



**HAL**  
open science

## Evaluation of a two-step clustering method for radar pulse deinterleaving

Louis Lesieur, Jean-Marc Le Caillec, Ali Khenchaf, Vincent Guardia

► **To cite this version:**

Louis Lesieur, Jean-Marc Le Caillec, Ali Khenchaf, Vincent Guardia. Evaluation of a two-step clustering method for radar pulse deinterleaving. 2023 IEEE Conference on Antenna Measurements and Applications (CAMA), Nov 2023, Gênes, Italy. pp.1047-1051, 10.1109/CAMA57522.2023.10352731 . hal-04357435

**HAL Id: hal-04357435**

**<https://hal.science/hal-04357435>**

Submitted on 29 Apr 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Evaluation of a two-step clustering method for radar pulse deinterleaving

Louis Lesieur  
*Lab-STICC, UMR CNRS 6285*  
*ENSTA Bretagne*  
Brest, France  
louis.lesieur@ensta-bretagne.org

Jean-Marc Le Caillec  
*Lab-STICC, UMR CNRS 6285*  
*IMT Atlantique*  
Brest, France  
jm.lecaillec@imt-atlantique.fr

Ali Khenchaf  
*Lab-STICC, UMR CNRS 6285*  
*ENSTA Bretagne*  
Brest, France  
ali.khenchaf@ensta-bretagne.fr

Vincent Guardia  
*Thales DMS*  
Brest, France  
vincent.guardia@fr.thalesgroup.com

**Abstract**—In electronic warfare, deinterleaving is the task which sorts radar pulses in order to separate radar waveforms, without prior knowledge. This paper intends to evaluate two versions of a two-step deinterleaving method. It starts with a RF-PW clustering, then tries to merge some clusters by defining a distance related to their set of pulse times of arrival. After a parameter optimization, the method is evaluated with five metrics on a dataset consisting of simulated frequency-agile and PRI-modulated signals, and on a real data observation.

**Index Terms**—electronic warfare, deinterleaving, radar pulses

## I. INTRODUCTION

**Context.** Electronic Warfare (EW) receivers are passive systems aiming to detect and identify emitters present in the environment. Identifying the received pulses, and possibly identifying the (friend or foe) source of this emission is a crucial point in modern military operations. In fact, in this kind of mission, an airplane is illuminated by a huge number of radar or non-radar (e.g. communication) pulses. The EW receiver processes the signal to extract and characterize the flow of radar pulses, which are interlaced, turning our aim of identifying a source not straightforward. The role of deinterleaving involves sorting them according to the emitter by processing pulses of a time window so as to form groups.

Radar pulses are characterized by a set of parameters measured by the receiver. The most significant are:

- the time of arrival (TOA),
- direction of arrival (DOA),
- pulse amplitude (PA),
- pulse width (PW),
- radio frequency (RF).

The TOA ( $\mu s$ ) is the pulse time of arrival, that is the time when the amplitude rising edge exceeds a threshold, its confidence interval is directly related to the receiver specifications. PA is the pulse amplitude, as measured by the receiver, it depends on the energy given by the emitter and the attenuation through

the atmosphere. PW (in  $ns$  or  $\mu s$ ) is frequently defined as the duration of the pulse when the amplitude is above  $-3dB$  of the maximum amplitude. This parameter is chosen by the emitter, as the RF. In EW, the mean frequency lies in the  $[1 - 40]GHz$  interval [1].

To describe a waveform, other parameters are of interest, such as the pulse repetition interval (PRI). It quantifies the duration between the emission of consecutive pulses by an emitter. It is generally measured in  $\mu s$  or  $ms$ .

**History.** Various deinterleaving methods have been developed, each one taking advantage of some parameters.

The first set of methods aims to exploit TOA and PRI. The principle of these methods is to assume the existence of waveforms having a certain PRI, then extract relevant pulses. The CDIF method [2] uses differential times of arrival (DTOA) of multiple degrees to build a cumulated histogram. A variant [3] does not cumulate degrees and uses a different threshold.

A second set of methods deals the deinterleaving problem as a clustering problem in a multi-dimensional space. All the parameters can be considered. The authors of [4] do cluster in the space (RF, PA, PW). Clustering can also be used as a pre-processing aiming to separate pulses that cannot be associated, for instance in case of a significant frequency gap. The authors of [5] process in two steps, with a Fuzzy-ART clustering on DOA/RF, then a DTOA histogram to de-interlace each cluster. Mottier et al. [6] present an original approach, which consists in doing a two-step clustering.

Recently, machine learning algorithms have been proposed, using a great diversity of architectures : SVM [7], RNN [8], [9], CNN [10], Transformer [11].

This article proposes an evaluation of a two-step clustering method introduced in Mottier et al. [6], which does not present quantitative result.

We begin by a presentation of the method in section II. In section III, we detail the experimental dataset and the evaluation metrics. Lastly, results are presented in section IV.

## II. METHOD

The selected methodology is first proposed by Mottier et al. [6]. Only TOA, PW and RF are used in this method, each pulse is associated with its Pulse Descriptor Word (PDW)  $x = (t, p, f)$ .

### A. Step 1

**Pre-processing.** Before the clustering in the PW-RF plane, a scaling is necessary in order to make  $p$  (ns) and  $f$  (GHz) consistent, so the time dimension is not flattened by the frequency dimension. For this pre-processing, min-max normalization is used, with parameters chosen by hand accordingly with the frequency bandwidth of interest.

**DBSCAN.** Then, in the first step, pulses are placed in the RF-PW plane, and a clustering is applied. DBSCAN [12] is a density-based clustering algorithm, it is an unsupervised method, which means the labels are not needed to perform. But it needs two parameters which have to be carefully fixed : the neighborhood radius  $\epsilon$  and a minimum number of points  $MinPts$ . Basically, DBSCAN defines clusters as a set of close points. If two points have a distance less than  $\epsilon$ , and if their respective number of neighbors is over  $MinPts$ , then they belong to the same cluster.

**Discussion.** It is a question of grouping pulses with similar pulse width and frequency. The parameters are chosen at the same time as other parameters explained further, in a Bayesian search on training data. This step provides as a temporary output, the integer vector  $y^{(1)}$  of clusters. From now on, the clusters made in this step will not be divided. Here, the strong hypothesis is that the pulses of a cluster come from the same emitter, such a cluster is said to be pure. Some points can be classified as outliers, they will stay as so until the end of the processing. The final number of targets identified will not exceed the number of clusters.

### B. Step 2

**Motivation.** Some waveforms can have frequency agility, that is to say the radar emits pulses of different frequencies successively. Pulses of such an emitter are divided into several clusters after an ideal PW-RF clustering. The second step aims to merge those clusters.

**Discussion.** Each cluster is characterized by its set of pulse times of arrival (TOAs). The clustering algorithm is hierarchical temporal clustering, using optimal transport distances between sets of TOAs. Indeed, most radars are rotating, and the distribution of pulses of an emitter, in terms of TOA over an observation window, corresponds to one or several lobes. Thus it is possible to think of a temporal similarity, regarding TOA distribution of each cluster. The authors use optimal transport distance to do so. Intuitively, clusters with lobes occurring at similar times have a narrow distance. This notion of distance allows one to use agglomerative hierarchical clustering as the final step. It merges some integers of  $y^{(1)}$  to build the final output  $y$ .

### C. Step 2, an alternative

An alternative to the second clustering is also proposed and evaluated, we refer to it as version 2 in this article.

**Binary vector.** It requires to compute a binary temporal vector, the size of which is equal to 100 for all clusters. This vector  $b$  is a discrete representation of the 150 ms observation time window, where each bit corresponds to a 1.5 ms. The vector aims to represent the time interval when the lobe is visible, we call this interval "time support". We complete the zero-initialized vector with ones between the beginning and end of the lobe. For larger observation windows, over than a few seconds, several lobes of a cluster can be seen, it would require to separate them first.

**Jaccard distance.** A distance is computed between two binary vectors. We chose Jaccard distance, based on Intersection over Union similarity, as we want matching lobes to have minimum distance, in order to be merged then:

$$d_J(b_1, b_2) = 1 - \frac{|b_1 \cap b_2|}{|b_1 \cup b_2|}$$

Two lobes with no intersection have maximum distance.

**DBSCAN.** DBSCAN is then applied again, with precomputed distance matrix.

## III. EXPERIMENTAL SETUP

### A. Waveforms

For this experiment, we use a training set of simulated data to find optimal parameters, and a test set for evaluation.

**Frugality.** The use of simulated data is justified by data frugality. Unlike other domains such as image classification, which has no issue finding large ready-to-use datasets, we face a data frugality issue in electronic warfare. That is due to the technical difficulty to correctly label real data. Another aspect of data frugality is that we do not have knowledge of many waveforms. Using them for training may cause a discrepancy in terms of generalization capacity in the case of supervised learning methods. In this part, we explain how we overcome these issues.

**Data generation.** We design a waveform random generator with expert knowledge. To limit the inserting of bias in our waveforms, we have to

- generate waveforms that do not violate radar theory,
- respect the orders of magnitude,
- represent the range of PRI modulation types.

**Primary parameters.** The simulated parameters are TOA, PW, RF. Amplitude and direction of arrival are not considered in our method. Finally, the value domain of PW is reduced to 3 values in order to avoid trivial solutions such as separating waveforms only based on PW.

**Secondary parameters.** All radars are supposed to rotate, and the lobe duration is mostly between 10 ms and 100 ms. The PRI values of a dataset can vary with a factor of 100. The considered PRI modulation types are 25% fixed, 25% jitter, and 50% stagger which can have 4 to 15 pulses in the stagger pattern.

## B. Observations

**Datasets.** We create two datasets to answer our needs of training and evaluation. For each, a waveform collection is first created, which is then used to create a dataset  $D = (X_k)_k$  of scenes, each scene  $X = (x_i)_i$  being a time series of pulses.  $D^{(train)}$  is intended to optimize parameters, it contains 400 waveforms to a total of 100 scenes.  $D^{(test)}$  is used for the final evaluation, and also contains 400 waveforms to a total of 100 scenes.

**Observation scenes.** We define a scene as the observation of a subset of the waveforms set, on a 150 ms time window. This duration implies that waveforms only have one observed lobe, because the rotation period is greater than 150 ms. The number of waveforms is uniformly sampled between 1 and 8. Every waveform is randomly shifted along the temporal origin, and there is no pulse loss. A noise is added on parameters to simulate the impact of the receiver previous processes.

## C. Metrics

**ARI.** The main metric of our experiment is a clustering metric, frequently used in research about deinterleaving [4], [10] : Adjusted Rand Index (ARI). This index makes deinterleaving a binary classification problem of all pulse pairs : two pulses belong to the same waveform, or not. Rand index is the percentage of correctly classified pairs, and the adjusted index gives a normalization of it, so as to obtain 0 after a random clustering and 1 after a correct clustering.

**More metrics.** In addition, we derive metrics from fuzzy loss functions of Gasperini et al. [10] : purity, non-fragmentation, and detection, that have the advantage to be interpretable. They are computed from a confusion matrix, created with a number of lines equal to the number of emitters, and a number of columns equal to the number of predicted emitters. Purity evaluates how much each predicted cluster is composed of the same waveform. Fragmentation evaluates how much pulses of a waveform are divided in several classes. Plus, detection allows us to verify that each waveform is represented by a main cluster. The metrics should be considered together, to avoid trivial solutions which do not take into account the problem complexity.

**Reject rate.** Finally, we use a reject rate, for pulses classified as outliers in step 1.

## D. Optimization

**Parameters.** In the experiment, we search for optimal parameters for the two versions of the method. Both versions share the same first step, the RF-PW clustering. This step has two parameters to be optimized :  $\epsilon_1$  and  $NPts_1$ . For the second step, version 1 computes optimal transport distances between TOAs, then executes an agglomerative hierarchical clustering which needs a value  $t_{distance}$  for the distance criterion. Version 2 computes IoU distances, then makes another DBSCAN clustering with  $NPts_2 = 1$ . The only parameter to set is  $\epsilon_2$ . In the end each version of the method has 3 parameters to be optimized.

**Scores.** We use Bayesian optimization to find optimal the parameters. Several scores  $s$  are tested as objective functions, with reject rate  $r$ , ARI  $a$ , and Gasperini et al. derived metrics which of purity  $p$ , non-fragmentation  $nf$  and detection  $d$ :

$$s = (1 - r) \cdot a \quad (\text{A})$$

$$s = (1 - r) \cdot \frac{p + nf + d}{3} \quad (\text{G})$$

$$s = (1 - r) \cdot \left( \frac{a}{2} + \frac{p + nf + d}{6} \right) \quad (\text{A+G})$$

Multiplication by  $(1-r)$  is justified by the necessity to have a reject rate as low as possible. All the other components of the equations are "higher is better" metrics. Score A tries to optimize ARI, G tries to optimize all Gasperini metrics with equal contributions, and A+G is a mean between A and G.

## IV. RESULTS AND DISCUSSION

### A. Results

**Simulated data.** The results on simulated data are presented in Table I. First, we note that in all cases the reject rate is almost zero, and purity is quite high. This is an indication that step 1 clustering is well optimized. Then, it is obvious that there is a trade-off between non-fragmentation and detection. G score tends to favor non-fragmentation whereas A and A+G scores tend to favor detection. It is possible that setting coefficients for metrics inside G score would balance the final results, for instance by making detection more important. Finally, version 1 is better than version 2 regarding non-fragmentation. This can be explained because version 2 algorithm merges two clusters if their lobe is occurring at the same time, not considering PRI modulations inside the lobes. Version 1 benefits from optimal transport to avoid merging clusters whose PRI modulations are different. The advantage in performance of version 1 is to be balanced with its cost in computing time, higher than version 2.

TABLE I  
RESULTS ON SIMULATED DATA AFTER OPTIMIZATION

Version	Score	Reject	ARI	Purity	Non-Frag	Detection
1	A	0.02	0.91	0.98	0.83	0.95
	A+G	0.01	0.88	0.98	0.81	0.96
	G	0.00	0.85	0.93	1.00	0.83
2	A	0.00	0.85	0.98	0.72	0.98
	A+G	0.00	0.85	0.98	0.73	0.97
	G	0.00	0.85	0.93	1.00	0.83

**Real data.** Additionally to the results on simulated data, we apply two-step clustering method to the real data observation shown in Fig. 1. There are four emitters identified by a color and an identifier. Orange is a fixed frequency radar, blue is a tight-band agile radar. Green and red are large-band stagger radars with overlapping frequencies. Three lobes are also overlapping over time.

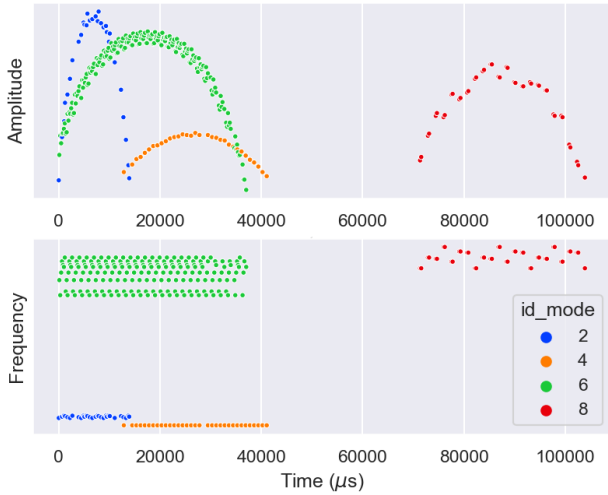


Fig. 1. Four emitters

We choose to apply the method version 1 optimized by A score, because it has maximum ARI in our experiments. The results are shown in Table II. Five emitters are predicted. The two less frequency-agile radars are successfully isolated. However, one radar is fragmented in 3 classes. The last radar is not even detected, step 1 failed to make clusters and classified all its pulses as outliers, noted as -1 in Fig. 2.

TABLE II  
RESULTS ON REAL DATA OBSERVATION SCENE

Version	Reject	ARI	Purity	Non-Frag	Detection
1	0.14	0.38	1.00	0.60	0.75

### B. Discussion

**Step 1 options.** Several approaches can pretend to handle step 1 correctly. In our experiment, we perform a min-max

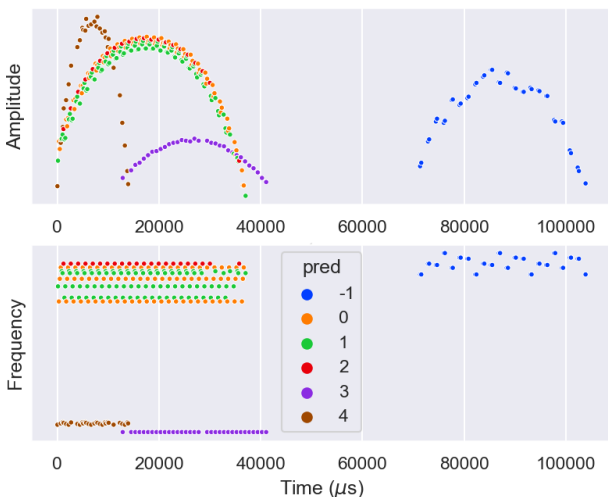


Fig. 2. Prediction : five emitters (0 to 4) and outliers (-1)

normalization. Another candidate is standardization on the train set. It is also possible to not do any normalization, and perform one-dimension DBSCAN on parameters one by one.

**Failing cases.** The failing cases are easily identifiable thanks to our metrics, they are the following. If step 1 fails, this can result in low purity or low detection rate. Low purity can occur when the clustering algorithm fails to separate emitters, when their parameters are close enough. The error can then propagate in step 2. Low detection rate is the other negative consequence of step 1 failure : it is the case in which a radar is not represented by a main cluster, as shown in Fig. 2. If step 2 fails, this can result in a high fragmentation rate, as in Fig. 2.

**Interpretability.** The two-step clustering deinterleaving methodology has the advantage to be very simple, and as opposed to deep learning methods, it is not a black box. A human agent can easily interpret the results of each step.

**Use case of time support.** It is interesting to notice a use case of the Jaccard distance between time support vectors, proposed for version 2. Sometimes the frequencies emitted by a radar can be received at different levels of amplitude, because of specific sea conditions for instance. In this case, several lobes can be observed, with simultaneous peaks. The detection threshold makes their time support different, that is to say their duration of lobe is unequal, as illustrated in Fig. 3 (red). The trick is to consider only the top part of the lobes, lets us say over -3dB of the maximum amplitude (green). Now the method would succeed, as the Jaccard distance between such time vectors would be close to zero.

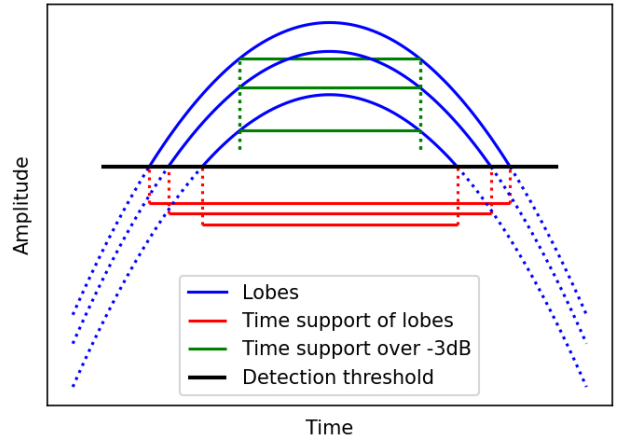


Fig. 3. Two ways of computing time support of lobes

**Long-term deinterleaving.** Finally, the TOA-based distances between RF-PW clusters presented in this article can be more than useful on longer time windows. In a 150 ms window, it is possible that 2 lobes occurring at the same time is a coincidence. In this case, our method would fail. However a close distance would be much more significant in a 10 s window, with several lobes matching exactly. This information could be used to track deinterleaved objects through time.

## V. CONCLUSION AND PERSPECTIVES

After a presentation of the context of electronic warfare and deinterleaving, we explained the selected methodology of two-step clustering. Step 1 consists of a RF-PW clustering, and step 2 defines a distance between clusters in order to merge some of them based on their TOAs. While the method by Mottier et al. uses an optimal transport distance and executes an agglomerative clustering, we proposed to use the Jaccard distance between binary vectors representing the time support of lobes. The parameters of the versions are set after a Bayesian optimization on simulated data. Then, we evaluated them with five metrics on simulated and real data. To conclude, the methodology performs well on simulated data but has difficulties to process real data. A perspective to enhance this deinterleaving method would be to train them on more realistic data or even real data directly. Lastly, it would benefit from the exploitation of other parameters such as pulse amplitude and PRI characterization.

## ACKNOWLEDGMENT

This work is part of a collaboration between Thales DMS and Lab-STICC.

## REFERENCES

- [1] Schleher, D. Curtis. *Electronic warfare in the information age*. Artech House, Inc., 1999.
- [2] Mardia, H. K. "New techniques for the deinterleaving of repetitive sequences." *IEE Proceedings F (Radar and Signal Processing)*. Vol. 136. No. 4. IET Digital Library, 1989.;
- [3] Milojević, D. J., and B. M. Popović. "Improved algorithm for the deinterleaving of radar pulses." *IEE Proceedings F (Radar and Signal Processing)*. Vol. 139. No. 1. IET Digital Library, 1992.
- [4] Mahmud, Shad. "Deinterleaving pulse trains with DBSCAN and FART." (2019).
- [5] Ata'a, A. W., and S. N. Abdullah. "Deinterleaving of radar signals and PRF identification algorithms." *IET radar, sonar & navigation* 1.5 (2007): 340-347.
- [6] Mottier, Manon, Gilles Chardon, and Frédéric Pascal. "Deinterleaving and Clustering unknown RADAR pulses." *2021 IEEE Radar Conference (RadarConf21)*. IEEE, 2021.
- [7] Mu, Hao, Jie Gu, and Yaodong Zhao. "A Deinterleaving Method for Mixed Pulse Signals in Complex Electromagnetic Environment." *2019 International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, 2019.
- [8] Liu, Zhang-Meng, and S. Yu Philip. "Classification, denoising, and deinterleaving of pulse streams with recurrent neural networks." *IEEE transactions on aerospace and electronic systems* 55.4 (2018): 1624-1639.
- [9] Chao, Wang, et al. "A Radar Signal Deinterleaving Method Based on Semantic Segmentation with Neural Network." *IEEE Transactions on Signal Processing* 70 (2022): 5806-5821.
- [10] Gasperini, Stefano, et al. "Signal clustering with class-independent segmentation." *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020.
- [11] Can, Oğul, et al. "Blind Deinterleaving of Signals in Time Series with Self-attention Based Soft Min-cost Flow Learning." *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021.
- [12] Ester, Martin, et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." *kdd*. Vol. 96. No. 34. 1996.