds to a sudden CD that is certainly related to a defective lot of raw material with a different inner constraint. The material properties affect the power consumption of the machine tool during the manufacturing of these parts. It was determined that it comes from the wearing of the mechanical stop that enables a precise loading of the raw material during the milling process. It has also been noticed that no CD occurs over v_f. It is still measured to enhance the accuracy of the VM model, as the sensor investment was already done. Therefore, one upstream CD and two visible inside CDs are identified, as

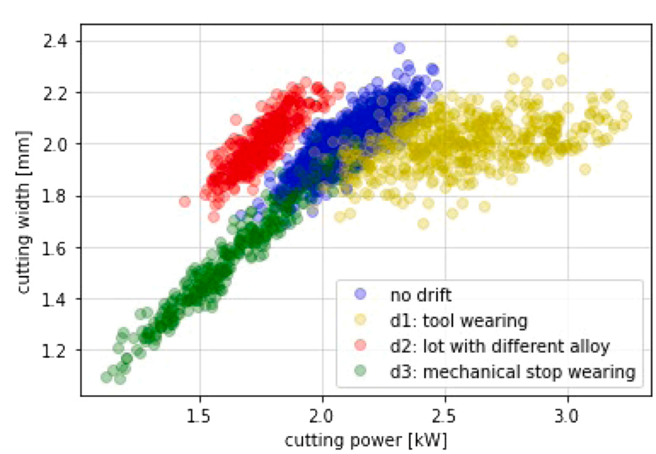

Fig. 7. Cutting width in function of the cutting power for the different CD.

shown in Fig. 8.

4.6. Solution pre-selection

From the previous stage, the following information concerning the involved CD was identified:

- No hidden CD.
- Presence of an incremental upstream CD (d_3).
- Presence of two visible inside CDs, one gradual (d_1) and one sudden (d_2).

4.7. Sampling decision system

Since no hidden CD was detected, it has been decided to use an active strategy that includes a safeguard with a low passive sampling frequency, having as a period half of the acquired signal length (15000 parts), which is activated only when no CDs are detected by the active module over that period. Concerning active CD detection, it has been decided to compare two algorithms: a statistical method (CUSUM) [31] and a clustering method (OLINDDA) [25]. The clustering method is robust to outliers, which gives it a greater range for tuning its sensitivity. The presence of noise and outliers added to the quiet high amplitude of the different CDs motivate the use of a relatively low sensitivity for CD detection.

4.8. Updating system

For the updating of the multi-layer perceptron, a moving window is used. First, the slope of the CDs is relatively important, making them easier to detect but requiring higher flexibility and thus a smaller window size. Second, as the magnitude of the CDs can be important, the number of new points to insert in the window at each update should be larger than one. By contrast, as the window is relatively short, the number of new points cannot be too high. Lastly, as there is an inside CD, the older point is removed at each update. This entire process makes it possible to define a short range of interest for the moving window size and number of points needed for updates, all of which will be tested.

4.9. Solution selection and optimization

4.9.1. Optimization problem definition

The metrics defined by the enterprise depends on the cost of each

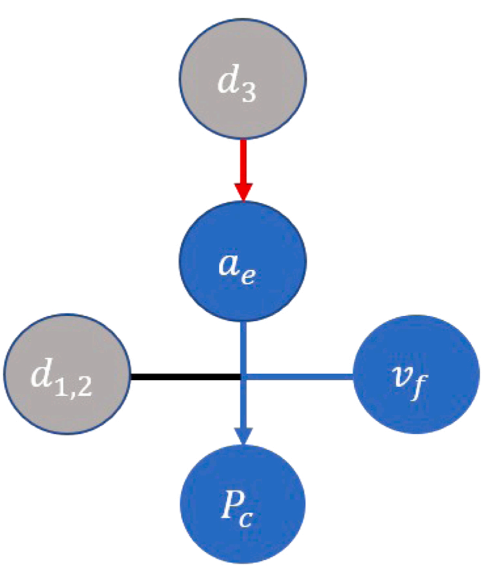

Fig. 8. Updated directed graph.

measure (c_m) and the cost of a poor estimation (c_{MAE}). Each measure costs 31 USD, and each millimeter of MAE costs 6 USD. The optimization problem aims to minimize the sum of the two costs $(C^T X)$; the variables of the optimization are the number of measures (n_m) and the mean absolute error of the produced part of a lot (MAE). These values are taken in the batch, including all the evaluated solutions. The enterprise cannot measure more than 1000 parts over a lot of 30,000 because of the measuring machine availability constraint. The optimization problem is therefore defined as follows:

$$\min_X (C^T X), \text{with } C = \begin{bmatrix} \frac{c_m}{n_p} \\ c_{MAE} \end{bmatrix}, X = \begin{bmatrix} n_m \\ MAE \end{bmatrix}, \text{ under } n_m < 1000$$

$\mathbf{c_m}$ **[USD/measure]:** cost per measure	$\mathbf{c_m = 31}$
$\mathbf{c_{mae}}$ **[USD/mm/part]:** cost of the bad estimation	$\mathbf{c_{mae} = 6}$
$\mathbf{n_p}$ **[parts]:** number of produced parts in the validation set	$\mathbf{n_p = 30000}$
$\mathbf{n_m}$ **[measures]:** number of measures	value to optimize
MAE [mm]: mean square error of a lot of $\mathbf{n_p}$ parts	value to optimize

4.9.2. Off-line solutions evaluation

On the tuning phase, for each algorithm it was necessary to evaluate different sets of hyperparameters, including the parameters for updating the VM algorithm, which are the moving windows size and number of new points added to the moving window. More than 300 combinations have been tested offline for both drift detection algorithms. Fig. 9 shows the MAE for each set of hyperparameters, over 30,000 produced parts in function of the number of measures based on the dataset acquired previously. In this case, it can be observed that extended CUSUM reaches better performances compared to OLINDA.

4.9.3. Solution selection

As presented in Fig. 10, the best approach for the enterprise, according to the cost and constraints previously defined, is extended CUSUM with an MAE of 0.03 mm and a number of measures equal to 114 over the test period. The total cost due to the poor estimation and the number of measures per produced part is equal to $C^T X$, which in this case is 0.30 USD/part. The size of the moving window used to update the estimator is 10 and the number of new points measured before each update is 3.

In order to test the performance of the selected algorithm, a test set of 1000 measures, one every 10 produced parts, is collected. Due to its small size, this test set only contains d_2 and d_3.

Fig. 11 shows the absolute error of the VM algorithm estimation at each manufactured part for the validation and test set. The blue and orange lines of the upper graphs represent, respectively, the absolute error of the VM algorithm estimation with and without an update. The dotted line indicates the instant where a CD is detected, a measurement is triggered, and an update of the estimator is made.

On the test phase, 45 parts are measured, and the resulting MAE is 0.0455 mm, for a cost of 0.41 USD/part. The solution seems to generalize correctly and is validated.

5. Discussion

The current section is devoted to presenting some key discussion points regarding the industrial implementation of such a methodology. The limits and potential adaptation of the methodology are also discussed.

The phenomenon of CD has its origin in the dynamic nature of manufacturing systems and in general any system that changes over time. More specifically, in manufacturing systems, CDs are generally occurring because of the deterioration and wear of components, tools, and material, but also from the quality of the input material, human

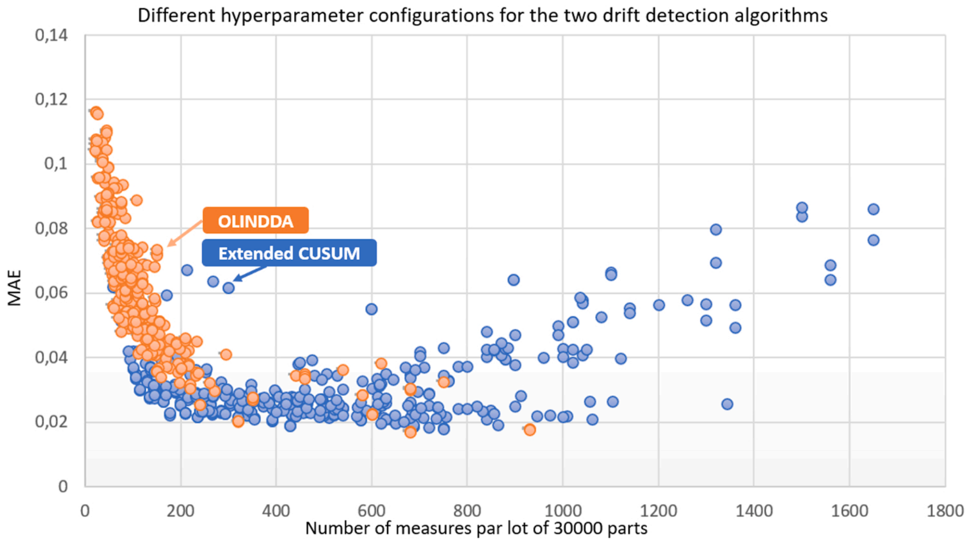

Fig. 9. Offline solutions evaluation graph.

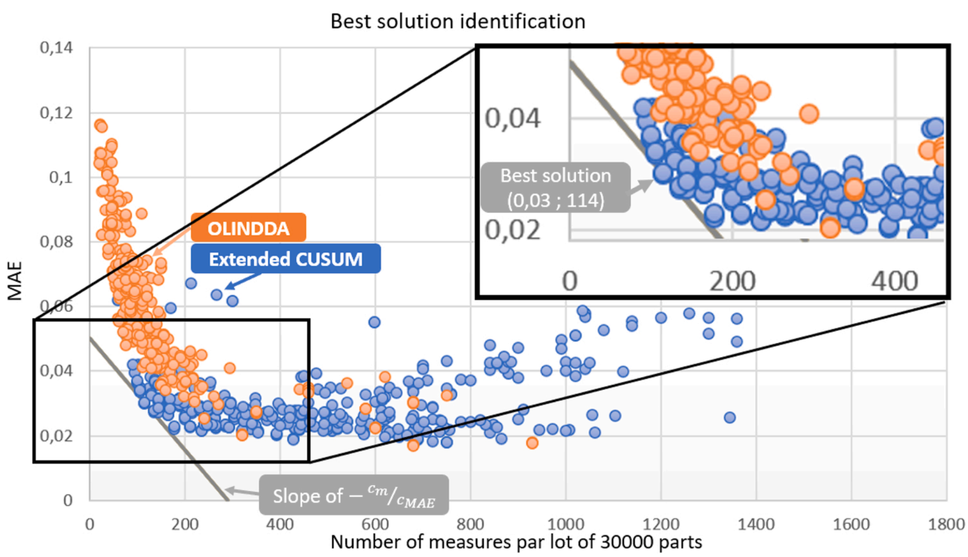

Fig. 10. Identification of the optimal solutions.

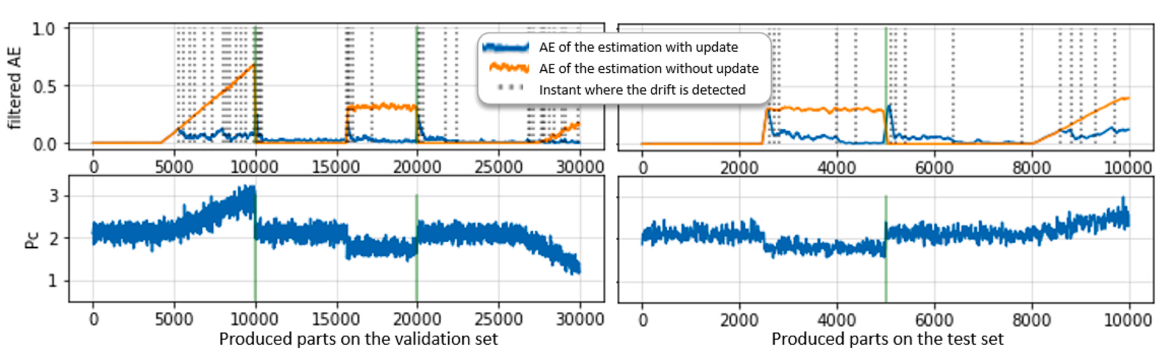

Fig. 11. Validation and test set of the selected solution.

error, and environmental changes such as temperature or humidity in the shop floor, and so on. Additionally, in the same spirit, feedback loops such as sensor data might change slightly over time because of the deterioration of the sensor itself, sending varying information. In an industrial environment, CDs occur systematically, which makes their consideration fundamental to ensure the maintenance of industrial data-based models.

Most of the Industry 4.0 technologies such as machine learning, artificial intelligence, digital twins, virtual metrology, predictive maintenance, and zero-defect manufacturing are data-driven technologies

[32]. Currently, most of the data-driven solutions are designed and developed to be static, which means that it is ready for deployment after the training of the corresponding model, without considering the CDs that will occur as time passes. The proposed methodology enables the adaptation of existing static models to dynamic ones. Taking the CDs into consideration, the flexibility and adaptability of the models is increased significantly, which consequently increases long-term accuracy. In other words, the proposed methodology gives a general approach to ensuring the maintenance of data-based models for the industry.

The presented methodology proposes to take into consideration CD characteristics to design solutions that consider the practical issues of solution implementation. For the first stage of the methodology, CD identification, the participation of application experts appears to be crucial. The better the CD identification, the better the model maintenance and, thus, the smaller the measuring and accuracy costs. However, in some cases the CD identification can be difficult to perform, such as when several CDs impact the system simultaneously. In those cases, identification tools and methods could reach their limits. Therefore, further research could be done to develop new methods to enhance data-based CD identification capability. In terms of CD characterization, the literature is filled with diverse terminology that does not serve the interests of a uniform methodology. The new proposed CD types aim to facilitate the CD identification. Further research still needs to be undertaken on the normalization of CD geometric characteristics.

The second stage of the methodology, pre-selection solution, makes it possible to pre-select various solutions by choosing a suitable SDS and US to deal with the involved CD. Currently, the procedure to build solutions is to study the literature to find other works dealing with similar case studies. However, two similar applications can involve different CDs. There are not two identical industrial environments for the same process, so the CDs are obviously different. This reveals the limit of the previous methodology and the need to analyze the CD characteristics when selecting a solution. Indeed, if the algorithms are selected or defined by the involved CD, then they would not be application-dependent anymore. Therefore, one will be able to pre-select solutions of interest based only on the CD that must be addressed. SDS is the key component for handling potential CDs. The passive strategy is the simplest approach to implement. Most of the implementations include a single timer; however, having one timer per CD could be interesting to develop. In a general way, further research on developing techniques for choosing the sampling frequency could benefit not only the CD handling field, but also the industrial quality control field with the optimization of the batch measurement frequency. For active strategies, the implementation must integrate a timer safeguard in industrial environments. Indeed, CDs can be missed when doing the CD identification, or new CDs can appear over time. There are few methods to optimize the safeguard sampling frequency, so there is still space for improvement. Finally, the hybrid strategy appears to be the best choice, in general, for industry environments due to the high number of CDs.

The developed framework considers online updates; the update is made immediately once an example is available. However, in the literature incremental updates are described in which the model is updated once the entire window of new data instances is sampled [33–35]. This raises the question of when to update, as determined by the size of that window, which makes it possible to fully dissociate the SDS from the US. The main advantage of this technique is to minimize the computational cost at the price of a slower reaction to CDs. The computational cost could also be minimized by updating the model in cascade. Incremental updates could become interesting when training a specialized estimator as it is done in ensemble learning. Ensemble learning has not been discussed in this paper as it is a special case. Ensemble learning has the unique feature of being able to update by removing the specialized estimator and adding new estimators specialized with a newer batch of examples [36]. It is also possible to store old models in “sleeping mode” in case the concept comes back [37]. Such methods require memory to be able to stock the different concepts; the updating is then more about model management than real updating. The most efficient models are activated while inefficient ones are deactivated. This seems promising as it can deal with recurrence. However, such a method requires identifying concept signatures to be able to recognize when a sleeping model will be useful. Thus, this approach seems somewhat futuristic, and it is not a priority for future research. In a general way, this methodology has shown that updating mechanisms depend on different CDs’ characteristics, which further reinforces the need to know the type and characteristics of the involved CDs in a problem.

The solutions developed with this methodology are suitable in most cases. However, when working in extreme constraints environments, this methodology might not be adequate as it is. Four different cases representing different scenarios of the optimization problem are presented in Fig. 12 as a way to discuss alternative solutions. The two questions at the basis of the scenarios are as follows:

- Are all samples measurable?
- How much does it cost to measure a sample?

If the samples to measure are unlimited and their measurements are costless, then there is no need for a model. Indeed, the measurements can be performed whenever they are required without caring about the relative cost. Hence, having an estimation model has no advantage over the physical measurement, and it does not make any sense to have one.

If the measurement frequency is limited but the measurement is free, the model enables measurement continuity while the real values are not available. As the measurement is free and more accurate than the model, measurements will be done every time possible. However, continuously updating the model with new points can be detrimental as the updating approaches; it is not possible to merely add the new data in the training set, but rather to exchange them. It can happen that the new examples hold less original information than the replaced ones. In this architecture, the SDS does not define when to measure but rather when to update. In this vein, active learning algorithms are suitable as they give as much relevant information as possible to the model. The US is thus connected in the SDS, which make sense in this case. Therefore, optimizing the forgetting mechanism of the US is fundamental in this case.

If the measurements are costly, the budget becomes a constraint and a ratio between acceptable model performance variation and acceptable measurement budget appears to be a good indicator for sampling decision strategy optimization. Most of the time, the first one is often hidden in the CD detector’s hyperparameters for active strategies, whereas the second one is used to design the time-based measurement frequency of passive strategies. This ratio is similar to the one used by manufacturers for classical product quality control, with the key difference that it is not product quality that is out of tolerance but estimation accuracy. The measurement budget is, in general, an incentive, not a constraint [38]. If it is a constraint, the problem enters the class of costly and limited measurement problems [13]. The methodology proposes in this research is made to deal with this type of scenario.

When the measurement is costly and only partially available, a queue

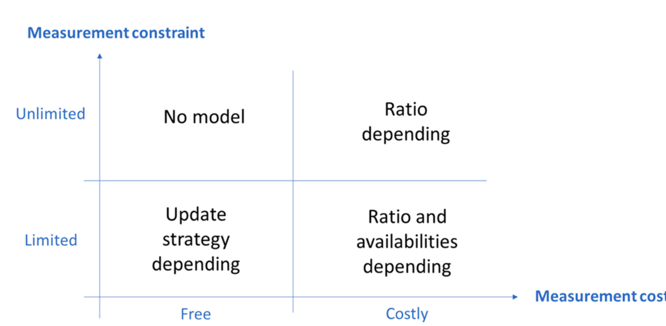

Fig. 12. Measurement constraint graph.

layer must be added to the strategy. Indeed, the time between the measurement order of the SDS and the actual measurement is unknown in this scenario. Next, the desired measurement must be queued. The queuing importance may depend on the measurement priority. Moreover, the ratio between the measurement frequency and the measurement availabilities can help greatly in defining the measurement strategy. If the measurement availability is low, it would be detrimental to miss the opportunity of measuring. To deal with this, one would want to complement a classical CD detection algorithm that does estimation with a forecasting CD prediction that is able to optimize the sampling strategy accounting for the measurement limitation. Further research on sampling strategies with limited and costly measurements would enhance the literature and be helpful for many applications.

With the last stage of the methodology, optimization, and solution selection, a clear vision and a good understanding on the problem are required. Indeed, it is mandatory to be able to define the right constraints and optimization function. Moreover, update process depends on the used estimation model. Therefore, it is recommended to consider the full operational framework when choosing a solution. This means considering the choice of the model with the SDS and US choices.

Finally, the proposed methodology enables the selection of the optimal solution based on CD characteristics, which is a novelty in the field. The methodology could be enhanced in the future if the link between CD characteristics and solutions were better developed in the literature.

6. Conclusion

This paper presents a new methodology to ensure the maintenance of industrial data-based models. The solution is based on the identification of CD characteristics and the development of an optimal solution to achieve the main objective, thus preserving the accuracy of the industrial data-based model. This new vision enables the definition of new types of CDs with a new approach to detect them. In light of this methodology, the optimal approach to ensure the data-based model accuracy, based on the measurement cost and the cost of bad estimation, can be selected. In the first stage, CD characteristics are identified. In the second stage, the methodology facilitates the preselection of some solutions of interest based on the CD characteristics. The solutions are defined by an SDS that decides when to measure new examples, and a US that describes how to update the data-based model. In the last stage, an optimization procedure makes it possible to select the most suitable elements leading to the optimal solution. The methodology is illustrated through a simulation.

In terms of further research opportunities, the links between solution elements selection and CD characteristics needs to be developed more. Additionally, more advanced methods to identify CD characteristics should be explored. Finally, the methodology should be adapted for extreme constraints such as limited measurement possibility.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.jmsy.2022.03.015.

References

[1] Psarommatis F, Sousa J, Mendonça JP, Kiritsis D. Zero-defect manufacturing the approach for higher manufacturing sustainability in the era of industry 4.0: a position paper. Int J Prod Res 2021;60:73–91. https://doi.org/10.1080/00207543.2021.1987551.

[2] Grevenitis K, Psarommatis F, Reina A, Xu W, Tourkogiorgis I, Milenkovic J, et al. A hybrid framework for industrial data storage and exploitation. Procedia CIRP 2019;81:892–7. https://doi.org/10.1016/j.procir.2019.03.221.

[3] Psarommatis F, May G, Dreyfus PA, Kiritsis D. Zero defect manufacturing: state-of-the-art review, shortcomings and future directions in research. Int J Prod Res 2020;58:1–17. https://doi.org/10.1080/00207543.2019.1605228.

[4] Chen G, Wang P, Feng B, Li Y, Liu D. The framework design of smart factory in discrete manufacturing industry based on cyber-physical system. Int J Comput Integr Manuf 2020;33(1):79–101. https://doi.org/10.1080/0951192X.2019.1699254.

[5] Kamble SS, Gunasekaran A, Parekh H, Mani V, Belhadi A, Sharma R. Digital twin for sustainable manufacturing supply chains: current trends, future perspectives, and an implementation framework. Technol Forecast Soc Change 2022;176:121448. https://doi.org/10.1016/j.techfore.2021.121448.

[6] Tao F, Qi Q, Liu A, Kusiak A. Data-driven smart manufacturing. J Manuf Syst 2018;48:157–69. https://doi.org/10.1016/j.jmsy.2018.01.006.

[7] Zhang C, Chen D, Tao F, Liu A. Data driven smart customization. Procedia CIRP 2019;81:564–9. https://doi.org/10.1016/j.procir.2019.03.156.

[8] F. Psarommatis, A dynamic scheduling tool and a methodology for creating digital twin of manufacturing systems for achieving Zero Defect Manufacturing 2021.

[9] Krawczyk B, Minku LL, Gama J, Stefanowski J, Woźniak M. Ensemble learning for data stream analysis: a survey. Inf Fusion 2017;37:132–56. https://doi.org/10.1016/j.inffus.2017.02.004.

[10] Gama J, Zliobaite I, Bifet A, Pechenizkiy M, Bouchachia A. A survey on concept drift adaptation. ACM Comput Surv 2014;46:1–37. https://doi.org/10.1145/2523813.

[11] Faria ER, Gonçalves IJCR, de Carvalho ACPLF, Gama J. Novelty detection in data streams. Artif Intell Rev 2016;45(2):235–69. https://doi.org/10.1007/s10462-015-9444-8.

[12] Dreyfus PA, Psarommatis F, May G, Kiritsis D. Virtual metrology as an approach for product quality estimation in Industry 4.0: a systematic review and integrative conceptual framework. Int J Prod Res 2021;60:742–65. https://doi.org/10.1080/00207543.2021.1976433.

[13] Zliobaite I, Bifet A, Pfahringer B, Holmes G. Active learning with drifting streaming data. IEEE Trans Neural Netw Learn Syst 2014;25(1):27–39. https://doi.org/10.1109/TNNLS.2012.2236570.

[14] A. Haque, L. Khan, and M. Baron, "SAND: Semi-supervised adaptive novel class detection and classification over data stream," 30th AAAI Conf. Artif. Intell. AAAI 2016, no. May 2015, pp. 1652–1658, 2016, doi: ⟨10.1007/978–3-319–18032-8⟩.

[15] Sun L, Ji Y, Zhu M, Gu F, Dai F, Li K. A new predictive method supporting streaming data with hybrid recurring concept drifts in process industry. Comput Ind Eng 2021;161:107625. https://doi.org/10.1016/j.cie.2021.107625.

[16] Guo H, Zhang S, Wang W. Selective ensemble-based online adaptive deep neural networks for streaming data with concept drift. Neural Netw 2021;142:437–56. https://doi.org/10.1016/j.neunet.2021.06.027.

[17] Liu Z, Loo CK, Pasupa K, Seera M. Meta-cognitive recurrent kernel online sequential extreme learning machine with kernel adaptive filter for concept drift handling. Eng Appl Artif Intell 2020;88:103327. https://doi.org/10.1016/j.engappai.2019.103327.

[18] Gonçalves PM, De Carvalho Santos SGT, Barros RSM, Vieira DCL. A comparative study on concept drift detectors. Expert Syst Appl 2014;41(18):8144–56. https://doi.org/10.1016/j.eswa.2014.07.019.

[19] Santos SGTC, Barros RSM, Gonçalves PM. A differential evolution based method for tuning concept drift detectors in data streams. Inf Sci 2019;485:376–93. https://doi.org/10.1016/j.ins.2019.02.031.

[20] Liu W, Zhang H, Ding Z, Liu Q, Zhu C. A comprehensive active learning method for multiclass imbalanced data streams with concept drift. Knowl-Based Syst 2021;215:106778. https://doi.org/10.1016/j.knosys.2021.106778.

[21] Webb GI, Lee LK, Goethals B, Petitjean F. Analyzing concept drift and shift from sample data. Data Min Knowl Discov 2018;32(5):1179–99. https://doi.org/10.1007/s10618-018-0554-1.

[22] Wassermann S, Cuvelier T, Mulinka P, Casas P. Adaptive and reinforcement learning approaches for online network monitoring and analysis. IEEE Trans Netw Serv Manag 2021;18(2):1832–49. https://doi.org/10.1109/TNSM.2020.3037486.

[23] Cheng FT, Hsieh YS, Chen CF, Lyu JR. Automated sampling decision scheme for the AVM system. Int J Prod Res 2016;54(21):6351–66. https://doi.org/10.1080/00207543.2015.1072649.

[24] Lu J, Liu A, Dong F, Gu F, Gama J, Zhang G. Learning under concept drift: a review. IEEE Trans Knowl Data Eng 2019:2346–63. https://doi.org/10.1109/TKDE.2018.2876857.

[25] Spinosaa EJ, de Carvalhoa AP de LF, Gamab J. Novelty detection with application to data streams. Intell Data Anal 2009;13(3):405–22. https://doi.org/10.3233/IDA-2009-0373.

[26] Masud M, Gao J, Khan L, Han J, Thuraisingham BM. Classification and novel class detection in concept-drifting data streams under time constraints. IEEE Trans Knowl Data Eng 2011;23(6):859–74. https://doi.org/10.1109/TKDE.2010.61.

[27] Khamassi I, Sayed-Mouchaweh M, Hammami M, Ghédira K. Discussion and review on evolving data streams and concept drift adapting. Evol Syst 2018;9(1):1–23. https://doi.org/10.1007/s12530-016-9168-2.

[28] G. Hulten, L. Spencer, and P. Domingos, 2001. "Mining time-changing data streams," *Proc. Seventh ACM SIGKDD Int. Conf. Know* l. D*iscov.* Data Min., vol. 18, pp. 97–106, 2001, doi: 10.1145/502512.502529.

[29] Kuncheva LI, Žliobait I. On the window size for classification in changing environments. Intell Data Anal 2009;13(6):861–72. https://doi.org/10.3233/IDA-2009-0397.

[30] Akbar A, Khan A, Carrez F, Moessner K. Predictive analytics for complex IoT data streams. IEEE Internet Things J 2017;4(5):1571–82. https://doi.org/10.1109/JIOT.2017.2712672.

[31] Alippi C, Roveri M. Just-in-time adaptive classifiers - part I: detecting nonstationary changes. IEEE Trans Neural Netw 2008;19(7):1145–53. https://doi.org/10.1109/TNN.2008.2000082.

[32] Kumar L, Ibne Hossain NU, Fazio SA, Awasthi A, Jaradat R, Babski-Reeves K. A data driven decision model for assessing the enablers of quality dimensions: context of industry 4.0. CIRP J Manuf Sci Technol 2021;35:896–910. https://doi.org/10.1016/j.cirpj.2021.10.003.

[33] Gama J, Zliobaite I, Bifet A, Pechenizkiy M, Bouchachia A. A survey on concept drift adaptation. ACM Comput Surv 2014. https://doi.org/10.1145/2523813.

[34] Song Y, Zhang G, Lu H, Lu J. A noise-tolerant fuzzy c-means based drift adaptation method for data stream regression. IEEE/ACM Trans Comput Biol Bioinform 2016;13:549–56. https://doi.org/10.1109/FUZZ-IEEE.2019.8859005.

[35] Song Y, Lu J, Lu H, Zhang G. Fuzzy clustering-based adaptive regression for drifting data streams. IEEE Trans Fuzzy Syst 2020;28(3):544–57. https://doi.org/10.1109/TFUZZ.2019.2910714 (Mar.).

[36] Bouchachia A. Incremental learning with multi-level adaptation. Neurocomputing 2011;74(11):1785–99. https://doi.org/10.1016/j.neucom.2010.12.017.

[37] Katakis I, Tsoumakas G, Vlahavas I. Tracking recurring contexts using ensemble classifiers: an application to email filtering. Knowl Inf Syst 2010;22(3):371–91. https://doi.org/10.1007/s10115-009-0206-2.

[38] Shan J, Zhang H, Liu W, Liu Q. Online active learning ensemble framework for drifted data streams. IEEE Trans Neural Netw Learn Syst 2019;30(2):486–98. https://doi.org/10.1109/TNNLS.2018.2844332.