Multiple-F0 tracking based on a high-order HMM model

Wei-Chen Chang, Alvin W.Y. Su, Chunghsin Yeh,, Axel Roebel, Xavier Rodet

MULTIPLE-F0 TRACKING BASED ON A HIGH-ORDER HMM MODEL

Wei-Chen Chang and Alvin W. Y. Su

Dep. of Computer Science and Information Engineering
National Cheng-Kung University, Tainan, Taiwan
bff@csie.ncku.edu.tw

Chunghsin Yeh, Axel Roebel and Xavier Rodet

Analysis/Synthesis team
IRCAM/CNRS-STMS Paris, France
cyeh,roebel,rod@ircam.fr

ABSTRACT

This paper is about multiple-F0 tracking and the estimation of the number of harmonic source streams in music sound signals. A source stream is understood as generated from a note played by a musical instrument. A note is described by a hidden Markov model (HMM) having two states: the attack state and the sustain state. It is proposed to first perform the tracking of F0 candidates using a high-order hidden Markov model, based on a forward-backward dynamic programming scheme. The propagated weights are calculated in the forward tracking stage, followed by an iterative tracking of the most likely trajectories in the backward tracking stage. Then, the estimation of the underlying source streams is carried out by means of iteratively pruning the candidate trajectories in a maximum likelihood manner. The proposed system is evaluated by a specially constructed polyphonic music database. Compared with the frame-based estimation systems, the tracking mechanism improves significantly the accuracy rate.

1. INTRODUCTION

The fundamental frequency, or F0, is an essential descriptor of harmonic sound signals. For the analysis of music signals, multiple-F0 estimation is a "necessary evil" because musical notes played by various instruments usually sound simultaneously. The development of a multiple-F0 estimation system is relevant to a wide range of applications such as source separation, music information retrieval, and automatic music transcription. A difficult problem in estimating the F0s of concurrent sources is the estimation of the number of sources, called *polyphony inference*. This paper proposes a method for inferring the polyphony in a single frame, and introduces a tracking mechanism to estimate the number of source streams in music signals. In this paper, "tracking" means to extract continuous F0 trajectories of the underlying sources.

Multiple-F0 tracking is closely related to source stream forming. The extraction of continuous F0 trajectories gives rise to the related streams of harmonic sources, whereas the extraction of source streams brings about the related F0 trajectories. There exist mainly two approaches to multiple-F0 tracking or source stream forming: (1) *tracking followed by clustering* (TfC) and (2) *clustering followed by tracking* (CfT). TfC performs partial tracking across analysis frames and then groups the related partials into source streams [1] [2] [3] [4]. On the contrary, CfT groups the partials into hypothetical sources in each analysis frame which are then tracked across frames [5] [6] [7]. Conceptually, one may regard TfC as a "horizontal then vertical" process, whereas CfT is a "vertical then horizontal" process. Another approach is to segregate the source streams by integrating the clustering and the tracking in a joint manner [8].

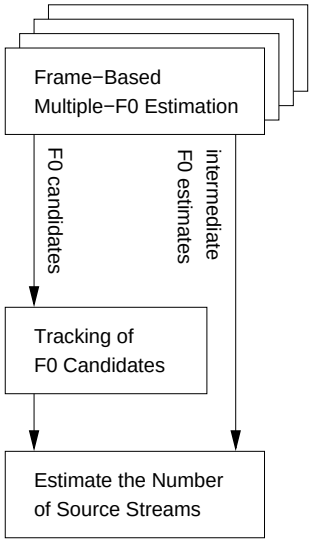


Figure 1: Overview of the proposed multiple-F0 tracking system.

The proposed multiple-F0 tracking system follows the CfT approach. In each analysis frame F0 candidates are extracted, of which their combinations are evaluated to estimate the number of sources along with the related F0s, called the *intermediate F0 estimates*. The F0 candidates are then connected across the frames to establish *candidate trajectories*. The reason to establish candidate trajectories beforehand is that the connection of the intermediate F0 estimates usually form broken segments of the underlying source streams. Candidate trajectories are more complete, which provides a good initial estimate of the source streams. Moreover, the tracking of F0 candidates requires fewer computations compared with partial tracking that is usually carried out in the TfC approach. The candidate tracks are then pruned to yield the final source streams according to the intermediate F0 estimates. The advantage of the proposed tracking system is its generic architecture. Given a frame-based F0 estimation system with a reasonable accuracy in polyphony inference, it is simple to "plug in" the frame-based system into the proposed tracking architecture.

This paper is organized as follows. In **Section 2**, the frame-based F0 estimation system is described, focusing on the polyphony inference algorithm. In **Section 3**, the candidate tracking algorithm is presented, which takes care of the missing candidates by using a high-order HMM. In **Section 4**, a track pruning algorithm is presented which iteratively excludes excessive source streams. Finally, the proposed system is evaluated and the possible improvements are discussed.

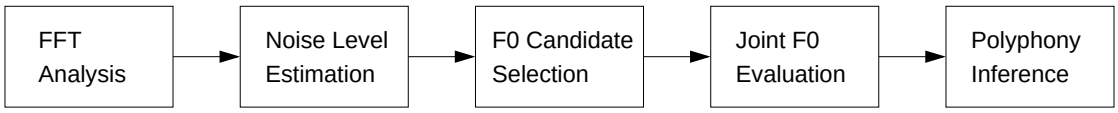


Figure 2: Overview of the frame-based multiple-F0 estimation system.

2. FRAME-LEVEL POLYPHONY INFERENCE

A previously developed frame-based multiple-F0 estimation algorithm [9] [10] lays a foundation for the tracking system (see Figure 2). In each analysis frame, FFT (Fast Fourier Transform) is applied to the observed signal to obtain the instantaneous spectrum. The observed sound signal is modeled as a sum of several harmonic sources and noise, where each harmonic source is modeled as a sum of sinusoids. The spectral peaks are considered sinusoids or noise generated by this signal model. To estimate multiple F0s, the number of sources is to be inferred. If the noise part is not estimated beforehand, the number of sources can be overestimated when unnecessary sources are simply used to explain the noise. Therefore, a noise level estimation algorithm has been developed [11] to distinguish sinusoidal peaks, considered to be the partials of harmonic sources, from noise peaks. Once the spectral peaks are classified according to the estimated noise level, the partials of a set of hypothetical sources should match most of the sinusoidal peaks. To evaluate the plausibility of a set of hypothetical sources, a score function has been proposed [12]. The score function is based on three assumptions concerning the physical properties of harmonic instrument sounds: (1) spectral match with low inharmonicity; (2) spectral smoothness; and (3) synchronous amplitude evolution within a single source. Because the number of combinations grows exponentially with the number of F0 candidates as well as the polyphony, a candidate selection algorithm has been developed to effectively select the F0 candidates, aiming at reducing the number of candidates [10]. The estimation of the number of sources is handled by a polyphony inference algorithm, which is to be detailed in the following.

The strategy is to progressively increase the polyphony hypothesis M and calculate the score of all possible combinations of F0 candidates. The scoring of hypothetical combinations is used to select the most plausible ones, among which the best combination is determined by iteratively verifying the related F0 hypotheses to consolidate the estimates. The estimation of the largest polyphony possible N_{max} is based on the *score improvement* [13]. All the top-five combinations (ranked by the score function) of all polyphony hypotheses, denoted by $\{\mathcal{C}_m\}_{m=1}^{N_{max}}$, are retained for the consolidation of the F0 estimates, denoted by $\mathcal{F}$.

The inference algorithm (see **Algorithm 1**) begins with listing the individual F0 hypotheses found in $\mathcal{C}_{N_{max}}$, denoted by $\mathcal{H}$, in order of their individual salience, which is derived from the individual score weighted by the appearing "frequency" in $\mathcal{C}_{N_{max}}$. Beginning with the most likely F0 hypothesis, each hypothesis is consecutively combined with the current estimate $\mathcal{F}$ and its contribution is verified by the previously calculated score criteria. If an F0 hypothesis (to be added) is higher in frequency than the lowest one previously selected, it is considered *valid* if it either improves the envelope smoothness of the hypothetical sources that have partials overlapping with its partials, or explains a significant amount of salient peaks. On the other hand, if an F0 hypothesis (to be added) is lower in frequency than the lowest one previously selected, it is considered valid provided that it explains a significant amount of salient peaks. Otherwise, it is considered a spurious source that is composed of noise. When an F0 hypothesis meets the requirements for a valid estimate, it is removed from the hypothesis list $\mathcal{H}$ and added into the set of the F0 estimates $\mathcal{F}$. During the progressive increase of the polyphony hypothesis M, the algorithm searches for the matched combinations in $\{\mathcal{C}_m\}_{m=1}^{N_{max}}$. When a matched combination is no longer found, the consolidation process stops. The polyphony is thus inferred along with the estimated F0s.

To measure the salience of an added F0 hypothesis, it is proposed to verify its *effective salience*, denoted by E_{eff}. The salience of a spectral peak, called *peak salience*, is defined as the sum of linear amplitudes of all the related spectral bins. Accordingly, the salience of an F0 hypothesis is defined as the sum of the peak salience of the peaks assigned to this hypothetical source [12]. An F0 hypothesis is considered valid if its effective salience is larger than the salience of noise, called *noise salience* E_{noise}, which is defined as the sum of the peak salience of the classified noise peaks. The estimated sources should explain most of the sinusoidal peaks such that the reduction of the *residual salience*, denoted by ΔE_R, is larger than the noise salience. The residual salience is defined as the sum of the peak salience of the remaining peaks that are yet to explain. This condition, $\Delta E_R > E_{noise}$, is important for the validation of a NHRF0 (non-harmonically related F0) hypothesis because its non-overlapping partials should match a significant amount of salient peaks.

The *improvement of spectral smoothness* is an important requirement for the validation of HRF0s (harmonically related F0s) because adding a HRF0 hypothesis usually improves the smoothness of the spectral envelopes of the currently selected sources. Since an additional HRF0 tends to improve the resulting spectral smoothness as well, it is necessary to put a constraint on the improvement of spectral smoothness. To achieve this goal, it is proposed to observe the variation of the score criterion MBW (Mean BandWidth) [12]. MBW evaluates the energy spread, that is, the bandwidth [14], of the spectral envelope of a hypothetical source. A smooth envelope results in a small MBW value because the high-frequency components in its spectrum are less dominant than the low-frequency ones. The improvement of spectral smoothness is required to exceed what can be allowed for harmonic instrument sounds. To learn the threshold of MBW as the allowed improvement of a spectral envelope, selected instrument samples of RWC Musical Instrument Sound Database [15] are used. Given an observed partial sequence of a harmonic sound, the hypothetical sources of the F0s at the partial frequencies are considered the HRF0 hypotheses. For each HRF0 hypothesis, the decrease of MBW, denoting ΔMBW, are evaluated. ΔMBW is the difference of MBW before, denoted by mbw_o, and after, denoted by mbw_s, smoothing out [1] the partials of a HRF0 hypothesis. For each analysis instance, mbw_o of the correct F0 and mbw_s

[1] A smoothed out partial is replaced by the amplitude interpolation of its adjacent partials.

Algorithm 1: polyphony inference

input : The list of F0 hypotheses $\mathcal{H} = \{\text{F0}_1, \text{F0}_2, \cdots, \text{F0}_J\} = \{\text{F0}(j)\}_{j=1}^{J}$ in order of salience along with the top-five combinations for all polyphony hypotheses $\{\mathcal{C}_m\}_{m=1}^{N_{max}}$

output: The inferred polyphony M with the F0 estimates $\mathcal{F}$

Initialization of the F0 estimates $\mathcal{F} = \{\varnothing\}$ and $M \longleftarrow 0$
Initialization of residue $E_R \longleftarrow 1$
while $J > 0$ **do**
  for $c = 1$ *to* J **do**
    if $\{F0_c \bigoplus \mathcal{F}\} \bigcap \mathcal{C}_{M+1}$ *and* $E_{eff}(F0_c) > E_{noise}$ **then**
      if *F0$_c$ is higher than any F0 in* $\mathcal{F}$ **then**
        if $\max(\{\Delta MBW\}_{m=1}^{M}) > \Delta MBW_{model}$ **then** `/* smoother envelope */`
          $\mathcal{F} \longleftarrow \text{F0}_c \bigoplus \mathcal{F}$
          update M and E_R
          break the For loop
        else
          if $\Delta E_R > E_{noise}$ **then** `/* reduction of residual salience larger than noise salience */`
            $\mathcal{F} \longleftarrow \text{F0}_c \bigoplus \mathcal{F}$
            update M and E_R
            break the For loop
          end
        end
      else
        if $\Delta E_R > E_{noise}$ **then**
          $\mathcal{F} \longleftarrow \text{F0}_c \bigoplus \mathcal{F}$
          update M and E_R
          break the For loop
        end
      end
    end
  end
  if *any F0 is added to* $\mathcal{F}$ **then**
    remove the selected F0 from $\mathcal{H}$
    $J \longleftarrow J - 1$
  else
    termination of the While loop
  end
end
ps: $\bigoplus$ stands for "combined with"

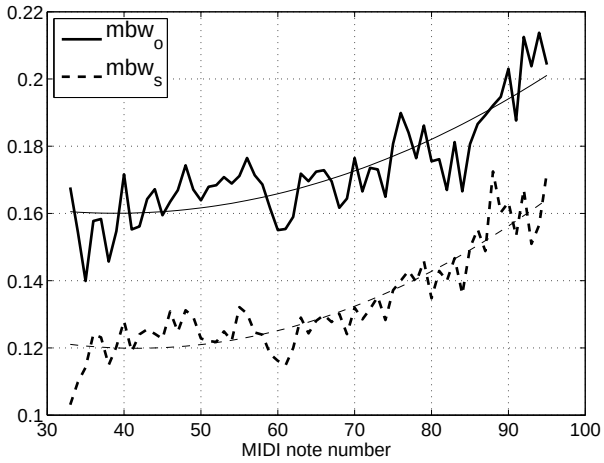


Figure 3: MBW comparison between those of the original spectral envelopes and those the smoothed spectral envelopes. The two thin lines are second-order polynomial functions fitting the trained MBW data.

of the HRF0 hypothesis that results in the maximal ΔMBW are retained. For each musical note, the calculated mbw_s and mbw_o are averaged for all the analysis instances of all the instruments (see Figure 3). They are further modeled, as a function of the MIDI note numbers, using a second-order polynomial. The threshold for the improvement of spectral smoothness is then defined as $\Delta\text{MBW}_{model} = (mbw_o - mbw_s)/mbw_o$.

3. TRACKING OF F0 CANDIDATES USING A HIGH-ORDER HMM

To associate F0 candidates across frames into related tracks, it is suggested to consider the tracking within a probabilistic framework based on a higher-level model. For music signals, the use of a *note event model* facilitates the depiction of a single note's behavior [16] [6]. A note event model can be represented as a hidden Markov model (HMM) which describes the temporal evolution of a single note as a sequence of states changing from frame to frame [17]. The states are often modeled into several: attack, sustain, release, and silence. The proposed note model is a simplified version which contains only two states, the attack state and the sustain state (see Fig. 4(a)). The tracking of F0 candidates can thus be understood as decoding multiple optimal paths in a three-dimensional trellis structure (see Fig. 5). Notice that the attack layer in the trellis structure allows the initialization of a trajectory at any analysis frame. The traditional algorithms like forward-backward algorithms or Viterbi algorithm are not appropriate in this case. This is because there is not just a single best path to be decoded but several, of which the number is unknown. In addition, the length of each trajectory is unknown, which is to be determined as well. In the following, a tracking algorithm is proposed, based on a high-order [2] HMM and a forward-backward tracking scheme.

[2] In this paper, the order represents the number of preceding frames to which the dependency of the current frame is related.

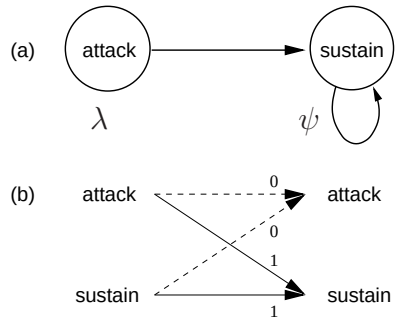


Figure 4: Proposed HMM note model: (a) graphical representation of the note model with the attack probability λ and the sustain probability ψ; (b) state transition weight matrix.

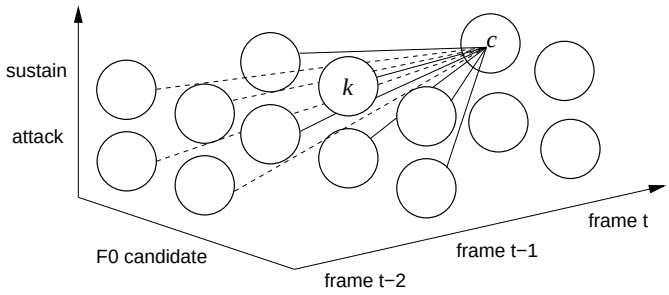


Figure 5: Illustration of the trellis structure of the HMM model of order 2. Each node represents the hidden state of a note, either *attack* or *sustain*. The solid lines connect the candidate c in the frame t to the candidates in the frame $t-1$, whereas the dash lines connect the candidate c in the frame t to the candidates in the frame $t-2$.

3.1. Forward propagation of connection weights

Since the trajectories to estimate are of different lengths, the proposed tracking algorithm does not intend to normalize their probabilities. The probability of a trajectory is associated with the *weights* propagated from node to node within the related path. The weight of a node is characterized by the observation probability emitted by a hidden state of a note. The probability emitted by the attack state is a scalar λ. The observation probability emitted by the sustain state is defined by the following Gaussian distribution:

$$\psi(\Delta_f) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{\Delta_f^2}{2\sigma^2}) \tag{1}$$

where Δf denotes the frequency difference in *tone* between two subsequent F0 candidates. σ is set 0.25, which corresponds to one-quarter tone. The transition between nodes is allowed for "attack to sustain" and "sustain to sustain" with equal probability (see Fig. 4(b)). In order to retrieve complete trajectories using the available F0 candidates, it is proposed to use a high-order HMM such that missing candidates can be taken care of and separate segments related to one single trajectory can be rejoined. Denoting a node by its coordinate in the trellis structure n(frame, candidate, state), the propagated weight from $n(t-d,k,p)$ to $n(t,c,q)$ can thus be defined:

$$\begin{aligned}&\gamma(n(t,c,q)|n(t-d,k,p)) = \\ &\alpha \cdot \omega(d) \cdot \Gamma(t-d,k,p) + (1-\alpha)\cdot\psi(\Delta_f), \quad q \text{ is sustain.}\end{aligned} \tag{2}$$

where $\Gamma(t-d,k,p)$ denotes the *forward propagated weight* at the node $n(t-d,k,p)$ which is related to the probability of the partially observed path from frame 1 to frame $t-d$. $\Gamma(t-d,k,p)$ is initialized to λ for all the nodes in the attack layer. For $t-d<0$, $\Gamma(t-d,k,p) = 0$. The order of the HMM note model is then determined by d. Assuming that the information from a closer frame is more reliable than that from a more distant frame, it is proposed to apply a weighting function

$$\omega(d) = \frac{1}{d^s} \tag{3}$$

which decreases with the distance d. The exponential weighting parameter s is to be determined. The introduction of the weighting parameter α is to adjust the dependency of the current node $n(t,c,q)$ on the preceding information available up to the node $n(t-d,k,p)$. The connection resulting in the maximal propagated weight is considered the most likely and is stored as a pointer to the "winning node" for the later backward tracking:

$$I_{max}(t,c,q) = \underset{d,k,p}{\operatorname{argmax}}\, \gamma(t,c,q|t-d,k,p) \tag{4}$$

Accordingly, the forward propagated weight at the current node $n(t,c,q)$ is updated:

$$\Gamma(t,c,q) = \gamma(t,c,q|I_{max}(t,c,q)) \tag{5}$$

3.2. Iterative backward tracking of candidate trajectories

At the forward tracking stage, the propagated weights $\Gamma(t,c,q)$ and the related back pointers $I_{max}(t,c,q)$ of all the nodes are recorded. Since multiple back pointers may point to one node, these back pointers attained during the forward tracking result in tree structures whose paths from the "roots" to the "leaves" are possible trajectories. At the backward stage, it is proposed to iteratively extract the F0 candidate trajectories by finding the most likely paths from the leaves to the roots. Starting from all the nodes in the sustain layer, the tracking connects the preceding nodes in a backward sense until a node in the attack layer is reached. In this way, a trajectory, starting in the attack state and ends in the sustain state, is retrieved. The nodes are marked "visited" once they are selected for the candidate trajectories. The visited nodes are no longer used for the tracking of the consecutive trajectories. The order in which the trajectories are extracted can thus give rise to different tracking results. In each analysis frame, it is reasonable to search in order of the propagated weights Γ of the nodes because a large weight is evidence of high probability. However, the exclusion of the visited nodes implies that the intersection of F0 trajectories are not allowed, which is left as a future possibility for improvements. An F0 candidate is then uniquely associated with only one trajectory.

4. ESTIMATE THE NUMBER OF SOURCE STREAMS

Assuming that the set of F0 candidate trajectories includes all the underlying source streams, the estimation of the source streams is in fact the *pruning* of the candidate trajectories. The final estimate

of the source streams shall be coherent with the polyphony (number of sources) and the intermediate F0 estimates determined by **Algorithm 1**. The polyphony $M(t)$ provided by **Algorithm 1** will be understood as an observation of the true, unknown polyphony. By means of experimental investigation it is found that the two-sided asymmetrical nearly exponential distribution (see Fig. 6(a)) models the probability of the polyphony error ΔM of **Algorithm 1** [10].

The candidate trajectories are to be pruned in a manner such that the likelihood of the observed polyphony $\{M(t)\}_{t=1}^{T}$ is maximized, where T is the number of frames. At each iteration of pruning, the current state of the polyphony of the candidate trajectories is considered as the *estimated polyphony* $\hat{M}$. Accordingly, the problem is to maximize the log likelihood $p(M|\hat{M})$ for all observed frames. Assuming that $p(M|\hat{M})$ can be completely described by $p(\Delta M)$, the log likelihood of the current set of candidate trajectories is calculated by means of

$$L = \sum_{t=1}^{T} \log p_t(\Delta M) \tag{6}$$

The maximization of the log likelihood L is carried out by iteratively pruning the candidate trajectories. However, the order of pruning plays an important role because the removal of a trajectory can effect the consecutive ΔMs for the related frames and consequently alter the evaluated likelihoods. It is proposed to prune the candidate trajectories in order of the *accordance ratio*

$$R = \frac{\text{number of intermediate F0 estimates in trajectory T}_k}{\text{length of trajectory T}_k} \tag{7}$$

which is the percentage of the number of the intermediate F0 estimates contained in a candidate trajectory T_k. The accordance ratio R of a candidate trajectory measures its salience. An F0 candidate trajectory matching fewer intermediate F0 estimates is considered less probable to be a valid source stream.

The objective of the pruning process is to maximize the global inference likelihood L. The pruning of the candidate trajectories is expected to first increase L when the trajectories related to spurious sources are removed. When the trajectories are overly pruned, L is expected to decrease. The investigation of several instances shows that during the iterative pruning process L does follow a parabolic-like curve of which the maximum, denoted by L_{max}, is observable (see Fig. 6(b)). To trace L_{max}, it is necessary to evaluate more iterations after L_{max} is reached such that the maximum can be located. In the application concerned, the pruning process will continue until the log likelihood becomes smaller than the initial likelihood (when no trajectories are pruned). Early stopping strategies may be used to shortcut the procedure. However, due to the fact that the last removals are done much faster because fewer candidates have to be evaluated, it seems that this optimization is less important. The pseudo code of the proposed algorithm is listed in **Algorithm 2**.

A testing example of the proposed tracking system is demonstrated in Fig. 7. This example is a piece of synthesized music comprised of four instruments: flute, oboe, clarinet and bassoon. A rectangular box represents the time-frequency boundaries of the related note. After the F0 candidate trajectories are established (see Fig. 7(a)), they are pruned, according to the intermediate F0 estimates (see Fig. 7(b)), using **Algorithm 2** to yield the final estimate of the source streams (see Fig. 7(c)).

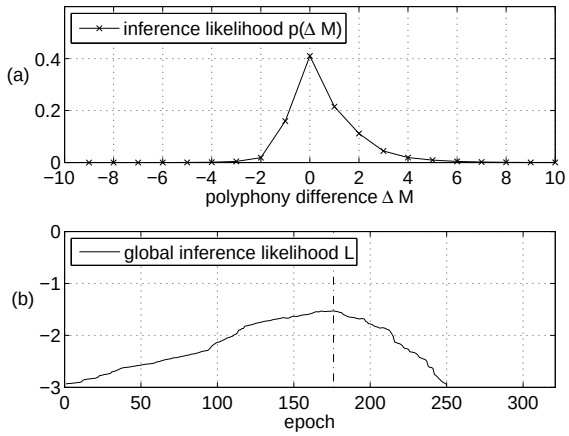


Figure 6: (a) Polyphony inference likelihood of the frame-based F0 estimator. (b) An example of log likelihood observations during the iterative pruning process. The dash line indicates where L_{max} occurs. The range of epoch is shown up to the initial number of candidate trajectories.

5. EVALUATION RESULTS

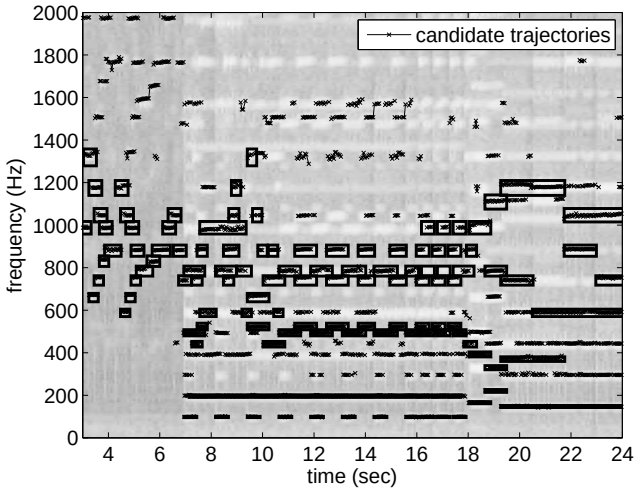
In order to evaluate a multiple-F0 estimation system, a systematic method has been proposed to create a polyphonic music database [18]. The idea is to make use of the great amount of existing MIDI files and music instrument sound samples to render synthesized polyphonic music. Care has been taken to split MIDI tracks to ensure that separate notes in a track do not overlap after rendering. In this way, ground truth can be more reliably established by a single-F0 estimator. This method is reproducible, extensible and interchangeable. Most importantly, the ground truth is verifiable. 26 pieces have been prepared for the study of multiple-F0 tracking.

There are several parameters to be trained: the initial probability of the attack state λ, the order d of the HMM, the related exponential parameter s, and the weighting parameter α. They are trained, on 13 pieces of synthesized music, using the evolutionary algorithm [19]. The best parameter set obtained is $(\lambda, d, s, \alpha) = (0.5, 2, 1.3, 0.52)$. Using this parameter set, another 13 pieces are used for the evaluation. The proposed tracking system is compared with two versions without the tracking mechanism: the MIREX' 07 version [9] and the thesis version [10]. The *overall accuracy* rate is used as the evaluation metrics [20]:

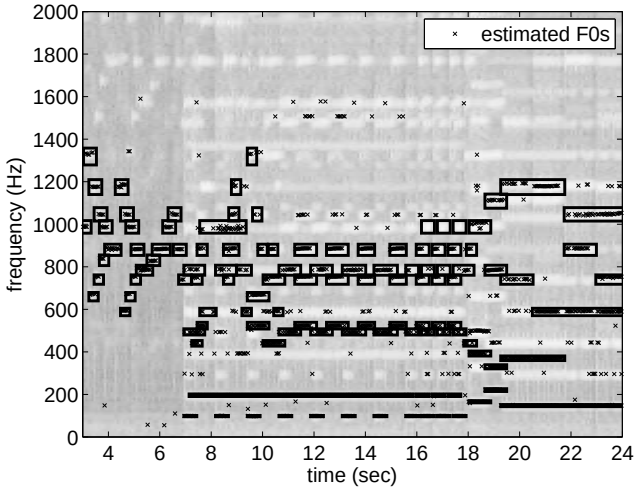
$$Acc = \frac{N_{corr}}{N_{corr} + N_{miss} + N_{subs} + N_{inst}} \tag{8}$$

where N_{corr} denotes the number of correctly estimated notes, N_{miss} denotes the number of missing notes, N_{subs} the denotes the number of substitution notes, and N_{inst} denotes the number of insertion notes. N_{corr} is often called *True Positives*; N_{miss} is often called *False Negatives*; N_{subs} and N_{inst} together are often called *False Positives*. In this test, concurrent sources with their F0s related to the same note are regarded as one single source. They are evaluated on a frame-by-frame basis, and a correct estimate should not deviate from the ground truth by more than 3%.

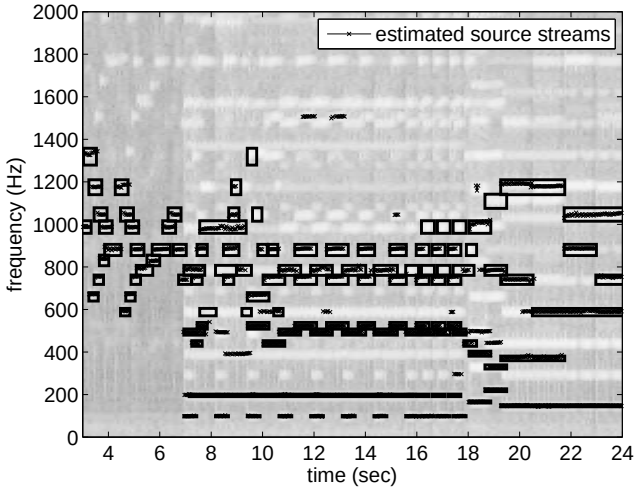
The accuracy rates of the three systems are compared from polyphony 1 to polyphony 6 (see Fig. 8). The average accuracy



(a) Tracking of F0 candidates using a high-order HMM model.



(b) Polyphony inference using the frame-based F0 estimation system.



(c) The pruning of candidate trajectories according to the estimated F0s.

Figure 7: A testing example of the proposed tracking system. The crosses mark the related F0s. The rectangular boxes represent the time-frequency boundaries of the ground truth notes.

Algorithm 2: Source stream estimation

input : A set of candidate trajectories $\mathcal{T}_0 = \{\mathrm{T}_k\}_{k=1}^{K}$, the observed polyphony $\{M(t)\}_{t=1}^{T}$, the accordance ratios $\{R(k)\}_{k=1}^{K}$.

output: The most likely source streams along with the related F0s.

Initialization of the estimated polyphony $\{\hat{M}\}_{t=1}^{T}$ related to $\mathcal{T}_0$;
Initialization of the log likelihood L_0;
Initialization of the epoch: $i \longleftarrow 1$;
$flagContinue \longleftarrow true$;
while $flagContinue$ **do**
  select the target trajectory: $\hat{k} \longleftarrow \underset{k \in \mathcal{T}_{i-1}}{\operatorname{argmin}} R(k)$;
  remove the selected trajectory: $\mathcal{T}_i \longleftarrow \mathcal{T}_{i-1} \ominus \mathrm{T}_{\hat{k}}$;
  update $\{\hat{M}\}_{t=1}^{T}$ according to the remaining trajectories $\mathcal{T}_i$;
  calculate the log likelihood L_i;
  if $L_i > L_0$ **then**
    $i \longleftarrow i + 1$;
  else
    $flagContinue \longleftarrow false$;
  end
end
select the best epoch: $\hat{i} \longleftarrow \underset{i}{\operatorname{argmax}} L_i$;
return $\mathcal{T}_{\hat{i}}$ as the final estimates of source streams;

rates of the MIREX'07 version, the thesis version, and the tracking version are 56.56%, 64.75%, and 69.79%, respectively. The MIREX'07 version has a slightly different polyphony inference algorithm and it is tuned to bias low-polyphony. However, its accuracy in the estimation of high polyphony is not satisfactory. The thesis version uses **Algorithm 1**, which improves significantly the accuracy in the estimation for the polyphony higher than 3. The proposed tracking system in fact uses the F0s estimated by the thesis version. For all the polyphony, the tracking improves the accuracy rates of the thesis version, especially for the polyphony higher than 3. The overall amelioration manifests the effectiveness of the proposed tracking scheme.

Further improvements are expected to reduce and "equalize" the error rates of the tracking system (see Fig. 9). The insertion note error increases with the decreasing polyphony, whereas the missing note error increases with the increasing polyphony. It is preferable to reduce both tendencies such that the system performs equally well for all polyphony. There are several possibilities to improve the tracking algorithm. First of all, the attack state requires a transient or onset feature to generate its observed probability λ. The initialization of λ with a fixed probability can not distinguish the correct onset, which may even cause the trajectories to connect to the partials of other sources. Secondly, the iterative pruning process can be improved by, for instance, taking care of the notes played in harmonic relations, called *octave streams*. Since the octave streams are often of high accordance ratios, further verification is necessary. Thirdly, the nodes should be allowed to share in different paths in the backward tracking stage. The intersection of source streams often occurs when, for instance, a singing voice, is involved. Fourthly, the use of $p(\Delta M|M)$, a polyphony-dependent inference likelihood, is expected to be more

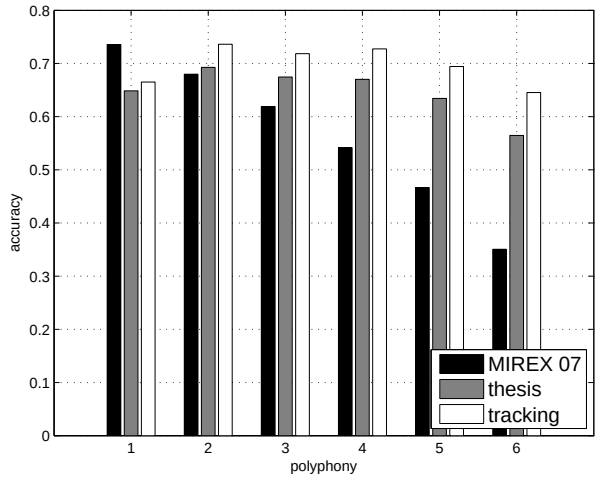


Figure 8: The evaluation results of three systems. Two versions of the frame-based F0 estimation system: (1) MIREX'07 version; (2) thesis version; and (3) thesis version using the proposed tracking algorithms.

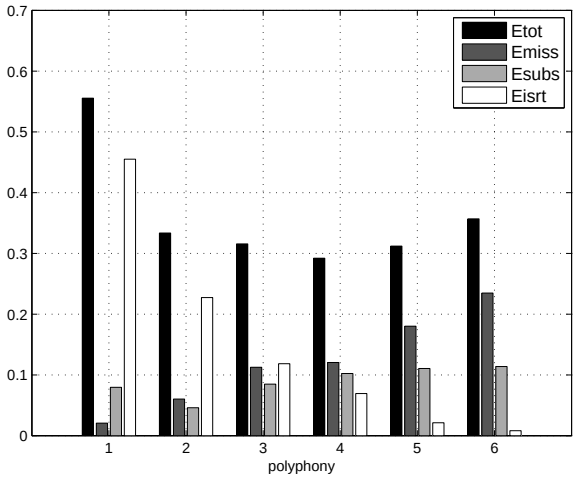


Figure 9: The error rates of the proposed tracking system: total error E_{tot}, missing note error E_{miss}, substitution note error E_{subs}, and insertion note error E_{isrt}.

realistic. Lastly, it is possible to rearrange the processing modules in the proposed system architecture to yield a better performance. For example, the tracking of candidate trajectories can be carried out before they are jointly evaluated. The tracking of candidate trajectories could eliminate spurious F0 candidates that are usually extracted around the transients.

6. CONCLUSION

Algorithms for the estimation of the number of sources in music signals have been presented. In the case of the frame-based F0 estimation, the presented polyphony inference algorithm verifies a hypothetical source according to the energy it explains and the spectral smoothness it improves. The forward-backward tracking mechanism then connects the F0 candidates into continuous trajectories. The use of a high-order HMM model allows the search of the most likely paths to "jump" across frames. The estimation of the number of source streams is realized by the pruning of candidate trajectories according to the F0s estimated by the frame-based estimation system in a maximum likelihood manner. The proposed tracking mechanism not only fuses the low-level signal descriptors like F0s into a high-level representation as note streams, but also improves the accuracy of estimation of the related F0s. The system architecture is easy to implement by plugging in a frame-based multiple-F0 estimation system. Further studies for improvements include the estimation of precise onsets, the verification of octave streams, and the handling of crossing trajectories.

7. ACKNOWLEDGMENT

The authors would like to express their gratitude towards the National Science Council, Taiwan, R.O.C. for funding Mr. Chang under the "Graduate Student Study Abroad Program".

8. REFERENCES

[1] D. K. Mellinger, *Event Formation and Separation in Musical Sound*, Ph.D. thesis, Department of Computer Science, Stanford University, December 1991.

[2] Keith D. Martin, "A blackboard system for automatic transcription of simple polyphonic music," *MIT Media Laboratory Perceptual Computing Section Technical Report*, , no. 385, July 1996.

[3] Andrew D. Sterian, *Model-Based Segmentation of Time-Frequency Images for Musical Transcription*, Ph.D. thesis, Department of Electronic Engineering, University of Michigan, 1999.

[4] M. Lagrange and G. Tzanetakis, "Sound source tracking and formation using normalized cuts," in *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'07)*, Honolulu, HI, USA, April 15-20 2007, vol. 1, pp. I–61–I–64.

[5] M. Wu, D.L. Wang, and G.J Brown, "A multipitch tracking algorithm for noisy speech," *IEEE Trans. on Speech and Audio Processing*, vol. 11, no. 3, pp. 229–241, 2003.

[6] M. Ryynänen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'05)*, Mohonk, NY, USA, 2005.

[7] E. Vincent, "Harmonic and inharmonic nonnegative matrix factorization for polyphonic pitch transcription," in *Proc. of IEEE, Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'08)*, Las Vegas, 2008.

[8] H. Kameoka, T. Nishimoto, and S. Sagayama, "Audio stream segregation of multi-pitch music signal based on time-space clustering using gaussian kernel 2-dimensional model," in *Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal*

Processing (ICASSP '05), Philadelphia, PA, USA, March 18-23 2005, vol. 3, pp. iii/5–iii/8.

[9] C. Yeh, "Multiple f0 estimation for mirex 2007," 2007, The 3rd Music Information Retrieval Evaluation eXchange (MIREX'07).

[10] C. Yeh, *Multiple fundamental frequency estimation of polyphonic recordings*, Ph.D. thesis, Université Paris VI, 2008, to appear.

[11] C. Yeh and A. Röbel, "Adaptive noise level estimation," in *Proc. of the 9th Int. Conf. on Digital Audio Effects (DAFx-06)*, Montreal, Canada, September 18-20 2006, pp. 145–148.

[12] C. Yeh, A. Röbel, and X. Rodet, "Multiple fundamental frequency estimation of polyphonic music signals," in *Proc. of IEEE, Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'05)*, Philadelphia, 2005, vol. 3, pp. iii/225 – iii/228.

[13] C. Yeh, A. Röbel, and X. Rodet, "Multiple f0 tracking in solo recordings of monodic instruments," in *120th AES Convention*, Paris, France, May 20-23 2006.

[14] Loen Cohen, *Time-Frequency Analysis*, Prentice Hall, 1995.

[15] Masataka Goto, "Rwc music database: Music genre database and musical instrument sound database," in *Proc. of the 4th International Conference on Music Information Retrieval (ISMIR 2003)*, Baltimore, Maryland, USA, October 27-30 2003, pp. 229–230.

[16] N. Orio and F. Dechelle, "Score following using spectral analysis and hidden markov models," in *Proc. of International Computer Music Conference (ICMC'03)*, Singapore, 2003.

[17] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 2, no. 77, pp. 257–286, 1989.

[18] C. Yeh, N. Bogaards, and A. Roebel, "Synthesized polyphonic music database with verifiable ground truth for multiple f0 estimation," in *Proc. of the 8th International Conference on Music Information Retrieval (ISMIR'07)*, Vienna, Austria, September 23-27 2007, pp. 393–398.

[19] Hans-Paul Schwefel, *Evolution and Optimum Seeking*, Wiley & Sons, New York, 1995.

[20] Graham E. Poliner and Daniel P.W. Ellis, "A discriminative model for polyphonic piano transcription," *EURASIP JASP*, 2006.