Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial
ISSN: 1137-3601
revista@aepia.org
Asociación Española para la Inteligencia Artificial
España

Gomez-Meire, Silvana; Reboiro-Jato, Miguel; Fajardo, Carlos H.; Olivieri, David; Fdez-Riverola, Florentino

A Robust Algorithm for Forming Note Complexes

Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial, vol. 14, núm. 47, 2010, pp. 27-37
Asociación Española para la Inteligencia Artificial
Valencia, España

Inteligencia Artificial 47(2010), 27-37
doi: 10.4114/ia.v14i47.1565

INTELIGENCIA ARTIFICIAL

http://erevista.aepia.org/

A Robust Algorithm for Forming Note Complexes

Silvana Gomez-Meire, Miguel Reboiro-Jato, Carlos H. Fajardo, David Olivieri, Florentino Fdez-Riverola

Department of Computer Sciencie
University of Vigo - Ourense (Spain)
{sgmeire,mrjato,cfajardo,dolivieri,riverola}@uvigo.es

Abstract We describe a new algorithm to transcribe musical note complexes from polyphonic piano music. Our method is a spectrogram based algorithm, which uses a robust peak detection scheme and forms note complexes by multiple sample conditional probability in the context of a time-frequency based finite state space approach, where note onsets are determined implicitly by changes in subharmonics. This paper provides a brief summary of some of the key algorithms in our method.

Keywords: Music transcription, note identification, signal analysis.

1 Introduction

The musical transcription is defined as the process by which, from hearing a musical piece it can be possible to reconstruct the sequence of notes forming the score. In other words, it is to obtain a symbolic representation of the piece that contains all the musical aspects of the same, in addition to identification of the note, set the tone, rhythm and duration.

Given that the automatic musical transcription problem has profound complexity, we believe there are several essential ingredients that algorithms addressing this problem must possess in order to be effective. First a robust low-level analysis of the audio signal is the key to determining notes, since we believe that the problem is data driven and well described by spectrogram based analysis. A second requirement is that it should build up a solution by using all data available. This is particularly inspired by Brown and Sterians [3] [6] view of *multiple hypothesis*.

A third requirement for musical transcription algorithms is that it should be effective for any instruments. Thus, it must be able to determine note onsets for cases where abrupt transitions in the time domain are not present. This requirement puts into question suggests that a more reliable algorithm would be derived from the frequency domain or perhaps from other transform methods.

Finally, we believe that certain implementation issues are important, especially in allowing *(1)* possibility for producing transcription in near real-time and *(2)* providing a framework which is extendible and can easily incorporate new algorithms. Related to the first of these issues, our algorithm processes audio file streams in blocks, and the finite state approach allows us to save note complexes in dynamic structures until they are complete. The second point is more difficult to demonstrate, however, we have used standard C++ object oriented design methodology and have heavily relied upon STL data structures, thus making the code portable and easily extendible for different algorithms [2].

Main Contributions of Paper: We describe a new set of algorithms for transcription of polyphonic music. The method we present is based upon transitions between states, which represent note complexes.

ISSN: 1988-3064(on-line)

Since our method is a spectrogram method, we describe the low-level peak detection algorithm we developed, and the peak to note association algorithm. We also show how transitions from one state are based upon a conditional probability so that decisions of whether a note belongs to a note complex is deferred until more information is available. In this manner, we eliminate spurious signals and our method is robustly determines groups of notes. Another aspect of our method is that the onset detection of notes is done in the frequency domain, so that we are not confined to particular instruments such as the piano, where the time domain signal provides sharp transitions between notes.

2 Foundations

In almost all algorithms reported in the literature for music transcription, there exist several common themes, particularly: *(1)* either a (short-time) fourier and/or wavelet transformation of the acoustic input signal is performed, converting into the time-frequency or space-scale plane, *(2)* information from the time-frequency is used to select peaks using some *ad-hoc* method, *(3)* onset detection is determined, and *(4)* the association of partials is made to notes. The difference between the reported work is the way in which each of these methods are implemented. Here we describe the details of the algorithms that we have developed in performing these common steps.

Principle of Superposition: As is well accepted, and indeed implicit in all other methods [4], is the principle of superposition. For a polyphonic instrument such as the piano, despite many complications of sound production through the baseboard and inharmonicity effects, the problem produces stationary eigenstates ([1], [5]) which add linearly to produce the full time domain signal.

2.1 Definitions and Time Scales

For the purpose of our discussion, there are several different time scales and concepts which we should like to define carefully. First, the common well known concepts hold: *(1) onset/offset times:* time at which a note begins/ends, *(2) note* n_k, is related to the fundamental frequency of oscillation and all the associated harmonics $\{\omega_k\}$ where $\omega_n = n\omega_0$, and the fundamental ω_0 is perceived as the pitch, *(3) timbre:* acoustic attribute of a note, which in general is more important to synthesis than transcription, as described.

Proper Sample Size: Before describing a note complex, it is interesting to describe the different time scales and sampling scales of interest. First, the sampling rate is 44.1 KHz, with a sampling interval, which we call the *proper sample*, of the distribution approximately 2.5 ms, however is a value that we can adjust. The sampling rate is sufficient to assure that we are free from aliasing effects. The STFT was defined with a Chebyshev window with 100dB of sidelobe attenuation. The spectral neighborhood size for computing refined partial power and frequency estimates was chosen to be 5points, and representing a bandwidth of 13.5Hz.

The implementation of the algorithm is intimately tied to the *proper* sample time scale, as we can see in figure 1, since it is about this size that we apply the entire algorithm. We make frequency transformation and perform peak detection, note association and determination of note complexes. Furthermore, each proper sample contains some points of the previous proper sample, following the principle of superposition, in order to avoid loss of information.

Definition 1 (Note Complex) *We define a note complex,* $G_s = G_s(t_1 \cdots t_k|\{n_1, \cdots n_k\})$ *as the set of notes* $n_j^{(k)}(t_i, t_f) \in G_s$*, which are voiced during the time frame* Δt_τ*, or equivalently from sample* k *to* $k + \tau$*. Furthermore, the notes* $n_j^{(k)}(t_i, t_f)$ *have onset/offset times* t_i *and* t_f *during the time interval* Δt_τ*.*

Finite State or Note Complex Scale: The note complex, G_s in our definition, is the exact equivalent to our definition of a finite state, or what we call the *transcription state* S which is the present state of the transition. Figure 2 shows the transitions from a state representing the present note complex $G(t_k) \longrightarrow G(t_{k+\tau+1})$, with conditional 2-state probability $P(G_k|G_{k+\tau+1})$.

In summary, the time scales for our algorithm are: *(1)* the *proper* sample size Δt_k, *(2)* onsets/offsets (t_i, t_f) times for individual note $n_j^{(s)}$ in a note complex $G_s(t_k \cdots t_{k+\tau})$, and *(3)* time between note complex

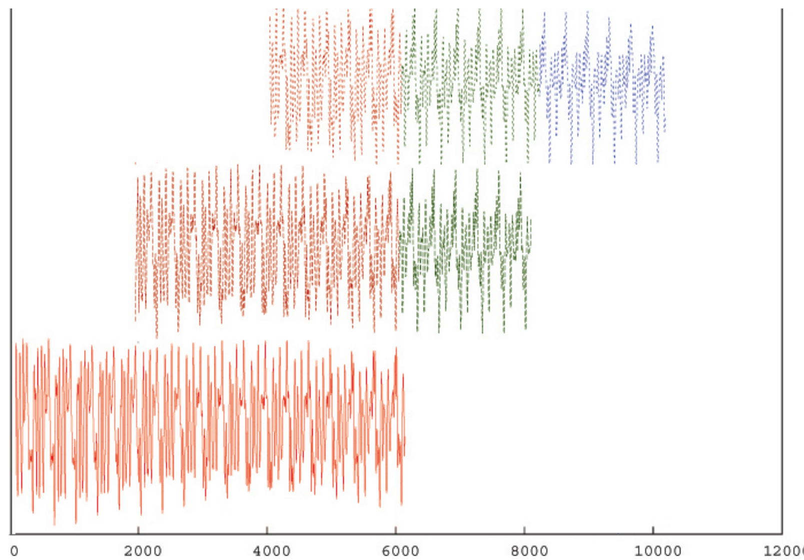

Figure 1: Three proper sample that shows the superposition between them.

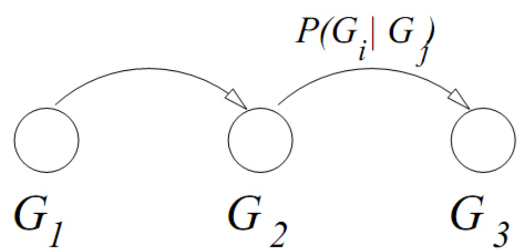

Figure 2: Finite state showing transitions probability $P(G_s|G_{s+1})$ from note complex G_s to G_{s+1}.

transitions $G_s \rightarrow G_{s+1}$. We shall describe in subsequent sections how these are important in algorithms definition.

2.2 Data Association and Inference

As described, an important time scale in the problem is given by the *proper* sample boundary, since it is granularity of note resolution. Moreover, at each evaluation we obtain peaks and perform a data association for obtaining notes. Information from the previous *proper* sample to decide to include or exclude a potential note in G_s.

3 The Algorithms and Implementation

As is the case in many musical transcription systems, Figure 3, the basic building blocks are: *(1)* the low-level signal analysis, or front-end processing, *(2)* the subharmonic association or note analysis phase, and *(3)* the backend notation and transcription phase.

In the first block, the first issue to do is reading the audio (.wav) file and getting preparing the number of points in the sample. The `FrontEnd` block is responsible for obtaining a list of reliable peaks, and the

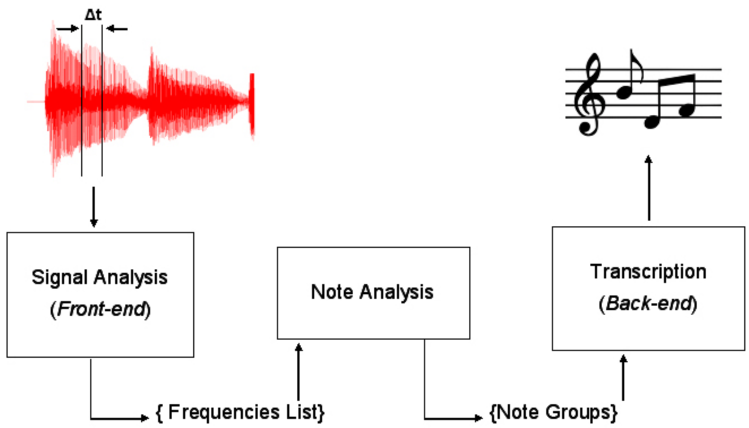

Figure 3: Musical Transcription System.

function `Note Analysis` contains all the algorithms for associating peaks to notes, which are determined from rules based upon prior samples. Each of these blocks shall now be described in subsequent sections.

3.1 Low-level Signal Analysis

We have found that the low level signal analysis is responsible for obtaining robust transcription results in subsequent algorithms in the system. The specific parameters of our front-end system, implemented with the FFTw library, are: *(a)* a timeslice window of approximately 2.5 ms gives good overall results for even fast notes, *(b)* the windowed STFT chosen is a Chebyshev, with 100dB sidelobe attenuation, and *(c)* accurate threshold based peak detection to nearly 5kHz.

Peak Threshold function For obtaining the peaks, we developed an ad-hoc thresholding algorithm, which consists of: *(a)* a moving average $\mathcal{M}$ and *(b)* a nonlinear fitting function $\mathcal{F}$ to adjust for the background power level across a large frequency range. Thus, once the full threshold curve $T(\omega) = \mathcal{M} + \mathcal{F}$ is obtained, the peaks are identified as those points greater than $T(\omega)$.

The background power spectrum: We know from physics, that the power spectrum always has the form: $P(\omega) = a_0 + (a_1 * \omega) \exp(-a_2 * \omega)$ Thus, we can model very well both the low and high frequency background if we could fit this function to the actual spectrum. Our procedure consist in obtaining the constants a_i in the above formula by using the well known Levenberg Marquadt nonlinear fitting algorithm. So that the algorithm converges to appropriate values for a_i, we have chosen initial guesses by simple pre-calculations directly from the power spectrum $P(\omega)$. We have found that it is more important that the fit accurate at high frequency, so we have payed special attention to heavily weight the initial guesses by the average value of the *asymptotic* power spectrum, so we set $a_0 \approx \sum_{k=N-m}^{N} |E_k|$.

The moving average: The moving average provides an excellent centered average the function. A simple way of writing the moving average at point x_j is $\bar{x}_j = 1/(2p+1) \sum_{k=j-p}^{j+p} x_k$. After experimenting with several orders of the moving average, we found that the 5-point and 7-point averages give the best results, while higher order tend to be far too slowly varying.

Finally, it is necessary to eliminate adjacent points. The algorithm is a standard linked list operation, which consists of bracketing the adjacent nodes, finding the maximum value within the bracketed set, and eliminating all but the maximum. A typical example of the peak detection algorithm is shown in Figure 4. The result of these algorithms is to produce the list (set): $F = \{p_o, \cdots, p_n\}$.

3.2 Peaks/Note Association Algorithms

Once the set of peaks F_k are obtained from the power spectrum of proper sample k, we must associate these peaks to subharmonics of notes n_j, which may eventually make up a note complex G_s. In peak/note association, the frequency of each peak ω_i is tested against every other $\omega_n \in F$ in order to determine

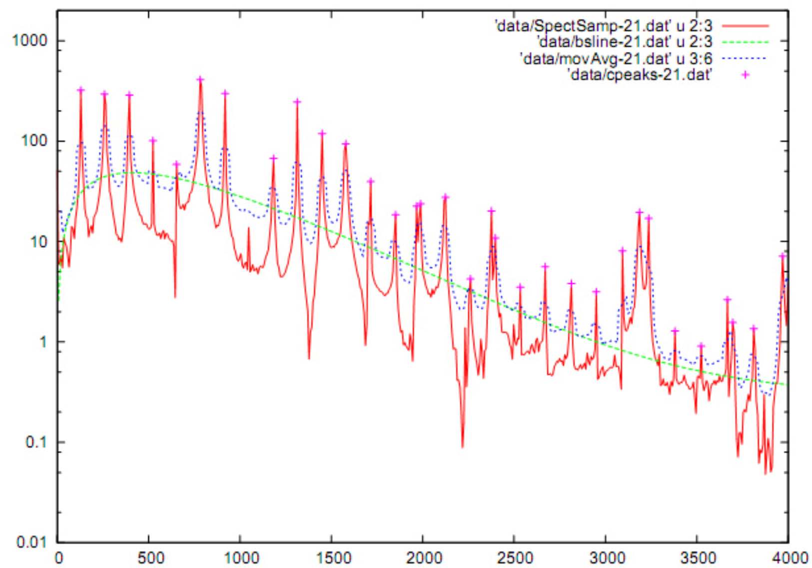

Figure 4: Example power spectrum for peak detection.

if the relation $\omega_n = n\omega_i$. Once a complete enumeration of all such groupings of multiples is performed, the fundamental frequency is determined, and hence the note n_j through the linear relation $n(\omega) \approx 2/log(2)log(\omega/440)$.

The note association is greatly reduced by starting with a subset of potential *candidate* peaks C. A simple selection method is to search through the list of all frequencies in $F_k = \{(\omega_0, a_0), \cdots (\omega_n, a_n)\}$ with associated amplitudes a_j and choose the set of ω_j, for $j < n$ where the relative amplitude is greater than a predetermined threshold $a_j^* = a_j/a_{max} > a_t$. It should be mentioned that this simple selection criteria works fairly well for piano music but has not been fully tested for other instruments which may exhibit missing fundamentals ω_o.

Definition 2 (Candidates List C_k) *A subset of notes obtained from the original set of spectrum peaks, F. The selection is a dynamic programming technique which eliminates all other possible combinations and only selects optimal candidates. Subsequent iterations will only consider these candidate peaks as starting points for constructing notes.*

Once the candidate list C_k is obtained, we use this list as potential fundamental frequencies $\omega_{0,j}$ for notes n_j and enumerate through the entire list F_k looking for multiples of it in a similar manner as described, with $\omega_n = n\omega_{0,j} \pm \delta\omega$, where in practice we allow for a radius of error $\delta\omega$. The resulting list is referred to as the *Preliminary list* P_k, because there may be *apparent* notes n_j constructed from the association process which are really spurious or accidental signals. We formalize the definition of the *preliminary list* in the following way:

Definition 3 (Preliminary List P_k) *A subset of notes, at sample k, obtained from peak to note association. This list contains potential notes which must be verified by examining conditional probabilities of previous preliminary lists from the $k-1$ sample, P_{k-1}. We calculate the conditional probability for the $j-th$ note $n_j^k \in P_k$ by observing the set of features θ.*

Definition 4 (The Feature Vector θ) *Observable parameters for each note $n_j^{(k)}$ that we obtain from the kth sample, that include power spectrum properties, number and distribution of subharmonics, energy values and specific amplitude information. These parameters are used for calculating transition probabilities.*

3.2.1 Parameters

The parameters θ of the note make a reference to physical observable values between the subharmonics. These values are easily calculated from information we obtain from the spectrum of the sample.

- Spectrum values: *(a)* Note value (n_ρ), which is the integer value of the note and the frequency using fitted model $n(f) = a_0 \log(a_1 f/440)$, with constants $a_0 = 30.26$ and $a_1 = 0.99$, *(b)* the sample number and time (s_k, t_k), which is the *proper* sample number and the time, *(c)* Amplitude and frequency (a_j^ρ, f_j^ρ) of note n_ρ in the note complex C, *(d)* the number of harmonics (N_h) for note n_ρ.
- Energy derived values: *(a)* E_t^ρ, the total energy obtained by summing amplitude of the subharmonics, $E_t^\rho = \sum_k |a_k^\rho|^2$ for note n_ρ, *(b)* energy of the fundamental and of the maximum are $E_{fo}^\rho = |a_{fo}|^2$ and $E_{fmax}^\rho = |a_{fmax}|^2$ respectively, *(c)* the relative energy is given by $E_r^\rho = E_t^\rho / E_{fmax}$, and *(d)* the gradient of the total energy, ∇E_t is used to test for the onset repeated notes.
- Distribution of subharmonics: An important heuristic for associating notes to note complexes is based upon the distribution of absent harmonics. There are three quantities of this type which are of interest: *(a)* the total number $n_\phi = \sum_k \phi_k$, η of missing harmonics (considered in a consecutive series), *(b)* individual structure functions ξ, where we define

$$\xi_i(k) = \sum_k^{N_h} \phi_k H(k_j - k_i) = \begin{cases} 1 & k_j \leq k \leq k_i \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

and *(c)* a weighted distribution η_ϕ, given through the definition: $\eta_\phi = \sum_{k_m \in \{\phi_k\}} w(k) k_m$, with $k_m \in \{\phi_k\}$ and $w(k) = exp(-\lambda * k)$

Upon low-level processing of the subsequent sample $k+1$, we perform an update step. In particular, we rely upon the transition probability matrix R_k between the preliminary lists P_{k-1} and P_k.

Definition 5 (Transitional Probabilities) *Given a set of preliminary notes, $P_k = \{n_1^k, n_2^k, \cdots, n_j^k\}$ obtained from the proper sample k and its immediate predecessor $P_{k-1} = \{n_1^{k-1}, n_2^{k-1}, \cdots, n_j^{k-1}\}$, where $n_j^k = n_j^k(\{\theta_i\})$ is the j-th note in the k sample and depends upon the observable parameters $\{\theta_i\}$, we define the matrix R_k of conditional probabilities:*

$$R_k = \left\{ r_1\left(n_1^k(\theta)|n_1^{k-1}(\theta)\right), \cdots, r_j\left(n_j^k(\theta)|n_j^{k-1}(\theta)\right) \right\} \tag{2}$$

The elements $r_{i,j}$ of this matrix are formed by considering $\{\theta_i\}$, of each n_j^k, depend on the value of this parameter and the previous sample.

$$r_j\left(n_j^k(\{\theta_i\}|n_j^{k-1}(\{\theta_i\}\right) = 1 - \frac{|n_j^k(\{\theta_i\}) - n_j^{k-1}(\{\theta_i\})|}{\max(n_j^k, n_j^{k+1})} \tag{3}$$

3.2.2 Heuristic Decision Rules

Given the subset of parameters of the preliminary note p_j, and the associated conditional probabilities r_j, we define a series fo generic heuristic rules that will permit us to determine the note and whether they can form a part of the A_k list. Table 1 shows the parameters and nominal rules empirically determined, for harmonic distribution and energy determination.

The last case shows two special cases which pose ambiguities if not careful: *(1) case of determining octaves:* where we have found that fourth harmonic must obey the condition $a_4 \geq 0.5a_0$, which unambiguously determines the correct octave, and *(2) case of repeated notes:* since our method is based upon the idea that onsets are determined by changes in the set of harmonics, we need to observer $\nabla E \geq 0$ and $r(E_t) \geq 0$.

The *subset* of notes from the preliminary list P_k which have been confirmed from the probabilities of R_k, are then promoted to the *Best note* list, or the A_k list. We define this list in the following manner:

Table 1: Empirically determined Heuristic Rules for inclusion in $\mathrm{A_k}$.

Type	Parameters	Nominal Rule
Harmonic Dist.	$Card(N_h) > N_h^{(\mathrm{opt})}$	$N_h^{(\mathrm{opt})} = 6$ nominal
	cond. prob. $r(N_h)$	$r(N_h) > 0.75$
	$\xi_2(k)$	$\xi_2(k) \leq 1$
	$r(\xi_2(k))$	$r(\xi_2(k)) \geq 0.5$
	η_ϕ	$\eta_\phi \leq 10$
	$r(\eta_\phi)$	$r(\eta_\phi) \geq 0.7$
Energy	Tot.rel. $E_r(n_j) = \frac{\sum_i a_i}{a_{max}}$	$E_r^{(\mathrm{opt})} \geq 0.2$
	cond. prob. $r(E_r)$	$r(E_r) > 0.4$
	∇E	$\langle \nabla E \rangle$
	Octaves	$a_4 \geq 0.5 a_0$
	Repeated notes	$\nabla E \geq 0$, $r(E_t) \geq 0.3$

Definition 6 (α-List or Best Note List (A_k)) *The set of notes $\{n_\rho\}$, that form the list $\mathcal{R}_k$ -the best notes in the actual proper sample-, obtained from P_k which have been selected by decision rules based upon the list $\mathcal{R}_k$, that contain the best notes in the actual proper sample. Furthermore, the notes $\{n_\rho\}$ constitutes a subset of the note complex G_s during sample k.*

Operationally, the notes in A_k are considered notes and their timing information is saved for back-end processing. It is necessary to update the α-List from the decision rules and selecting the notes n_ρ from P_k which will be *promoted* to A_k. As indicated in the definition, A_k list represents all the notes in the note complex G_s during time Δt_k, so $G_s(\Delta t_k) = A_k$. At $k+1$ more notes can *enter* into G_s.

Mathematically, the state of A_k for sample k is obtained from the function $A_k = A_k(A_{k-1}, R_k, \mathcal{R}_k)$. In particular, we can write the *state equation for* A_k in terms of the quantity $\mathcal{R}_k$, which is the result list of the heuristic decision rule.

$$\mathrm{A_k} = \begin{cases} \mathrm{A_{k-1}} \cup \kappa & \mathrm{card}(\mathcal{R}_k) \geq \mathrm{card}(\mathcal{R}_{k-1}) \\ & \text{where } \kappa = \mathcal{R}_k - \mathrm{A_{k-1}} \\ \mathrm{A_{k-1}} - \kappa' & \text{otherwise} \\ & \quad \text{where } \kappa' = \mathrm{A_{k-1}} - \mathcal{R}_\mathrm{k} \end{cases} \tag{4}$$

The transition from a note complex G_s to G_{s+1} corresponds to the list $A = \emptyset$, that is there are no valid notes present in the signal (when an onset takes place). All the notes in G_s that were voiced can now be written to the back-end processor. We have used an additional list, referred to as the B_k list, for temporarily storing these notes $n_\rho \in G_s$ and their associated onset/offset times t_i and t_f respectively.

Definition 7 (Back-end B_k List) *A temporary storage list which contains the entire note group which exists during a time Δt. It is what gets passed to the back-end processor for writing out a musical score.*

$$\mathrm{B_k} = \begin{cases} \emptyset & \mathrm{A_{k-1}} = 0 \\ (\mathrm{A_{k-1}} - \mathrm{A_k}) \cup \mathrm{B_{k-1}} & \text{otherwise} \end{cases} \tag{5}$$

Once the note complex is written to B_k, the full back-end stage is called for writing the musical notation, which in our case is done by hand-crafted scripts for GNU Lilypond.

3.3 Algorithm Operation

A demonstration of the algorithm for a hypothetical case useful to demonstrate the steps. Consider the following definitions: *(a)* an individual note $n_i^{(j)}$, as before, is represented with two indices, i and j, where j is the note complex, and i is an indices counting the number of notes which enter the alpha queue, (but may not necessarily be a final note), *(b)* the time t_k is the fundamental time tick; it is the absolute time in the segment measured in the middle of the proper sample, and *(c)* $(\Delta t)_j$ is the duration of a note group, while Δt_i is the duration of an individual note.

Table 2 shows a hypothetical sequence of samples and the associated detected notes with the state of each of the queues in the algorithm described above.

Table 2: State of the Queues..

time	**Prelim.Note**	**Lists** A_k,B_k
t_1	$\{n_1, \mathbf{n_2}, \mathbf{n_3}\}$	$A_1 = \{n_2\}$,$B_1 = \{\}$
t_2	$\{n_1, \mathbf{n_2}, \mathbf{n_3}, n_4\}$	$A_2 = \{n_2, n_3\}$, $B_2 = \{\}$
t_3	$\{\mathbf{n_2}, \mathbf{n_3}\}$	$A_3 = \{n_2, n_3\}$, $B_2 = \{\}$
t_4	$\{n_2, n_3, n_5, n_6\}$	$A_4 = \{n_2, n_3\}$, $B_2 = \{\}$
t_5	$\{\mathbf{n_5}, \mathbf{n_6}\}$	$A_5 = \{\}$, $B_5 = \{n_2(t_1, t_4)$ $n_3(t_1, t_4)\}$
t_5	$\{\mathbf{n_5}, \mathbf{n_6}\}$	$A_5 = \{n_5, n_6\}$, $B_5 = \{\}$
t_6	$\{\mathbf{n_5}, \mathbf{n_6}\}$	$A_6 = \{n_5, n_6\}$, $B_6 = \{\}$
t_7	$\{\mathbf{n_6}, \mathbf{n_7}\}$	$A_7 = \{n_6\}$, $B_7 = \{n_5(t_5, t_7)\}$
t_8	$\{\mathbf{n_6}, \mathbf{n_7}, n_8\}$	$A_8 = \{n_6, n_7\}$, $B_8 = \{n_5(t_5, t_7)\}$
t_9	$\{\mathbf{n_6}, \mathbf{n_7}, n_9\}$	$A_9 = \{n_6, n_7\}$, $B_9 = \{n_5(t_5, t_7)\}$
t_{10}	$\{\}$	$\{\}$ $\{n_5(t_5, t_7),$ $n_6(t_5, t_9), n_7(t_5, t_9)\}$

The following is a short description of a typical situation of how the algorithm works.

- time $\mathbf{t_1}$: from the frequencies and from the subharmonic grouping, three potential notes are placed into the preliminary list. Imagine that of these three notes, n_1 has a low probability, determined by the number of subharmonics pertaining to it; notes n_2 and n_3 have a high probability of being real notes, yet only n_2 has a sufficiently high probability for entering into the A_k list directly, so it is copied into the A_k list.

- time $\mathbf{t_2}$; probability of n_1 is the same, however the note n_2 and n_3 are confirmed by previous observations; also another note n_4 is a potential candidate and must wait for further samples before entering into A_k.

- time $\mathbf{t_3}$: n_4 does not appear so is eliminated.

- time $\mathbf{t_4}$: change of notes; some energy of n_2 and n_3 is present but weak compared to n_5 and n_6; algorithm defers decision to include in A_k.

- time $\mathbf{t_5}$: A_k is emptied and its contents are copied to the B_k list; note complex has concluded and can be written; preliminary note list P_k contains notes n_5 and n_6. Since they were present previously, they enter the A_k complex.

- time $\mathbf{t_6}$: n_5 and n_6 appear again, so they remain in A_k list.

- time $\mathbf{t_7}$: note n_5 disappears from P_k, and it so it gets subtracted from A_k list and gets written to B_k with onset t_5 and offset time t_7.

- time $\mathbf{t_8}$: note n_7 is confirmed and written to A_k; the note n_8 enters into P_k with low probability since it has few subharmonics a low amplitude fundamental; a decision for inserting n_8 into A_k is deferred until the next sample.

- time $\mathbf{t_9}$: the decision defer n_8 into A_k is justified since it is no longer present; the note n_9 appears but not yet included into A_k.

- time $\mathbf{t_{10}}$: all notes disappear, since there are no observed subharmonics; the condition $A_k = \emptyset$ signals writing all contents to B_k and calling the backend processor. The list contains the full note complex $G_s = \{n_5, n_6, n_7\}$ with the onset/offset times indicated in the table.

What is not shown here are numbers indicating how the decisions are actually made to include notes in A_k which come from P_k. This is the subject of the next subsection.

3.4 Results

A real example, Figure 5, is describing using the next tables. The audio sample has been recording with a Yamaha electric piano.

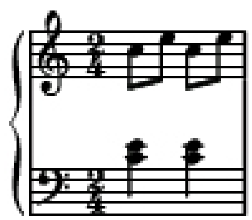

Figure 5: Musical segment example.

The first table, Table 3 show the functioning of the front-end of the algorithm. It shows the couple frequency/amplitude for the first group of notes, the number of subharmonic, and the present notes.

Table 3: Example of a samples and frequency/amplitude for the first four harmonics.

Sample (*ind*)	*Note* (N_h)	*Subharmonic* (f, a)
69 (0)	-9 (23)	$\{(266,444)(524,1005)(790,109)(1048,290)\}$
69 (1)	3 (11)	$\{(524,1005)(990,154)(1579,199)(1995,98)\}$
69 (2)	7 (9)	$\{(660,558)(1313,125)(1845,117)(2512,7)\}$
...	...	...
77 (0)	-5 (24)	$\{(330,328)(660,884)(990,230)(1313,102)\}$
77 (1)	3 (18)	$\{(524,1600)(990,230)(1579,297)(1995,151)\}$
77 (2)	7 (18)	$\{(660,884)(1313,102)(1845,71)(2577,7)\}$
...	...	...

In Table 4, you can see the diferent values from the observable parameters that we use for forming the R-matrix. We can see the probabilities in Table 5 .

Table 4: Representative samples for the observable parameters.

n_s	n_ρ	f_j^ρ	N_h	E_t	E_f	n_ϕ	η_ϕ	(ξ_1, ξ_2, ξ_3)
69	-9	266	23	2510	444	0	0	(0, 0, 0)
69	3	524	11	1482	1005	4	7.9	(0, 0, 4)
69	7	660	9	827	558	5	10.5	(0, 2, 3)
72	-9	258	12	1573	259	7	11.7	(0, 0, 7)
72	3	524	10	1082	761	4	8.1	(0, 1, 3)
72	7	660	8	425	219	7	15.3	(0, 4, 3)
...	...	...	...	...	...	...	...	...

Finally, Table 6 shows the time sample, the preliminary lists and A_k and B_k list as described throughout the paper.

Table 5: Example of matrix elements of R.

n_s	$r(f_j^\rho)$	$r(N_h)$	$r(E_t)$, $r(E_f)$	$r(n_\phi)$, $r(\eta_\phi)$	$r(\xi_1)$, $r(\xi_2)$, $r(\xi_3)$	$r(\nabla E)$
69	-9	1	(0.5, 0.5)	(1, 1)	(1, 1, 1)	1170
69	3	0.9	(0.4, 0.4)	(0.7, 0.7)	(1, 1, 0.75)	760
69	7	0.9	(0.4, 0.4)	(0.8, 0.7)	(1, 0.5, 1)	438
72	-9	0.6	(0.6, 0.6)	(0.8, 0.7)	(1, 1, 0.8)	-912
72	3	0.8	(0.6, 0.6)	(0.7, 0.7)	(1, 0, 1)	-633
72	7	0.6	(0.4, 0.3)	(0.2, 0.2)	(1, 0, 0.6)	-493
...	...	...	...	...	...	...

Table 6: Example of status of the major lists for note complexes.

t_S	P_{k-1}	P_k	A_k,	B_k
68	$\{\}$	$\{-9,-1,3,7\}$	$\{\}$	$\{\}$
69	$\{-9,-1,3,7\}$	$\{-9,3,7\}$	$\{-9,3,7\}$	$\{\}$
70	$\{-9,3,7\}$	$\{-9,3,7\}$	$\{-9,3,7\}$	$\{\}$
75	$\{-9,3,7\}$	$\{3,7\}$	$\{3,7\}$	$\{-9\}$
76	$\{3,7\}$	$\{3,7,15\}$	$\{3,7\}$	$\{-9\}$
77	$\{3,7,15\}$	$\{-5,3,7\}$	$\{-5,3,7\}$	$\{-9\}$
82	$\{-5,3,7\}$	$\{-5\}$	$\{\}$	$\{-9,7,3,-5\}$
83	$\{-5\}$	$\{-9,3,7\}$	$\{-9,3,7\}$	$\{\}$
...	...	...	...	...

4 Conclusions

Our method for constructing note complexes accurately identifies notes from the association of peaks to notes and using deferred decision making based upon conditional probabilities of physical observable between samples.

It is interesting to emphasize that for the determination of the notes included in a segment, we only analyze the information of the present proper sample and some data of the previous one. Also, the probability matrix definition to calculate the heuristic values of each note becomes of great utility to take decisions. All parameters can be observed in the frequency domain. It means that it is easy to apply the method to another instruments with the only condition of knowing the unique profile of each one.

On the other hand, we can obtain an accuracy between the 70% for complex polyphonic samples and 98% for the simplest ones. These results establish our method on the level of success obtained with another research works. Furthermore, we can solve with some success simple cases of determining octaves and repeated notes problems. Although, it is necessary to keep working in it.

A relevant aspect for us in order to continue our work is to implement *adaptive own samples* inspired by the concept of adaptive meshes of finite element. The idea is to increase the sample size for segments with very long notes in the timeline, reducing it for those segments with short notes.

According with the situation aforementioned, the main idea with the proposed ensemble forecasting aproach is to use a two steps algorithm implementation that, in a first approximation, analyze the energy of the time domain signal contour and then decide when to vary the proper sample size for later frequency analysis.

References

[1] B. Bank. Physics-based sound synthesis of the piano. Master's thesis, Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology, 2000.

[2] S. Gomez-Meire. Resumen de tesis: Transcripcion de musica polifonica para piano basada en la resolucion de grupos de notas y estados *Inteligencia Artificial*, 14(45):44–47, 2010. doi: 10.4114/ia.v14i45,1090.

[3] M. Puckette J. Brown. A high resolution fundamental frequency determination based on phase changes of the fourier transform. *Journal of the Acoustical Society of America*, 94:662–667, 1993. doi: 10.1121/1.406883.

[4] A. Klapuri. *Signal processing Methods for the Automatic Transcription of Music*. PhD thesis, Department of Information Technology, Tampere University of Technology, 2004.

[5] M. Leca L. Rossi, G. Girolami. Identification of polyphonic piano signals. *Acustica*, 83(6):1077–1084, 1997.

[6] A. Sterian. *Model Based Segmentation of Time-Frequency Images for Musical Transcription.* PhD thesis, University of Michigan, 1999.