Analysis of Sound for Additive Synthesis: Tracking of Partialis Using Hidden Markov Models

Ph. Depalle, G. Garcia and X. Rodet
IRCAM, 31 rue Saint Merri, 75004 Paris, France
Tel: (33.1) 44.78.46.45, Fax: (33.1) 42.77.29.47
e-mail: pho.garcia, rod@ircam.fr

Abstract

We present a sinusoidal partials tracking method, based on Hidden Markov Models (HMM), for additive synthesis of sound. It allows to track time-varying partials as well as constant ones, and has no harmonicity constraint at all. In addition, it is able to easily track crossing partials. Several interesting examples will be shown, as well as a complete description of the HMM-tracker.

1. Introduction

We present here a HMM-based tracker of sinusoidal partials, for additive synthesis of sound [Garcia 92]. Additive synthesis is one of the most powerful synthesis techniques. It is based on a model where each sound signal is a sum of sinusoids with time-varying frequency and amplitude (Rodet 92). These parameters are generally supplied to the synthesizer as breakpoint time-functions. Such functions are interpolated at the sampling rate to obtain instantaneous parameter values.

When working with natural sounds, these breakpoint time-functions can be estimated from the original signal by an automatic analysis. This is basically performed in three steps: short-term spectral estimation, spectral peak detection, and spectral peak continuation. In IRCAM's additive synthesis environment, the first step is implemented by a sliding window FFT, and the second one by extraction of local maxima followed by a Lagrange interpolation. A more detailed description of this is presented in [Depalle 93].

The third step of the analysis procedure is by far the more critical one. Existing approaches [Mc Auley 86], [Serra 89] work well enough for some categories of sounds (harmonic, voiced, and slow time-varying sounds), but fail in presence of multiple harmonic structures, inharmonics, crossing partials, voiced/unvoiced transitions, and strong frequency variations. Our aim has been to cope with all these problems by globally optimizing the set of detected trajectories. More explicitly, all trajectories in a given time interval will be detected simultaneously, after evolution of continuity criteria in the whole set of peak frames. This differs from existing methods, which generally consider that tracking has been done up to the previous frame, and examine each peak of the current frame to eventually associate it to an existing trajectory.

The tracking problem has been formulated in terms of a hidden Markov model. Thus, we find the optimal set of trajectories by looking for the highest probability state sequence, by means of the Viterbi algorithm [Rabiner 86]. This leads to very good tracking performances, as it will be shown in section 5. In addition, the method can be applied without any modification to formant tracking.

In sections 2 and 3, we describe the HMM-tracker, in section 4 we discuss the principles of the algorithm and in section 5 we present some examples.

2. Method description

The model of trajectory we use in our HMM tracker is intuitively defined as a time-sequence of peaks which satisfies continuity constraints on parameter slopes. Consequently, our method needs to identify trajectories whose amplitude and frequency slope evolve smoothly in time. The use of parameter slopes rather than parameter values enables to track time-varying partials as easily as constant ones, and solves the problem of detecting crossing trajectories. To formalize our HMM tracker, let us introduce some definitions:

- A trajectory is labelled by an index \( i > 0 \).
- The time series of peak frames is referenced by subscript \( k \). At time \( k \), there are \( h_k \) peaks. The frequency and amplitude of peak \( j \) in frame \( k \) are denoted \( f_k(j) \) and \( a_k(j) \) respectively.
- A peak which belongs to a trajectory is called an useful peak; otherwise it is called a spurious peak.
- An index \( I_k(j), 1 \leq j \leq h_k \), is associated to each peak of frame \( k \). Indices of useful peaks are the label of the trajectory to which they belong, and indices of spurious peaks are null.
- \( I_k \) is the vector of indexes at time \( k \).
- The transition probability from a state \( x \) at time \( k \) to a state \( y \) at time \( k+1 \) is denoted by \( O_k(x,y) \).

Then, the elements of the HMM are defined as follows: at time \( k \), the state \( S_k \) is the ordered pair of vectors \( (h_k, I_k) \), and the observation \( O_k \) is the ordered pair of scalars \( (f_k, a_k) \).
Defining state and observation with ordered pairs is based on the fact that we want to retrieve the right set of trajectories as the globally optimal succession of elementary trajectories. The more elementary trajectory is a straight line which takes two points. In this way, slopes appear as the relevant parameters to compute the transition probabilities between states.

Some important remarks need to be done here:
Peak parameters are not considered as observations, but define the transition probabilities; they are parameters of the model. With the above definition of the observation, we retain only the combinatorial aspect of the peak stretching problem. As a consequence, for a given state there is only one possible observation. On the other hand, a given observation can be produced by several states, and this is why states remain hidden. In other words, the observation probabilities for a given state are equal to unity for one particular observation, and null for all others. Consequently, only transition probabilities need to be computed. For example, all possible states for the observation (3, 2) are:

- States with no trajectory:

- States with one trajectory:

- States with two trajectories:

Fig 1: All possible states for the observation (3, 2).

3 Evaluation of state transition probabilities

According to the intuitive model of trajectory proposed below, transition probabilities must favour state transitions which provide:
- the best continuity of parameter slopes between useful peaks,
- the largest discontinuity of parameter slopes between spurious peaks.

Figure 2 shows an example of this: the first transition must be more probable than the second one, because it defines more smooth trajectories.

Fig 2: Two possible state transitions.

Let us consider a transition from a state \( x \) at time \( k-1 \) to a state \( y \) at time \( k \). It is clear that frames \( k', k-1 \) and \( k-2 \) are concerned. For each peak \( j \) of frame \( k \), we compute an individual score \( \theta_k(j) \) in the following manner:

- If state \( x \) defines peak \( j \) as useful, we look for peaks \( t \) and \( r \) of frames \( k-1 \) and \( k-2 \) respectively, such that

\[
I_k(j) = I_{k-1}(t) = I_{k-2}(r)
\]

Then, a gaussian function is used to evaluate the continuity of frequency and amplitude slopes, for the trajectory defined by these three peaks:

\[
\theta_k(j) = \exp\left(-\frac{(\Delta f_t - \Delta f_r)^2}{\sigma^2_1}\right) \times \exp\left(-\frac{(\Delta a_t - \Delta a_r)^2}{\sigma^2_2}\right)
\]

where \( \sigma_1, \sigma_2 \) are input parameters and

\[
\Delta f_t = f_k(j) - f_{k-1}(t), \Delta a_t = a_k(j) - a_{k-1}(t), \Delta f_r = f_{k-1}(t) - f_{k-2}(r), \Delta a_r = a_{k-1}(t) - a_{k-2}(r).
\]

- If state \( y \) defines peak \( j \) as spurious, we look for peaks \( t \) and \( r \) of frames \( k-1 \) and \( k-2 \) respectively, such that

\[
\left(\Delta f_t - \Delta f_r\right)^2 \text{ is minimum.}
\]

The discontinuity of the trajectory defined by these three peaks is evaluated as:

\[
\theta_k(j) = \begin{cases} 
1 & - (1 - \mu) \exp\left(-\frac{(\Delta a_t - \Delta a_r)^2}{\sigma^2_2}\right) \\
1 & - (1 - \mu) \exp\left(-\frac{(\Delta f_t - \Delta f_r)^2}{\sigma^2_1}\right) 
\end{cases}
\]

where \( \mu \) (\( 0 \leq \mu \leq 1 \)) is an input parameter.

Figure 3 illustrates the computation of individual scores \( \theta_k(j) \) for a given state transition, concerning useful and spurious peaks.

Fig 3: Evaluation of individual scores \( \theta_k(j) \). Then, a global score is computed as
\[ \alpha_{k}(x,y) = \frac{\rho_{k}(x,y)}{\sum_{i \in \Omega_{k}} \rho_{k}(x,z)} \]

4. Algorithm

The computational complexity of the algorithm depends strongly on the number of combinations of indexes in the vectors \( \Omega_{k} \), which determines the number of states. It also depends on the number of frames on which trajectories have to be optimised. In order to limit this complexity (otherwise the method is not practicable) we use the Viterbi algorithm on a window of \( T \) frames length which slides frame by frame, and we introduce some constraints on the index combinations. For instance, births and deaths of trajectories are disallowed in the states, forcing the state sequences to have a constant number of trajectories. To take into account births and deaths on the whole set of frames, we use a higher level procedure. Births are detected by looking for partials in current window which did not exist in the preceding one; deaths are found by looking for partials in the preceding window which disappear in the current one. Consequently, the length \( T \) fixes the shortest duration of a partial.

Other constraints can be eventually introduced at the state level, such as: limiting frequency or amplitude slopes, limiting the number of line crossings, selecting a reduced set of possible number of trajectories, etc.

The detection of births and deaths is schematised in figure 4.

![Diagram of birth and death detection](image)

Fig. 4: Birth and death detection.

5. Experimental results

The following figures show some experimental results obtained by applying our method to simulated and real data.

- [Crossing of partials: three sinusoids embedded in white noise (SNR = 0 dB)](image)
- [Tracking of partials of a voice “glissando”](image)
- [Percussive inharmonic sound (Tabla dha)](image)
6. Conclusion

The previous examples show that a purely combinatorial HMM-based tracker is quite performant for additive analysis/synthesis in a musical context. The presented method has two key principles:

- The use of parameter slopes rather than parameter values, to evaluate the continuity of trajectories. This allows to track variable frequency and variable amplitude partials, and solves the problem of frequency line crossing.
- The global optimization, in frequency and time, of the set of detected partials, which is a consequence of the HMM formulation of the method. This ensures a certain coherence in tracking multiple partial structural patterns, and provides a very good voicing discrimination: the algorithm stops tracking during unvoiced regions since it can’t find any continuous trajectory.

Our HMM-tracker has been implemented in a modular way so that constraints can be chosen before running it. We are currently working on the reduction of the computational cost of the algorithm by refining the set of constraints. We are also looking for a more efficient type of algorithm, to take advantage of the factored structure of transition probabilities. In particular, we are examining the possibility of substituting the Viterbi algorithm to eliminate computation redundancy.

References


