Journal of Machine Learning Research 14 (2013) 2213-2238 Submitted 11/12; Revised 3/13; Published 8/13

Segregating Event Streams and Noise with a Markov Renewal Process Model

Dan Stowell DAN.STOWELL@EECS.QMUL.AC.UK
Mark D. Plumbley MARK.PLUMBLEY@EECS.QMUL.AC.UK
Centre for Digital Music
Queen Mary University of London
London, E1 4NS, United Kingdom

Editor: Samy Bengio

Abstract

We describe an inference task in which a set of timestamped event observations must be clustered into an unknown number of temporal sequences with independent and varying rates of observations. Various existing approaches to multi-object tracking assume a fixed number of sources and/or a fixed observation rate; we develop an approach to inferring structure in timestamped data produced by a mixture of an unknown and varying number of similar Markov renewal processes, plus independent clutter noise. The inference simultaneously distinguishes signal from noise as well as clustering signal observations into separate source streams. We illustrate the technique via synthetic experiments as well as an experiment to track a mixture of singing birds. Source code is available.

Keywords: multi-target tracking, clustering, point processes, flow network, sound

1. Introduction

Various approaches exist for the task of inferring the temporal evolution of multiple sources based on joint observations (Mahler, 2007; Van Gael et al., 2009). They are generally based on a model in which sources are continuously observable, in the sense that they are expected to emit/return observations at every time step (though there may be missed detections). Yet there are various types of source for which observations are inherently intermittent, and for which this intermittence exhibits temporal structure that can be characterised as a point process. Examples include sound event sequences such as bird calls or footsteps (Wang and Brown, 2006), internet access logs (Arlitt and Williamson, 1997), pulsars in astronomy (Keane et al., 2010) and neural firing patterns (Bobrowski et al., 2009). Intermittent observations are also often output from *sparse representation* techniques, which transform signals into a representation with activations distributed sparsely in time and state (Plumbley et al., 2010).

In this paper we describe a generic problem setting that may be applied to such data, along with an approach to estimation. We are given a set of timestamped data, and we assume each datum is produced by one of a set of similar but independent signal processes, or by a "clutter" noise process, with known parameters. We do not know the true partitioning of the data into sequences each generated by a single process, and wish to infer this. We do not know how many processes are active, and we do not assume that each process produces the same number of observations, or observations at the same time points.

This specific type of clustering problem has applications in various domains. For example, when sparse representation techniques are used for source separation in time series, they often yield a set of atomic activations which must be clustered according to their underlying source, and preferably to discard any spurious noise activations (Plumbley et al., 2010). Temporal dependence information may help to achieve this (cf. Mysore et al. 2010). Timestamped data such as internet access logs often contain no explicit user association, yet it may be desirable to group such data by user for for further analysis (Arlitt and Williamson, 1997). In computational audio scene analysis, it is often the case that sound sources emit sound only intermittently during their presence in the scene, yet it is desirable to track their temporal evolution.

1.1 Related Work

To our knowledge, this particular problem setting has not been directly addressed in the literature. Temporal data is most commonly treated using a model of sources which update continuously, or synchronously at an underlying temporal sampling rate. Pertinent formulations for our purposes include the infinite factorial hidden Markov model (infinite FHMM) of Van Gael et al. (2009), or the probability hypothesis density filter (PHD filter) (Mahler, 2007), both of which infer an unknown number of independent Markov sources. FHMMs assume that the underlying sources are not intermittent during their lifetime, and also that they persist throughout the whole observation period. Pragmatically, intermittent emissions may be handled by incorporating silence states, though to implement arbitrary-duration silence states may require additional workarounds such as multiple parallel/sequential silences. The PHD filter allows for stochastic missed detections but not for structured intermittency.

Among techniques which do not assume a synchronous update, graph clustering approaches such as normalised cuts have similarities to our approach (Shi and Malik, 2000). In particular, Lagrange et al. (2008) apply normalized cuts in order to cluster temporally-ordered data. However, the normalised cuts method is applied to undirected graphs, and Lagrange et al. (2008) use perceptually-motivated similarity criteria rather than directed Markov dependencies as considered herein. Further, the normalized cuts method does not include a representation of clutter noise, and so Lagrange et al. (2008) perform signal/noise cluster selection as a separate postprocessing step. In the present work we include an explicit noise model.

In automatic speech recognition, segmental models or fragment-decoding models are inferred using a combinatorial graph search through temporal observations (Glass, 2003; Barker et al., 2005), and thus have resonances with the method we will develop here. However they address only a single-source problem. (Barker et al. 2005 considers multi-talker background noise but only one foreground source.)

Our problem setting also exhibits similarities with that of structure discovery in Bayesian networks (Koivisto and Sood, 2004). However, in that context the dependency structure is inferred from correlations present in multiple observations from each vertex in the structure. In the present case we have only one observation per vertex, plus the partial ordering implied by temporality.

In the following we develop a model in which an unknown number of point-process sources are assumed to be active as well as Poisson clutter, and describe how to perform a maximum likelihood inference which clusters the signal into individual identified tracks plus clutter noise. We then demonstrate the performance of the approach in synthetic experiments, and in an experiment analysing birdsong audio.

2. Preliminaries

Throughout we will consider sets of observations in the form $\{(X,T)\}$ where X is state and T is time. A Markov renewal process (MRP) generates a sequence of such observations having the Markov property:

$$\begin{aligned} &P(\tau_{n+1} \leq t, X_{n+1} = j \mid (X_1,T_1),\ldots,(X_n = i,T_n)) \\ &\qquad = P(\tau_{n+1} \leq t, X_{n+1} = j \mid X_n = i) \qquad\qquad \forall n \geq 1, t \geq 0, i, j \in \mathcal{S} \end{aligned}$$

where τ_{n+1} is the time difference $T_{n+1} - T_n$. Note that τ is not explicitly given in observations $\{(X,T)\}$, but can be inferred if we know that a particular pair of observations are adjacent members within a sequence.

We will have cause to represent our data as a *network flow* problem (Bang-Jensen and Gutin, 2007, Chapter 3). A *network* is a graph supplemented such that each arc A_{ij} has a *lower capacity* l_{ij} and *upper capacity* u_{ij}, and a *cost* a_{ij}. A *flow* is a function $x : A \to \mathcal{R}_0$ that associates a value with each arc in the network. We will be concerned with integer flows $x : A \to \mathcal{Z}_0$. A flow is *feasible* if $l_{ij} \leq x_{ij} \leq u_{ij}$ for all A_{ij} in the graph, and for all vertices (except for any source/sink vertices) the sum of the inward flow is equal to the sum of the outward flow. For any flow we can calculate a total cost as the sum of $a_{ij}x_{ij}$ over all A_{ij}. We define the *value* of a feasible flow to be the sum of x_{ij} over all arcs leading from source vertices.

The standard terminology of flow networks associates capacities, flows and costs with arcs but not vertices. However, in the following we will have cause to associate such attributes with vertices as well as with arcs. This can be implemented transparently by the standard technique of *vertex expansion*, in which each vertex is replaced by an in-vertex and an out-vertex, plus a single arc between them which bears the associated attributes (Bang-Jensen and Gutin, 2007, Section 3.2.4).

3. Mixtures of Markov Renewal Processes with Clutter Noise

For the present task, we consider MRPs which are time-limited: each process comes into being at a particular point in time (governed by an independent Poisson process with intensity $\lambda_b(X)$), and after each observation it may "die" with an independent death probability $p_d(X)$. Otherwise it transitions to a new random state-and-time according to the transition distribution $f_x(X,\tau)$. The overall system to be considered is not one but a set of such time-limited MRPs, plus a separate Poisson process that generates clutter noise with intensity $\lambda_c(X)$. The MRPs are independent but share common parameters. We will refer to the overall system (including the noise process) as a *multiple Markov renewal process* system or *MMRP*, in order to clarify when we are referring to the whole system or to a single MRP.

We receive a set of N observations in the form $\{(X,T)\}$ and we assume that they were generated by an MMRP for which the process parameters are known, but the number K of MRPs is unknown as well as the allocation of each observation to its generating process. We assume that each observation is generated either by one MRP or by the noise process. Given these observations as well as model parameters $f_x(X,\tau)$, λ_b, p_d, λ_c, there are many ways to cluster the observations into $K \in [0,N]$ non-overlapping subsets to represent the assertion that each cluster represents all the emissions from a single MRP, with H of the observations not included in any cluster and considered to be noise. The

overall likelihood under a chosen clustering is given by

$$\text{likelihood} = \prod_{k=1}^{K} p_{\text{MRP}}(k) \prod_{\eta=1}^{H} p'_{\text{NOISE}}(\eta)$$

where $p_{\text{MRP}}(k)$ represents the likelihood of the observation subsequence in cluster k being generated by a single MRP, and $p'_{\text{NOISE}}(\eta)$ represents the likelihood of a single observation datum under the noise model. (A set of clusters is arbitrarily indexed by $k \in [1, K]$.)

In order to find the maximum likelihood solution, we may equivalently divide the likelihood expression through by a constant factor, to give an alternative expression to be maximised. We divide by the likelihood that *all* data were generated by the noise process, to give the likelihood ratio:

$$\text{L} = \prod_{k=1}^{K} \frac{p_{\text{MRP}}(k)}{p_{\text{NOISE}}(k)} \tag{1}$$

where for notational simplicity we use $p_{\text{NOISE}}(k)$ as the joint likelihood of all observations contained within cluster k under the noise model. This likelihood ratio L will shortly be seen to be a convenient expression to optimise—in particular because the likelihoods for the H data points labelled as noise do not need to be considered in (1) since their likelihood ratios are 1 (they have the same likelihood in the numerator and denominator).

The component likelihood ratio for a single cluster k is given by

$$\frac{p_{\text{MRP}}(k)}{p_{\text{NOISE}}(k)} = \frac{p_b(X_{k,1}) \cdot p_d(X_{k,n}) \cdot \prod_{i=2}^{n_k} f_{X_{k,i-1}}(X_{k,i}, T_{k,i} - T_{k,i-1})}{\prod_{i=1}^{n_k} p_c(X_{k,i})}$$

where $(X_{k,i}, T_{k,i})$ refers to the ith observation assigned to cluster k, this cluster having n_k observations indexed in ascending time order. The term $p_b(\cdot)$ refers to the likelihood associated with a single observation under the Poisson process parametrised by λ_b, and similarly for $p_c(\cdot)$ for the clutter process parametrised by λ_c.

The overall likelihood ratio L tells us the relative likelihood that the observation set was generated by the selected clustering of signals and noise, as opposed to the possibility that all observations were generated by clutter noise. Our goal is to find the clustering that yields the highest likelihood ratio, and therefore the set of MRP track identities that is most likely to originate from signal rather than noise.

3.1 Network Flow Representation

For any observation set of non-trivial size, there is a combinatorial explosion of possible clusterings available and enumerating them all is intractable. In this subsection we propose to transform the problem into an equivalent problem of network flow, which can be addressed using graph theoretic techniques.

To maximise the likelihood ratio, we can equivalently minimise its negative logarithm, which we will consider as a "cost" for any particular solution. We define additive component costs for

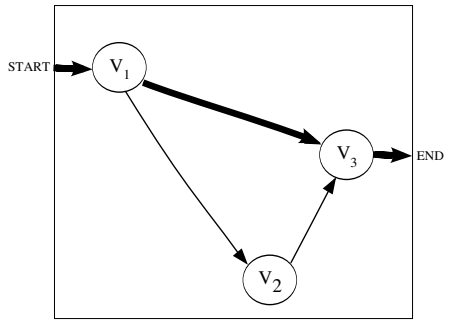

Figure 1: Simple illustration of a path within a network that might correspond to a single MRP sequence. Time increases along the horizontal axis. The bold arrows indicate a path from the first to the third datum (the second datum being left out of the corresponding cluster). The thin arrows indicate an alternative possible path.

birth, death, transition and clutter respectively as:

$$a_b(X) = -\log p_b(X), \tag{2}$$
$$a_d(X) = -\log p_d(X), \tag{3}$$
$$a_t(X, X', \tau) = -\log f_X(X', \tau), \tag{4}$$
$$a_c(X) = \log p_c(X), \tag{5}$$

which leads to the following expression for the overall cost under a particular cluster assignment:

$$-\log(\mathrm{L}) = \sum_{k=1}^{K} \left(a_b(X_{k,1}) + a_d(X_{k,n}) + \sum_{i=2}^{n_k} a_t(X_{ik,i-1}, X_{k,i}, T_{k,i} - T_{k,i-1}) + \sum_{i=1}^{n_k} a_c(X_{k,i}) \right). \tag{6}$$

The Markov structure of transitions, as well as this representation as additive costs, permit a natural representation as a problem defined on a directed graph. If we construct a directed graph with observations as vertices and possible transitions as arcs, then every possible path in the graph (from any vertex to any other reachable vertex) corresponds to one potential MRP cluster (Figure 1). A set of K paths corresponds to a set of K MRP clusters. To reflect the assumption that each observation is generated by no more than one MRP, we require that a vertex can be a member of no more than one path in such a set. Vertices not included in any of the paths correspond to noise observations.

Given our restriction that a vertex can be included in no more than one path, the problem of finding a mutually compatible set of MRP clusterings is equivalent to solving a particular kind of *network flow* problem (Bang-Jensen and Gutin, 2007, Chapter 3). In our case, the concept of a flow will be used to pick out a set of arcs in the graph corresponding to a possible clustering, by

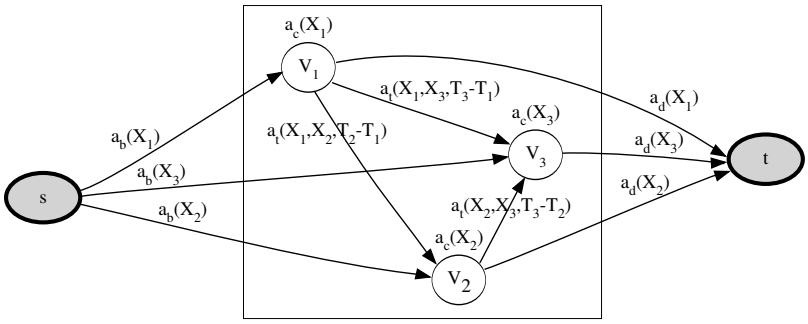

Figure 2: Constructing the weighted flow network for a set of three observations.

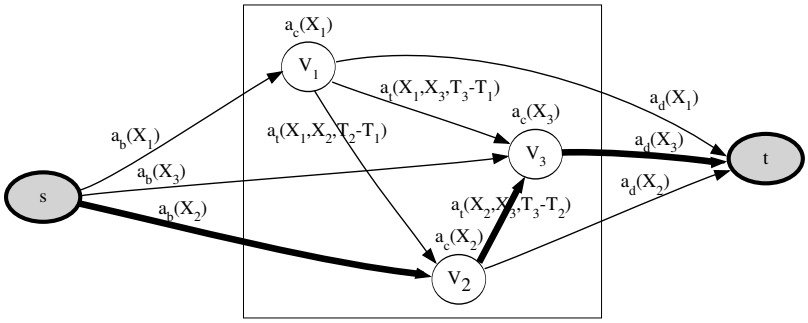

Figure 3: The network of Figure 2, with a single-path flow indicated (*s-2-3-t*).

associating each arc with a value 1 or 0 indicating whether the arc is included in the clustering. Therefore, in addition to the requirement that the flow is integer-valued, all arcs will be defined to have unit capacity: $l_{ij} = 0, u_{ij} = 1$ for all A_{ij}. To reflect our assumption that each observation can be included in only one cluster, we will also specify unit capacities for all vertices.

It remains to specify how we can associate the costs (2)–(5) with the network such that we can solve for the minimum-cost solution to (6). Transition costs will be associated with arcs, and clutter costs with vertices, but in order to include birth and death costs we must modify the network by adding a single "source" vertex with an outward arc to all other vertices, and a single "sink" vertex with an inward arc from all other vertices, and by requiring that no other vertices act as sources or sinks (i.e., in a feasible flow, their inward and outward flows must balance). We then associate birth costs with arcs from the source and death costs with arcs to the sink. This means that all feasible flows in our network will be composed of paths which consist of one single birth cost, plus a sequence of clutter and transition costs, and a single death cost. The source and sink have infinite capacity, allowing for solutions with unbounded K.

Putting these considerations together, constructing the directed graph proceeds as follows:

- A unit-capacity vertex V_i is created corresponding to each observation (X_i, T_i). The clutter noise cost $a_c(X_i)$ is associated with this vertex.

- A unit-capacity arc A_{ij} is created corresponding to each possible transition between two observations such that $T_i < T_j$. The transition cost $a_t(X_i, X_j, T_j - T_i)$ is associated with this arc.

- A "source" vertex s is added, with one arc A_{si} leading from s to each of the observation vertices. The birth cost $a_b(X_i)$ is associated with each arc A_{si}.

- A "sink" vertex t is added, with one arc A_{it} leading from each of the observation vertices to t. The death cost $a_d(X_i)$ is associated with each arc A_{it}.

The temporal ordering of observations means that the graph will contain no cycles.

An illustration of the network constructed for a set of three observations is given in Figure 2. It is clear that any path from the source s to a sink t (we call this an (s,t)-path) visits a sequence of vertices representing a temporal sequence of observations. In the case given in Figure 2, seven different (s,t)-paths are possible, and various combinations of these can form a feasible flow. For example the flow along the single path *s-2-3-t* highlighted in Figure 3 represents the possibility that the observations X_2 and X_3 were generated by a single MRP while X_1 is clutter: the costs associated with flow along that path (the *path flow*) are related to the birth of *2*, the transition from *2* to *3*, and the death of *3*, plus the clutter noise costs. The cost associated with any single-path flow corresponds to one of the K top-level summands in Equation (6). Since in our case each (s,t)-path carries one unit of flow, the *value* of each feasible flow is the number of paths it contains, and corresponds to the number of MRP processes inferred in the data. The total *cost* of each feasible flow is the sum of the path costs contained, and corresponds to the sum calculated in Equation (6).

3.2 Inference

The minimum cost flow in a network constructed according to our scheme corresponds to the clustering with maximum likelihood ratio. So to perform inference we can use existing algorithms that solve minimum-cost flow problems. The *value* of the minimum-cost flow, which gives the number of MRP sources inferred, may be any integer between 0 and N.

Full inference: We use the Edmonds-Karp algorithm (Bang-Jensen and Gutin, 2007, Chapter 3), which iteratively searches for single paths in a *residual network* representation and does not get trapped in local optima. The Edmonds-Karp algorithm is often used to find maximum-value flow but can be used to optimise cost in our case of binary capacities.

The asymptotic time complexity of the Edmonds-Karp search relates to the number of vertices and arcs as $O(|V||A|^2)$. The number of vertices is closely related to the number of observations N; since we generate an arc for every possible transition between a pair of observations, $|A|$ may be on the order of N^2 in the worst case. Hence we add a constraint in constructing the arcs which is reasonable in many applications: we assert that transitions have an upper limit in the size of the time step, and so we do not create arcs for time separations above some threshold $\tau_{\max}$. The cardinality $|A|$ is then on the order of NB where B is the maximum number of observations within a time window of size $\tau_{\max}$ (and often $B << N$).

Greedy inference: If faster search is required at the cost of optimality, greedy search strategies are available. One such strategy is to repeatedly apply a minimum-cost path algorithm to the network, at each iteration taking the resulting path as an identified cluster and removing its vertices from the network before the next iteration. Since the graph is acyclic, finding a minimum-cost path can be performed very efficiently with order $O(|A|+|V|)$ at each iteration (Bang-Jensen and Gutin, 2007, Section 2.3.2); however there is no guarantee of optimality since the overall minimum-cost flow is not guaranteed to be composed of path flows of lowest individual cost. In our experiments

we will compare this greedy search empirically against the optimal search (using the same $\tau_{\max}$ threshold for both).

In the present work we primarily consider offline (batch) inference. However, online inference is possible within the same framework, in which new observations are received incrementally by updating the graph as observations arrive. The Edmonds-Karp search cannot be used on such a dynamic network, except by re-starting the search from scratch upon update. Alternative strategies such as those based on cycle-cancelling can be used to provide an updateable inference (Bang-Jensen and Gutin, 2007, Section 3.10.1). The speed of cycle-cancelling relative to Edmonds-Karp may depend on the nature of the data; we implemented both and found the cycle-cancelling relatively slow.

Thus far we have considered inference using a single set of MMRP model parameters, encoded as the costs in (6). It may be of value to evaluate the same data under different MMRP models, in situations where multiple types of MRP process (having different parameters) may be active. Multiple parametrisations cannot be represented together in a single flow network since they would assign conflicting costs to arcs. To accommodate incompatible costs is equivalent to the "multi-commodity" extension of the minimum-cost flow problem, which is NP-complete (Even et al., 1975). However, if the clutter noise model is held constant between two different MMRP inferences, then the two likelihood ratios calculated by (1) can be divided through to give a likelihood ratio between the two. This allows us to choose between possible MMRP models although not to combine them in a single clustering.

To summarise the MMRP inference described in this section: given a set of observations plus MRP process parameters and noise process parameters, one first represents the data as a flow network, with added source and sink nodes, and with costs representing component likelihoods (Section 3.1). One then applies a minimum-cost flow algorithm to the network, such as Edmonds-Karp (which we use for "full inference") or a suboptimal greedy search. Each (s,t)-path in the resulting minimum-cost flow represents a single cluster (a single MRP sequence) in the maximum-likelihood result, while the nodes which receive no flow represent data to be labelled as noise.

4. Experiments

We have described a multiple Markov renewal process (MMRP) inference technique which takes an MRP model, an iid clutter noise model and a set of timestamped data points, and finds a maximum-likelihood partition of the data into zero or more MRP sequences plus clutter noise. In the following, we will first illustrate its properties with a synthetic experiment designed to explore robustness (Section 4.2). We then apply MMRP inference in two experiments based on applications to audio tracking tasks: a synthetic experiment based on a well-known test of auditory "streaming" (Section 4.3), and an experiment to track multiple singing birds in an audio mixture (Section 4.4). However, we must first consider how to evaluate algorithm outputs.

4.1 Evaluation Measures

To judge the empirical performance of our inference procedure, we must determine whether it can correctly separate signal from noise, and whether it can correctly separate each individual MRP sequence into its own stream. MMRP inference can be considered as a clustering task and could be evaluated accordingly. However, the noise cluster is qualitatively different from the MRP clusters,

and the transitions within MRP sequences are the latent features of primary interest, so we will focus our evaluation measures on signal/noise separation and transitions.

In the following our statistics will be based on the standard F-measure (Witten and Frank, 2005, Chapter 5), which summarises precision and recall as follows:

$$\begin{aligned} F &= 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \\ &= \frac{2t_+}{(2t_+ + f_- + f_+)} \end{aligned} \tag{7}$$

where t_+ is the number of true positive detections, f_+ the number of false positive detections (noise data labelled as signal), and f_- the number of false negative detections (signal data labelled as noise).

However, the task for which our MMRP inference is designed is not an ordinary classification task: the signal/noise label for each ground-truth datum can be treated as a class label to be inferred, but the individual signal streams to be recovered do not have labels. To quantify performance we use the F-measure in two ways. The first (which we denote F_{SN}) evaluates the signal/noise classification performance without considering the clustering. The second (which we denote F_{sigtrans}) evaluates the performance at recovering the *pairwise transitions* that are found in the ground-truth signals, that is, the arcs in the true dependency graph underlying the data. In order to make the two measures relatively independent, we measure F_{sigtrans} only on event pairs that have been correctly classified as signal, since otherwise false-positive noise events could have a strong influence on both (see Figure 4). Thus, in the following we use F_{SN} to measure signal/noise separation and F_{sigtrans} to determine whether inference is correctly recovering separate streams.

4.2 Synthetic Experiment I: MMRP Generated Data

We designed a synthetic experiment to generate data under the MMRP model described in previous sections, with user-specified parameters including birth intensity, death probability, and clutter noise intensity. The test was conducted with state X defined on a discrete alphabet and continuous time T, and the transition network among states and times was algorithmically generated as follows: for each state, a random subset of possible next states was selected, with the number of out-arcs dependent on the user-specified sparsity of the transition model. The weights of the out-arcs were sampled as a multinomial distribution (sampled from a symmetric Dirichlet distribution with $\alpha = 1$). Each out-arc was also associated with a density on the size of the time gap to the next event, taking a log-normal distribution with a mean randomly sampled from a log-normal parent distribution. To create an observation set, a set of birth events and clutter events were sampled independently from their Poisson distributions, and then each birth event was used as the starting point to sample a single $\{(X,T)\}$ sequence using the death probability and the transition network. The intensities for the birth process and the noise process were uniform across the alphabet of states, and so in the following we parametrise them by their intensity along the time axis only. Similarly, in the present experiment we held death probability as uniform across state. Observation sets and noise events were sampled within a time window of fixed duration. We used a signal-to-noise ratio (SNR) parameter to control the intensity of noise observations (λ_c) in relation to that of signal observations:

$$\lambda_c = \frac{\lambda_b}{p_d} \cdot \text{SNR}.$$

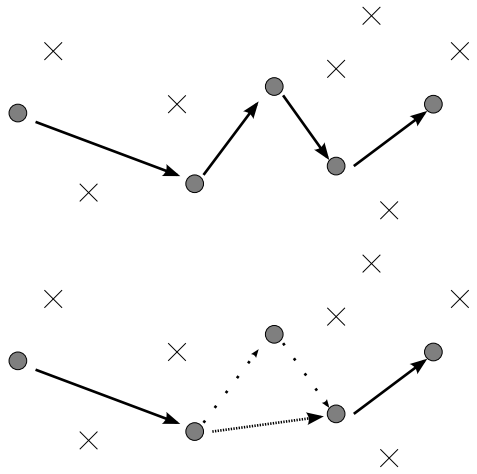

Figure 4: Illustration of errors reflected in F_{sigtrans}. The upper diagram shows a hypothetical ground-truth transition through a sequence of five observations (circles) accompanied by clutter noise (crosses). The lower diagram shows what would happen if inference missed one of those observations out of the chain, resulting in one false-positive (dashed arrow) for a transition that does not exist in the ground-truth. If the "skipped" observation is labelled as noise then the two false-negative arcs (dotted arrows) would not be considered in F_{sigtrans} (its omission would already be represented in the F_{SN} statistic): considering the false-positive and the two true-positives and applying (7), the F_{sigtrans} value then is $\frac{2}{3}$. If the "skipped" observation is labelled as signal then the false-negative arcs are also considered and F_{sigtrans} is $\frac{2}{5}$.

The factor of p_d appears as well as the birth intensity (λ_b) because the SNR relates to the count of all signal observations (not just births), and for a fixed death probability we have a geometric distribution over the number of detections per birth with expected value $1/p_d$.

To evaluate performance of our inference applied to such data, we repeatedly generated observation sets as described, and ran both the greedy and full inference algorithms on the data. Unless otherwise stated, for all synthesis runs we used the following parameters: alphabet size 10, SNR 0 dB, birth intensity 0.2 per second, death probability 0.1, observation duration 40 seconds. (We also ran tests with alphabet size 100, obtaining very similar results, and so we have not included those.) In each case a transition network was generated with a sparsity of 50%, and the parent distribution for the transition time densities was a log-normal centred on 1 second with a standard deviation of 1; distributions for each transition arc were log-normal with mean sampled from that parent distribution and a standard deviation of 0.1.

The chosen setting for death probability implies an expected chain length of 10 emissions for a single MRP source. Together with the the birth intensity and SNR this implies that a typical generated observation set would consist of 160 observations, half being signal and half noise. Empirically, each of our observation sets had a mean polyphony (the number of simultaneously active sources) varying from around 0.1 to 4.5, with substantial variation in the polyphony during the course of each (generally 0–10).

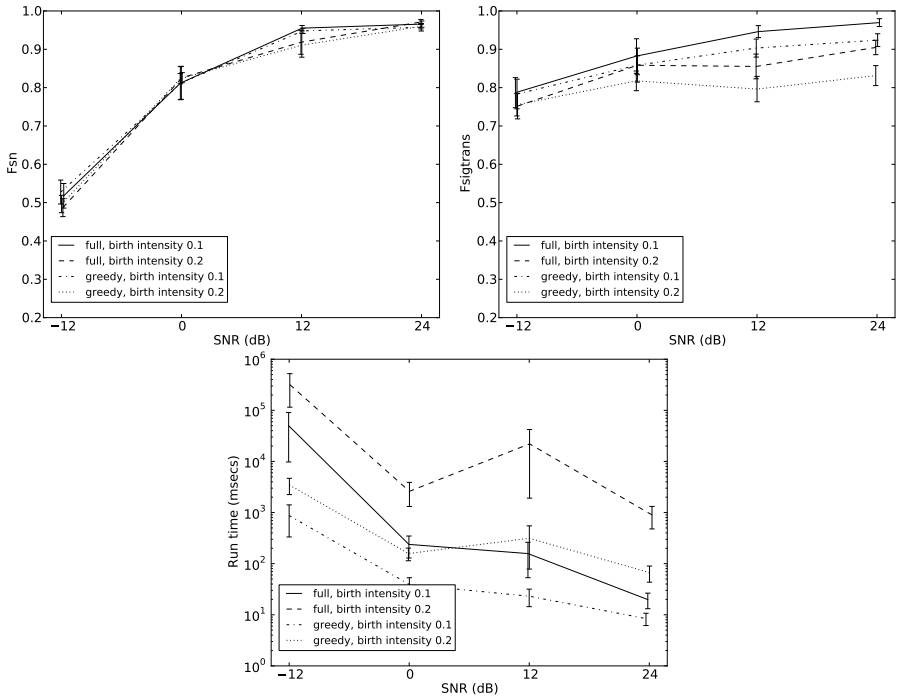

Figure 5: Performance of the full and greedy inference algorithms with varying SNR. Plots show the F-measure for signal/noise separation (F_{SN}, upper left) and signal transitions (F_{sigtrans}, upper right), as well as the measured runtime (lower). Means and confidence intervals are shown, taken over 20 independent runs. States are defined on an alphabet of size 10, and transition sparsity set at 10% or 50% (alternating runs). In this plot, we also compare birth intensities (λ_b) of 0.1 and 0.2.

Results were evaluated using the F_{SN} and F_{sigtrans} measures described in Section 4.1. For our first test, the algorithms were supplied with the true model parameters p_b, f_X, p_d, p_c to calculate (6). Figure 5 shows how performance varies with SNR. In this synthetic experiment, the separation of signal and noise (measured by F_{SN}) is strong at high SNRs and falls off a little as the SNR approaches zero. With very adverse SNR (-12 dB) performance drops off noticeably. This is the case for both the full and greedy algorithms. The F_{sigtrans} measure shows a milder decline with SNR, but also notable differences between the full and greedy inference, with a consistent benefit in accurate recovery of transitions if the full inference is used. We also show the measured runtime in Figure 5: the increased accuracy of full inference in recovering signal transitions comes at a cost of increased runtime, especially under adverse SNRs (because of the larger number of noise events generated).

In order to study the sensitivity of inference to misspecified or unknown parameters, we also ran the same test but with systematically misspecified parameters for inference. This is important not only because we seek a robust algorithm, but also because parameters such as the birth density and death probability together imply approximate expectations about the level of polyphony in the signal. Since one advantage of our approach is that it infers an arbitrary number of signal sequences

in the data, we are interested to determine whether the correctness of these parameters is crucial for successful inference.

Results are shown in Figure 6. We see that both algorithms (greedy and full) are robust to poor estimation of the birth density, death probability and SNR. The advantage of full over greedy inference is maintained at around five percentage points in F_{sigtrans} through most of these varied conditions.

We also tested misspecification of transition probabilities. In order to create a controllable amount of misspecification we implemented a stochastic degradation of the transition density information: given a degradation parameter $d \in [0,1]$, for every state in the transition table with probability d we resampled the set of out-arc weights; and then for each out-arc separately, with probability d we resampled the mean of its log-normal density over time. This gave a stochastic corruption of the transition probability which could range from moderate to very strong. Results (Figure 6) show that misspecification of the transition probabilities exhibits a strong effect compared against the other variations: the algorithms are relatively robust up to around 10% degradation, but F_{sigtrans} in particular falls dramatically when the transition probabilities become badly corrupted. This reflects the fact that the transition probabilities encode the key structural distinction between signal and noise, *and* the key information that one could use to disambiguate two co-occurring signal streams.

We also investigated how inference may degrade when conditions fail to match some of the assumptions of the model: in many applications there may be missed detections, or noise may not be truly independent but exhibit correlations with the signal. Figure 7 shows the performance of inference as these issues are progressively introduced into the data. Missed detections were simulated by omitting observations at random; noise correlations were simulated by selecting a controllable fraction of the noise observations, and modifying those noise observations to have the same state and very similar time position as a randomly-selected signal datum. The algorithms appear moderately robust to such problems: F_{SN} progressively deteriorates as the proportion of issues increases, but F_{sigtrans} exhibits notable strong declines down toward chance performance with strong degradation. However, the algorithms (both greedy and full) are robust to moderate violations of the assumptions.

However, we also noticed that correlated noise led to a significant increase in algorithm runtime. This is plotted in Figure 8, showing that correlated noise beyond 25% can lead to run-times which are orders of magnitude longer, even though the data under consideration has the same number of observations and the same ratio of signal and noise observations. This occurs in the full algorithm, and also in the greedy algorithm though with less severity. We propose that the reason for this is that when the flow network includes many search paths which are extremely similar—for example differing only in the choice of a particular signal datum or a competing noise datum, both at the same location and thus with the same likelihood—then this can create a combinatorial explosion of paths that must be explicitly searched. Standard network search algorithms use branch-and-bound-type optimisation to avoid explicit recursion into many of the candidate paths (Papadimitriou and Steiglitz, 2000). This optimisation ignores search paths which have no possibility of improving on a locally cached result, and so speeds up search while still finding the global optimum. The effect of this optimisation is weakened when many paths have very similar costs, and search time increases in practice even though the formal size of the problem is no different.

To summarise the observations made in this experiment, we find that MMRP inference is generally quite robust to variations in conditions and model parameters. The greedy algorithm achieves performance close to that of the full algorithm in most cases, although the full algorithm consistently

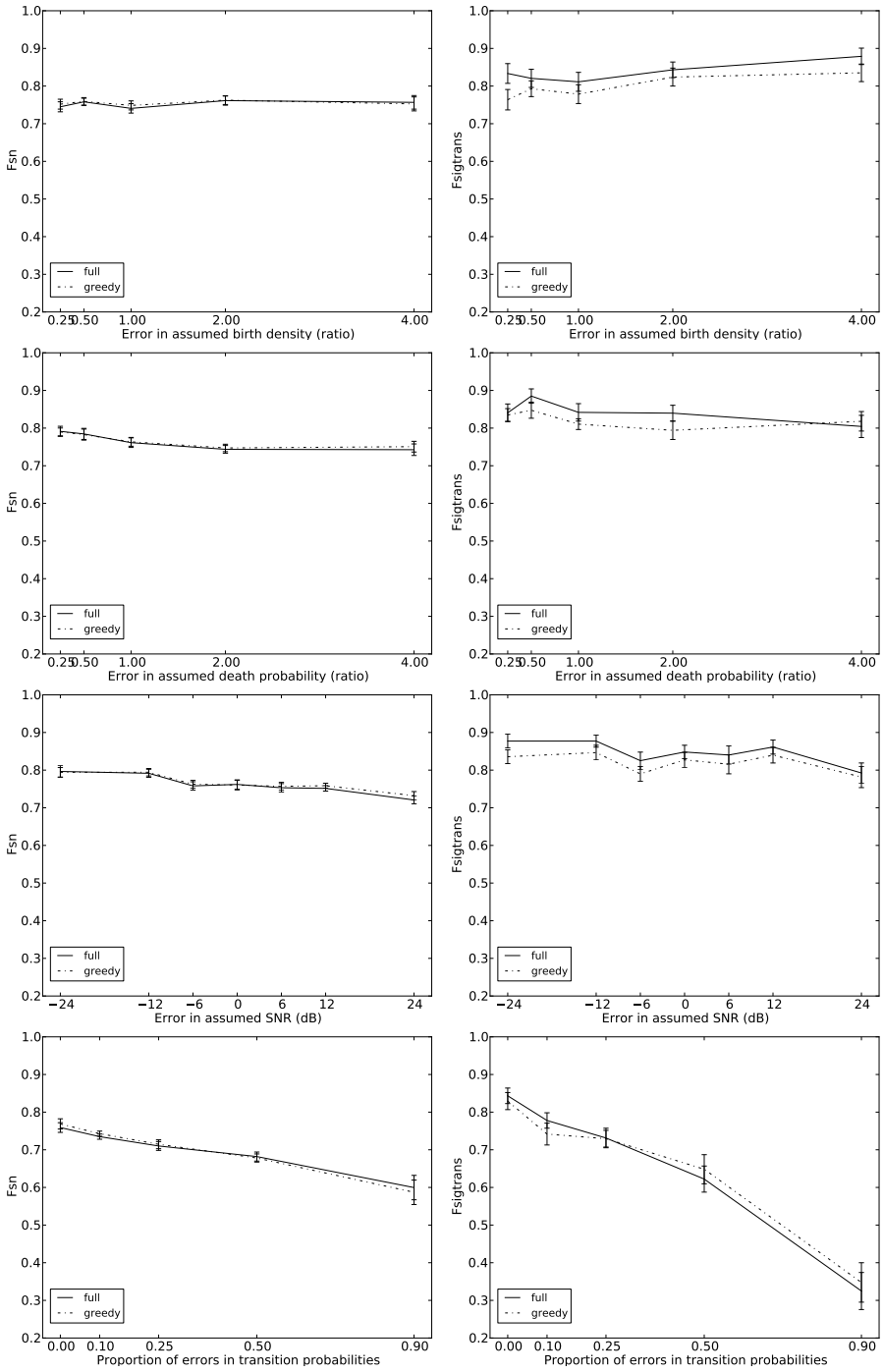

Figure 6: Sensitivity of inference to misspecified parameters. Plots are as in Figure 5 but showing how performance varies with mismatch between the true and specified parameters for the birth density, death probability, SNR, and transition density. SNR is fixed at 12 dB for all plots, except in the SNR plot for which we average over runs with true SNR $\in 0, 6, 12$ dB to confirm that SNR sensitivity does not vary strongly with SNR.

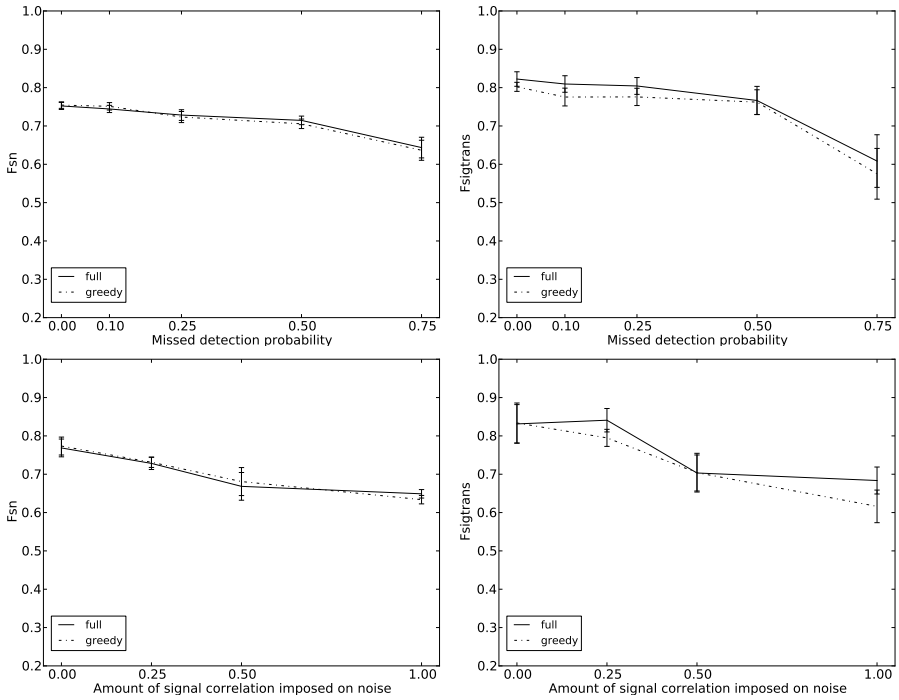

Figure 7: Sensitivity of inference to missed data and correlated noise. Plots are as in Figure 6 but showing how performance varies when some detections are missed, and when noise is not independent but correlated with signal.

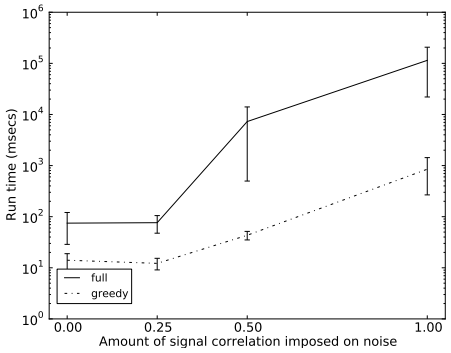

Figure 8: Algorithm run-time for the correlated-noise test of Figure 7.

achieves stronger F_{sigtrans} in all but some strongly adverse conditions. This shows empirically that the greedy algorithm has a tendency to find local optima, and the results suggest these local optima reflect not so much issues in signal/noise discrimination but "crossed wires" in MRP sequences. The most critical parameter for successful MMRP inference appears to be the transition probability structure rather than assumptions about birth/death probabilities, which accords with our intuition that the Markov structure of the sequences is the source of the discriminative power. As well as the

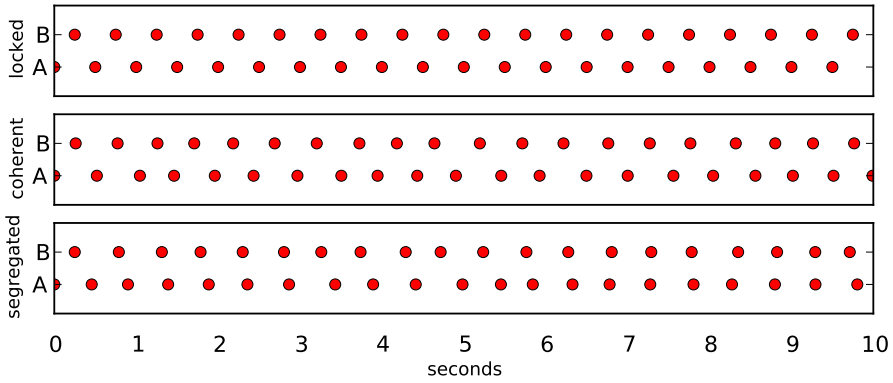

Figure 9: Examples of sequences generated by strict locked ABABAB repetition (top), and by similar generators but with time offsets affected by process noise reflecting either coherent (ABABAB, middle) or segregated (A_A_A_ and _B_B_B, bottom) dependency structure.

transition structure, another important consideration is signal/noise correlation, which in the present experiment can lead to impaired F_{sigtrans} results as well as notably increased computation time.

4.3 Synthetic Experiment II: Auditory Streaming

To illustrate the relevance of our algorithm to the multi-source tracking required in tasks such as machine listening (or computational auditory scene analysis), we next consider a synthetic experiment inspired by the classic "audio streaming" experiments used to explore human auditory grouping of sound sequences (Winkler et al., 2012). In this context the MRP model might be taken to represent not necessarily a model of how event sequences were generated, but a compact model of expectations about event sequences that can be used for computational tasks such as auditory streaming.

A strictly alternating sequence of the form ABABAB…, where A and B are different tones (Figure 9, top row), can be interpreted either as a single alternating sequence (the "coherent" interpretation) or as a simultaneous but out-of-phase pair of constant sequences (the "segregated" interpretation). Various factors can lead an observer to prefer one interpretation or the other; here we focus on the case where drift in the timing of the events makes one or the other model more likely (Cusack and Roberts, 2000, Experiment 2). If the sequences drift such that the phase of the As and Bs remain in constant relationship (Figure 9, second row), this is consistent with a "coherent" alternating generator, though may by chance be generated by a "segregated" pair of generators. If the sequences drift such that the phase relationship is not maintained (third row), then this is inconsistent with the "coherent" model but consistent with the "segregated" model. We can generate data with these properties and observe how the MMRP inference behaves under the assumptions of each model.

For our synthetic experiment we defined two separate MRP transition models (one "coherent" and one "segregated") to emit values in a one-dimensional state space $\mathcal{X} \in \mathbb{R}$. Each model was

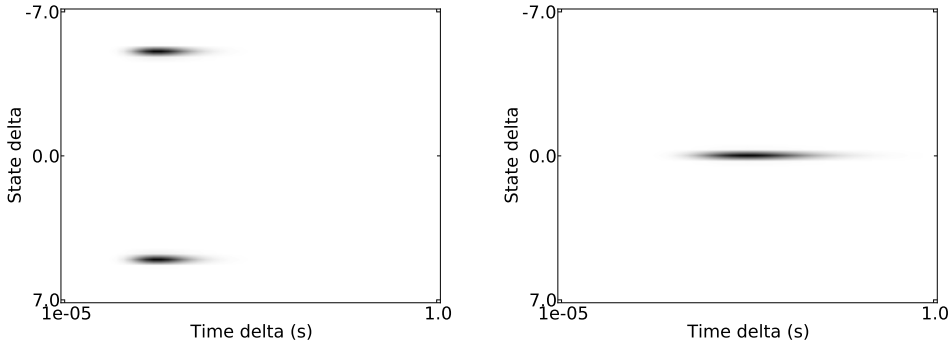

Figure 10: MRP transition probability densities for the two synthetic models: coherent (left) and segregated (right). The coherent model uses two Gaussians centred on 0.25 sec and ± 5 in state, while the segregated model uses one Gaussian centred on 0.5 sec and 0 in state. For each Gaussian, the standard deviation is 0.1 in state and 0.2 in log-time-delta.

specified by a Gaussian mixture probability distribution defined on state-delta and log-time-delta:

$$P(\tau_{n+1} \leq t, X_{n+1} = j \mid X_n = i) = f(X_{n+1} - X_n, \log \tau_{n+1}) .$$

Figure 10 illustrates the transition models. Time differences here are modelled as log-Gaussian to reflect a simple yet perceptually plausible model for lower-bounded time intervals. The variance of the Gaussian components leads to process noise, and the two models tend to output different sequences in general. We also define a "locked" model for generation only, which generates a strict ABABAB sequence with no process noise. Its emissions could in principle be explained by either of the two other models.

These models served two roles in our experiment, to synthesise data and to analyse it. For synthesis, we generated four simultaneous sequences each with a random offset in state space, and we also added iid Poisson clutter noise in the same region of state space, whose intensity is held constant within each run to create a given SNR. In the case of the segregated model, each generator was a pair of such models, independent except for the initial phase and offset, generating As and Bs as was done in Figure 10. In this experiment we did not use probabilistic births or deaths during synthesis, instead generating a fixed polyphony lasting the whole of the excerpt. For MMRP inference we used fixed parameters derived from the SNR value and an arbitrary death probability of 0.033. The following relationships show how to derive the birth and clutter likelihood parameters from the SNR value expressed as a ratio:

$$p_b = \frac{\mathrm{SNR} \cdot p_d}{1 + \mathrm{SNR}} , \qquad p_c = \frac{1}{1 + \mathrm{SNR}} . \tag{8}$$

The factor of p_d enters into the calculation of p_b for the reasons described in Section 4.2.

The first column of Figure 11 shows the results of generating data under the locked, coherent and segregated models, with two generated sequences present in each case. The second column

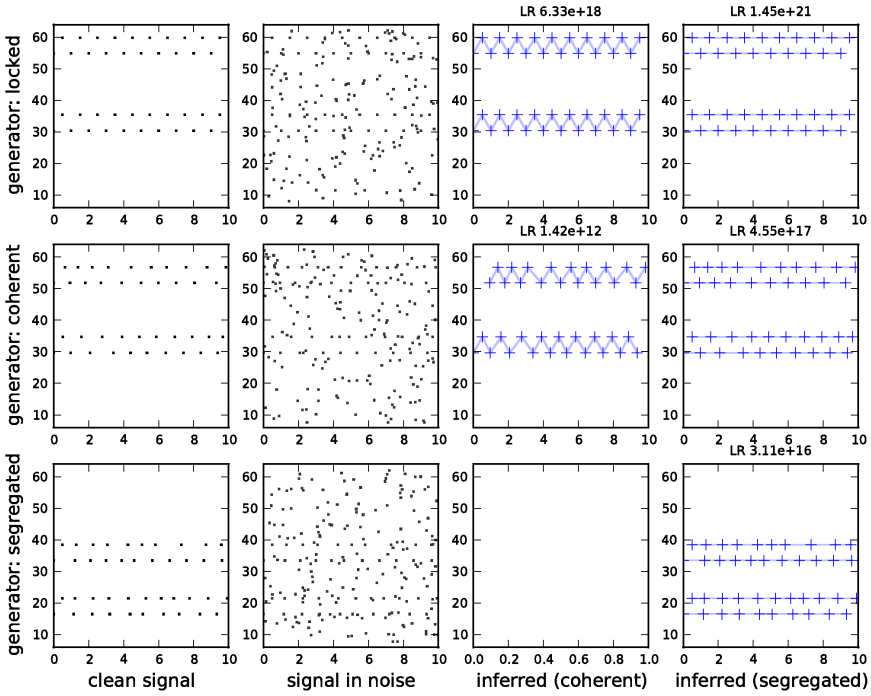

Figure 11: Results of generating observations under the locked, coherent or segregated model (in each row), and then analysing them using the coherent model or the segregated model (final two columns). Note that we have selected an example with clear pitch separation between streams, for visual clarity: in general, and in our tests with four streams, sequences often overlap in pitch and are not so obviously separable.

shows the sequences with added clutter noise at an SNR of -12 dB. The final two columns show the maximum-likelihood signal sequences inferred under the coherent and the segregated model. The MMRP inference typically extracts clear traces corresponding to the ground-truth signals, even in strongly adverse SNR. It is visually evident in the first column that the generated sequences in the middle row have some drift in their rate, but stay in order, while the As and Bs in the bottom row drift relative to each other and do not maintain order. This leads to unlikely emission sequences as judged by the coherent model, and so the coherent model finds the maximum-likelihood solution to be that with no sequences (the blank plot in the figure). Inference using the segregated model extracts traces in all three cases, since the phase-locked drift of the coherent model is not unlikely under the segregated model.

To evaluate MMRP inference in this case, we ran this process multiple times, varying the SNR level and whether the true SNR was known to the algorithm. When not known, the SNR estimate was arbitrarily held fixed at 0 dB. For each setting we conducted 20 runs and recorded the F_{SN} and F_{sigtrans} statistics. Figure 12 illustrates the results, showing broad consistency with the previous experiment. Recovery performance is very strong in all but the most adverse conditions, in most cases

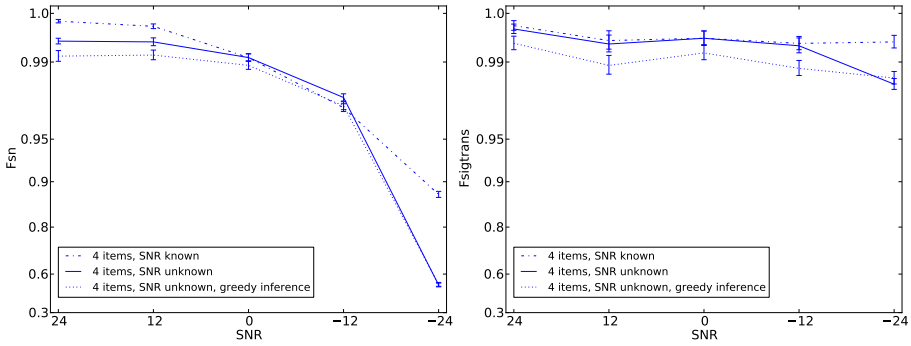

Figure 12: F-measure for signal/noise separation (F_{SN}) and transitions (F_{sigtrans}). The ground truth in each case is a combination of four ABABAB streams, generated via the coherent or segregated cases (20 runs of each type). Means and standard errors are shown; the vertical axis is reverse-log-scaled so that the results very near 1.0 can be distinguished.

being well above 0.95. For these particular scenarios, signal/noise discrimination is impaired under the strongest condition tested (SNR -24 dB), but under other conditions the recovery is good, and F_{sigtrans} remains strong in all runs. As in the previous experiment, full inference shows a consistent advantage over the greedy inference, though this tails off at -24 dB SNR. In this test, knowledge of the true SNR gives a further boost in the performance of MMRP inference.

4.4 Birdsong Audio Experiment

Many natural sound sources produce signals with structured patterns of emissions and silence, for example birdsong or footsteps. As in the previous experiment inspired by auditory streaming, if we model these natural sound sources with an MRP then our inference procedure should be able to separate multiple simultaneous "streams" of emissions. In the following experiment we studied the ability of our inference to perform this separation in data derived from audio signals containing multiple instances of a species of bird common in many European countries, the Common Chiffchaff (Salomon and Hemim, 1992). Chiffchaff song consists of sequences of typical length 8–20 "syllables". Each syllable is a pitched note consisting of a downward chirp to a briefly-held tone in the region of 5–8 kHz. Syllables are separated by around 0.2–0.3 seconds. The exact note sequence has not to our knowledge been studied in detail; it appears to exhibit only short-range dependency, and is thus amenable to analysis under Markovian assumptions.

4.4.1 Data Preparation

To aid reproducibility, we used recordings from the Xeno Canto database of publicly-available bird recordings.[1] We located 25 recordings of song of the Chiffchaff (species *Phylloscopus collybita*) recorded in Europe (excluding any recordings marked as having "deviant" song or uncertain species identity; also excluding *calls* which are different from *song* in sound and function). The recordings used are listed in Table 1. We converted the recordings to 44.1 kHz mono wave files, high-pass filtered them at 2 kHz, and normalised the amplitude of each file. File durations varied from 8.5

1. Available at `http://www.xeno-canto.org/europe`.

ID	Country	ID	Country
XC103404	pl	XC48263	no
XC25760	dn	XC48383	de
XC26762	se	XC54052	it
XC28027	de	XC55168	fr
XC29706	se	XC56298	de
XC31881	nl	XC56410	ru
XC32011	nl	XC57168	fr
XC32094	no	XC65140	es
XC35097	es	XC77394	dk
XC35974	cz	XC77442	se
XC36603	cz	XC97737	uk
XC36902	nl	XC99469	pl
XC46524	nl		

Table 1: Chiffchaff audio samples used in our data set, giving the Xeno Canto ID and the country code. Each recording can be accessed via a URL such as `http://www.xeno-canto.org/XC103404`, and the data set is also archived at `http://archive.org/details/chiffchaff25`.

seconds to many minutes, so to create a set of independent audio samples which could be mixed together to create mixtures with overlapping bouts of song, audio files were each trimmed automatically to their highest-amplitude 8.5-second segment. Source code for these preprocessing steps are published along with the full code.[2]

Each audio file was analysed separately to create training data; during testing, audio files were digitally mixed in groups of one to five files.

In order to convert an audio file into a sequence of events amenable to MMRP inference, we used spectro-temporal cross-correlation to detect individual syllables of song, as used by Osiejuk (2000). We designed a spectrotemporal template using a Gaussian mixture (GM) to represent the main characteristics of a single Chiffchaff syllable, a downward chirp to a briefly-held note (Figure 13). The GM was modelled on a Chiffchaff recording from Xeno Canto which was not included in our main data set (ID number XC48101). Then to analyse an audio file we converted the file into a spectrogram representation (512 samples per frame, 50% overlap between frames, Hann window), and converted the GM to a sampled grid template with the same time-frequency granularity as the spectrogram, before sliding the grid template along the time axis and along the frequency axis (between 3–8 kHz), evaluating the correlation between the template and spectrogram at each location. Correlation values were treated as detections if they were local peaks with value greater than a threshold correlation of 0.8.

Such cross-correlation detection applied to an audio file produces a set of observations, each having a time and frequency offset and a correlation strength (Figure 14). It typically contains one detection for every Chiffchaff syllable, with occasional doubled detections and spurious noise detections. When applied to mixtures of audio, this produces data appropriate for MMRP inference.

2. Available at `https://code.soundsoftware.ac.uk/projects/markovrenewal`.

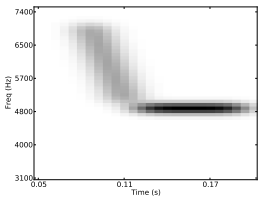

Figure 13: Template used for spectro-temporal cross-correlation detection. The downward and horizontal bars have equal total weight; the latter appears darker because shorter. The template is a manually-constructed Gaussian mixture model having 40 components. It is then used for signal pre-processing both during training and testing.

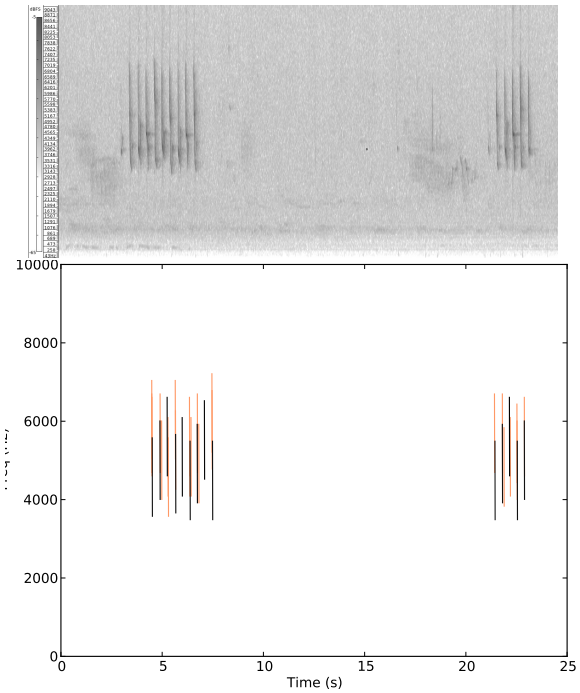

Figure 14: Example of cross-correlation detection: excerpt of spectrogram shown (top), and the corresponding detections (bottom). In the lower image, bold lines represent detections treated as "signal" in the filtering used for training, while the fainter lines represent detections used to train the noise model.

Note that the noise detections often have relatively strong signal correlations, as seen in Figure 14. From our first experiment (Section 4.2) we expect this to have an effect primarily on runtime, though it may also be an issue for performance. We will consider this in light of the results.

In order to derive a Gaussian mixture model (GMM) transition probability model from monophonic Chiffchaff training data, for each audio file in a training set we filtered the observations automatically to keep only the single strongest detection within any 0.2 second window. This time

limit corresponds to the lower limit on the rate of song syllables; such filtering is only appropriate for monophonic training sequences and was not applied to the audio mixtures used for testing. The filtered sequences were then used to train a 10-component GMM with full covariance, defined on the vector space having the following four dimensions:

- log(frequency) of syllable one
- log(frequency) of syllable two
- log(magnitude ratio between syllables)
- log(time separation between syllables)

This GMM then served as the transition probability distribution to be used in inference. We also trained a separate GMM to create a noise model, taking the set of observations that had been discarded in the above filtering step and training a 10-component GMM with full covariance to fit an iid distribution to the one-dimensional log(frequency) data for the noise observations.

4.4.2 INFERENCE FROM AUDIO MIXTURES

In order to test whether the MMRP approach could recover syllable sequences from audio mixtures, we performed an experiment using five-fold cross-validation. For each fold we used 20 audio files for training, and then with the remaining five audio files we created audio mixtures of up to five signals, testing recovery in each case. For each mixture file, we applied spectro-temporal cross-correlation as described above, then performed both full and greedy inference using the empirically-derived signal and noise GMMs to provide densities/intensities for transition and clutter. We used fixed probabilities for p_d, inferred from the empirical average sequence length in the training data, and p_b, inferred using (8) with a default SNR estimate of 0 dB.

To provide a low-complexity baseline showing the recovery quality using only the marginal properties of the signal and noise, we also created a simple baseline system which treated both signal and noise as iid one-dimensional log(frequency) data, using maximum likelihood to label each observation as either signal or noise. The baseline system then clustered together observations that were identified as signal and were separated by less than 0.7 seconds (a duration chosen to reflect the 0.2–0.3 sec gap sizes in Chiffchaff song, with tolerance for occasional missed detections).

We tested each of these approaches using mixtures of one, two, three, four or five of the test recordings. As in the previous experiment, we measured the F_{SN} statistic to evaluate signal/noise separation, and the F_{sigtrans} statistic to evaluate the performance at recovering separate sequences.

Results are shown in Figure 15. Broad outcomes are similar to those of the previous experiments. Signal/noise discrimination is very similar between full and greedy inference, and remains steady as the polyphony increases. Again, though, the full inference shows a general advantage over greedy inference in the correct recovery of transitions. This pattern is consistent across all the polyphony levels tested, except the case of just one bird, in which there is no occasion for the greedy method to make mistakes by crossing one bird's track with another, so it achieves the same performance as full inference.

We also note that all the MMRP inference runs exhibit a significant and very strong improvement over the baseline, both for F_{SN} and F_{sigtrans}. This shows that the transition information learnt from the training data is indeed pertinent in this application example.

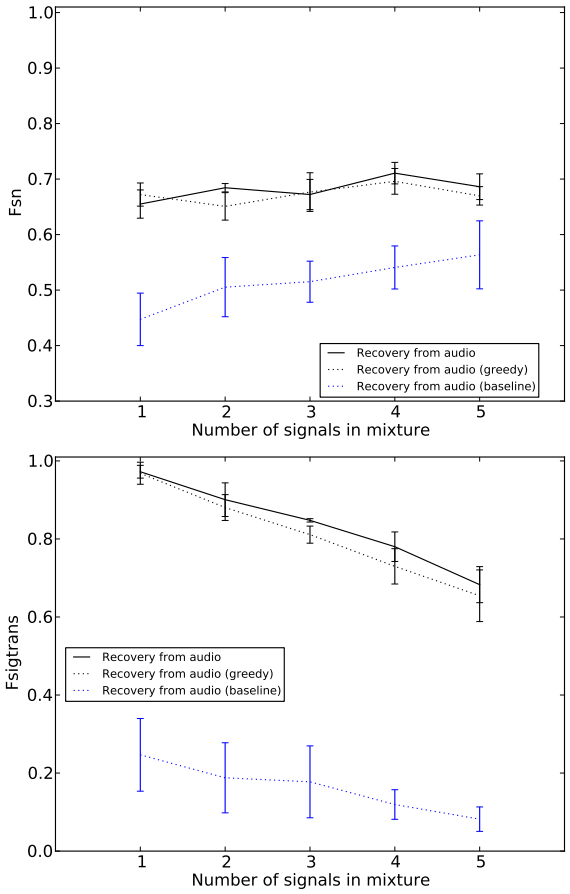

Figure 15: The F_{SN} and F_{sigtrans} evaluation measures for the Chiffchaff audio analyses. Means and standard errors are shown taken over the five folds of the cross-validation.

However, in this experiment the levels of F_{SN} obtained are much lower than in the purely synthetic experiments. A likely reason for this is the front-end we use to detect events in audio: the simple cross-correlation technique may extract slightly different events when applied to a mixture as opposed to the monophonic recordings. Another potential issue is whether the fitted GMMs generalise well from training to test data. In order to explore these factors we ran the same test with full inference, but with some variation on the front-end analysis:

Ideal recovery: To simulate ideal-case recovery, instead of using the audio mixture we simply pooled the signal and noise observations that had been derived from the test set's individual mono analysis, then performed MMRP inference as in the audio recovery case.

Ideal recovery, synthetic noise: To simulate ideal recovery but with more adverse noise conditions, we proceeded as in the ideal case, but also added extra clutter noise at 0 dB. To do this, we created a copy of every observation in the test set, but assigned it an independent random time position, thus creating noise with the same frequency distribution as the true signal but uncorrelated in time.

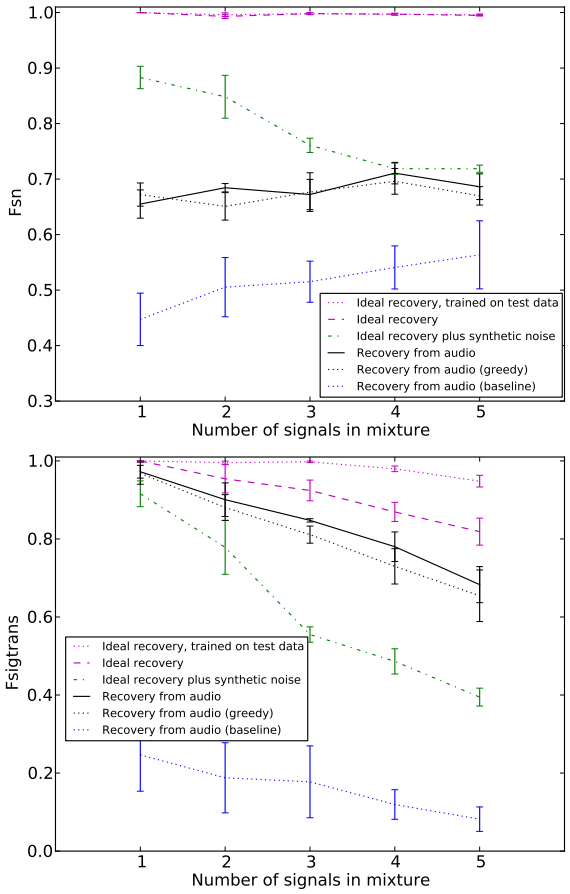

Figure 16: As Figure 15, with ideal-recovery results superimposed.

Ideal recovery, tested on training set: To measure an "upper limit" on performance and probe the generalisation capability of the algorithm, we proceeded as in the ideal case, but used GMMs trained on the actual test files to be analysed rather than on the separate training data.

Results, superimposed with the results from the standard detection approach, are shown in Figure 16. Very strong performance is achieved in the noiseless "ideal recovery" cases, achieving results similar to those in the synthetic experiments. The small size of the difference between training on the test data and on the training data (in particular for F_{SN}) indicates that the algorithm can generalise across the data used in our experiment. However, the F_{sigtrans} measure shows a notable boost when trained on test data, which may reflect some degree of heterogeneity in transition distributions in the recordings. Resolving the similarity of sequences in birds across different geographical locations is of interest to bioacoustics researchers (see, e.g., Mahler and Gil 2009), but at present there are not the large annotated databases that would facilitate such analyses of similarity for a single species.

When synthetic noise is added to the ideal-recovery case, performance is reduced considerably. The F_{SN} measure approaches that of the more realistic case, while F_{sigtrans} is even more strongly impaired. This indicates that the detection front-end in the more realistic case is indeed a bottleneck

for performance, but its impact on F_{sigtrans} is not as severe as on F_{SN}. Note that our synthetic noise is temporally decorrelated from the signal, whereas the noise present in recovery from audio mixtures shows quite strong correlations (Figure 14). Our results indicate that in this experiment the noise correlation is not a major impediment to recovery from audio, since the uncorrelated noise induces consistently worse performance in F_{sigtrans}, and a similar level of performance in F_{SN} at high polyphony.

Taken together, these results show that the practical task of retrieving detections from audio mixtures has a significant effect on algorithm performance, but that MMRP inference still performs strongly in simultaneously inferring signal/noise discrimination and clustering signals into tracks. We are exploring more sophisticated bird syllable detection to improve on these results (Stowell et al., 2013). As in the synthetic experiments, in the present experiment the full MMRP inference shows a consistent F_{sigtrans} benefit over the greedy inference, although this must be balanced against the additional runtime cost.

5. Conclusions

In this paper we have investigated the problem of segregating timestamped data originating in multiple point processes plus clutter noise. We developed an approach to inferring structure in data produced by a mixture of an unknown number of similar Markov renewal processes (MRPs) plus independent clutter noise. The inference simultaneously distinguishes signal from noise as well as clustering signal observations into separate source streams, by solving a network flow problem isomorphic to the MMRP mixture problem. Our method is general and has very few free parameters.

In experiments we have shown that inference can perform very well even under high noise conditions (as far as -12 dB SNR, depending on application). The full optimal MMRP inference incurs a higher complexity than a greedy approach, but generally achieves a more accurate recovery of the event-to-event transitions present in the data. In a synthetic experiment, we explored the robustness of inference, and found that good performance is possible despite misspecification of parameters such as the birth density and noise level. Inaccurate specification of the MRP transition probability structure can impair performance, as can correlated noise, though inference is still robust to mild corruptions of these types. Correlated noise can also incur high run-times because of its effect on the graph search.

To illustrate applications of the technique, we then conducted two experiments related to audio recognition tasks. In an experiment based on the "auditory streaming" paradigm, we showed that MMRP inference can recover polyphonic event streams from noisy observations, applying different MRP generative models to implement different expectations about the streams to be recovered. Then in an experiment on birdsong audio data we showed strong performance, albeit with a dependence on the quality of the underlying representation to recover events from audio data.

The inference in the present work is limited to models without hidden state and with only single-order Markov dependencies. These limitations arise from the combinatorial ambiguity in MMRP mixtures (unlike ordinary Markov models) over which is the immediate predecessor for each observation. Future work will aim to find techniques to broaden the class of models that can be treated in this way.

Reproducible research: Python source code for our implementation and our experiments is freely available online.[2]

Acknowledgments

DS and MP are supported by EPSRC Leadership Fellowship EP/G007144/1.

References

M. F. Arlitt and C. L. Williamson. Internet web servers: Workload characterization and performance implications. *IEEE/ACM Transactions on Networking*, 5(5):631–645, 1997. doi: 10.1109/90.649565.

J. Bang-Jensen and G. Gutin. *Digraphs: Theory, Algorithms and Applications*. Springer Verlag, London, 1st edition, 2007. URL http://www.cs.rhul.ac.uk/books/dbook/.

J. P. Barker, M. P. Cooke, and D. P. W. Ellis. Decoding speech in the presence of other sources. *Speech Communication*, 45(1):5–25, 2005. doi: 10.1016/j.specom.2004.05.002.

O. Bobrowski, R. Meir, and Y. C. Eldar. Bayesian filtering in spiking neural networks: Noise, adaptation, and multisensory integration. *Neural Computation*, 21(5):1277–1320, 2009. doi: 10.1162/neco.2008.01-08-692.

R. Cusack and B. Roberts. Effects of differences in timbre on sequential grouping. *Attention, Perception, & Psychophysics*, 62(5):1112–1120, 2000. doi: 10.3758/BF03212092.

S. Even, A. Itai, and A. Shamir. On the complexity of time table and multi-commodity flow problems. In *16th Annual Symposium on Foundations of Computer Science*, pages 184–193. IEEE, 1975. doi: 10.1109/SFCS.1975.21.

J. R. Glass. A probabilistic framework for segment-based speech recognition. *Computer Speech & Language*, 17(2):137–152, 2003. doi: 10.1016/S0885-2308(03)00006-8.

E. F. Keane, D. A. Ludovici, R. P. Eatough, M. Kramer, A. G. Lyne, M. A. McLaughlin, and B. W. Stappers. Further searches for rotating radio transients in the Parkes Multi-beam Pulsar Survey. *Monthly Notices of the Royal Astronomical Society*, 401(2):1057–1068, 2010. doi: 10.1111/j.1365-2966.2009.15693.x.

M. Koivisto and K. Sood. Exact Bayesian structure discovery in Bayesian networks. *The Journal of Machine Learning Research*, 5:549–573, 2004. URL http://jmlr.csail.mit.edu/papers/v5/koivisto04a.html.

M. Lagrange, L. G. Martins, J. Murdoch, and G. Tzanetakis. Normalized cuts for predominant melodic source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 16 (2):278–290, 2008.

B. Mahler and Diego Gil. *The Evolution of Song in the Phylloscopus Leaf Warblers (Aves: Sylviidae): A Tale of Sexual Selection, Habitat Adaptation, and Morphological Constraints*, volume 40, pages 35–66. 2009. doi: 10.1016/S0065-3454(09)40002-0.

R. P. S. Mahler. *Statistical Multisource-Multitarget Information Fusion*. Artech House, Boston/London, 2007.

G. Mysore, P. Smaragdis, and B. Raj. Non-negative hidden Markov modeling of audio with application to source separation. In *Proceedings of the International Conference on Latent Variable Analysis and Signal Separation (LVA / ICA)*, volume 6365/2010, pages 140–148, St. Malo, France, 2010. doi: 10.1007/978-3-642-15995-4_18.

T. S. Osiejuk. Recognition of individuals by song, using cross-correlation of sonograms of Ortolan buntings emberiza hortulana. *Biological Bulletin of Poznań*, 37(1 Suppl):39–50, 2000.

C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, New Jersey, new edition, 2000.

M. D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M. E. Davies. Sparse representations in audio and music: From coding to source separation. *Proceedings of the IEEE*, 98(6):995–1005, 2010. doi: 10.1109/JPROC.2009.2030345.

M. Salomon and Y. Hemim. Song variation in the Chiffchaffs (*Phylloscopus collybita*) of the Western Pyrenees–the contact zone between the *collybita* and *brehmii* forms. *Ethology*, 92(4):265–282, 1992. doi: 10.1111/j.1439-0310.1992.tb00965.x.

J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

D. Stowell, S. Muševič, J. Bonada, and M. D. Plumbley. Improved multiple birdsong tracking with distribution derivative method and Markov renewal process clustering. In *Proceedings of the International Conference on Audio and Acoustic Signal Processing (ICASSP)*, 2013. preprint arXiv:1302.3642.

J. Van Gael, Y. W. Teh, and Z. Ghahramani. The infinite factorial hidden Markov model. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems (NIPS)*, volume 21, pages 1697–1704. 2009.

D. L. Wang and G. J. Brown, editors. *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. IEEE Press, New York, 2006.

I. Winkler, S. Denham, R. Mill, T.M. Bőhm, and A. Bendixen. Multistability in auditory stream segregation: a predictive coding view. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 367(1591):1001–1012, 2012. doi: 10.1098/rstb.2011.0359.

I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco, CA, USA, 2nd edition, 2005.