Deinterleaving of Discrete Renewal Process Mixtures with Application to Electronic Support Measures

Jean Pinsolle, Olivier Goudet, Cyrille Enderli, Sylvain Lamprier, Jin-Kao Hao

Abstract—In this paper, we propose a new deinterleaving method for mixtures of discrete renewal Markov chains. This method relies on the maximization of a penalized likelihood score. It exploits all available information about both the sequence of the different symbols and their arrival times. A theoretical analysis is carried out to prove that minimizing this score allows to recover the true partition of symbols in the large sample limit, under mild conditions on the component processes. This theoretical analysis is then validated by experiments on synthetic data. Finally, the method is applied to deinterleave pulse trains received from different emitters in a RESM (Radar Electronic Support Measurements) context and we show that the proposed method competes favorably with state-of-the-art methods on simulated warfare datasets.

Index Terms—Deinterleaving; Renewal process; Maximum likelihood estimation; Electronic support measure; Radar.

I. INTRODUCTION

WE consider the following problem. A sequence $z = (z_i)_{i \in \mathbb{Z}}$ of symbols has been observed during a time window $[\![0, T]\!]$, with $T \in \mathbb{N}^*$. The symbols z_i of the sequence arrive at successive integer time steps $t_i \in \mathbb{Z}$, with the first symbol of the observation window being indexed as z_0: $0 \leq t_0 < t_2 < \cdots \leq T$. t is the sequence of arrival times observed until time T. Each symbol of z is drawn from a finite set $\mathcal{A}$ (alphabet). The underlying generative model P of this sequence is assumed to be a set of $m > 0$ Markov processes $G_{\Pi}(\{P^e\}_{e \in E(\Pi)})$, where $E(\Pi)$ is a set of emitters and $m = |E(\Pi)|$ is the number of different emitters in this set. Each P^e, with $e \in E(\Pi)$, is an independent random process for emitter e, generating symbols in the sub-alphabet $A_e \subset \mathcal{A}$ with their corresponding arrival times. $\Pi = \{A_e\}_{e \in E(\Pi)}$ is the partition of $\mathcal{A}$ into the sub-alphabets A_e, which are assumed to be non-empty and disjoint. Given a sample (z, t) from P, observed until time T, and without prior knowledge about the number of sub-alphabets A_e (number of emitters), the deinterleaving problem of interest is to reconstruct the original partition $\Pi = \{A_e\}_{e \in E(\Pi)}$ of the alphabet.

J. Pinsolle and C. Enderli are with Thales DMS France SAS, 2 Av. Jean d'Alembert, 78190 Trappes, France (e-mail:jean.pinsolle@fr.thalesgroup.com; cyrille-jean.enderli@fr.thalesgroup.com).

O. Goudet, S. Lamprier, J.K. Hao (Corresponding author) is with LERIA, Université d'Angers, 2 Boulevard Lavoisier, Angers 49045, France (e-mail: sylvain.lamprier@univ-angers.fr; olivier.goudet@univ-angers.fr; jin-kao.hao@univ-angers.fr).

This problem is motivated by an application in the context of electronic warfare, where an airborne ESM (Electronic Support Measure) receiver is located in an environment composed of many radars. This receiver intercepts pulse trains from multiple transmitters over a common channel. Each pulse is characterized by a frequency, which can be associated with a symbol z_i in alphabet $\mathcal{A}$ (using a clustering method in frequency space), and a time of arrival (TOA), which can be discretized and assigned an integer value t_i. Other parameters such as pulse duration, power, or angle of arrival, may also be available, but in this work we restrict attention to the time and frequency information. In this context, the sub-alphabets A_e correspond to the frequency channels used by the transmitters. They realize a partition of the total processed frequency band, which corresponds to the case where transmitters avoid mutual interference by controlling their access to the radio spectrum. The cases where different transmitters share common frequency channels is not addressed here. It corresponds to the case of non disjoint sub-alphabets which is out of the scope of this work. From the recorded pulse train, the aim of our problem is to determine how many radars are present in the environment and what signals are emitted by each radar.

The literature on deinterleaving RADAR signals is abundant. It was first performed in the literature by calculating histograms of received time differences [1], [2] or by estimating the phases of the interleaved pulse trains [3]. This type of method has recently been revisited in [4], who proposed an agglomerative hierarchical clustering method combined with optimal transport distances. Such clustering methods deliver fast outcomes, but might fail in environments with complex radar patterns, potentially causing a performance degradation.

To overcome these limitations, another stream of work is based on inferring mixtures of Markov chains [5]. In these methods, a clustering algorithm is first applied to group the different pulses into different clusters, or letters, which correspond to the alphabet $\mathcal{A}$. Then, in a second step, these letters are partitioned into different sub-alphabets to identify the different emitters that could have generated the observed sequence of letters.

These methods are mainly based on two types of pulse train modeling in the literature: Interleaved Markov Process (IMP) and Mixture of Renewal Processes (MRP).

Modeling with IMP was introduced for the first time in [5]. This model does not directly take into account the arrival times of pulses, but rather considers their order of arrival. The overall interlaced sequence produced by the different

transmitters and received in a single channel is modeled using a switching mechanism, whose role is to alternate the sequence of transmissions by the different emitters within the observed sequence. The inference of an IMP is generally carried out using non-parametric methods, by maximizing a global penalized likelihood score. When sub-alphabets are disjoint, i.e., each symbol can only be transmitted by a single transmitter, a deinterleaving scheme has been proposed in [6]. The authors proved the convergence of this method under mild conditions on switching and component processes, and provided that all possible partitions of letters can be evaluated. However, as the exploration of all possible partitions grows exponentially with the number of different symbols, the evaluation of all candidate partitions may be unfeasible in practice. Therefore, different search heuristics have recently been proposed to explore the partition space: an iterated local search in [6], [7], and a memetic algorithm in [8]. Meanwhile, [9] has proposed a variant of the IMP model introduced in [6] for the case of non-disjoint sub-alphabets inferred with an expectation maximization (EM) algorithm. The asymptotic consistency of this deinterleaving scheme is not proven.

These IMP models do not explicitly take into account the delay between the different signals in the deinterleaving scheme, but only their order of arrival. This feature makes them robust to disturbances that may occur during arrival time measurements, but does not take advantage of all available information. However, this information can be critical to the deinterleaving process because the times between different pulses generally follow regular, identifiable distributions.

Therefore, another category of methods has emerged in the literature based on inference of mixtures of Markov renewal processes. In these models, there is no switching mechanism, and the global generation model consists of a set of independent Markov renewal chains, or semi-Markov processes, which have been studied in detail in the discrete case, notably in [10]. Estimation of mixtures of semi-Markov chains has been carried out in [11] for continuous data or discrete data generated by a negative binomial distribution. The authors used an expectation maximization (EM) algorithm. Another method for deinterleaving a mixture of renewal processes has recently been proposed in [12]. This model is designed for continuous data and relies solely on temporal information. The proposed model assumes that the number of emitters is known in advance and that there are no labels on individual pulses.

In this paper, we aim to achieve the best of both worlds, namely to take into account all available information, such as pulse characteristics and arrival times, by modeling a mixture of renewal processes as in [11]. We also aim to propose a consistent deinterleaving scheme, as it has been done for IMP models [6], enabling the true partition to be recovered with probability tending towards one when sufficient data is provided. We would also like to take advantage of recent advances to explore the symbol partition space proposed by [8] and applied to the deinterlacing of RADAR pulse trains.

This paper is organized as follows. Section II presents the renewal processes mixture model. Section III describes the estimation of the parameters and the deinterleaving method to retrieve the partition of the symbols. Section IV presents the experimental setting and a first experimental analysis of the consistency of the score. Section V reports on empirical results of the proposed method compared to the state of the art. Section VI discusses the contribution and presents some perspectives for future work.

II. MIXTURE OF RENEWAL PROCESSES

The underlying generative model P of a sequence (z, t) is assumed to be a set $G_\Pi(\{P^e\}_{e \in E(\Pi)})$ of $m > 0$ independent Markov processes.

A. Markov renewal chains

Given a sequence z of symbols in the alphabet $\mathcal{A}$, for each process P^e associated to the sub-alphabet A_e, we use $z[A_e]$, or just z^e, to present the sub-string of the sequence z obtained by deleting all symbols not in A_e and $t[A_e]$ (or just t^e) the corresponding sequence of arrival times.

Each process P^e, with $e \in E(\Pi)$, is modeled as a Markov renewal chain (Z^e, T^e) with:
- the chain Z^e with state space A_e, generating a sequence of symbols $(z^e_l)_{l \in \mathbb{Z}}$.
- the chain T^e generating the strictly increasing arrival times t^e_l of all z^e_l.

For each process P^e, we introduce also the chain $X^e = (x^e_l)_{l \in \mathbb{Z}}$, corresponding to the sojourn time in each state of the chain Z^e. For all $l \in \mathbb{Z}$, $x^e_l = t^e_l - t^e_{l-1} > 0$.

In the context of Radar Electronic Support Measurement (RESM), the arrival times of the individual pulses, and consequently the interval between them, are multiples of integers. The arrival times are quantified by the digital receiver, with a time resolution corresponding to the Least Significant Bit (LSB). Thus every time measurement is multiple of this LSB. Therefore, for each letter a in a sub-alphabet A_e, we assume that each sojourn time is drawn in a finite set K^a of strictly positive integers. In the following, for a given emitter $e \in E(\Pi)$, we also note $K_e = \bigcup_{a \in A_e} K^a$ as the union of sojourn time sets of all symbols from A_e.

Each chain (Z^e, T^e) satisfies for all $l \in \mathbb{Z}$ the following markov assumption:

$$P^e((z^e_l, t^e_l)|(z^e_{<l}, t^e_{<l})) = \\ = P^e(z^e_l|z^e_{l-1}) \times P^e(t^e_l - t^e_{l-1}|z^e_{l-1}). \quad (1)$$

where $(z^e_{<l}, t^e_{<l})$ is the history of emission of transmitter e before event l.

Thus, after a given event at t^e_l, the next symbol z^e_{l+1} is independent from the chosen delay x^e_{l+1}. Also the delay x^e_{l+1} only depends on the previously emitted symbol z^e_l.

We make these assumptions in order to reduce the number of parameters to be estimated and therefore speed up the likelihood calculations, although in practice this is not always true, since some agile radar waveforms may consist of a periodically repeated synchronized time-frequency pattern. In some other cases, (1) may hold at least approximately. We assume (and empirically confirm in our context) that the likelihood expression derived from (1) is robust enough to produce good deinterleaving performance in practice.

Moreover, we assume that each process P^e is time homogeneous. Therefore $P^e(z^e_{l+1}|z^e_l)$ and $P^e(t^e_{l+1} - t^e_l|z^e_l)$ are independent from l.

We denote by $\mathbf{P}^e = (p^e_{i,j})$ the transition matrix of Z^e, with for $i,j \in A_e$ $p^e_{i,j} = P^e(z^e_{l+1} = j|z^e_l = i), l \in \mathbb{N}$, and for $l \in \mathbb{N}$, $i \in A_e$, $k \in K^i$, we introduce the quantity $q^e_i(k) = P^e(x^e_{l+1} = k|z^e_l = i)$.

Assumption $(\mathcal{P})$. We assume that $p^e_{i,j} > 0$ for $i,j \in A_e$.

Assumption $(\mathcal{Q})$. For each $i \in A_e, k \in K^i$, we assume that $q^e_i(k) > 0$.

Assumptions $(\mathcal{P})$ and $(\mathcal{Q})$ allow us to establish the uniqueness of generative model representations. The proof of Theorem 1 is given in the appendix of the online supplement.

Theorem 1. *Given a generative model $P = G_\Pi(\{P^e\}_{e \in E(\Pi)})$, corresponding to a partition Π of the alphabet $\mathcal{A}$, under assumptions $(\mathcal{P})$ and $(\mathcal{Q})$, if $P = G_{\Pi'}(\{P^e\}_{e \in E(\Pi')})$ for some partition Π', then $\Pi = \Pi'$ must hold.*

III. DEINTERLEAVING SCHEME

In this section, the estimation of the different components of the proposed discrete renewal process mixture is described and a consistent negative log-likelihood score is derived.

A. Semi-Markov Chain Estimation

For each transmitter e with sub-alphabet A_e, and given a sequence (z,t) observed during a time window $[\![0,T]\!]$, we define the following quantities:

- $N^e_i(T)$, the number of symbols i in the observed sequence $z^e = z[A_e]$ until time T.
- $N^e_{i,j}(T)$, the number of transitions from i to j, observed in the sequence z^e until time T.
- $N^e_i(k,T)$, the number of transitions observed from i to any other letter in z^e until time T, with sojourn time equal to k.
- $N^e(T)$, the total number of symbols in z^e .

For a transmitter e, let $(z^e, t^e) = (z^e_i, t^e_i)_{i=0}^{N^e(T)-1}$ the ordered sequence of events generated by e during the observation window according to (1). Let also (z^e_{-1}, t^e_{-1}), with $t^e_{-1} < 0$, be the last event generated by e directly preceding the observation window. The conditional likelihood of the observed sequence (z^e, t^e) during $[\![0,T]\!]$ given (z^e_{-1}, t^e_{-1}) is defined as:

$$\mathcal{L}^T_-(z^e, t^e) = P^e(z^e_0|z^e_{-1}) \frac{P^e(t^e_0 - t^e_{-1}|z^e_{-1})}{R^e_{-1}(|t^e_{-1}| - 1)}$$

$$\prod_{l=1}^{N^e(T)-1} P^e(z^e_l|z^e_{l-1})P^e(t^e_l - t^e_{l-1}|z^e_{l-1})R^e_{N^e(T)-1}(u^e_T), \quad (2)$$

where $R_i(.)$, the survival function of sojourn time in state i, is defined for $k \in \mathbb{N}$ by $R_i(k) = P(x^e_{i+1} > k|z^e_i)$, and $u^e_T = T - t^e_{N^e(T)-1}$.

However, (z^e_{-1}, t^e_{-1}) is unknown, and its bayesian estimation would imply a complex marginalization, intractable in our deinterleaving setting. Rather, we rely on the following approximation, which ignores start and end of the observation window without biasing the process (see proposition 6 in the appendix of the online supplement):

$$\mathcal{L}^T_\sim(z^e, t^e) = \prod_{l=1}^{N^e(T)-1} P^e(z^e_l|z^e_{l-1})P^e(t^e_l - t^e_{l-1}|z^e_{l-1}) \quad (3)$$

For each transmitter e, we define the empirical estimators of the coefficients $p^e_{i,j}$ of the transition matrix $\mathbf{P}^e$ and the empirical estimators of the coefficients $q^e_i(k)$ of the distributions of the sojourn times, respectively by $\hat{p}^{e,\sim}_{i,j}(T) := \frac{N^e_{i,j}(T)}{N^e_i(T)}$ and $\hat{q}^{e,\sim}_i(k,T) := \frac{N^e_i(k,T)}{N^e_i(T)}$, if $N^e_i(T) \neq 0$. If $N^e_i(T) = 0$, we set $\hat{p}^{e,\sim}_{i,j}(T) := 0$ and $\hat{q}^{e,\sim}_i(T,k) := 0$.

Proposition 1. *The estimators $\hat{p}^{e,\sim}_{i,j}(T)$ and $\hat{q}^{e,\sim}_i(k,T)$ maximize the approached log-likelihood function $\log(\mathcal{L}^T_\sim(z^e, t^e))$.*

When using the estimators $\hat{p}^{e,\sim}_{i,j}(T)$ and $\hat{q}^{e,\sim}_i(k,T)$, the maximum of the approached conditional likelihood of the sequence (z^e, t^e) is denoted $\widehat{\mathcal{L}}^T_\sim(z^e, t^e)$.

Given an observed sequence (z^e, t^e), generated by transmitter e and observed until time T, we denote by $H^T(z^e, t^e) = -\log(\widehat{\mathcal{L}}^T_\sim(z^e, t^e))$, the approached maximum negative log-likelihood for transmitter e.

We have

$$H^T(z^e, t^e) = H^T_Z(z^e, t^e) + H^T_X(z^e, t^e), \quad (4)$$

with $H^T_Z(z^e, t^e)$ the term due to the state transitions for transmitter e,

$$H^T_Z(z^e, t^e) = - \sum_{i,j \in A_e} N^e_{i,j}(T)\log \frac{N^e_{i,j}(T)}{N^e_i(T)}, \quad (5)$$

and $H^T_X(z^e, t^e)$ the term due to the distribution of sojourn time for transmitter e:

$$H^T_X(z^e, t^e) = - \sum_{i \in A_e} \sum_{k \in K^i} N^e_i(k,T) \log \frac{N^e_i(k,T)}{N^e_i(T)}. \quad (6)$$

By convention in (5), given $T > 0$, if for some i and j, we have $N^e_{i,j}(T) = 0$, i.e. there is no transition from i to j in the observed sub-sequence z^e until time T, then we set $N^e_{i,j}(T)\log \frac{N^e_{i,j}(T)}{N^e_i(T)} := 0$. In the same way, in (6), if for $i \in A^e$, and $k \in K^i$, $N^e_i(k,T) = 0$, we set $N^e_i(k,T) \log \frac{N^e_i(k,T)}{N^e_i(T)} := 0$.

B. Global model estimation and deinterleaving scheme

As all transmitters are independent, the *global* approached likelihood of the sequence (z,t) observed until time T, and related to a partition of symbols $\Pi = \cup_{e \in E(\Pi)} A_e$, is given by

$$\mathcal{L}^T_{\sim,\Pi}(z,t) = \prod_{e \in E(\Pi)} \mathcal{L}^T_\sim(z^e, t^e). \quad (7)$$

We use $H^T_\Pi(z,t) = -\log \widehat{\mathcal{L}}^T_{\sim,\Pi}(z,t)$ to denote the corresponding empirical entropy. Thus, we have,

$$H_\Pi^T(z,t) = \sum_{e \in E(\Pi)} H^T(z^e, t^e), \qquad (8)$$

with $H^T(z^e, t^e)$ given by Equation (4).

Given a sequence (z,t) observed during the time window $[\![0,T]\!]$, the proposed deinterleaving scheme corresponds to finding the partition minimizing a penalized entropy score function of the form:

$$C_\Pi^T(z,t) = H_\Pi^T(z,t) + \gamma m \log(n), \qquad (9)$$

with γ a non-negative (penalization) constant, m the number of transmitters and n the size of the observed sequence z until time T. Such penalization in $\log(n)$ [13] aims to take into account the fact that the number of parameters, related to the number of transmitters, varies with the considered partition. Given a candidate partition Π, if simultaneous emission of different symbols from the same alphabet A_e occurs in the observed sequence, the partition Π is considered incompatible with the process that generated the sequence (the same process cannot emits several symbols at the same time). In that case, we arbitrarily set $C_\Pi^T(z,t) := +\infty$ to discard that wrong candidate partition.

Assumption $(\mathcal{K})$. Let $K_e = \bigcup_{z \in A_e} K^z$ be the set of all sojourn times for transmitter e. For all transmitter $e \in E(\Pi)$, associated to the underlying generative process $G_\Pi(\{P^e\}_{e \in E(\Pi)})$, the greatest common divisor (gcd) of the set K_e is equal to 1.

Under this additional assumption, we show with the following theorem that the deinterleaving scheme allows to retrieve the *true* partition when T goes toward infinity, with an exhaustive search in the space of all possible partitions of the alphabet $\mathcal{A}$. This proof is given in the appendix (online supplement). Hypothesis $(\mathcal{K})$ is quite relevant for radar emitters: to resolve ambiguities on pulse repetition interval range measurement, pulse repetition intervals are often taken purposely coprime which in turn translates into coprimed sojourn time.

Theorem 2. *Let* $P_\Pi = G_\Pi(\{P^e\}_{e \in E(\Pi)})$ *and let* Π' *be a partition of* $\mathcal{A}$ *such that* $\Pi' \neq \Pi$. *Then, if* P_Π *verifies assumptions* $(\mathcal{Q})$, $(\mathcal{K})$ *and* $(\mathcal{P})$, *we have*

$$\lim_{T \to \infty} \mathbb{P}_{(z,t) \sim P_\Pi}\left(\frac{1}{T}(C_{\Pi'}^T(z,t) - C_\Pi^T(z,t)) > 0\right) = 1. \quad (10)$$

This is also true for sequences (z,t) *sampled from* $P_{\Pi|\Pi'}$, *with* $P_{\Pi|\Pi'}$ *a process restricting sequences from* P_Π *to the subset of possible sequences for* $P_{\Pi'}$.

The second part of the theorem shows that the result not only holds due to the incompatibility of partitions. Even if, for instance, no simultaneous emission of different symbols occurs in the observed sequence (which can be rather unlikely for some scattered sets of emission delays), our deinterleaving process is able to identify the true partition if the observed sequence is sufficiently long.

This result is valid for all non-negative values of the penalization constant γ in (9), and in particular when $\gamma = 0$, as it is experimentally confirmed in Section IV using synthetic data generated according to the model described in Section

II. However, we experimentally observe in practice, as shown in Section V, that using $\gamma > 0$ can improve the results when applying the deinterleaving scheme in the ESM domain, whose incoming data does not exactly match the proposed model.

C. Solving the Combinatorial Problem in the Space of Partitions

Given an alphabet $\mathcal{A}$ and a sample (z,t) observed until time T, finding the partition Π minimizing the score C_Π^T given by (9) requires to solve a combinatorial problem in the search space of the alphabet partitions given by

$$\Omega_\mathcal{A} = \{\Pi | \mathcal{A} = \bigcup_{e \in E(\Pi)} A_e$$
$$\wedge \forall(e,e') \in E(\Pi)^2, e \neq e' \implies A_e \cap A_{e'} = \emptyset\}.$$

The size of this search space can be calculated exactly with the Bell number:

$$B_{|\mathcal{A}|} = \sum_{k=0}^{|\mathcal{A}|} \frac{1}{k!} \sum_{i=0}^{k} (-1)^i \binom{k}{i} (k-i)^{|\mathcal{A}|}. \qquad (11)$$

For a small number of letters (less than 10), as the size of $\Omega_\mathcal{A}$ remains reasonable (less than $B_{10} = 115975$), we can use an exhaustive search in the space of all partitions. This can be done efficiently using the algorithm proposed in [14].

However, since this space search grows exponentially with the number of letters, an exhaustive search is in general not feasible in a reasonable amount of time.

Therefore, when the number of letters is greater than 10, we propose to use an heuristic to find the best partition Π in the huge search space $\Omega_\mathcal{A}$. We adapted the recent memetic algorithm MAAP, which was previously used for a deinterleaving problem in [8], seen as a particular grouping problem. Like MAAP, our proposed algorithm, called TEDS for *Temporal Estimation Deinterleaving Scheme*, uses a population of two different candidate solutions (partitions). As described in Algorithm 1, it alternates between two phases during the search. In the first phase, a local search procedure (called TabuAP) is performed to improve the partitions in the population. In the second phase, crossovers (GLPX) are performed between partitions to generate new candidate partitions, which are further improved by the TabuAP procedure in the next generation. The algorithm is efficiently implemented with incremental evaluation techniques, taking advantage of the fact that the likelihood score of each sub-alphabet A_i can be calculated independently. The appendix of the online supplement gives more details on the implementation of the TabuAP and GLPX procedures.

IV. FIRST EXPERIMENTAL ANALYSIS

The first experiment studies the ability of the deinterleving scheme to retrieve the *true* partition, when the data are generated according to the *ideal* framework described in Section II, with a collection of independent renewal processes, and when it is possible to perform an exhaustive search in the space $\Omega_\mathcal{A}$ of the partitions. We first describe the data generating process used for these experiments. Then we present an experimental analysis of the consistency of the score.

Algorithm 1 TEDS -Temporal Estimation Deinterleaving Scheme (inspired by MAAP)

1: **Input**: sequence (z, t) observed until time T, with z drawn from alphabet $\mathcal{A}$.
2: **Output**: Best partition Π_{best} of $\mathcal{A}$ found so far.
3: $\Pi_1, \Pi_2, \Pi_{best} \leftarrow$ random draws in partition space $\Omega_{\mathcal{A}}$
4: **while** stop condition is not met **do**
5: $\quad \Pi'_1 \leftarrow \text{TabuAP}(\Pi_1, (z, t))$
6: $\quad \Pi'_2 \leftarrow \text{TabuAP}(\Pi_2, (z, t))$
7: $\quad$ **if** $C^T_{\Pi'_1}(z, t) < C^T_{\Pi_{best}}(z, t)$ **then**
8: $\quad\quad \Pi_{best} \leftarrow \Pi'_1$
9: $\quad$ **end if**
10: $\quad$ **if** $C^T_{\Pi'_2}(z, t) < C^T_{\Pi_{best}}(z, t)$ **then**
11: $\quad\quad \Pi_{best} \leftarrow \Pi'_2$
12: $\quad$ **end if**
13: $\quad \Pi_1 \leftarrow \text{GLPX}(\Pi'_1, \Pi'_2, (z, t))$
14: $\quad \Pi_2 \leftarrow \text{GLPX}(\Pi'_2, \Pi'_1, (z, t))$
15: **end while**
16: return Π_{best}

A. Synthetic dataset generation

The datasets are based on synthetic sequences (z, t) of size n generated by a set of m renewal processes $G_\Pi(\{P^e\}_{e \in E(\Pi)})$. Given a length n and an alphabet $\mathcal{A}$, a *scenario*, i.e a sequence (z, t), is generated as described below. Each parameter is randomly drawn from uniform distributions on the specified interval or space.

- From an alphabet $\mathcal{A}$, a ground truth partition $\Pi_{truth} = \{A_e\}_{e \in E(\Pi_{truth})}$ of the alphabet $\mathcal{A}$ is drawn in the search space $\Omega_{\mathcal{A}}$. $m = |E(\Pi_{truth})|$ is the number of transmitters associated with this partition.
- For each transmitter $e \in E(\Pi_{truth})$, with corresponding sub-alphabet A_e, a probabilistic transition matrix $\mathbf{P}^e = (p^e_{i,j})$ of size $|A_e| \times |A_e|$ is drawn with non-zero coefficients in order to satisfy assumption $(\mathcal{P})$.
- For each symbol a in $\mathcal{A}$, a number of sojourn time states is drawn in $[\![1, K]\!]$, then different sojourn times k_a are drawn in the interval $[\![1, L]\!]$, such that assumption $(\mathcal{K})$ is verified. K and L are two hyper-parameters of the generator, set up to the values $m + 1$ and $|\mathcal{A}| + 1$ respectively (arbitrary choice). Then, for each symbol a in $\mathcal{A}$, a probability $q_a(k_a)$ of each sojourn time k^a, is drawn such as $\sum_{k^a \in K^a} q_a(k_a) = 1$, and $\forall k^a \in K^a, q_a(k_a) > 0$, in order to meet assumption $(\mathcal{Q})$.
- An initial state symbol z^e_0 of each process P^e is drawn in A_e.
- For each transmitter P^e a sequence $(z^e_l, t^e_l)_{l \in \mathbb{N}}$ is generated using the initial state z^e_0, the transition matrix $\mathbf{P}^e$ and the distribution of sojourn times for each symbol $a \in A_e$.
- The m sequences (z^e_l, t^e_l) are merged to build a sequence (z, t) of size n, with increasing times of arrival $0 \leq t_0 \leq t_1 \leq \cdots \leq t_{n-1} \leq T$.

Note that the number of transmitters is drawn between 1 and the number of symbols $|\mathcal{A}|$ (see first point), but it is not equivalent to a uniform random draw in $[\![1, |\mathcal{A}|]\!]$ since it depends on the topology of the partition space. Still, the prob-
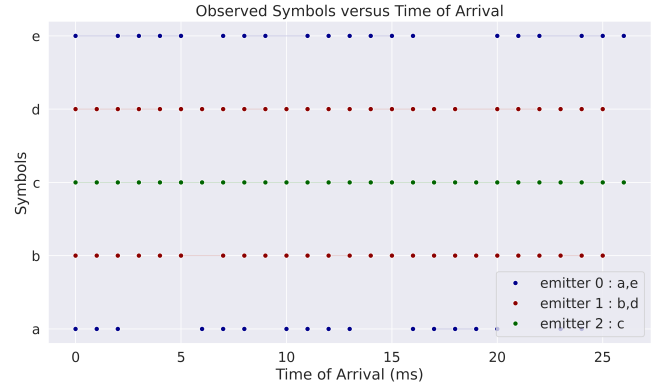

Fig. 1. Display of data generated for a scenario of size 200, with 5 different symbols coming from 3 different emitters. The y-axis represents the symbols, the x-axis their arrival times. The ground truth is $\Pi_{truth} = \{a, e\} \cup \{b, d\} \cup \{c\}$: emitter 0 in blue emits symbols a and e, emitter 1 in red emits symbols b and d, emitter 2 in green emits symbol c.

ability to draw a partition with a high number of transmitters m increases as the number of symbols $|\mathcal{A}|$ increases.

Using this synthetic dataset generator, denoted as $G(\mathcal{A}, n),$[1] we first study different scenarios with different sizes n and a low number of symbols. This allows us to experimentally verify that the ground truth partition Π_{truth} can be retrieved with probability one, when n goes toward infinity and when an exhaustive search in the space of all partitions is performed.

Figure 1 illustrates an example of generated scenario with $|\mathcal{A}| = 5$ and $n = 200$. Here the ground truth is $\Pi_{truth} = \{a, e\} \cup \{b, d\} \cup \{c\}$, there are 3 different transmitters, each distinguished by its color.

B. Empirical verification of the consistency of the score

In order to analyse the robustness of our deinterleaving scheme, we study different configurations with a number of symbols $|\mathcal{A}| \in \{3, 5, 7, 9\}$ and a length n ranging in the interval $[\![0, 5000]\!]$. For each configuration $(|\mathcal{A}|, n)$, 1000 different datasets $(\mathcal{A}, \Pi_{truth}, z, t)$ are independently generated.

For each of these datasets, an exhaustive search of the partition $\hat{\Pi}$ minimizing the score $C^T_{\hat{\Pi}}(z, t)$, given by Equation (9) and computed with $\gamma = 0$, is performed.[2] At the end of this process, if $\hat{\Pi}$ is equal to Π_{truth} (up to a permutation of the different sub-alphabets), we consider it a success, otherwise we consider it a failure.

Figure 2 shows the average success rate of this procedure computed for 1000 independent scenarios for each configuration $(|\mathcal{A}|, n)$.

These results confirm the convergence of the model even without penalization on the score (when $\gamma = 0$), as stated by Theorem 2. Indeed, we observe on this figure that for all scenarios with different numbers of symbols, that the success

---

[1]The generated datasets are available at https://github.com/JeanPinsolle/renewal_processes.

[2]The number of candidate partitions are respectively $5, 52, 877, 21147$ when the numbers of letters are respectively $|\mathcal{A}| = 3, 5, 7, 9$ (cf. Equation (11)). It is therefore possible to enumerate all the partitions in a reasonable amount of time in this case.

rate converges toward 100% when the sequence length n of the observed data (z, t) increases.

We also observe a decrease in convergence speed as alphabet size increases. For alphabet sizes of $3, 5, 7, 9$, the success rate reaches 99% for sequence lengths n of $460, 875, 1920, 2395$, respectively. This can be explained by the greater number of parameters to be estimated as the alphabet size increases.
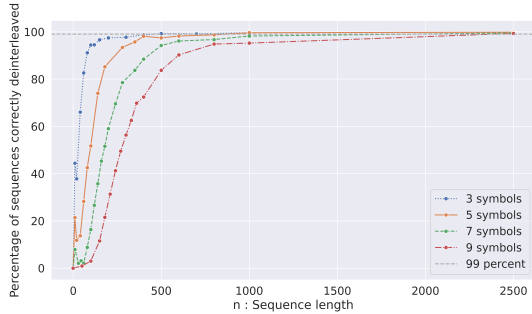

Fig. 2. Average success rate of the proposed deinterleaving scheme when an exhaustive search in the partition space is performed and displayed for different numbers of symbols and different sequence sizes n. The horizontal line represents the 99% threshold of correctly deinterlaced sequences.

V. EXPERIMENTAL VALIDATION

The goal of this section is to evaluate the performance of the proposed algorithm TEDS in comparison with state-of-the-art algorithms of the literature. The TEDS algorithm corresponds to the evaluation of the score presented in Section III-B coupled with the memetic algorithm mentioned in Section III-C for the exploration of the partition space. We first describe the baseline algorithms, followed by their hyper-parameter settings used in the experiments. We then report the experimental results obtained on the synthetic datasets with 5, 10, 20 and 50 symbols coming from the generator presented in the last section, followed by the results on datasets coming from an ESM data generator, which simulates realistic situations with mobile radar warning receivers.

A. State-of-the-art algorithms

For comparison with the state of the art, the following algorithms have been re-implemented in Python 3 from the authors' original papers:

- The improved SDIF algorithm from [15]. SDIF is an algorithm first proposed in [2], which is a well-known reference in the RADAR community. It aims to identify different letter sub-sequences based on the calculation of histograms of observed sojourn times.
- OT: the deinterlacing algorithm proposed in [4]. In this algorithm, a matrix of optimal transport distance is first computed between all sub-sequences of symbols.[3] Then, according to this distance matrix, a hierarchical clustering method is applied to group the different sequences.

[3]We used the POT Python library for this implementation https://pythonot.github.io/.

- IMP: the deinterleaving scheme proposed in [6] with the exploration of the partition space developed by [7]. In this approach, the underlying generative model of the sequence is assumed to be an *interleaved Markov process* $P = \mathcal{I}_\Pi(P_1, \ldots, P_m; P_w)$, where $m > 0$ is the unknown number of different transmitters, P_i is an independent component random process for transmitter i, generating symbols in the sub-alphabet $A_i \subset \mathcal{A}$, P_w is a random switch process over transmitters. The vector $\mathbf{k} = (k_1, \ldots, k_m; k_w)$ corresponds to the different orders of the components and switch Markov processes. The authors proposed a deinterleaving scheme, given by minimizing a global cost function:

$$(\hat{\Pi}, \hat{\mathbf{k}}) = \underset{(\Pi, \mathbf{k})}{argmin} \; C^I_{(\Pi, \mathbf{k})}(z), \tag{12}$$

with $C^I_{(\Pi, \mathbf{k})}(z)$ a global penalized maximum likelihood (ML) entropy score of the sequence z of size n (sorted by increasing times of arrival t) under the IMP model defined as

$$C^I_{(\Pi, \mathbf{k})}(z) = \sum_{i=1}^{m} \hat{H}_{k_i}(z[A_i]) + \hat{H}_{k_w}(\sigma_\Pi(z)) + \beta \kappa \log n, \tag{13}$$

with $\hat{H}_{k_i}(z[A_i])$ the ML entropy of each process P_i, $\hat{H}_{k_w}(\sigma_\Pi(z))$ the ML entropy of the switch process, β a constant and κ the number of free parameters in the model.

B. Parameter settings

In TEDS, the penalization parameter γ in Equation (9) is set to 0 for the first set of experiments on synthetic data (Section V-C), while it is calibrated on a train set to the value of 19 for the ESM dataset considered in Section V-D. For the tabu search procedure, the tabu tenure parameter α is set to the value of 0.6 (as in [8]). The maximal number of iterations for each tabu search is set to 50. Table I summarizes the parameter setting for our algorithm and the different competitors. For each algorithm and each scenario, a maximum computation time of one hour is retained.

TABLE I
PARAMETER SETTINGS

| Parameter	Description	Value
TEDS		
nb_{iter}	Nb iterations of local search	50
α	Tabu tenure parameter	0.6
γ	ML Penalization parameter	0 (sec V-C), 19 (sec V-D)
IMP [7]		
nb_{iter}	Nb iterations local search	50
r	Radius of random jump	2
β	ML Penalization parameter	0.5 (sec V-C), 0.1 (sec V-D)
OT [4]		
s	Threshold silhouette score	-0.1 (sec V-C), 0.2 (sec V-D)
SDIF		
ϵ	Precision for histograms	0.05
X	Threshold peaks histograms	0.9
c	Threshold PRI transform	0.5
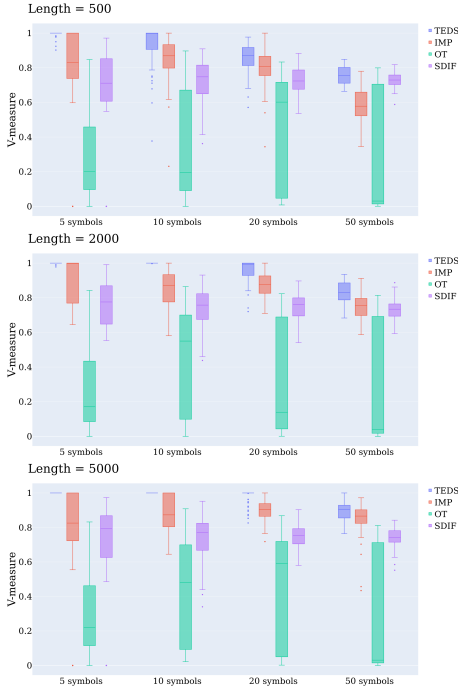
Fig. 3. Comparison of the deinterleaving algorithms on synthetic data of size $n = 500$, 2000 and 5000. Tests were made for 5, 10, 20 and 50 symbols. Each box represents the distribution of the V-measure scores obtained by the compared algorithms and computed for 100 independent scenarios.

C. Experiments on synthetic datasets

We first consider the synthetic datasets presented in Section IV-A with alphabets $\mathcal{A}$ of size 5, 10, 20 and 50 and observed sequences of size $n = 500, 2000, 5000$. Figure 3 displays with different boxplots the distribution of the V-measure scores obtained for 100 independent scenarios by each compared method (SDIF, OT, IMP, and TEDS) for each configuration $(|\mathcal{A}|, n)$.

The V-measure score is a classical metric used to evaluate the quality of a partitioning result. It corresponds to the harmonic mean of the measures of homogeneity and completeness of the partition of symbols in comparison with the ground truth [16]. The higher the scores, the better the results.

First we observe that TEDS (in blue) always returns the ground truth partition for the scenarios with 5 and 10 symbols and sequences with 5000 data points (V-measure scores always equal to 1). For other configurations, the number of data points is not sufficient to guarantee a consistent estimation of the score associated to each partition. Unsurprisingly, the worst results are obtained for scenarios with the highest number of symbols, $|\mathcal{A}| = 50$, and the lowest numbers of data points, $n = 500$.

Overall, we observe that TEDS (in blue) dominates all the competitors for all configurations. The second best algorithm is IMP (in red), which also uses a deinterleaving scheme based on a Markov representation of the data. Algorithms that are not based on a Markov model return partitions far from the ground truth, as evidenced by the low V-measure scores obtained by SDIF and OT for the different scenarios.

Regarding the computation time required to find the best

partition, TEDS and IMP require the highest computing time, because they need to compute thousands of entropy scores during the exploration of the partition space.

D. Electronic Warfare experiments

In this section we present an application on datasets coming from an Electronic Warfare data generator which simulates realistic situations with an airborne ESM (Electronic Support Measure) receiver in an environment composed of many radars whose number and positions can be selected in the simulation. One configuration corresponds to a random draw in a realistic radar library and a random draw in their relative phasing. For each configuration, we generate a dataset $\mathcal{D}$ consisting in a sequence of intercepted pulse parameters, including their corresponding frequency (CF) and time of arrival (ToA). The *ground truth* partition Π_{truth} (i.e., the association of each pulse to each transmitter) is given by the simulator but is assumed unknown. The objective is then to retrieve Π_{truth} from the data.

1) Data description: Data are composed of 100 scenarios generated with the ESM simulator. One scenario, illustrated in Figure 4, corresponds to a manually selected window lasting few seconds in a simulation of several minutes. The windows $[\![0, T]\!]$ during which the pulses are observed for each scenario, were selected regarding two criteria: two different transmitters cannot transmit in the same frequency (because otherwise we are in the case of non-disjoint alphabets) and all transmitters cannot cease to transmit in the time window in order to meet as much as possible the hypothesis $(\mathcal{P})$ and $(\mathcal{Q})$ made for the data generating process (see Section II). Scenarios were generated with or without missing pulses and with or without noises, the missing rate can reach 60% and missing pulses are randomly drawn according to a uniform law. The length of the sequences observed in the different scenarios ranges from 90 to 8600, and the number of transmitters varies from 1 to 11.

2) Preprocessing step: Contrary to synthetic data, the number of symbols is not fixed and is deduced from the observed data with the method explained below.

A preprocessing of the data is first performed to obtain the alphabet $\mathcal{A}$ from the dataset, as it is done in [7]. It consists in clustering pulses with the DBSCAN algorithm [17] based on their frequency. Then, each cluster obtained is associated with a symbol in $\mathcal{A}$. We then obtain a sequence of frequency symbols z of size n with their corresponding arrival times t.

The ϵ-neighborhood parameter of DBSCAN corresponds to our precision parameter and is a fixed number of the order of the frequency measured in MHz. After this pulse clustering into the different groups (symbols), we obtain a number of symbols ranging from 7 to 53 in the different scenarios.

Figure 4 illustrates an example of data generated with the ESM simulator. The first plot represents the frequency of the observed pulses (y-axis) according to their time of arrival (x-axis) as they are fed into the algorithm. The second plot shows the result of the pre-processing step, where the x-axis represents the arrival times, but the y-axis shows the symbols made with frequency values (frequency values are not considered anymore: only symbol grouping remains). Ground

Fig. 5. Comparison of deinterleaving algorithm on simulated PDWs. One box represents the distribution of the V-measure scores of an algorithm on the 50 test scenarios.
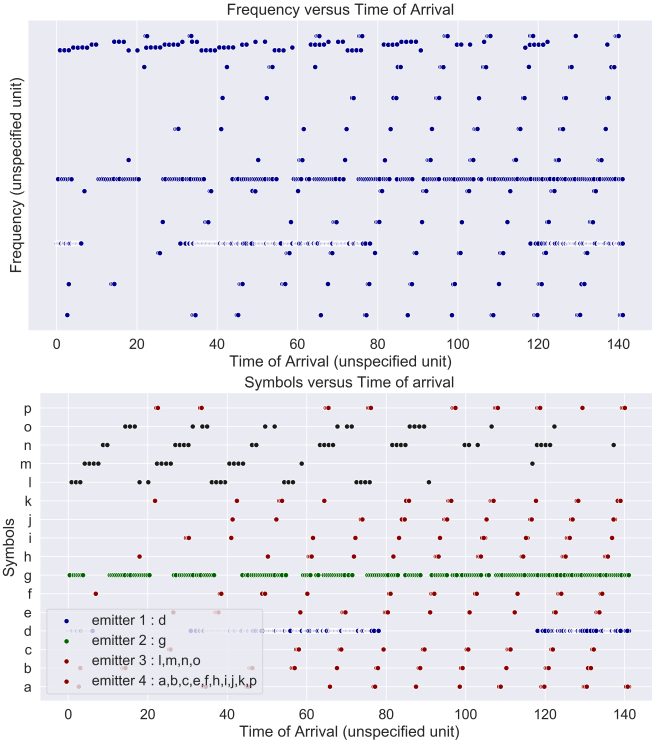
Fig. 4. For both plots the x-axis is the time of arrival of observed data. The y-axis in the first plot shows the frequency values as they are observed. The second plot shows the result of the preprocessing step where the y-axis represents the symbols when clustering pulses based on their frequency. Ground truth is given: each color represents one emitter. There are four emitters. Here we can see emitter 1 (in blue) emits only symbol d, while emitter 4 (red) emits 10 symbols $a, b, c, e, f, h, i, j, k, p$.

truth is represented, with four emitters and each one associated with one color.

3) Hyperparameter calibration: We first performed a calibration of the main critical parameters of the different algorithms in order to maximize the average V-measure score on a train set composed of the first 50 instances. With this process, the penalization parameter γ of TEDS in the score computation (cf. Equation (9)) is set to 19. The penalization parameter β of IMP is set to 0.1. Three parameters were optimized for SDIF, $\epsilon = 0.05$, threshold $x = 0.9$, and a second threshold needed for the PRI (Pulse Repetition Interval) transform, $\alpha = 0.5$. For the OT optimal transport algorithm, the hierarchical cluster cut is defined as a function of a silhouette score with a threshold set at $s = 0.2$. These parameters are gathered in Table I.

4) Validation of the performances: With the calibrated parameters, we ran the compared algorithms on the 50 remaining validation instances. Figure 5 displays the distribution of the V-measure scores obtained by the four algorithms on the 50 test scenarios.

TEDS and IMP obtain the best results, with a slightly better median obtained by TEDS compared to IMP, even if a statistical test indicates that the difference in scores between TEDS and IMP is not significant. TEDS returned 17 perfectly deinterleaved scenarios over 50 against 16 for IMP. These two algorithms do not retrieve the ground truth for the same
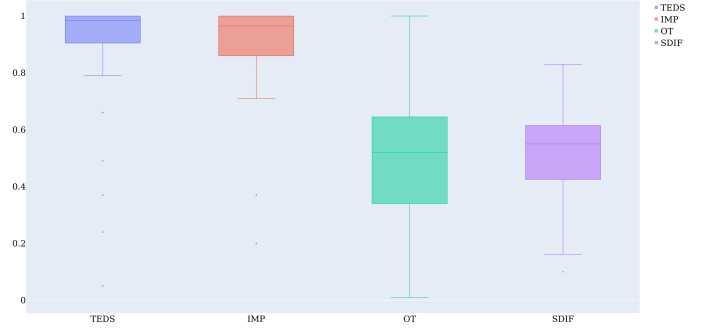
scenarios, which highlights to some extend the complementary nature of these two approaches for this type of data. Note that OT and SDIF make less restrictive hypothesis than we made for the data selection i.e., $(\mathcal{P})$, $(\mathcal{Q})$ and no stop of emission. Therefore, their results do not fully reflect their true efficiency.

VI. DISCUSSION AND PERSPECTIVES

The main contribution of this paper is to propose a new deinterleaving method for interleaved pulse trains, which exploits both negative log-likelihood minimization of the symbol sequence and negative log-likelihood minimization of their sojourn times distribution. The choices made in constructing the model (mixture of independent renewal processes, entropy evaluation and penalty term) are supported by theoretical analysis. We have shown that minimizing the proposed score allows us to recover the true partition associated with the generation of the observed sequence in the large sample limit. This theoretical result is confirmed by an experimental analysis of the score consistency. A comparison with other algorithms on synthetic data and electronic warfare data from a realistic simulator confirmed that TEDS is competitive with state-of-the-art algorithms for the deinterleaving task and scales favorably, in terms of computing time required, to the number of transmitters in the generative process.

This work opens up various perspectives for future research. First of all, we have observed that some of the assumptions made in this paper can be violated in reality. In particular, we assumed in this work that the sub-alphabets are disjoint, which is not always the case in realistic scenarios, because different transmitters may send pulse with the same frequency. Secondly, we have assumed that all radars in the environment constantly emit, but it may happen that a radar emits no symbol for an extended period of time. The model could take this into account by introducing, for example, a temporary "off" state for each radar, during which it emits no pulses.

REFERENCES

[1] C. L. Davies and P. Hollands, "Automatic processing for ESM," 1982.

[2] H. Mardia, "New techniques for the deinterleaving of repetitive sequences," in *IEE Proceedings F (Radar and Signal Processing)*, IET, vol. 136, 1989, pp. 149–154.

[3] T. L. Conroy and J. B. Moore, "On the estimation of interleaved pulse train phases," *IEEE Transactions on Signal Processing*, vol. 48, no. 12, pp. 3420–3425, 2000.

[4] M. Mottier, G. Chardon, and F. Pascal, "Deinterleaving and Clustering unknown RADAR pulses," en, in *2021 IEEE Radar Conference (RadarConf21)*, Atlanta, GA, USA: IEEE, May 2021, pp. 1–6.

[5] T. Batu, S. Guha, and S. Kannan, "Inferring mixtures of markov chains," in *Learning Theory*, J. Shawe-Taylor and Y. Singer, Eds., red. by D. Hutchison et al., vol. 3120, Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 186–199.

[6] G. Seroussi, W. Szpankowski, and M. J. Weinberger, "Deinterleaving finite memory processes via penalized maximum likelihood," *IEEE Transactions on Information Theory*, vol. 58, no. 12, pp. 7094–7109, 2012.

[7] G. Ford, B. J. Foster, and S. A. Braun, "Deinterleaving pulse trains via interleaved markov process estimation," in *2020 IEEE Radar Conference (RadarConf20)*, IEEE, 2020, pp. 1–6.

[8] J. Pinsolle, O. Goudet, C. Enderli, and J.-K. Hao, "A memetic algorithm for deinterleaving pulse trains," in *Evolutionary Computation in Combinatorial Optimization, Lecture Notes in Computer Science 13987*, L. Pérez Cáceres and T. Stützle, Eds., Cham: Springer Nature Switzerland, 2023, pp. 66–81.

[9] A. Minot and Y. M. Lu, "Separation of interleaved markov chains," in *2014 48th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, USA: IEEE, Nov. 2014, pp. 1757–1761.

[10] V. S. Barbu and N. Limnios, *Semi-Markov chains and hidden semi-Markov models toward applications: their use in reliability and DNA analysis*. Springer Science & Business Media, 2009, vol. 191.

[11] H. Cardot, G. Lecuelle, P. Schlich, and M. Visalli, "Estimating finite mixtures of semi-markov chains: An application to the segmentation of temporal sensory data," *Journal of the Royal Statistical Society Series C: Applied Statistics*, vol. 68, no. 5, pp. 1281–1303, 2019.

[12] J. Young, A. Høst-Madsen, and E.-M. Nosal, "Deinterleaving of mixtures of renewal processes," *IEEE Transactions on Signal Processing*, vol. 67, no. 4, pp. 885–898, 2019.

[13] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978, ISSN: 00905364. Accessed: Jun. 13, 2024. [Online]. Available: http://www.jstor.org/stable/2958889.

[14] D. E. Knuth et al., *The art of computer programming*. Addison-Wesley Reading, MA, 1973, vol. 3.

[15] Y. Liu and Q. Zhang, "Improved method for deinterleaving radar signals and estimating pri values.," *IET Radar, Sonar and Navigation*, vol. 12, no. 5, pp. 506–514, 2018.

[16] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2007, pp. 410–420.

[17] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, vol. 96, 1996, pp. 226–231.

Online Supplement
Deinterleaving of Discrete Renewal Process Mixtures with Application to Electronic Support Measures

Jean Pinsolle*[†], Olivier Goudet[†], Cyrille Enderli*, Sylvain Lamprier[†] and Jin-Kao Hao[†]

* Thales DMS France SAS, 2 Av. Jean d'Alembert, 78190 Trappes, France

{jean.pinsolle,cyrille-jean.enderli}@fr.thalesgroup.com

[†] LERIA, Université d'Angers, 2 Boulevard Lavoisier, Angers 49045, France

{olivier.goudet,sylvain.lamprier,jin-kao.hao}@univ-angers.fr

APPENDIX

This online supplement presents the proofs of the theorems and propositions (parts A–C) as well as the local search procedure (TabuAP) and the crossover operator (GLPX) used Algorithm 1 (part D).

A. Proof of Theorem 1

Proof. Let Π' be a partition such that $P = G_{\Pi'}(\{P^e\}_{e \in E(\Pi')})$. Let us assume that $\Pi \neq \Pi'$ and let us show that we arrive at a contradiction.

First case. We claim that there exist a sub-alphabet $A_{e_{\Pi'}} \in \Pi'$, such that $|A_{e_{\Pi'}}| \geq 2$ and $A_{e_{\Pi'}} \notin \Pi$. Here $e_{\Pi'} \in E(\Pi')$ denotes an emitter generating symbols in the sub-alphabet $A_{e_{\Pi'}}$.

$A_{e_{\Pi'}}$ contains at least two symbols a and b, such that a is in a sub-alphabet $A_{e_{\Pi}} \in \Pi$ and $b \notin A_{e_{\Pi}}$. Thus, there exists $A_{f_{\Pi}} \in \Pi$, with $f_{\Pi} \in E(\Pi)$ and $f_{\Pi} \neq e_{\Pi}$, such that $b \in A_{f_{\Pi}}$ with $A_{f_{\Pi}} \neq A_{e_{\Pi}}$.

We denote by k_{min} the least possible delay between two symbols a that P can generate (i.e., $k_{min} = \min_{k \in K^a}(k)$). Since $P = G_{\Pi'}(\{P^e\}_{e \in E(\Pi')})$, b cannot be emitted after a in a time delay lower than k_{min}. Since $P = G_{\Pi}(\{P^e\}_{e \in E(\Pi)})$, P can generate a sequence $z[A_{e_{\Pi}} \cup A_{f_{\Pi}}] = z_{t_a}..z_{t_b}..z_{t_{a'}}$, with $z_{t_a} = a$, $z_{t_b} = b$, $z_{t_{a'}} = a$, where $t_{a'} - t_a = k_{min}$ and $t_a < t_b \leq t_{a'}$, under assumptions $(\mathcal{P})$ and $(\mathcal{Q})$. This is always true since $P^{e_{\Pi}}$ can generate any sub-sequence corresponding to an arbitrary number λ of repetitions of symbol a, with time delay k_{min} separating each occurrence. For a sufficiently high λ, f_{Π} can emit symbol b during the same interval, whatever $K_{e_{\Pi}}$ and $K_{f_{\Pi}}$.

Thus, either $t_b = t_{a'}$, which is impossible for Π' since arrival times of symbols from a single transmitter must be strictly increasing (by definition of a renewal process as given in Section II-A), or it contradicts the previous assertion that states that $t_b - t_a \geq k_{min}$.

Second case. We claim that there exists a sub-alphabet $A_{e_{\Pi}} \in \Pi$, such that $|A_{e_{\Pi}}| \geq 2$ and $A_{e_{\Pi}} \notin \Pi'$. $A_{e_{\Pi}}$ contains at least two symbols a and b, such that a is in a sub-alphabet

$A_{e_{\Pi'}} \in \Pi'$ and $b \notin A_{e_{\Pi'}}$. Thus, there exists $A_{f_{\Pi'}} \in \Pi'$, with $f_{\Pi'} \in E(\Pi')$ and $e_{\Pi'} \neq f_{\Pi'}$, such that, $b \in A_{f_{\Pi'}}$ with $A_{f_{\Pi'}} \neq A_{e_{\Pi'}}$. Using the same reasoning as in the first case, we arrive at a contradiction, which concludes the proof.
□

B. Proof of Proposition 1

Proof. This proof is very similar to the proof of Proposition 4.1 in [1]. First we have

$$\forall \, i \, \in A_e \sum_{k \in K^i} q_i^{e,\sim}(k) = 1 \tag{14}$$

Let $(\lambda_i)_{i \in z[A_e]}$ be real coefficients. According to (3) and (14), the approached log-likelihood can be written in the form

$$\log(\mathcal{L}_{\sim}^T(z^e, t^e)) = \sum_{i,j \in A_e} N_{i,j}^e(T) \log p_{i,j}^{e,\sim}$$
$$+ \sum_{i \in A_e} \left(\sum_{k \in K^i} N_i^e(k,T) \log q_i^{e,\sim}(k) + \lambda_i(1 - \sum_{l \in K^i} q_i^{e,\sim}(l)) \right) \tag{15}$$

When deriving (15) with respect to $q_i^{e,\sim}(k)$, we observe that a maximum is obtained in $\hat{q}_i^{e,\sim}(k) = \frac{N_i^e(k,T)}{\lambda_i}$.

By using (14), we obtain

$$1 = \sum_{k \in K^i} q_i^{e,\sim}(k) = \sum_{k \in K_i} \frac{N_i^e(k,T)}{\lambda_i} = \frac{N_i^e(T)}{\lambda_i} \tag{16}$$

Thus, the value of $q_i^{e,\sim}(k)$ maximizing (15) is $\hat{q}_i^{e,\sim}(k) = \frac{N_i^e(k,T)}{N_i^e(T)}$.

With the same method, we derive that the value of $p_{i,j}^{e,\sim}$ maximizing (15) is $\hat{p}_{i,j}^{e,\sim}(T) = \frac{N_{i,j}^e(T)}{N_i^e(T)}$.
□

C. Proof of Theorem 2

In order to prove Theorem 2, we first show that the generative model G_Π defined as a set of Markov renewal chains can be represented as an ergodic finite-state-machine (FSM) source.

Using the notation proposed in [2], an FSM over an alphabet A is defined by the triplet $\mathcal{F} = (\mathcal{S}, s_0, f)$ where $\mathcal{S}$ is a finite set of states, $s_0 \in \mathcal{S}$ is a fixed initial state and $f : \mathcal{S} * A \to \mathcal{S}$ is the *next-state function*. The definition of an FSM source is completed from an FSM with the addition of a conditional probability distribution $P_{\mathcal{F}}(.|s)$ with each state s of $\mathcal{S}$, and a probability distribution $P_{\mathcal{F}}^{init}(.)$ on the initial state s_0. The FSM source generates a sequence of states s by choosing an initial state s_0 according to $P_F^{init}(.)$, then by choosing an action a_t according to $P_F(.|s_{t-1})$, and transitions to the state $s_t = f(s_{t-1}, a_t)$.

1) Markov renewal chain as an FSM: We first observe that a Markov renewal chain (Z^e, T^e) can be represented as an FSM source. For a transmitter e generating a sequence of symbols in A_e, we define the state space $\mathcal{S}^e \in A_e \times D^e$, with D^e the set of integers corresponding to the time since a letter of A_e has been transmitted. $D^e = \{0, 1, \ldots, \max_{a \in A_e, k \in K^a} k - 1\}$ is a finite set of integers.

A state $s \in \mathcal{S}^e$ corresponds to a pair $s = (i, d)$, with $i \in A_e$, the last symbol transmitted by the emitter e since a duration of $d \in \mathbb{N}$ units of time.

We introduce also the set $A_e^* = A_e \cup *$, with $*$ a new symbol, indicating that the previous symbol $i \in A_e$ lasts one unit of time longer but that there is no transition toward a new letter.

For a Markov renewal chain (Z^e, T^e), we consider the FSM $\mathcal{F}^e = (\mathcal{S}^e, s_0^e, f^e)$, with finite state space $\mathcal{S}^e \in A_e \times D^e$, initial state s_0 and next-state function f^e.

For the definition of the next-state function f^e, given a state $s = (i, d)$ and $a \in A_e^*$, we have

$$f^e(s, a) = \begin{cases} (i, d+1) & \text{if } a = * \\ (a, 0) & \text{if } a \in A_e \end{cases}$$

Note that, following a process different from P^e on states S^e (which we consider in the following for the proof of Theorem 2), it is possible to get actions at state s that should be unavailable from that state. This is the case for instances when $a = *$ while $d \geq k_{max}$, with $k_{max} > \max_{b \in A_e, k \in K^b} k$ an arbitrarily delay limit set for the transmitters (allowing to consider finite state sets in the following). In that cases, we consider that $f^e(s, a) = s$ to stay inside S^e.

To complete the definition of an FSM source, for each state $s = (i, d)$ (with $i \in A_e$ and $d \in \mathbb{N}$) and symbol $a \in A_e^*$ we define the conditional distribution as

$$P_{\mathcal{F}^e}(a|(i,d)) = \begin{cases} 1 - h_i(d+1) & \text{if } a = * \\ p_{i,j}^e h_i(d+1) & \text{if } a = j \in A_e, \end{cases}$$

with $h_i(t) = \frac{q_i^e(t)}{\sum_{k \in K^i, k \geq t} q_i^e(k)}$, the instantaneous risk of occurrence of a transition at time t for the letter i.

By abuse of notation and for simplicity, we denote $P_{\mathcal{F}^e}(s_{t+1}|s_t) = P_{\mathcal{F}^e}(a_{t+1}|s_t)$ the probability of choosing an action a_{t+1} given s_t, which leads to the deterministic transition to the state $s_{t+1} = f(s_t, a_t)$.

Proposition 2. *Under assumptions $(\mathcal{Q})$, $(\mathcal{K})$ and $(\mathcal{P})$, each FSM $\mathcal{F}^e = (\mathcal{S}^e, s_0^e, f^e)$ for $e = 1, \ldots, E$, is ergodic.*

Proof. Let $K_e = \cup_{z \in A_e} K^z$, the set of all sojourn times for transmitter e. Let $K^z(d)$ be the set of sojourn times for letter z greater or equal to d: $K^z(d) = \{k \in K^z, k \geq d\}$.

First, we observe that all the states $s \in S^e$ of the FSM $\mathcal{F}^e = (\mathcal{S}^e, s_0^e, f^e)$ built from the Markov renewal chain (Z^e, T^e) are recurrent. Indeed, according to assumptions $(\mathcal{Q})$ and $(\mathcal{P})$, for $i, j \in A^e$, $l \in \mathbb{N}$ we have $P^e(z_{l+1}^e = j | z_l^e = i) > 0$, and for $i \in A^e$, $k \in K^i$, $P^e(x_{l+1} = k | z_l^e = i) > 0$. Therefore, for each state $s = (i, d) \in \mathcal{S}^e$, there is always a positive probability to return to this same state (i, d) in a finite number of time steps.

Given a state (i, d), the number of time steps allowing to return to this same state (i, d) with positive probability, is $\Delta T = k_d^i + \sum_{j=1, \ldots, |K_e|} \lambda_j k_j^e$, with $k_d^i \in K^i(d)$, $k_j^e \in K_e$ and the λ_j's are non-negative integer coefficients. Given a state $(i, d) \in \mathcal{S}^e$, we denote $T^e(i, d)$ this set of transition times ΔT.

Given a state (i, d), let us show that the greatest common divisor (gcd) of the set $T^e(i, d)$ is equal to 1.

Let us assume that $gcd(T^e(i, d)) = r$, with $r > 1$ and let us show that we arrive at a contradiction. As $gcd(T^e(i, d)) = r$,

$$\forall (\lambda_1, \ldots, \lambda_{K_e}) \in \mathbb{N}^{|K_e|}, \exists \mu \in \mathbb{N}, k_d^i + \sum_{j=1,\ldots,|K_e|} \lambda_j k_j^e = \mu r.$$
(17)

In particular, when all λ_j's are equal to 0, for $k_d^i \in K^i(d)$, there exists $\mu_0 \in \mathbb{N}$, such that $k_d^i = \mu_0 r$. Thus r is a divisor of $k_d^i \in K_e$. If K_e contains only k_d^i, then according to assumption $(\mathcal{K})$, $k_d^i = 1$, and we directly have $r = 1$ which is a contradiction. Otherwise $|K_e| > 1$. Thus, given any other sojourn time $k_j^e \in K_e$, such that $k_j^e \neq k_d^i$, using Equation (17), we know that there exists $\mu_j > \mu_0$, such that we have also $k_d^i + k_j^e = \mu_j r$, and thus we have $k_j^e = (\mu_j - \mu_0)r$. Then r is a divisor of k_j^e. Thus, $r > 1$ is a divisor of all sojourn times $k_j^e \in K_e$, which is impossible because $gcd(K_e) = 1$, according to assumption $(\mathcal{K})$.

Therefore, given a state $(i, d) \in \mathcal{S}^e$, we have shown that $gcd(T^e(i, d)) = 1$, thus each state $s \in \mathcal{S}^e$ is aperiodic.

Moreover each pair of states, $s_i, s_j \in \mathcal{S}^e$, communicate with each other, i.e., when starting from state s_i there is always a positive probability to arrive in state s_j in a finite number of time steps. Therefore, $\mathcal{F}^e$ is ergodic. $\square$

We now show the likelihood equivalence of both representations (Markov renewal chain and FSM source) for a single transmitter. Let $(z^e, t^e) = (z_l^e, t_l^e)_{l=0,\ldots,n^e-1}$ be an observed sequence of n^e symbols $z_l^e \in A_e$ with their corresponding time of arrival t_l^e, generated by P^e until time T, given (z_{-1}, t_{-1}) the last event before the observation. Let $s^e = (s_t^e)_{t=0,\ldots,T}$ be the observed sequence of $T + 1$ pairs $s_t^e \in \mathcal{S}^e$, such that for all $l = -1, 0, \ldots, n^e - 1, \forall t \in [\![t_l^e, t_{l+1}^e [\![, s_t^e = (z_l^e, t - t_l^e)$. If (z_{-1}^e, t_{-1}^e) is the last state before the observation, the corresponding last unobserved FSM state is $s_{-1}^e = (z_{-1}^e, |t_{-1}^e| - 1)$.

Proposition 3. *Given state s_{-1}^e at time $t = -1$, the conditional likelihood $L_{-,\mathcal{F}^e}^T(s^e)$ for the sequence $s^e = (s_i^e)_{i=0}^T$, generated by the FSM $\mathcal{F}^e$, is equal to the conditional likelihood $\mathcal{L}_-^T(z^e, t^e)$ of the corresponding sequence $(z_i^e, t_i^e)_{i=0}^{N^e(T)-1}$ given (z_{-1}^e, t_{-1}^e).*

Proof.

$$L_{-,\mathcal{F}^e}^T(s^e) = \prod_{t=-1}^{T-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e) \tag{18}$$

$$L_{-,\mathcal{F}^e}^T(s^e) = \prod_{t=-1}^{t_0^e-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e) \prod_{l=0}^{n^e-2} \prod_{t=t_l^e}^{t_{l+1}^e-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e)$$
$$\prod_{t=t_{n^e}^e-1}^{T-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e), \tag{19}$$

We compute the first term $\prod_{t=-1}^{t_0^e-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e)$. We have, if $t_0 > 0$

$$\prod_{t=-1}^{t_0^e-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e) \tag{20}$$
$$= \prod_{t=-1}^{t_0^e-2} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e) P_{\mathcal{F}^e}(s_{t_0}^e | s_{t_0-1}^e)$$
$$= \prod_{t=-1}^{t_0^e-2} (1 - h_{z_{-1}^e}(|t_{-1}^e| + t + 1)) P^e(z_0^e | z_{-1}^e) h_{z_{-1}^e}(t_0^e + |t_{-1}^e|)$$
$$= \frac{\sum_{k \geqslant |t_{-1}| + t_0} q_{z_{-1}^e}(k)}{\sum_{k \geqslant |t_{-1}|} q_{z_{-1}^e}(k)} P^e(z_0^e | z_{-1}^e) \frac{q_{z_{-1}^e}(t_0^e + |t_{-1}^e|)}{\sum_{k \geqslant |t_{-1}^e| + t_0} q_{z_{-1}^e}(k)}$$
$$= P^e(z_0^e | z_{-1}^e) \frac{P(t_0^e + |t_{-1}^e| \, | z_{-1}^e)}{R_{-1}^e(|t_{-1}^e| - 1)}$$

If $t_0 = 0$, (20) is immediate.

Now let us compute separately the terms $\prod_{t=t_l^e}^{t_{l+1}^e-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e)$ and $\prod_{t=t_{n^e}^e-1}^{T-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e)$.

For $l = 0, \dots, n^e - 2$, given states $s_{t_l^e}^e = (z_l^e = i, d = 0)$ and $s_{t_{l+1}^e}^e = (z_{l+1}^e = j, d = 0)$,

$$\prod_{t=t_l^e}^{t_{l+1}^e-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e) = [\prod_{t=0}^{t_{l+1}^e-t_l^e-2} 1 - h_i(t+1)] p_{i,j}^e h_i(t_{l+1}^e - t_l^e) \tag{21}$$
$$= R_i(t_{l+1}^e - t_l^e - 1) p_{i,j}^e h_i(t_{l+1}^e - t_l^e)$$

Thus,

$$\prod_{t=t_l^e}^{t_{l+1}^e-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e) \tag{22}$$
$$= p_{i,j}^e \sum_{k \in K^i, k > t_{l+1}^e - t_l^e - 1} q_i^e(k) \times \frac{q_i^e(t_{l+1}^e - t_l^e)}{\sum_{k \in K^i, k \geq t_{l+1}^e - t_l^e} q_i^e(k)}$$
$$= p_{i,j}^e q_i^e(t_{l+1}^e - t_l^e)$$
$$= P^e(z_{l+1}^e | z_l^e) \times P^e(t_{l+1}^e - t_l^e | z_l^e)$$

Furthermore, given a state $s_{t_{n^e}-1}^e = (z_{n^e-1}^e, d = 0)$,

$$\prod_{t=t_{n^e-1}^e}^{T-1} P_{\mathcal{F}^e}(s_{t+1}^e | s_t^e) = \prod_{t=0}^{T-t_{n^e-1}^e-1} (1 - h_{z_{n^e-1}^e}(t+1)) \tag{23}$$
$$= R_{n^e-1}^e(T - t_{n^e-1}^e)$$

Using (2), (19), (20), (22) and (23) we finally obtain:

$$L_{-,\mathcal{F}^e}^T(s^e) = \mathcal{L}_-^T(z^e, t^e). \tag{24}$$
$\square$

2) Collection of Markov renewal chains as a FSM: Given a partition $\Pi = \{A_{e_\Pi}\}_{e_\Pi \in E(\Pi)}$ of $\mathcal{A}$ into m non-empty and disjoint sub-alphabets, the global generative process G_Π corresponds to a set of m independent Markov renewal chains associated to each individual transmitter $e_\Pi \in E(\Pi)$, each emitting from its own sub-alphabet A_e. This process can be represented as a global FSM $\mathcal{F}_\Pi = (\mathcal{S}_\Pi, \mathbf{s}_0^\Pi, f_\Pi)$, with state set $\mathcal{S}_\Pi = \prod_{e \in E(\Pi)} \mathcal{S}^e$ corresponding to the Cartesian product of the states of the individual FSM of each transmitter $e \in E(\Pi)$ defined in previous section. Given a state $\mathbf{s} \in \mathcal{S}_\Pi$, with $\mathbf{s} = \{s^e\}_{e \in E(\Pi)}$, and an action $\mathbf{a} \in A_\Pi^*$, with $A_\Pi^* = \prod_{e \in E(\Pi)} A_e^*$ and $\mathbf{a} = (a^e)_{e \in E(\Pi)}$, we have $f_\Pi(\mathbf{s}, \mathbf{a}) = (f^e(s^e, a^e))_{e \in E(\Pi)}$, where f^e stands for the next state function from the corresponding individual FSM $\mathcal{F}^e$. For each state we define and $P_{\mathcal{F}_\Pi}(.|\mathbf{s}) = \prod_{e \in E(\Pi)} P_{\mathcal{F}^e}(.|s^e)$.

Proposition 4. *Under assumptions $(\mathcal{P})$, $(\mathcal{Q})$ and $(\mathcal{K})$, the FSM $\mathcal{F}_\Pi = (\mathcal{S}, \mathbf{s}_0, f)$ is ergodic.*

Proof. Under assumptions $(\mathcal{P})$, $(\mathcal{Q})$ and $(\mathcal{K})$, according to Proposition 2 each FSM $\mathcal{F}^e = (\mathcal{S}^e, s_0^e, f^e)$ for $e \in E(\Pi)$, is ergodic.

Therefore, for $e \in E(\Pi)$, there exists some n_0^e, such that for every integer n^e, greater than n_0^e, and for every pair of states $i, j \in \mathcal{S}^e$, there exists a i-to-j path of length n_0^e, a i-to-j path of length $n_0^e + 1$, a i-to-j path of length $n_0^e + 2$, and so on. We denote $n_0 = \max_{e \in E(\Pi)} n_0^e$. Thus, given any two states $\mathbf{s}_i = \{s_i^e\}_{e \in E(\Pi)}$ and $\mathbf{s}_j = \{s_i^e\}_{e \in E(\Pi)}$, there always exists a path of length n_0 allowing to reach $\mathbf{s}_j$ from $\mathbf{s}_i$. Therefore $\mathcal{F}_\Pi$ is ergodic. $\square$

Given a partition $\Pi = \{A_e\}_{e \in E(\Pi)}$ of $\mathcal{A}$ into m non-empty and disjoint sub-alphabets, let (z, t) be sequence of n symbols $z \in \mathcal{A}$ with their corresponding time of arrival t_l, generated by $P = G_\Pi$ and observed in the time window $[\![0, T]\!]$. In this sequence, each transmitter $e \in E(\Pi)$ generates a sequence $(z_l^e, t_l^e)_{l=0,\dots,n^e-1}$ of n^e symbols $z_l^e \in A_e$ with their corresponding time of arrival $t_l^e \in \mathbb{N}$ and their last non observed state (z_{-1}^e, t_{-1}^e). Let $\mathbf{s} = (\mathbf{s}_t)_{t=0,\dots,T}$ be the sequence of $T + 1$ states $\mathbf{s}_t = \{s_t^e\}_{e \in E(\Pi)} \in \mathcal{S}_\Pi$, such that $\forall e \in E(\Pi)$, $\forall l = -1, 0, \dots, n^e-1, \forall t \in [\![t_l^e, t_{l+1}^e]\!]$, $s_t^e = (z_l^e, t-t_l^e)$ given $s_{-1}^e = (z_{-1}^e, |t_{-1}^e| - 1)$.

Proposition 5. *Given a state $\mathbf{s}_{-1}$ at time $t = -1$, the conditional likelihood $L_{-,\mathcal{F}_\Pi}^T(\mathbf{s})$ of this sequence $\mathbf{s}$, generated by the FSM $\mathcal{F}_\Pi$, and observed until time T, is equal to the conditional likelihood $\mathcal{L}_{-,\Pi}^T(z, t)$ of the sequence (z, t) generated by P given $(z_{-1}^e, t_{-1}^e)_{e \in E(\Pi)}$.*

Proof. This is trivial, as all the transmitters are independent, and using Proposition 3. □

Now let Π be a partition associated to a generative model G_Π. The maximum of the exact and approached likelihood, given by (2) and (3), of the sequence (z,t) following (z_{-1}, t_{-1}) are respectively denoted as $\widehat{\mathcal{L}}^T_{-,\Pi}(z,t)$ and $\widehat{\mathcal{L}}^T_{\sim,\Pi}(z,t)$.

Proposition 6. *Given two partitions Π and Π', for any sequence (z,t) sampled from $P_{\Pi|\Pi'}$, we have:*

$$\frac{log(\widehat{\mathcal{L}}^T_{-,\Pi'}(z,t)) - log(\widehat{\mathcal{L}}^T_{\sim,\Pi'}(z,t))}{T} \to 0 \quad (25)$$

a.a.s as $T \to \infty$.

Proof. For any emitter $e \in E(\Pi')$, we denote by $\hat{p}^e$ and $\hat{q}^e$ the estimators that maximize the log-likelihood function:

$$
\begin{aligned}
log(\widehat{\mathcal{L}}^T_-(z^e, t^e)) = {}& log(\hat{p}^e_{z_{-1}, z_0^e}(T)) + log(\hat{q}^e_{z_{-1}}(t_0^e - t_{-1}^e, T)) \\
& - log(\widehat{R}^e_{-1}(|t_{-1}^e| - 1)) \\
& + \sum_{i \in A_e} \sum_{j \in A_e} N^e_{i,j}(T) log\, \hat{p}^e_{i,j}(T) \\
& + \sum_{i \in A_e} \sum_{k \in K_i} N^e_i(k,T) \, log\, \hat{q}^e_i(k,T) \\
& + log(\widehat{R}^e_{N^e(T)-1}(u_T)) \quad (26)
\end{aligned}
$$

with $u_T = T - t^e_{N^e(T)-1}$.

We first look at each estimator $\hat{q}^e_i$ for any symbol i from A_e, that are all independent from $\hat{p}^e$ in that maximization, and also independent from each other $\hat{q}^e_j$ whenever $j \neq i$. By construction, our algorithm considers all necessary delays as included in $K^{z^e_i}$ for each i. For a sufficiently large observation window $[\![0,T]\!]$, we thus have, for each transition starting from symbol i, a delay $k \in K^{z^e_i}$ that match the delay of the observed transition, and thus one different parameter $\hat{q}^e_i(k,T) > 0$. Since the sequence is sampled from a process that guarantees the feasibility for $P_{\Pi'}$, we know that two symbols from A_e cannot be emitted at the same time, which discards $k = 0$.

Recalling $\widehat{R}^e_i(x) = P(t^e_{i+1} - t^e_i > x | z_i)$, we can write :

$$\widehat{R}^e_{-1}(|t_{-1}^e| - 1) = \sum_{k \in K^{z^e_{-1}}, k > |t_{-1}^e| - 1} \hat{q}^e_{z_{-1}}(k,T). \quad (27)$$

and

$$\widehat{R}^e_{N^e(T)-1}(u_T) = \sum_{k \in K^{z^e_{N^e(T)-1}}, k > u_T} \hat{q}^e_{z_{N^e(T)-1}}(k,T). \quad (28)$$

For a given symbol i and a given delay $k \in K^{z^e_i}$, we note that $\hat{q}^e_i(k,T)$ can be part of the expression of $\widehat{R}^e_{-1}(|t_{-1}^e| - 1)$ (if $z^e_{-1} = i$ and $k > |t_{-1}^e|$) or $\widehat{R}^e_{N^e(T)-1}(u_T)$ (if $z^e_{N^e(T)-1} = i$ and $k > u_T$).

Using of the same method as in the proof for Proposition 1 with these additional factors, we obtain the following equation by canceling the derivative of the Lagrangian expression w.r.t. $\hat{q}^e_i(k,T)$:

$$\lambda_i = -\xi^e_{-1}(k,i) + \frac{I(k = t_0^e - t^e_{-1}) + N^e_i(k,T)}{\hat{q}^e_i(k,T)} + \xi^e_u(k,i) \quad (29)$$

where I(.) is the indicator function that equals 1 if the argument is true, 0 otherwise, and

$$\xi^e_{-1}(k,i) = \begin{cases} \frac{1}{\widehat{R}^e_{-1}(|t^e_{-1}| - 1)} & \text{if } z^e_{-1} = i \text{ and } k \geq |t^e_{-1}| \\ 0 & \text{otherwise,} \end{cases}$$

$$\xi^e_u(k,i) = \begin{cases} \frac{1}{\widehat{R}^e_{N^e(T)-1}(u_T)} & \text{if } z^e_{N^e(T)-1} = i \text{ and } k > u_T \\ 0 & \text{otherwise.} \end{cases}$$

Next, since $\xi^e_{-1}(k,i)$ and $\xi^e_u(k,i)$ are both upper-bounded by $\frac{1}{\hat{q}^e_i(k,T)}$, we observe that λ_i is bounded as:

$$\frac{N^e_i(k,T) + 2}{\hat{q}^e_i(k,T)} \geq \lambda_i \geq \frac{N^e_i(k,T) - 1}{\hat{q}^e_i(k,T)}.$$

Since $\hat{q}^e_i(k,T) \leq 1$, we know that $\lambda_i \geq N^e_i(k,T) - 1$. We also deduce that

$$\frac{N^e_i(k,T) + 2}{\lambda_i} \geq \hat{q}^e_i(k,T) \geq \frac{N^e_i(k,T) - 1}{\lambda_i}.$$

Considering that sequences are sampled from $P_{\Pi|\Pi'}$, and since we know that the process is ergodic from assumption $\mathcal{K}$, we have for every symbol i and delay in $k \in K^e_i$:

$$\frac{N^e_i(k,T)}{T} \to \frac{q^*_i(k)}{\delta^*_i} \text{ a.a.s as } T \to \infty$$

with δ^*_i the expected delay between two occurences of symbol i in the sequence and $q^*_i(k)$ the true probability of delay k from symbol i in the process $P_{\Pi|\Pi'}$. Thus, since δ^*_i and $q^*_i(k)$ are stationary, we get that both $N^e_i(k,T)$ and λ_i increase with T. We obtain that both bounds of $\hat{q}^e_i(k,T)$ converge in probability to $\frac{N^e_i(k,T)}{\lambda_i}$, thus: $\hat{q}^e_i(k,T) - \frac{N^e_i(k,T)}{\lambda_i} \to 0$ a.a.s as $T \to \infty$.

From the Karush–Kuhn–Tucker conditions, we know that: $\lambda_i(1 - \sum_{l \in K^i} \hat{q}^e_i(l,T)) = 0$, thus λ_i converges in probability toward $\sum_{l \in K^i} N^e_i(l,T)$ when T goes toward infinity. Finally, we get that: $(\hat{q}^e_i(k,T) - \frac{N^e_i(k,T)}{\sum_{l \in K^i} N^e_i(l,T)}) \to 0$ a.a.s as $T \to \infty$, which corresponds to the estimator obtained from the maximization of the approximation $\widehat{\mathcal{L}}^T_{\sim,\Pi'}(z,t)$.

Demonstrating the asymptotic equivalence of optima for $\hat{p}^e$ estimators is more direct. We can simply note that both likelihoods only differ from the first transition, that is ignored from $\widehat{\mathcal{L}}^T_{\sim,\Pi'}(z,t)$. The only possible difference thus can lies on $\hat{p}^e_{z_{-1}, z_0^e}(T)$, whose estimation from $\widehat{\mathcal{L}}^T_{-,\Pi'}(z,t)$ gives $\frac{N^e_{z_{-1}, z_0^e}(T) + 1}{N^e_{z_{-1}}(T) + 1}$, while we obtain $\frac{N^e_{z_{-1}, z_0^e}(T)}{N^e_{z_{-1}}(T)}$ from the maximization of $\widehat{\mathcal{L}}^T_{\sim,\Pi'}(z,t)$. Similarly as above, we can simply note that, for any $i \in A^e$ and $j \in A^e$:

$$\frac{N^e_i(T)}{T} \to \frac{1}{\delta^*_i} \text{ a.a.s as } T \to \infty$$

and:

$$\frac{N^e_{i,j}(T)}{T} \to \frac{p^*_{i,j}}{\delta^*_i} \text{ a.a.s as } T \to \infty$$

with $p_{i,j}^*$ the stationary probability for emitting j from i in process $P_{\Pi|\Pi'}$. Thus, as all components except T are constants, we get, recalling $\hat{p}_{z_{-1}^e, z_0^e}^e(T) = \frac{N_{z_{-1}^e, z_0^e}^e(T)+1}{N_{z_{-1}^e}^e(T)+1}$:

$$\frac{N_{z_{-1}^e, z_0^e}^e(T)+1}{N_{z_{-1}^e}^e(T)+1} - \frac{N_{z_{-1}^e, z_0^e}^e(T)}{N_{z_{-1}^e}^e(T)} \to 0 \text{ a.a.s. as } T \to \infty$$

Thus, we know that all estimators from both optimization converge asymptotically toward the same optima. We directly obtain that:

$$\frac{\log(\widehat{\mathcal{L}}_{-,\Pi'}^T(z,t)) - \log(\widehat{\mathcal{L}}_{\sim,\Pi'}^T(z,t))}{T} \to 0 \quad (30)$$

a.a.s. as $T \to \infty$, since every component from $\widehat{\mathcal{L}}_{-,\Pi'}^T(z,t)$ not present in $\widehat{\mathcal{L}}_{\sim,\Pi'}^T(z,t)$ converge to constant values depending on stationary true parameters of the process $P_{\Pi|\Pi'}$. □

Now using Theorem 1 and Propositions 4, 5 and 6 we prove Theorem 2.

Proof. For this proof we took inspiration from the proof of Lemma 10 in [2] and we use the representation of the generative model G_Π as a global ergodic FSM $\mathcal{F}_\Pi$, as described above. Let Π' be a partition of $\mathcal{A}$ such that $\Pi' \neq \Pi$.

Let respectively $\mathcal{F}_\Pi = (S_\Pi, s_0^\Pi, f_\Pi)$ and $\mathcal{F}_{\Pi'} = (S_{\Pi'}, s_0^{\Pi'}, f_{\Pi'})$ be the FSM representing $P_\Pi = G_\Pi(\{P^e\}_{e \in E(\Pi)})$ and $P_{\Pi'} = G_\Pi(\{P^e\}_{e \in E(\Pi')})$. Let $P_{\mathcal{F}_\Pi}$ and $P_{\mathcal{F}_{\Pi'}}$ be the respective probability distribution defined on states of the respective FSM $\mathcal{F}_\Pi$ and $\mathcal{F}_{\Pi'}$ to represent the processes.

First of all, let $\mathcal{F}^+ = (S^+, s_0^+, f^+)$ be a common refinement of $\mathcal{F}_\Pi$ and $\mathcal{F}_{\Pi'}$, such that there exist functions $g_\Pi : S^+ \to S_\Pi$ and $g_{\Pi'} : S^+ \to S_{\Pi'}$ allowing to recover states of both processes from S^+. That is, for any sequence (z,t) observed until time T and its corresponding state sequence $(s_t^+)_{t=0}^T$, the respective state sequences $(s_t^\Pi)_{t=0}^T$ and $(s_t^{\Pi'})_{t=0}^T$ satisfy $s_t^\Pi = g_\Pi(s_t^+)$ and $s_t^{\Pi'} = g_{\Pi'}(s_t^+)$, for any $t = 0 \dots T$. According to [2], it is always possible to construct a common refinement of two FSMs, whose state set S^+ is the Cartesian product of the two respective state sets. Thus, for any state $s_t^+ \in S^+$, we consider $s_t^+ = (s_t^\Pi, s_t^{\Pi'})$, with $s_t^\Pi \in \mathcal{S}^\Pi$ and $s_t^{\Pi'} \in \mathcal{S}^{\Pi'}$.

At any time step t, $s_t^\Pi \in \mathcal{S}_\Pi$ (resp. $s_t^{\Pi'} \in \mathcal{S}_{\Pi'}$) can be recovered from s_t^+, by considering the function g_Π (resp. $g_{\Pi'}$) that selects the corresponding part to Π (resp. Π') from s_t^+.

In the following, we note $P_\mathcal{F}^+$ the adaptation of the probabilities $P_\mathcal{F}$ to the states of the FSM $\mathcal{F}^+$, that is: $\forall s_i^+ \in \mathcal{S}^+, \forall a \in A_\Pi^*, P_{\mathcal{F}_\Pi}^+(a|s_i^+) = P_{\mathcal{F}_\Pi}(a|g_\Pi(s_i^+))$. Thus, $\forall (s_i^+, s_j^+) \in \mathcal{S}^+ \times \mathcal{S}^+, P_{\mathcal{F}_\Pi}^+(s_j^+|s_i^+) = \sum_{a \in A_\Pi^*} P_{\mathcal{F}_\Pi}(a|g_\Pi(s_i^+)) I(f^+(s_i^+, a) = s_j^+)$, with $I(.)$ the indicator function. One notes that to make $P_{\mathcal{F}_\Pi}^+$ irreducible, we consider in the following a set S^+ only containing states that are reachable from the process P_Π. Thus, $\mathcal{S}^+$ is a finite state space, since every transmitter owns a finite maximal emission delay. Also, as $P_{\mathcal{F}_\Pi}$ is ergodic according to Proposition 2 under assumptions $(\mathcal{P})$, $(\mathcal{Q})$ and $(\mathcal{K})$, and $P_{\mathcal{F}_\Pi}^+$ is ergodic under the same assumptions.

In that setting, the asymptotic normalized Kullbak-Liebler divergence relative to P_Π from $P_{\Pi'}$ can be written as:

$$
\begin{aligned}
D(P_\Pi||P_{\Pi'}) &= D(P_{\mathcal{F}_\Pi}^+||P_{\mathcal{F}_{\Pi'}}^+) \quad (31) \\
&= \sum_{s \in S^+} P_{\mathcal{F}_\Pi}^+(s) D(P_{\mathcal{F}_\Pi}^+(.|s)||P_{\mathcal{F}_{\Pi'}}^+(.|s)),
\end{aligned}
$$

with $D(P_{\mathcal{F}_\Pi}^+(.|s)||P_{\mathcal{F}_{\Pi'}}^+(.|s))$ the Kullback-Leibler divergence of the conditional distribution $P_{\mathcal{F}_\Pi}^+(.|s)$ from $P_{\mathcal{F}_{\Pi'}}^+(.|s)$. This provides us a powerful tool to compare the processes.

However, as already mentioned above, impossible actions for $P_{\Pi'}$ can be emitted from P_Π when for instance an emission delay is lower (resp. greater) than the minimal (resp. maximal) delay for the corresponding transmitters from Π'.

- First, regarding maximal delay, when following P_Π it may happen that a symbol z from a transmitter e of Π has not been observed since a long time d, because for instance other symbols from e were repeated many times after its last occurrence.
- On the other hand, actions can be blocked because they correspond to the simultaneous emission of symbols that shouldn't be observed at the same timestep, due to the fact they belong to the same sub-alphabet of Π'.

Such cases result in an undefined KL divergence, with some impossible transitions or incompatible states for Π'. Considering sufficiently long time windows, such bad events occur with probability 1 for every $\Pi' \neq \Pi$, for sequences generated from P_Π, due to the incompatibility result stated in theorem 1. As explained in section III-B, such situation gets $C_{\Pi'}^T(z,t) := +\infty$ when detected, which allows our algorithm to discard the corresponding partition. This respects the positive difference from the theorem, as $C_\Pi^T(z,t)$ always remain defined after a sufficiently large T.

However, such bad events can occur very rarely for some scattered sets of emission delays. In the following, we show that even for sequences feasible for both process P_Π and $P_{\Pi'}'$, our algorithm is able to correctly identify the best partition, after a sufficiently long period of time T. For this, we consider sequences sampled from $P_{\Pi|\Pi'}$, which corresponds to a process restricting P_Π to the subset of possible sequences for $P_{\Pi'}$.

Thus, we consider an alternative FSM $\mathcal{F}^* = (S^*, s_0^*, f^*)$, which corresponds to the common refinement $\mathcal{F}^+$ in which we remove from f^+ all transitions which are either impossible regarding Π or Π', or that lead to any $s \in \mathcal{S}^+$ from which every sequence to come back in s implies impossible actions regarding Π or Π'. $\mathcal{S}^* \subset \mathcal{S}^+$ contains only states that are reachable following possible transitions for both processes, starting from any state from $\mathcal{S}^*$. By definition, $s \in \mathcal{S}^*$ implies there exists a set of trajectories in $\mathcal{F}^*$ that allows us to leave and go back to s. In that FSM, following process $P_{\Pi|\Pi'}$ corresponds to considering $P_{\mathcal{F}_\Pi}^*$ as the probabilities from $P_{\mathcal{F}_\Pi}^+$ adapted to $\mathcal{F}^*$, where the probability mass of actions leading deterministically outside $\mathcal{S}^*$ from any state in $\mathcal{S}^*$ are evenly redistributed on other admissible actions regarding f^*. Using this, we have:

- By definition, starting from a state from $\mathcal{S}^*$, there always exists an alternative to the simultaneous emission of symbols from the same transmitter in $E(\Pi')$. These symbols can be emitted from their corresponding transmitters in $E(\Pi)$ at different timesteps. When such a delayed emission occurs, the corresponding transmitter recovers its whole set of possible delays, with gcd of 1, hence ensuring aperiodicity of the state in $\mathcal{F}^*$.
- By construction, as $\mathcal{S}^*$ is a finite set, there always exists a finite k_{max} sufficiently high to maintain aperiodicity of any state from $\mathcal{S}^*$ despite blocked transitions due to maximal delays.

Thus, if $\mathcal{S}^*$ is non-empty (some sequences can be generated by both processes when starting from s_0^Π), $P_{\mathcal{F}_\Pi}^*$ is ergodic under the same assumptions as above for the ergodicity of $P_{\mathcal{F}_\Pi}$ on $\mathcal{S}$. In that setting, the KL divergence relative to $P_{\Pi|\Pi'}$ from $P_{\Pi'}$ can be given as

$$
\begin{aligned}
D(P_{\Pi|\Pi'} || P_{\Pi'}) &= D(P_{\mathcal{F}_\Pi}^* || P_{\mathcal{F}_{\Pi'}}^*) = D(P_{\mathcal{F}_\Pi}^* || P_{\mathcal{F}_{\Pi'}}^+) \quad (32) \\
&= \sum_{s \in S^*} P_{\mathcal{F}_\Pi}^*(s) D(P_{\mathcal{F}_\Pi}^*(.|s) || P_{\mathcal{F}_{\Pi'}}^*(.|s))
\end{aligned}
$$

Let (z,t) be sequence of n symbols $z_l \in \mathcal{A}$ with their corresponding time of arrival $t_l \in \mathbb{N}$, generated by $P_{\Pi|\Pi'}$ and observed in the time window $[\![0,T]\!]$. In this sequence, each transmitter e, generates a sequence $(z_l^e, t_l^e)_{l=0,\ldots,n^e-1}$ of n^e symbols $z^e \in A_e$ with their corresponding time of arrival $t_l^e \in \mathbb{N}$. Let $\mathbf{s}^T = (\mathbf{s}_t)_{t=0,\ldots,T}$ be the sequence of $T+1$ vector states $\mathbf{s}_t \in S^*$, corresponding to the generation of (z,t) with $P_{\mathcal{F}_\Pi}^*$ on $\mathcal{F}^*$.

Let $\hat{P}_{\mathcal{F}_\Pi}^*(\mathbf{s}^T)$ be the probability estimator that maximizes the log-likelihood $\log(\hat{L}_{\mathcal{F}_\Pi}^{T,*}(\mathbf{s}^T)))$, with corresponding empirical entropy $H_{\mathcal{F}_\Pi}^{T,*}(\mathbf{s}^T) = -\log(\hat{L}_{\mathcal{F}_\Pi}^{T,*}(\mathbf{s}^T)))$. We have

$$
H_{\mathcal{F}_\Pi}^{T,*}(\mathbf{s}^T) = -\sum_{s,s' \in \mathbf{s}^T} N_{s,s'}(T) \log \hat{P}_{\mathcal{F}_\Pi}^*(s'|s), \quad (33)
$$

with $N_{s,s'}(T)$ the number of transitions from state $s \in S^*$ to state $s' \in S^*$ observed in the sequence $\mathbf{s}^T$ until time T. We have:

$$
\begin{aligned}
&\frac{1}{T}(H_{\mathcal{F}_{\Pi'}}^{T,*}(\mathbf{s}^T) - H_{\mathcal{F}_\Pi}^{T,*}(\mathbf{s}^T)) = \\
&\sum_{s \in \mathbf{s}^T} N_s/T \sum_{s' \in \mathbf{s}^T} [N_{s,s'}/N_s] \log[\hat{P}_{\mathcal{F}_\Pi}^*(s'|s)/\hat{P}_{\mathcal{F}_{\Pi'}}^*(s'|s)] \quad (34)
\end{aligned}
$$

As $\hat{P}_{\mathcal{F}_\Pi}^*(\mathbf{s}^T)$ is the probability estimator that maximizes the likelihood of $\mathbf{s}^T$, we have $N_s/T = \hat{P}_{\mathcal{F}_\Pi}^*(s)$ and $N_{s,s'}/N_s = \hat{P}_{\mathcal{F}_\Pi}^*(s'|s)$, since $\mathbf{s}^T$ was generated following $P_{\Pi|\Pi'}$. Thus:

$$
\begin{aligned}
&\frac{1}{T}(H_{\mathcal{F}_{\Pi'}}^{T,*}(\mathbf{s}^T) - H_{\mathcal{F}_\Pi}^{T,*}(\mathbf{s}^T)) \\
&= \sum_{s \in \mathbf{s}^T} \hat{P}_{\mathcal{F}_\Pi}^*(s) \sum_{s' \in \mathbf{s}^T} \hat{P}_{\mathcal{F}_\Pi}^*(s'|s) \log[\hat{P}_{\mathcal{F}_\Pi}^*(s'|s)/\hat{P}_{\mathcal{F}_{\Pi'}}^*(s'|s)] \\
&= D(\hat{P}_{\mathcal{F}_\Pi}^* || \hat{P}_{\mathcal{F}_{\Pi'}}^*). \quad (35)
\end{aligned}
$$

Next, as $H_{\mathcal{F}_\Pi}^{T,*}(\mathbf{s}^T) = -\log(\hat{L}_{\mathcal{F}_\Pi}^{T,*}(\mathbf{s}^T))$ and $H_{\mathcal{F}_{\Pi'}}^{T,*}(\mathbf{s}^T) = -\log(\hat{L}_{\mathcal{F}_{\Pi'}}^{T,*}(\mathbf{s}^T))$, by using (35) and Proposition 5, we obtain:

$$
\frac{1}{T}(-\log(\hat{\mathcal{L}}_{-,\Pi'}^T(z,t)) + \log(\hat{\mathcal{L}}_{-,\Pi}^T(z,t))) = D(\hat{P}_{\mathcal{F}_\Pi}^* || \hat{P}_{\mathcal{F}_{\Pi'}}^*). \quad (36)
$$

And thus,

$$
\begin{aligned}
&\frac{1}{T}(-\log(\hat{\mathcal{L}}_{\sim,\Pi'}^T(z,t)) + \log(\hat{\mathcal{L}}_{\sim,\Pi}^T(z,t))) = D(\hat{P}_{\mathcal{F}_\Pi}^* || \hat{P}_{\mathcal{F}_{\Pi'}}^*) \\
&+ \frac{1}{T}(\log(\hat{\mathcal{L}}_{-,\Pi'}^T(z,t)) - \log(\hat{\mathcal{L}}_{\sim,\Pi'}^T(z,t))) \\
&- \frac{1}{T}(\log(\hat{\mathcal{L}}_{-,\Pi}^T(z,t)) - \log(\hat{\mathcal{L}}_{\sim,\Pi}^T(z,t))). \quad (37)
\end{aligned}
$$

Now from (37) and Proposition 6, we deduce that the quantity $\frac{1}{T}(H_{\Pi'}^T(z,t) - H_\Pi^T(z,t))$ converges toward $D(\hat{P}_{\mathcal{F}_\Pi}^* || \hat{P}_{\mathcal{F}_{\Pi'}}^*)$ in probability when T goes toward infinity. So does $\frac{1}{T}(C_{\Pi'}^T(z,t) - C_\Pi^T(z,t))$, with $C_\Pi^T(z,t) = H_\Pi^T(z,t) + \gamma m \log(n)$, as the term $\frac{\gamma m \log(n)}{T}$ goes toward 0 as T goes toward infinity, as γ and m are positive constants and $n \leq T$.

Since the sequence $\mathbf{s}^T$ is generated with probability $P_{\mathcal{F}_\Pi}^*$, we have $D(P_{\mathcal{F}_\Pi}^* || \hat{P}_{\mathcal{F}_\Pi}^*) \to 0$
as $T \to \infty$. It thus remain to demonstrate that $D(P_{\mathcal{F}_\Pi}^* || \hat{P}_{\mathcal{F}_{\Pi'}}^*) > 0$ to conclude the proof.

Since Π' is, by assumption different from Π, and thus following Theorem 1 incompatible with P_Π, no valid assignment of parameters for $\mathcal{F}_{\Pi'}$ can generate $P_{\Pi|\Pi'}$, and, thus, $P_{\mathcal{F}_\Pi}^*$ is not in the set of valid parameters $V(\Pi') \subset V^*$ for $\mathcal{F}_{\Pi'}^*$, with V^* the whole set of valid parameters in $\mathcal{F}^*$. Said otherwise, we can distinguish two possible settings:

- At least two symbols z and z' belong to the same sub-alphabet A_{e_Π} in Π and to two respective different sub-alphabets $A_{e_{\Pi'}}$ and $A_{e'_{\Pi'}}$ in Π'. For any $s \in \mathcal{S}^*$, such that $s_\Pi^{e_\Pi} = (z',d')$, with $s_\Pi = g_\Pi(s)$, and $s_{\Pi'}^{e_{\Pi'}} = (z,d)$, with $s_{\Pi'} = g_{\Pi'}(s)$ (i.e., we have $d > d'$), $D(P_{\mathcal{F}_\Pi}^*(.|s) || \hat{P}_{\mathcal{F}_{\Pi'}}^*(.|s)) = 0$ implies $P_{\mathcal{F}_\Pi}^*(a_\Pi^{e_\Pi} = z|s) = P_{\mathcal{F}_{\Pi'}}^*(a_{\Pi'}^{e_{\Pi'}} = z|s)$ for any $(a_\Pi, a_{\Pi'}) \in A_\Pi^* \times A_{\Pi'}^*$ such that $P_{\mathcal{F}_\Pi}^*(a_\Pi|s) > 0$. This cannot be true for every such state. Two different states $(s,s') \in \mathcal{S}^{*2}$ can be equal on e_Π (i.e, $s_\Pi^{e_\Pi} = s_\Pi'^{e_\Pi}$), while having a different value for $e_{\Pi'}$ (e.g., $s_{\Pi'}^{e_{\Pi'}} = (z,d)$ and $s_{\Pi'}'^{e_{\Pi'}} = (z,d')$, with $d \neq d'$. From the definition of the probabilities of actions, which are based on the instantaneous risk of emission, we have: $P_{\mathcal{F}_{\Pi'}}^*(a_{\Pi'}^{e_{\Pi'}}|s) \neq P_{\mathcal{F}_{\Pi'}}^*(a_{\Pi'}^{e_{\Pi'}}|s')$. At the same time, since $P_{\mathcal{F}_\Pi}^*(a_\Pi^{e_\Pi}|s) = P_{\mathcal{F}_\Pi}^*(a_\Pi^{e_\Pi}|s')$, we can conclude that there exists some state $s \in \mathcal{S}^*$, such that $P_{\mathcal{F}_\Pi}^*(a_\Pi^{e_\Pi}|s) \neq P_{\mathcal{F}_{\Pi'}}^*(a_{\Pi'}^{e_{\Pi'}}|s)$.
- At least two symbols z and z' belong to two respective different sub-alphabets A_{e_Π} and $A_{e'_\Pi}$ in Π and to the same sub-alphabet $A_{e_{\Pi'}}$ in Π'. Same manner as above, $P_{\mathcal{F}_\Pi}^*(a_\Pi^{e_\Pi} = z|s)$ cannot be equal to $P_{\mathcal{F}_{\Pi'}}^*(a_{\Pi'}^{e_{\Pi'}} = z|s)$ for all s such that $s_\Pi^{e_\Pi} = (z,d)$, with $s_\Pi = g_\Pi(s)$, and $s_{\Pi'}^{e_{\Pi'}} = (z',d')$, with $s_{\Pi'} = g_{\Pi'}(s)$.

Thus, $D(P_{\mathcal{F}_\Pi}^* || \hat{P}_{\mathcal{F}_{\Pi'}}^*) > 0$ for any estimation of probabilities in $\mathcal{F}_{\Pi'}$, since $D(P_{\mathcal{F}_\Pi}^* || \hat{P}_{\mathcal{F}_{\Pi'}}^*) = 0$ only if $\forall s \in \mathcal{S}^*$ such that $P_{\mathcal{F}_\Pi}^*(s) > 0$ we have $D(P_{\mathcal{F}_\Pi}^*(.|s) || \hat{P}_{\mathcal{F}_{\Pi'}}^*(.|s)) = 0$, which is impossible for $\Pi \neq \Pi'$ as demonstrated above. Thus

$$
\lim_{T \to \infty} \mathbb{P}_{(z,t) \sim P_\Pi}(\frac{1}{T}(C_{\Pi'}^T(z,t) - C_\Pi^T(z,t)) > 0) = 1, \quad (38)
$$

which concludes the proof of Theorem 2.

□

D. Details of the procedures used in the memetic algorithm for alphabet partitioning

In this section, we describe in more detail the local search (TabuAP) and the crossover (GLPX) procedures used in Algorithm 1 (see Section III-C of the main paper).

1) TabuAP local search procedure: TabuAP for alphabet partitioning is adapted from [3]. Starting from an initial partition Π, it then iteratively replaces the current solution Π by a neighboring solution Π' taken from its one-move neighborhood $N(\Pi)$ until it reaches a maximum of nb_{iter} iterations.

A neighboring solution of a partition $\Pi = \cup_{e=1}^m A_e$ is generated by using the *one-move* operator, which displaces a symbol $a \in A_e$ to a different sub-alphabet A_f, such that $f \neq e$. A symbol $a \in A_e$ is allowed to be transferred to an existing group A_f for $f = 1, \ldots, m$, with $f \neq e$, or to a new group A_{m+1}. We use $\Pi \oplus < a, A_e, A_f >$ to denote the resulting neighboring partition after performing a one-move operation.

At each iteration, TabuAP examines the neighborhood and selects a best admissible neighboring solution Π' to replace Π. A neighboring solution $\Pi \oplus < a, A_e, A_f >$ built from Π is admissible if the associated one-move $< a, A_e, A_f >$ was not registered in the tabu list, which records the most recently performed moves. Each time a move is performed, its reversed move is added in the tabu list and forbidden during the $t = r(10) + \alpha|\mathcal{A}|$ next iterations (tabu tenure) where r is a random number uniformly drawn in $1, ..., 10$ and α is a hyperparameter of the algorithm set to 0.6.

In order to identify the best admissible partition in the neighborhood, we computer the global penalized scores given by (9) of all the neighboring partitions $\Pi \oplus < a, A_e, A_f >$ and retain the neighboring partition with the lowest score. For a move $< a, A_e, A_f >$ applied to the current partition Π and resulting in a new partition $\Pi' = \Pi \oplus < a, A_e, A_f >$, only the negative log-likelihood scores of the changing groups A_e and A_f need to be reevaluated.

2) Greedy Likelihood-based Crossover Operator: The GLPX crossover, adapted from [3], aims to transmit to the offspring the largest possible A_e sub-alphabets from both parents with as the lowest entropy as possible, because our problem is to minimize the global negative log-likelihood of the partition over the whole alphabet.

A GLPX score associated to a group of letters A_e is defined as $\frac{H^T(z^e, t^e)}{|A_e|}$, where $H^T(z^e, t^e)$ is the negative log likelihood for transmitter e given by (4) and $|A_e|$ is the size of the sub-alphabet A_e.

Given two parent partitions Π_1 and Π_2, the GLPX procedure performs two steps. First, it transmits to the offspring the sub-alphabet with the lowest GLPX score. After having withdrawn the letters of this sub-alphabet in both parents and recomputed all scores, it transmits to the offspring the sub-alphabet with the lowest GLPX score of the second parent. This procedure is repeated until all the letters of the alphabet $\mathcal{A}$ are assigned to the offspring. Note that this crossover is asymmetrical. Starting the procedure with parent Π_1 or parent Π_2 can produce different offspring partitions. Therefore when used in the MAAP algorithm described in Algorithm 1 to generate two new offspring partitions $\Pi_1 = GLPX(\Pi'_1, \Pi'_2, (z, t))$ and $\Pi_2 = GLPX(\Pi'_2, \Pi'_1, (z, t))$, the two offspring partitions Π_1 and Π_2 can be very different (in the sense of edit distance).

REFERENCES

[1] V. S. Barbu and N. Limnios, *Semi-Markov chains and hidden semi-Markov models toward applications: their use in reliability and DNA analysis*. Springer Science & Business Media, 2009, vol. 191.

[2] G. Seroussi, W. Szpankowski, and M. J. Weinberger, "Deinterleaving finite memory processes via penalized maximum likelihood," *IEEE Transactions on Information Theory*, vol. 58, no. 12, pp. 7094–7109, 2012.

[3] J. Pinsolle, O. Goudet, C. Enderli, and J.-K. Hao, "A memetic algorithm for deinterleaving pulse trains," in *Evolutionary Computation in Combinatorial Optimization, Lecture Notes in Computer Science 13987*, L. Pérez Cáceres and T. Stützle, Eds., Cham: Springer Nature Switzerland, 2023, pp. 66–81.