Energy Threshold Detection in Neutronic Signal by Automatic Classification and Linear Programming

Philippe Durand , Luan Jaupi, Dariush Ghorbanzadeh

***Abstract*—It is proposed in this paper the automatic detection thresholds in the neutronic noise from a nuclear reactor chamber. The proposed method combines a pattern recognition algorithm, known as dynamic cluster and a linear programming algorithm. Euclidean distance, often adopted by default, in the dynamic cluster algorithm; is here replaced by the distance L^1, also called least absolute deviations (LAD).**

***Index Terms*—EMC, Dynamic Cluster Algorithm, Linear Programming, Neutronic noise.**

I. Introduction

We are interested in this study at temporal representations of neutronic noise. This noise is recouped by magnetic tape, it is then digitized, allowing the computer processing.The Heart of the reactor of the nuclear power plant Bugey, consists of 157 fuel assembly of square section. Around this core are exposed four rooms detections; each of them has two overlapping sections (high and low). The main role of these rooms is to measure the neutronic flux delivered by the reactor core. Schematically, they consist of an enclosure filled with gas in which two electrodes immersed between which an electric field is established. When charged particles arrive, the ions are moved to each of the two electrodes according to their sign. These charges collected, are the neutronic signal. ionizing particles, being created by collisions between gas atoms and neutrons, the signal measuring the neutronic flux. The information extracted from this signal are the fluctuations of the mean value. We are interested in this noise (see fig. 1)

II. Nature of the experience achieved

Neutronic noise (fig. 1) can be considered as a random signal. When is varied by moving its vertical bars, the energy of the reactor, we can detect what results at the level of neutronic signal. One goal of our work is the analysis of the curves shown in Figure 2. The noise can be broken down into three parts. In the first part: $Z1$, reactor's energy is constant, the noise level can be considered stable around a mean value. In the second part $Z2$, we start down the bars, we detected several thresholds, for example two $s1$, $s2$ in picture three, corresponding to two displacements of the bars. These are the energies of these thresholds we propose to detect.

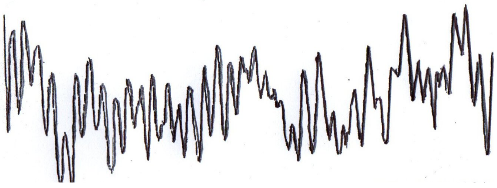

Fig. 1. Neutronic noise

Philippe Durand, Luan Jaupi, Dariush Ghorbanzadeh are with the Department of Mathematics (modélisation mathématiques et numériques), Conservatoire National des Arts et Métiers, 292 rue Saint martin, 75141 Paris FRANCE e-mail: philippe.durand@cnam.fr, leon.jaupi@cnam.fr, dariush.ghorbanzadeh@cnam.fr

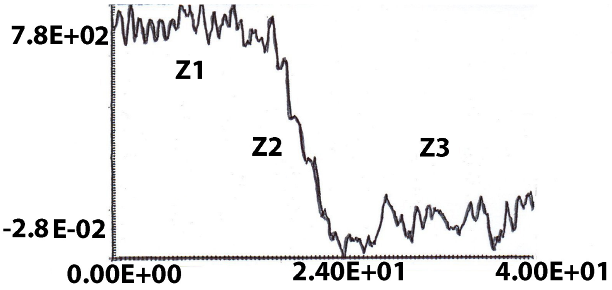

Fig. 2. Transition in neutronic noise

A. Pattern recognition segmentation problem.

The first problem is a localization problem: We need to detect the beginning and end of the bar movement .We then determined by the algorithm of dynamic clusters with three classes three most representative regression lines. At the convergence (last iteration of the algorithm), we obtain two points P_1 and P_2, which respectively gives the beginning and the end of the bar movement. The second problem is interested, in the transition region to determine the number of thresholds to be detected. Then adjusts the algorithm of dynamic clusters to the case of a variable number of classes (see fig. 3).

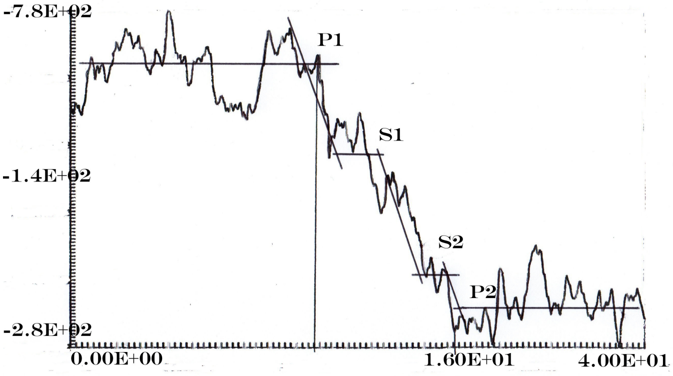

Fig. 3. Segmentation in neutronic noise

ISBN: 978-988-19253-4-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

III. The algorithm of dynamic clustering applied to our problem

The algorithm of dynamic clustering is introduced for the first time by Diday [2], [3]. This is a pattern recognition method as part of automatic classification of data.

A. The algorithm

Recall that the algorithm of dynamic clustering need the definition of a dissimilarity function acting on a point x and a class ω.

$$\forall x \in E, \forall \omega \in \mathcal{P}(E), (x, \omega) \to f(x, \omega) \in \mathbb{R}^{+} \tag{1}$$

Is arbitrarily chosen a starting partition $\mathcal{P}(E)$ of the set of representations E. Using the function of dissimilarity are reassigned the points, as to minimize the sum of dissimilarities. It also updates the partition Π So we compute the quantity:

$$g(\Pi) = \sum_{\omega \in \Pi} \sum_{\in \omega} f(x, \omega) \tag{2}$$

The algorithm stops when the partition obtained in step j is the one obtained in step $j + 1$.
The best partition is given by:

$$g(\Pi^{*}) = min(g(\Pi)) \tag{3}$$

It must be noted that the global minimum is hard to find, one is satisfied in many cases looking for a local minimum for this, the choice of the starting partition is determinant.

B. Measure of dissimilarity, and linear regression

Neutronic noise is decomposed into stable area around a mean value. A linear regression technique is used to approximate each stable area. Given a norm N of Euclidean space, we define the measure of dissimilarity D given by:

$$D((x_t, y_t), L_{a^j, b^j}) = N(y_t - (a^j x_t + b^j)) \in \mathbb{R}^{+} \tag{4}$$

In this equation each L_{a^j, b^j} is an element of the partition set at iteration j, Specifically, is regression line at step j. Then, we try to minimize:

$$\sum_{j=1}^{k} \sum_{t=1}^{P_j} N(y_t - (a^j x_t + b^j)) \tag{5}$$

In his thesis, Diday [2] show that the number of iterations is sensitive to the initial partition and the choice of a distance, that is what we will discuss in the following.We can already say that the use of the Euclidean norm would not be a good choice in the case of signals such as white noise. Indeed very remote values of the regression lines will be taken into effect. Our choice'll cover a distance L^1 norm, and transform a conventional linear regression problem into a linear programming problem.

IV. Linear programming

It is well known that a linear programming problem returns to solve the following problem:

$$\begin{cases} Min_x(c^T.x) \\ A'.x \leq B \\ x \geq 0 \end{cases} \tag{6}$$

If we add deviation variables,(augmented problem) the problem amounts to solving a linear system (with equalities constraints):

$$\begin{cases} Min_x(c^T.x) \\ A.x = B \\ x \geq 0 \end{cases} \tag{7}$$

In a linear programming problem is then distinguished feasible solutions that is to say those for which $A.x = B$ and that are optimal in the sense that $C^T x$ is optimum. Can be treated this augmented problem (only equalities constraints related to the matrix A) thanks to the tools of linear algebra in dimension n. It introduces the concept of feasible basis from under square matrices extracted.

A. Convex polyhedrom

Given a linear problem, ruled by the increased matrix linear constraints: A, if A has rank m, we say that B is a base, if it is a regular sub matrices of rank m. We are interested then the bases feasible solutions. They are obtained from the top of the convex polyhedron associated with the linear problem. Recall that:

Definition IV.1. A polyhedron is a closed polytope and a polytope is the intersection of a finite number of half-closed spaces. These are convex sets as finite intersection of convex.

The set (polytope or polyhedron S) has a number of vertices $\nu(S)$ equal to: $\begin{pmatrix} n \\ m \end{pmatrix}$

Let S be a convex set to any positive y, λ, called ray, the set of $x + \lambda y$

Theorem IV.1. *Any point of a convex polytope can be expressed as a linear combination of extreme points, to which is added eventually, a linear combination of extremal rays.*

Theorem IV.2. *The optimum of a linear function, on a convex, is attained in at least an extemal item. If it is achieved by many, it is also for any linear combination of extremal points.*

With the simplex algorithm, only explores vertices that reduce the cost function.

B. Symplex algorithm

We recall in this part symplex algorithme:, if B is a basis of A, A split into block matrix:

$$A = \begin{pmatrix} C_B & C_N \\ B & N \end{pmatrix} \tag{8}$$

1°) $B = B^0$ feasible basis init

2°) $I := I + 1$

3°) Calculate $\overline{b} = B^{-1}b$, $\pi = c_B.B^{-1}$, $\overline{c_N} = C_N - \pi.N$

4°) if $\overline{c_N} \leq 0$ end : the optimum is reached else if $\exists s$ s.t $\overline{c_s} > 0$ then

5°) compute $\overline{A_s}$=$B^{-1}A_s$ (A_s = (a_{is} column of A if $\overline{a_{is}} \leq 0$ for all i end: optimun unbounded. else $\widehat{x_s}$= $Min_{is.t\overline{a_{is}}>0}\frac{\overline{b_i}}{\overline{a_{is}}}$

6°) let x_t column r of $B^{-1}A_t = e_r$ then the new feasible base is $\widehat{B} + s - t$ and $B \leftarrow \widehat{B}$ and return to 2)

V. Linear regression and linear programming

The choice, which is adopted to measure the differences is the L^1 norm, The problem is a minimization in absolute value:

$$\sum_{j=1}^{k}\sum_{t=1}^{P_j}|y_t - (a^j x_t + b^j)| \qquad (9)$$

If we introduce deviation variables, linear regression treated problem is transformed into a linear programming problem [1], find a, b such that:

$$\begin{cases} ax_1 + b + \varepsilon_1 = y_1 \\ ax_2 + b + \varepsilon_2 = y_2 \\ ..... \\ ax_n + b + \varepsilon_n = y_n \end{cases} \qquad (10)$$

$$\sum_{i=1}^{n}|\varepsilon_i| minimum$$

We denote ε_i^+, ε_i^- the n_1 deviations positives and n_2 negatives.

$$\sum_{i=1}^{n}|\varepsilon_i| = \sum_{i=1}^{n_1}\varepsilon_i^+ + \sum_{i=1}^{n_2}\varepsilon_i^- \qquad (11)$$

VI. Application to the segmentation and thresholding neutronic noise

The different steps of the algorithm are summarized below:

A. Segmentation and thresholding program

1°) Step one: select a partition P_0 at random

2°) Step two: while $P_{n+1} \neq P_n$ using the measure of dissimilarity, $D((x_t, y_t), L_{a,b}) = |y_t - (ax_t + b)|$ recalculate the distance of the points to the lines regressions, and change classes; with the help of the simplex algorithm recalculate regression lines relating to the new partition.

3°) Step Three: give the results

We tested the algorithm on the example in Figure 4, the algorithm converges very quickly, in four iteration we obtain a very precise result, the non-recourse Euclidean distance saves several iteration, it eliminates parasites points: those too distant of classes kernels. This algorithm has been used twice. the first to determine the transition of beginning of the bars movements, the second time to determine the energy threshold.

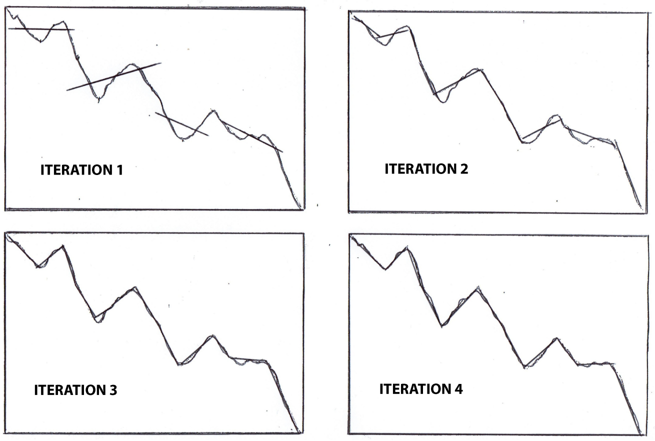

Fig. 4. Algorithm iterations

B. Applications to neutronic noise

The program defined in the previous section is used for two specific applications. The first is the automatic detection of the sequence of bar movements The algorithm is programmed to detect three classes and then the dynamic clustering algorithm has a partition noise in three disjunct subsets. The beginning of the bar movement is obtained by the intersection of the first two regression lines, respective cores of the two first class to the convergence of the algorithm. The intersection of the last two regression lines, detects the end time of the bar movement.

The second application is the detection of energy levels associated with displacements of the bars. In our modeling, we know the number of displacements. This allows to know the number of thresholds corresponding to the energy levels of the reactor. Corresponding partition can thus define of the noise in the algorithm of dynamic clusters. At the convergence of the algorithm, the horizontal lines gives the values of the different levels of energy obtained.

The results obtained by this method are exploitable and algorithm developed here converges faster due to the linear programming module that replaces the Euclidean distance.

VII. Conclusion

This study allowed us to test a method of automatic classification for thresholding neutron noise. This method has been implemented using the linear programming resources. Indeed the use of a suitable distance helps to speed the classification algorithm. The use of this technique can effectively detect the beginning and end of bar movement of a nuclear reactor; in the transitional phase, it also allow to quantify the energy of the signal at different threshold, depending on bar descents.

References

[1] N. Abdelmalek, "An Efficient Method for the Discrete Linear L1 Approximation Problem)," Mathematics of Computation, Vol. 29, number 131, pp. 844-850

[2] E. Diday, "Phd thesis", University Paris VI, 1972.

[3] E. Diday and J.C Simon, "Clustering Analysis" In Digital Pattern Recognition, Springer-Verlag, Berlin, 1976.

[4] M. Gondran, M. Minoux “Graphes and Algorithmes,” Lavoisier, 2009.

[5] I, Tomek “More on piecwise linear approximation” Computers and Biomedical Research, Vol 8, pp. 568, 572; 1975.