Indexation of Bench Test and Flight Data

Cynthia Faure[1], Jérôme Lacaille[2], Jean-Marc Bardet[3] and Madalina Olteanu[4]

[1,3,4] *SAMM (EA 4543), Paris 1 Panthéon Sorbonne University, Paris, 75634, France*
cynthia.faure@malix.univ-paris1.fr
jean-marc.bardet@univ-paris1.fr
madalina.olteanu@univ-paris1.fr

[2] *Safran Group, Moissy-Cramayel, 77550, France*
jerome.lacaille@safrangroup.com

ABSTRACT

An important amount of data is provided every day in Safran Aircraft Engines' test benches. Produced by thousands of sensors, test and flight data represent a big interest for engineers but manual analysis of the information is too complex, if not impossible. As specific data are extracted from the database, it is not unusual to miss interesting information when focusing on a given problem. Defining the data as a succession of labels where each label appears as transient or stabilized phases would be one way to solve the problem. The start and stop points of the different phases will be computed by an offline change-point detection algorithm. In order to detect potential crucial changes of characteristic variables, it is relevant to develop powerful algorithms. The Pruned Exact Linear Time (PELT) method (Killick R. & Eckley, 2012) is a parametric change-point detection which searches the optimal partition of a monovariate signal (temporal series of one variable) in our case (but can also be applied to a multivariate signal) with a linear complexity. This algorithm meets our expectations in many ways: robustness, fast computing and accurate results. Then on a multivariate aspect, patterns are built with parameters and initial conditions and, classified in a specific category with a map/reduce scheme. This clustering will allow different analysis: the comparison of different patterns with the definition of a distance and the research of a specific pattern in a large database. For example if an engine shows a specific engine temperature pattern after the test pilot changes the shaft rotation speed from one level to another, engineers may ask if this behavior is usual. If not, it should be very interesting to see if such pattern happens in the past on other engines or other tests and dig from the database the old documents related to those rare events and eventually the people concerned. The objective of this project is to progressively score and classify different patterns in an increasing database of labels. The first step was to implement the PELT algorithm. Then it is possible to identify the different transient phases extracted from small subsets of temporal measurements and compute models for each patterns. These codification of transient phase will lead to a classification into labels or topics. After defining enough patterns, each new record of measurements will be automatically classified.

1. INTRODUCTION

Safran Aircraft Engines' Datalab is an experimental team focused on evaluating new data technologies. The different solutions are challenged by specific studies producing analytic reports or effective prototypes. (Lacaille J., 2016). Three domains are adressed by the Datalab beyond prognostic and health monitoring (PHM) which are:

- Development: better understand the development process and optimize our knowledge base.
- Industry: optimize the design of the engine and the fabrication process.
- Operationnal flights: identify the usage of the engine during flights, link to wear, then maintenance and finally the possession cost.

The present paper describes our road map to build a methodology and tools to help interpreting data generated by development tests. Today's ground development tests are specified by design offices according to each specific needs: dynamic behavior, performance, acoustic, aerodynamic, etc. The bench test is designed by each party to identify specific behaviors but the number of sensors is so big that only part of the records are analysed and some patterns that may not have been in the present interest of engineers would not be observed. Specific data patterns are analyzed manually by experts of each field. But this analyze could be improved by an automatic

computation which would merge data of the different fields. A challenge is to automatically identify known patterns and detect unusual behaviors. Even if this goal may be reached, another problem is to use efficiently the results. For this purpose experts have asked to find a way to automatically send messages to concerned people (see Figure 1).
The next section (2. Methodology) describes the methodology and the tools we developed to solve this challenge in the long term. Section 3. Current work and Research summarizes our present work and results.
Safran Aircraft Engines is an engine manufacturer and definitely not a software editor and our power in data analysis stays limited. However, we have just build new entities in the Safran Group to address this specific domain of data analysis. They are the research center Safran Tech [1] and a new company named Safran Analytics. Safran Tech helps supervising PhDs and interactions with academic laboratories while Safran Analytics focuses more on business solutions. We clearly seek for all good ideas in this new domain and build a business environment able to mature them.

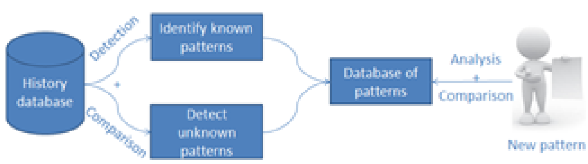

Figure 1. Our challenge: find a way to systematically scan data, build statistics such that experts may automatically be informed about patterns of their concern or about unusual behaviors.

2. METHODOLOGY

Problem summary
Development bench tests are specified in a document giving for each design office the goal of the test, the execution pattern, the awaited results and the demanding engineers. At the end of the test, each numeric record corresponds to a specified demand and if a visual expertise concludes to a specific behavior, the results are stored in logbook documents written by experts and linked to the numeric records. However, as previously described experts only seek for awaited patterns. Most unusual behaviors may not be observed manually as nobody is searching for them.
In summary, we need statistics on temporal patterns in a very huge database of engine tests and flight data where each test produces numerous high frequency records (from 1 Hz to 100 Hz to begin) of hundreds to thousands of measurements. These measurements come from different sets of sensors that differ from one test to another, from one engine to another; they do not even share the same name, the same position or the same unit. This may be seen as a problem similar to that of searching images or documents after a general indexation of patterns. The difference here resides in the diversity of dimensions and corresponding units per record but also in the physical relations between those dimensions, which are specific to the aerodynamic process.

Coding transient phases
A lot of studies have been already done on stabilized phases, only few of them are based on transient phases (Lacaille & Gerez, 2011) and (Lacaille, 2014). The lack of information about transient phases would be compensated with the analysis of all measurements, the identification and the labellization of the transient patterns. This leads to replace the numeric records by sequences of parameterized labels. Transforming multivariate numeric temporal records into sequences of labels will clearly help searching for apparition of a given label in a record that belongs to a given test of a given engine... The indexation of these records will be done by an offline change-point algorithm.
We replace the task of coding systematically all unknown patterns by an incremental solution. The expert defines a specific pattern; inserts the pattern in an analytic tool that defines a mathematical model whose parameters code the label. Then he executes a corresponding matching algorithm and adds results in a new label base. This iterative mode builds sequences of labels for each test record (Figure 2). This approach is not exhaustive but it let us focus beforehand on physically interesting patterns. Temporal unidentified behaviors may also be detected as unknown and draw our attention in a general way if recurrent. Moreover, if an expert creates a label it is easy to register his name as a person of interest when the matching algorithm detects the corresponding pattern.

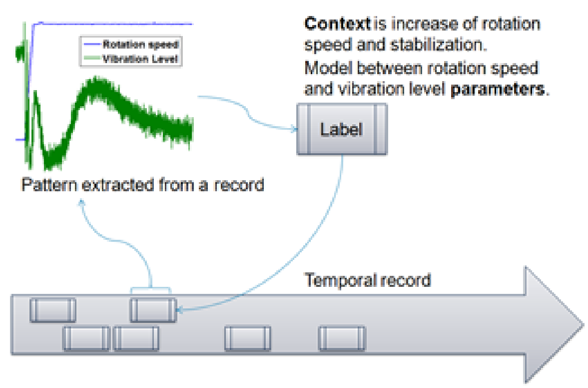

Figure 2. Incrementation of the code of the record sequence by adding new labels: Labels of temporal patterns may overlap, for example when looking at different sets of measurements.

Storage of all patterns
The bench test history is very huge: many tests are executed per day, each one corresponding to hundreds of records during more than one hour. This history database is a hardware and software solution adapted to high velocity storage and

it is clearly not adapted to data analysis (except on the fly computation during or just after storage). Using a distributed hardware, the solution of the indexation process (building the label sequences) and the search and statistics on the labels can be implemented as described below (Figure 3):

- Historic data will be accessed periodically (at leisure time) and temporarily transferred on a cluster.
- Matching algorithms automatically working in parallel on stored records will detect registered patterns and compute their characteristics. Detection results (labels) will update the sequences corresponding to each record improving the indexation of our history database.
- The last step is to build statistics on those sequences, hence to be able to search for apparition of labels or successive labels. The label sequences are also stored in a distributed environment that allows parallel execution of the search algorithms.

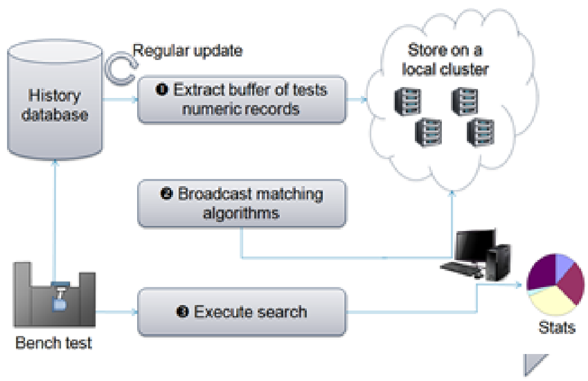

Figure 3. Illustration of the distributed process of indexation and query.

Analysis of a new test
During a new bench test (Figure 4), the numeric records will automatically passed as inputs to an algorithm searching for transient patterns with parameters, then go through the matching filters and lastly will produce a label sequence with labels detected in agreement with a likelihood threshold chosen by the experts. The search algorithm will check parts of this sequence including the label singletons themselves, hence producing a rank output among our data history. Thus, new records will generate a set of reports associated to each detected label. An alert is generated when the observation of the label behaves like an outlier; otherwise, a statistic presenting the rank position as a quartile (or p-value) will be given for information.
When the label base improves, it will be also possible to check for successive patterns acting like a sequence of events instead of just one pattern.

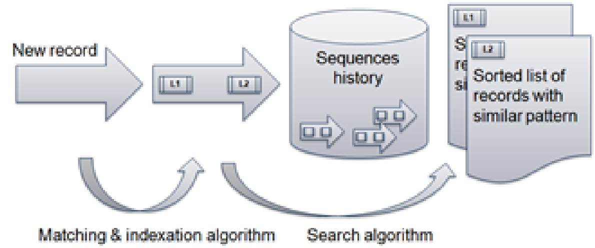

Figure 4. Matching algorithms indexes the database of records. Search algorithm finds similar patterns in the indexed database.

Identification of the experts
The ranking of a pattern identifies a list of similar observations that appear on past test records. Those records are linked to specification documents and some patterns may be linked to visual analysis and stored in the knowledge base (Figure 5). The frequently referenced authors and related services at the origin of those documents are automatically informed by the system. Topic classification of documents is also used to generate the report.

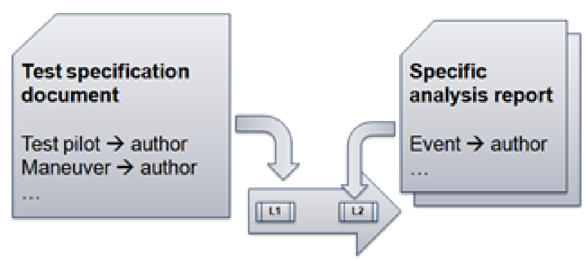

Figure 5. Meta data such as names and services at the origin of the test as well as specific analysis reports written by experts are linked to each test and each pattern.

3. CURRENT WORK AND RESEARCH

The main goals are the indexation of the temporal series satisfying the expert and the definition of a transient phase.

Indexation of temporal series

Detecting change-points in flight data may be seen as a first step in the identification of transient phases. Hence, the aim is to search for an efficient change-point detection method, with low computational cost and high performance. The chosen work is the off-line parametric change-point detection, with an unknown number of change-points. Training a parametric model that fits all available recorded features is not an easy task, mainly because of the heterogeneity of the data. To begin, we focused on data having a piecewise linear behavior. Hence, the algorithm to be further introduced will search for change-points in the slope.

Detecting change-points in the slope of a time-series can be achieved, for instance, by minimizing the least-squared residuals contrast, as described in the seminal paper (Bai & Perron, 1998). Since the number of change-points is unknown, a penalty term is usually added to the contrast function, as proposed in (Lavielle & Moulines, 2000). Then, the penalized contrast function may be minimized using various approaches based on dynamic programming.

Two of these algorithms are investigated here: optimal partitioning [OP] (see, for instance, (Jackson, 2005)) and the pruned exact linear time method [PELT] (Killick R. & Eckley, 2012). OP and PELT algorithms are both searching for the optimal partition minimizing the penalized contrast function, but the interest of PELT is a reduced computational time: the complexity of OP is quadratic, while the complexity of PELT is linear. Both methods require the a priori choice of a penalty term, usually the AIC or the BIC penalties. We will describe the different steps of the algorithm. The PELT method is a derivation of the OP method but with a better computational cost.

Let $Y = (Y_1, \dots, Y_N)$ be a sequence of random variables. For $t \in \{1, \dots, N\}$, suppose that Y_t is a function of $X_t \in \mathbb{R}^p$, where X_t is a random or deterministic vector. In our application Y is a monovariate signal and for now $X_t = (t, 1)$. X_t could also be the shaft speed or the altitude during the flight.

Also, assume there exists K^* (unknown) parametric changes in the relationship between (Y_t) and (X_t) : there exists

- An unknown vector $(\tau_1^*, \dots, \tau_{K^*}^*) \in \mathbb{N}^{K^*}$ such that $\tau_1^* < \tau_2^* < \dots < \tau_{K^*}^*$
- $K^* + 1$ unknown vectors $\theta_i^* \in \mathbb{R}^p$ satisfying $Y_t = g_{\theta_i^*}(X_t, \varepsilon_t)$ when $t \in \{\tau_i^* + 1, \tau_i^* + 2, \dots, \tau_{i+1}^*\}$, where by convention $\tau_0^* = 0$ and $\tau_{K^*+1}^* = N$.

This method aims at estimating K^*, $(\tau_1^*, \dots, \tau_{K^*}^*)$ and $(\theta_i^*)_i$, the parameters of the "true" model to be retrieved from an observed sample. In this model, $\theta_i^* = (\theta_i^{*(1)}, \theta_i^{*(2)})$ and $g_\theta(x, e) =< x, \theta > +e$, with $< \cdot, \cdot >$ the inner product. Let $\hat{\theta}_{u,v} = \arg\min_{\theta \in \mathbb{R}^p} \sum_{t=u+1}^{v} C\big(Y_t, X_t, \theta\big)$ be the estimate of θ computed in the time-interval $\{u + 1, ..., v\}$.

The off-line change-point detection strategy chosen here consists in minimizing in $(K, (\tau_i), (\theta_i))$ a penalized contrast defined by :

$$
\begin{aligned}
&(\hat{K}, \hat{\tau}_1, \dots, \hat{\tau}_{\hat{K}}) = \\
&\underset{K; \tau_1 < \tau_2 < \dots < \tau_K}{\arg\min} \left\{ \sum_{i=0}^{K} \sum_{t=\tau_i+1}^{\tau_{i+1}} C\big(Y_t, X_t, \hat{\theta}_{\tau_i, \tau_{i+1}}\big) + \beta f(K) \right\}
\end{aligned} \tag{1}
$$

where the cost function C to be minimized is the MDL (minimum description length), (Rissanen, 1978) and (Davis, Lee, & Rodriguez-Yam, 2006) :

$$
\begin{aligned}
C(Y_t, X_t, \hat{\theta}_{u,v}) &= 3\ln(v-u) + (v-u)\log(2\pi\hat{\sigma}^2) \\
\text{where } \hat{\sigma}^2 &= \frac{1}{v-u} \sum_{t=u+1}^{v} \big(Y_t - \hat{\theta}_{u,v}^{(1)} t - \hat{\theta}_{u,v}^{(2)}\big)^2
\end{aligned}
$$

The term $\beta f(K)$ is the penalty which prevents from overfitting. The choice of the penalty term is usually linear (for instance, AIC or BIC) in the number of break points.

The optimal partitioning [OP] algorithm used here for minimizing the penalized contrast in (1) was first described in (Jackson, 2005). The idea is to use dynamical programming in order to reduce the exponential complexity of an exhaustive search to a quadratic one. With the previous notations, a change-point occurs between the instants u and v if there exists some instant $u < l < v$ such that the cost computed for the interval $[u + 1, l]$ plus the cost computed for the interval $[l + 1, v]$ is less than the cost computed for the interval $[u + 1, v]$:

$$
\begin{aligned}
\sum_{t=u+1}^{l} C(Y_t, X_t, \hat{\theta}_{u,l}) + \sum_{t=l+1}^{v} C(Y_t, X_t, \hat{\theta}_{l,v}) + \beta \\
< \sum_{t=u+1}^{v} C(Y_t, X_t, \hat{\theta}_{u,v})
\end{aligned} \tag{2}
$$

Using this criterion, the OP algorithm scans the data iteratively and associates the optimal value of the penalized contrast, $F(Y_{1:u})$, to each subsample $Y_{1:u} = (Y_1, ..., Y_u)$. The proof that $F(Y_{1:u})$ contains the minimum of the penalized contrast for each $u = 1, ..., N$, may be sketched by the following recursion :

$$
\begin{aligned}
&F(Y_{1:N}) = \\
&\min_{u=1,\dots,N} \left\{ F(Y_{1:u}) + \sum_{t=u+1}^{N} C(Y_t, X_t, \hat{\theta}_{u,N}) + \beta \right\}
\end{aligned}
$$

The incremental part starts here. One should see that for each step u, the minimum cost is computed and this method reuses the previous computations to calculate the new one. One may see immediately that the time instant τ which gives the optimal $F(Y_{1:N})$ corresponds to the last change-point before N. The algorithm computes iteratively

$$
\begin{cases}
F_{1:N} = (F(Y_{1:1}), ..., F(Y_{1:N})) \\
CP_{1:N} = (CP(Y_{1:1}), ..., CP(Y_{1:N}))
\end{cases}
$$

where $F(Y_{1:u})$ is the minimum value of the penalized contrast and $CP(Y_{1:u})$ represents the last change point before u, for each subsample $Y_{1:u} = (Y_1, ..., Y_u)$, $u = 1, ..., N$. The optimal partition may be retrieved by scanning backwards $CP_{1:N}$. The OP method has a quadratic complexity although the PELT method previously introduced has a linear complexity.

Introduced in (Killick R. & Eckley, 2012), the PELT algorithm aims at reducing the computational complexity of the OP algorithm, while still retrieving the optimal solution. This is achieved by pruning the set of possible solutions when minimizing $F(Y_{1:u})$. Pruning is justified by the following property of change-points, proven in (Killick R. & Eckley, 2012): if equation (2) holds for some $u < l < v$ and if $F(Y_{1:u}) + \sum_{t=u+1}^{l} C(Y_t, X_t, \hat{\theta}_{u,l}) + \beta \geq F(Y_{1:l})$, then u can never be the last change-point before v. Pruning the set of possible change-points reduces the quadratic complexity of OP to a linear one. The algorithm, which consists in adding a supplementary step to the OP procedure, is summarized in *Procedure PELT*.

Procedure PELT

- Initialize $F(Y_{1:1}) = -\beta$, $R = \{1\}$ and $CP = NULL$
- For $u = 1, ..., N$, compute iteratively (forward) $F(Y_{1:u})$ and the corresponding break point : $\tau = \arg\min_{\tau \in R} F(Y_{1:u})$, $CP = CP \cup \tau$.
  Update the set of plausible change-points, R :
  $R \leftarrow \{v \in R \cup \tau : F(Y_{1:v}) + \sum_{t=v+1}^{\tau} C(Y_t, X_t, \hat{\theta}_{v,\tau}) + \beta \leq F(Y_{1:\tau})\}$
- Compute iteratively (backward) the optimal change points :
  $\hat{\tau}_K \leftarrow \arg\min_{1,...,N} F(Y_{1:N})$, $\hat{\tau}_{K-1} \leftarrow \arg\min_{1,...,\hat{\tau}_K} F(Y_{1:\hat{\tau}_K})$, ...

After a test session on simulated data with good results, an application of the PELT method on flight data has been made(see Figures 6 and 7):

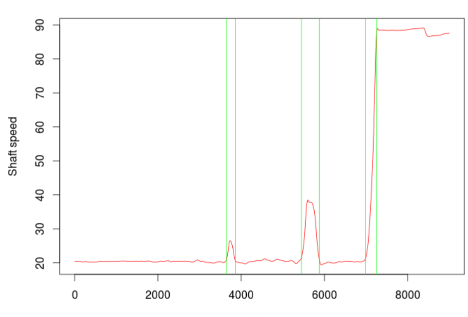

Figure 6. Results for PELT on the shaft speed.

PELT achieves the best trade-off between accuracy of results and computational time. Hence, this method was selected to be trained and tested on large real data. For illustration, an endogenous feature (the way the shaft speed changes) is represented in Figure 6. The behavior of this feature is mainly piecewise linear, hence the PELT algorithm previously described and used for simulations was used as such, but for more complex time-series, it is sufficient to select a more appropriate cost-function C and train exactly the same algorithm described in *Procedure PELT*.

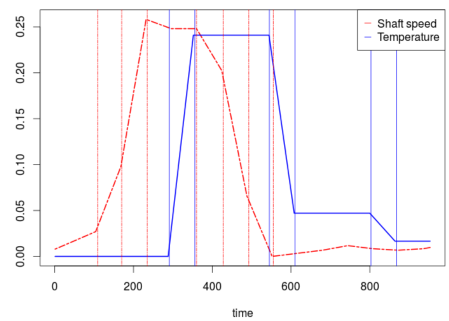

Figure 7. Results for PELT on binary commands.

In Figure 7, the results show two curves from the same flight during the same time lapse. On these data, the detected change-points are the expected one. In the analysis of this binary commands, we observe a similarity in the evolution of both curves with a certain delay. It suggests a relation of causality between the shaft speed and the temperature of the engine although this delay is only of a few seconds. This causality and this delay are interesting information that can benefit the experts.

Globally, the change-points are well detected, even the small ones. The meaning of each change-point is settled afterwards with the help of experts. A change-point may be observed, for instance, after an action of the pilot (pulling the lever). Detecting these change-points allows to assume causal relations between different features with some delay (a raise of the lever implies a raise of the shaft speed). Eventually, machine-learning techniques which will learn the "normal" change-points and set alarms for "abnormal" change-points may be applied afterwards in an operational context.

Recognition of transient patterns and classification

A transient phase is a temporal interval during which "something is observed". It is defined by the context that identifies a temporal interval and the behavior of the observations in that same interval.
For example, a transient interval may begin with the increase of the thrust level and end when the engine shaft reaches a stabilized speed and the engine temperatures stabilize also (Figure 8). In such specific event the pattern is defined by the "increase of thrust", it is parameterized by the amount of increase, the lever and probably the initial values of the engine shaft speed and temperatures.

Another example of pattern is still after an increase of thrust, but then after stabilization of the shaft speed, the vibration energy excited by this same shaft continues to change (Figure 9). In that case, the expert was positive that this problem

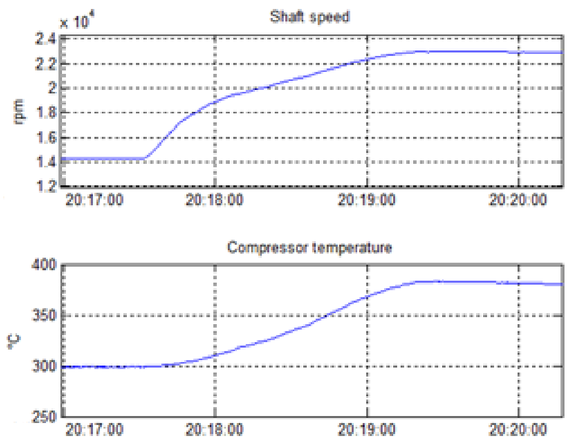

Figure 8. The thrust level change at 20:17:35, then the shaft speed reaches a new stabilized value with a concave pattern and the compressor temperature increases in a different way until it also reaches a new stabilized value.

already appear in old engine configurations but was not able to retrieve information from the knowledge base.

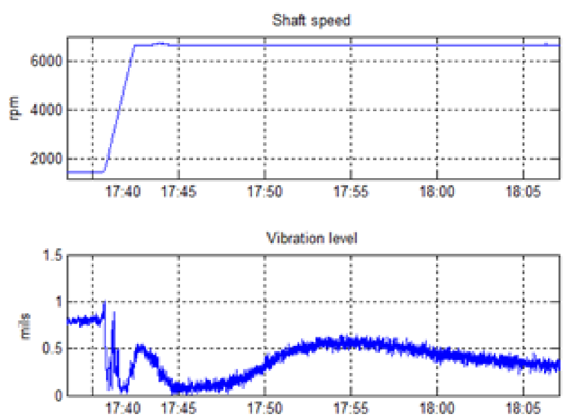

Figure 9. A maneuver increases the speed of the engine, when the speed stabilized, the vibration filtered around the speed shaft frequency continue to change.

To code this specific pattern we identify the exogenous measurements (shaft speed increase) and the endogenous observations (the way the filtered vibration energy changes). One way to define a transient phase is to select all change points detected by the PELT method in different variables of the same flight and, connect them when break points appear in the same lapse of time (see Figure 10). In this example, all change-points detected are illustrated in Figure 7. The start point of the transient phase happens to be the first detected point in the shaft speed and the last point is the last detected point in the temperature. The transient phase will be defined this way. It is not excluded that the last point of a transient phase would not be the start point of the next transient phase.

The indexation helps the detection of the start and end points

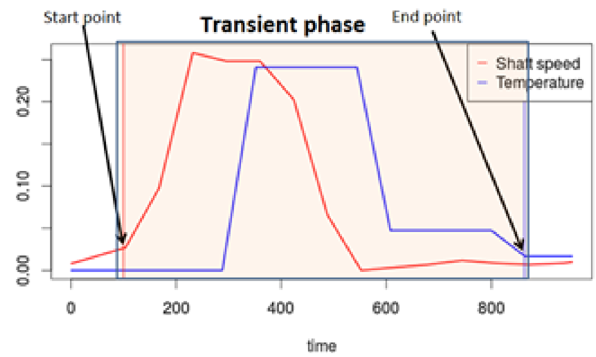

Figure 10. Representation of a transient phase on a bivariate signal.

of a transient phase. This phase will represent a pattern which will be coded with parameters and initial conditions. A lot of transient phases could happen during one test and, also among different tests it may be possible to observe the same kind of pattern. Therefore after collecting enough patterns, a classification is applied on these patterns and a sufficient database of patterns is created. In this database, the recurrent patterns and the unknown ones is stored and statistical studies are now available. For the classification, the detected patterns are compared through a matching algorithm.

We now have a tool that detects transient phases and classifies them into different topics. The main goal is to make this method run automatically everytime a new bench test is created. The detected signatures are stored in the existing database and the unknown patterns are sent to experts to look closely at them. If an expert has observed a specific pattern, this information is translated into a coded signature and a research of this type of pattern is done on the new database. Such work would never be possible manually, especially in high dimension. Some patterns may be easier to detect by their relation of causality. For example, stall of the compressor is observed by the opening of the TBV (Transfer Bleed Valves) that releases some pressure but is the result of regulation by the engine computer. The real pattern begins before the TBV opens and should be estimated by detection of a change in the compressor behavior, probably because of another action, for example an increase of electric power need, which is not a priori linked to this dramatic stall effect.

4. CONCLUSION

We are in the process of a wide indexation program of the flight data records for engines. The first step was the detection of change points, then recognition of transient phases. We implemented an algorithm that detect with a linear complexity the start and end points of a transient phase. The PELT method meets our expections in term of accuracy of results and computational time. The method is very flexible thanks to the cost function because depending the variable, it can change. The results on simulated and real data show that the change points are globally well detected.

The next step is labelling these patterns and, once we get some interesting number of labels and when we index a nice set of records, we could implement the search algorithm. Then we will produce label statistics and execute ranking computation by similarity on new records.

With the final result (data of all patterns), the possibility to check the normal or strange behavior of a variable will be available. In the mean time, we search to optimize our algorithms.

NOMENCLATURE

Y_t, X_t	sequence of random variable
N	size of sequence
t	time
K	number of change-points
τ_i	instant of break point
θ_i	parameter
$g_\theta(x, e)$	function
C	cost function
$u < l < v$	any instants of the sequence
$\beta f(K), \beta$	penalty term
$F(Y_{1:u})$	optimal value for the subsample $Y_{1:u}$
$CP_{1:u}$	last change point before u
$< .,. >$	inner product
AIC	Akaike Information Criterion
BIC	Bayesian Information Criterion

REFERENCES

Bai, J., & Perron, P. (1998). Estimating and Testing Linear Models with Multiple Structural Changes. *Econometrica*, *66*(1), 47–78. Retrieved from `http://www.jstor.org/stable/2998540` doi: 10.2307/2998540

Davis, R. A., Lee, T. C., & Rodriguez-Yam, G. A. (2006). Structural break estimation for nonstationary time series models. *Journal of the American Statistical Association*, *101*, 223-239.

Jackson, B. e. a. (2005). An algorithm for optimal partitioning of data on an interval. *IEEE Signal Processing Letters*, *12*(2), 105–108. doi: 10.1109/LSP.2001.838216

Killick R., F. P., & Eckley, I. (2012). Optimal detection of changepoints with a linear computational cost. *JASA*, *107*(500), 1590–1598. Retrieved from `http://arxiv.org/abs/1101.1438`

Lacaille, J. (2014). Robust monitoring of turbofan sensors. In *Ieee aerospace conference proceedings.* doi: 10.1109/AERO.2014.6836193

Lacaille, J., & Gerez, V. (2011). Online Abnormality Diagnosis for real-time Implementation on Turbofan Engines and Test Cells. *Phm*, 1–9.

Lacaille J., B. I. S. F. C., Bense W. (2016). Indexation of numeric bench test records, a big data vision. *IEEE Aerospace conference*.

Lavielle, M., & Moulines, E. (2000). Least-squares estimation of an unknown number of shifts in a time series. *Journal of Time Series Analysis*, *21*, 33–59. doi: 10.1111/1467-9892.00172

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*(5), 465–471. doi: 10.1016/0005-1098(78)90005-5

BIOGRAPHIES

Cynthia Faure is PhD student in statistics at the Statistique, Analyse, Modelisation Multidisciplinaire laboratory (SAMM) in Paris 1 University, Panthéon Sorbonne.

Jérôme Lacaille is an emeritus expert in algorithms for Safran international aeronautic group. He joined the Snecma Company in 2007 with responsibility for developing a health monitoring solution for jet engines. Jérme has a PhD from the Ecole Normale Supérieure (France) in Mathematics. He has held several positions including scientific consultant and professor. He has also co-founded the Miriad Technologies Company, entered the semiconductor business taking in charge the direction of the Innovation Department for Si Automation (Montpellier - France) and PDF Solutions (San Jose - CA). He developed specific mathematic algorithms that were integrated in industrial processes. Over the course of his work, Jérôme has published several papers on integrating data analysis into industry infrastructure, including neural methodologies and stochastic modeling as well as some industrial applications.

Jean-Marc Bardet is a full professor of applied mathematics in University Paris 1-Panthéon-Sorbonne since 2003. He completed his Ph.D. at the University Paris Sud-Orsay in 1997 under the supervision of D. Dacunha-Castelle. He also was an assistant professor in University Toulouse 3. His main thematics of research is "limit theorems, statistics and applications for time series and stochastic processes". He has published several articles in Annals of Statistics, Bernoulli, Stochastic Processes and Applications, Electronic Journal of Statistics, Journal of Time Series Analysis,...

Madalina Olteanu graduated the University of Bucarest in 2001, and obtained her Ph.D. in Applied Mathematics in December 2006, from Université Panthéon Sorbonne. Since Sept. 2007 she is an Assistant Professor at Université Paris 1 Panthéon-Sorbonne and member of the SAMM research team. Her main research interests are nonlinear time series, model selection, machine learning and applications to humanities and social sciences. She is author or co-author of more than 30 peerreviewed papers in international journals or conference proceedings.