MULTI-PITCH ESTIMATION FOR POLYPHONIC MUSICAL SIGNALS

P. Fernández-Cid, F.J. Casajús-Quirós

SSR, Escuela Técnica Superior de Ingenieros de Telecomunicación, UPM
Ciudad Universitaria s/n
28040-Madrid, SPAIN
http://www.gaps.ssr.upm.es/pablofc/index.html

ABSTRACT

Automatic Score Transcription goal is to achieve an score-like (notes pitches through time) representation from musical signals. Reliable pitch extraction methods for monophonic signals exist, but polyphonic signals are much more difficult, often ambiguous, to analyze.

We propose a computationally efficient technique for automatic recognition of notes from a polyphonic signal. It looks for correctly shaped (magnitude and phase wise) peaks in a, time and frequency oversampled, multiscale decomposition of the signal. Peaks (partial candidates) get accepted/discarded by their match to the window spectrum shape and continuity-across-scale constraints. The final partial list builds a resharpened and equalized spectrum. Note candidates are found searching for harmonic patterns. Perceptual and source based rejection criteria help discard false notes, frame-by-frame. Slightly non-causal postprocessing uses continuity (across a <150 ms. observation time) to kill too short notes, fill in the gaps, and correct (sub)octave jumps.

1. INTRODUCTION

Many methods have been reported for automatic monophonic speech and music pitch analysis. Applications include coding, recognition, time/pitch stretching, pitch-to-MIDI, etc. [3],[4].

These models fail if exposed to polyphonic signals where different voices or notes are mixed. Even the case of reverberant monophonic signals is prone to errors (reverb enlarges the effective note duration, and signal is not perfectly monophonic).

Bregman´s book [1] has been a turning point for a growing group of researches on Computational Auditory Scene Analysis, which try to build useful models of acoustic signal high-level interpretation. Physically-motivated ear models [6] are the preferred (though computationally intensive) signal processing front-end, sometimes implemented with wavelets. Their outputs feed 2D and 3D filters that enhance meaningful perceptual clues for source streaming (onset time, frequency and amplitude variation rate, etc. [2]). Streaming itself is accomplished by ad-hoc rules or AI (blackboard, multiagent) techniques for competition and collaboration among different grouping criteria. Harmonic FLLs, multiscale techniques and prediction-reconciliation schemes have also been proposed (see [5]).

In spite of the heavy computational load of these methods, results are (so far) not up to the job. Additionally, the harmonicity cue for grouping has only been briefly used, though it is a very salient characteristic of musical signals.

We propose a new front-end, and then concentrate in the use of harmonicity for correct grouping of partials. Our focus is on pitch transcription, but results are encouraging as a basis for source separation, using the partial list assigned to each pitch to feed an additive resynthesis system.

2. MULTISCALE SINUSOIDAL MODEL

We use a multiscale sinusoidal model for the analysis of the polyphonic musical signal. The final goal of this front-end is to obtain a representative set of partials (frequency, amplitude and phase) for each time frame. Resynthesis confirms the opportunity of this decomposition. Multiple fft (with different window lengths) are used to better discriminate the partials at the scale they fit the most. Results of each scale are combined, ending up with a single partial list for each time frame. Partial interference is taken into account.

One of the most remarkable characteristics of musical signals is the huge range of the fundamental frequency, spanning many octaves. Additionally, the partials from different notes in polyphonic signals can lay very close together or even collide. No single window can give a proper resolution over the whole range.

Timbral and dynamic behavior of musical signals is also to be considered. Acoustic musical instruments produce sound linked (by design) to our hearing capabilities. Very low pitched notes are difficult to be heard and they are usually played more statically, with longer note durations and lack of ornaments. In order to produce high pitched notes, both the instruments and the players are subject to a heavier effort, making this high notes more unstable and modulated. Strong sharp resonances are not uncommon in low/medium pitched notes, but are rare on high pitched notes because as the frequency rises so decreases our capability to discern the precise pitch, and we need a clear, well defined fundamental to perceive the proper note.

2.1 Filterbank

This discussion suggests the need of a multiscale front-end. Many researches parallel the known behavior of our ear using a constant-Q or 3rd octave filterbank coupled with non-linearities that try to simulate the cochlea. Mid and upper frequency channels of such decomposition mix different partials. With monophonic signals, this mix beats at the period of the fundamental. For polyphonic signals unrelated partials beat meaninglessly on a single channel.

The (windowed) fft approach warranties a uniform view of the spectral contents of a signal, the problem being the window selection compromise between frequency resolution and inside-window stability of the signal. Sinusoidal models (peak detection on an fft) are common for monophonic signal analysis.

Our front-end (fig.1) uses a time-aligned non-complementary filterbank. The fft of each output is a detailed (doubled spectral resolution) look at the first half of the previous spectrum. A multiscale peak search is made on this set of 4 spectra.

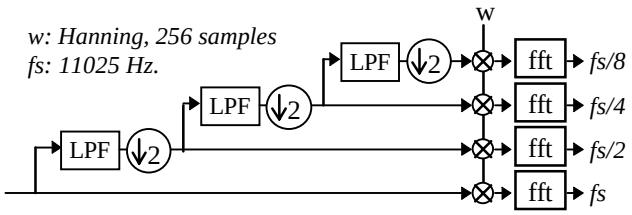

Figure 1. Filterbank diagram. LPF is half-band lowpass. A single Hanning window covers different time spans at each scale due to factor 2 decimation on LPF outputs.

Short windows are well matched to (usually unstable) high pitched partials. If they are high order harmonics of a low note they will be of little amplitude and of little importance for the global pitch of the note (the lack of frequency resolution is not a problem). If they are partials of a high pitched note, they will have noticeable energy and they will be sparse (distant to each other), so the lack of frequency resolution is neither a problem (also the log human pitch perception should be considered).

Long windows are only applied to lower spectral regions. They are needed to resolve the partials of low/mid notes. Sometimes partials lay so close together that very long windows are needed, but cannot be applied (the notes themselves don't hold for so long statically). That's why we only use 4 steps in the filterbank. Therefore, some partial collision should be allowed.

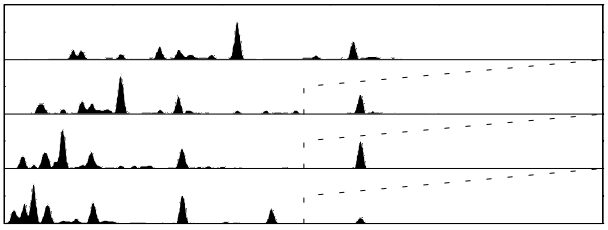

Figure 2. Spectra at 4 different scales. Bottom to top, each scale doubles frequency resolution but spans half the bandwidth (piano chord + soloist woodwind note).

We calculate the fft with an excess factor in the number of bins (8 times the number of time samples). This makes available a detailed shape for each peak (not just a couple of points as in a point per point fft). When two partials are very close to each other, the peaks of both partials can be still seen in the enlarged resolution fft, and an attempt can be made to discover both partials from the detailed shape of the spectrum in the nearbies.

Frames advance at 11.6 ms irrelevant of scale: frame overlap is 50% at the full band scale (128 samples at 11025 fs), and greater at successive scales.

The process of partial list creation has two steps. First, a partial list is found at each scale. Then results from the 4 scales are combined into a single final partial list.

2.2 Per-scale partial list detection

Initially every peak in the fft is found. Too low leveled peaks are discarded. For each of the remaining peaks the surrounding bins are compared with the theoretical peak -the main lobe of the window spectrum-. A 'quality of fit' for the peak is measured as:

$$\frac{\sum_{\xi}\left|S(k+\xi)-A\cdot W(\xi)\right|^2}{\sum_{\xi}\left|S(k+\xi)\right|^2} \quad \text{where} \quad A=\frac{\sum_{\xi}\left|S(k+\xi)\cdot W(\xi)\right|}{\sum_{\xi}\left|W(\xi)\right|^2}$$

S is the signal complex spectrum, W is the window complex spectrum, k is the bin index of the peak, and the sum spans a range ξ of bins that corresponds to the width of the main lobe of W. This formula is a partial-only-focused version of our previous harmonic matching algorithm for monophonic signals [3]. It constitutes a kind of quality of fit between the measured peak and the expected ideal peak, and is related to the (energy normalized) least square difference between real and ideal peaks.

If the quality is not good enough, maybe the partial is subject to disturbing influence of nearby partials. If another peak lies closer than the window main lobe width, a new opportunity for the peak to be confirmed is given. Instead of trying to emulate the candidate peak nearbies with a single window main lobe, two are summed located at the candidate peak and at the interfering peak (with the corresponding amplitude and phase). Then, the previous formula is reevaluated with the sum instead of W.

Too badly shaped peaks are eliminated from the initial temptative peak list. This initial peak selection (the decision level) is somewhat lax, so the remaining peaks are not still validated, but are subject to an across-scale confirmation.

2.3 Final partial list compilation

Validation comes from a combination of the surviving partials at the 4 scales. We begin studying the whole band scale, and then proceed to halved band scales in order.

If a partial exists simultaneously (at almost the same frequency) at the current scale and one or more consecutive scales, it gets immediately validated.

If the partial has no continuation in the next scale partial list, it can only be confirmed from the current scale. If it lies in the upper mid of the spectrum (the next scale has no support for this frequency) its 'quality of fit' parameter is checked to be better than a minimum (more restrictive than in the intra-scale partial validation). If it lies in the lower mid, the 'quality of fit' should be better that an even more restrictive minimum (the lack of a correspondent peak in the next scale should be compensated by a better quality of fit).

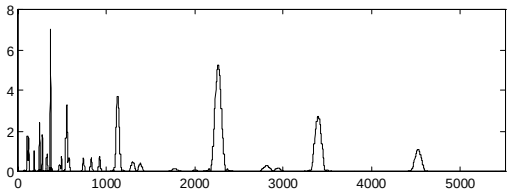

Figure 3. Final partial set for the previous example. Width of each partial is used here to show the scale at which they have been validated.

2.4 Masking

The results so far are then tested for masking effects. We build a mask by locating a bell shape at the bin of each confirmed partial. The bell height is that of the partial. The bell shape is:

$$\frac{100^{Hann(N)}-1}{99}$$

Hann(N) is the length N Hanning window and is used (point per point) as a power; N is calculated to span 1.5 semitones at each side of the peak. Final shape is similar to the response of Gammatone (≡cochlear) filters but simpler to compute.

The mask is not calculated as the sum of all the bells, as this would make the mask incorrectly tall in between close peaks. For each bin the highest single bell contribution is considered.

Partials whose amplitude does not surpass 97% of the value of the mask are discarded (most of the times second order lobes, sidelobes of modulated partials, or noisy peaks).

3. NOTE IDENTIFICATION

3.1 Partial grouping

The quality of the partial list obtained in the previous steps, has been assessed by resynthesis: it constitutes a valid representation of the original signal. But we need to classify partials into notes.

The 'harmonic sum' of the spectrum at equispaced bins has been used to test for prominent comb partial patterns, but leads to octave errors (subproducts combinations of real notes, etc.).

We tried the harmonic sum with various 'equalized' spectra (partial enhancement by means of different kinds of non linear filtering of the spectrum), and obtained the best results when substituting the original spectrum by a new 'synthetic' spectrum made from 'sharpened' peaks at the positions of the confirmed partials from the previous analysis. The height of the synthetic partials was not its amplitude, but 1 minus the 'quality of fit' parameter. This had 2 effects: a) the quality of fit of already confirmed partials is not too distant to zero, so the heights of synthetic partials get effectively equalized; b) the sharpening of the partials gives greater importance to the correct centering of the measured partials at the true multiples of the fundamental.

It became quite noticeable that it was the centering of the partials the primal criteria for grouping and we decided to change the grouping criteria from the modified harmonic sum to a new rule based one, which has proved much more reliable.

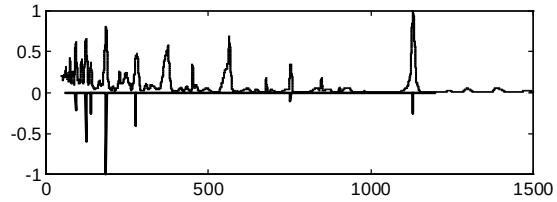

Figure 4. Traditional (spectral) harmonic sum (positive values) against resharpened and equalized partials harmonic sum (negative values; some false note discard criteria have also been applied -see later-).

For each note candidate in the desired pitch range, the list of confirmed partials is searched for those partials that lay closer to the 10 first true multiples of the fundamental (only the early harmonics contribute meaningfully to the sense of pitch).

Each note candidate can then be assigned an amplitude pattern, a 'centering' pattern and a partial list (index of partials temptatively assigned to the note). Once this selection (for a given note candidate) has been made, we can talk about 'harmonics' of the note candidate. If some harmonic is too off-centered, it is discarded and marked as being zero amplitude. At this step, partials can be claimed as harmonics by various notes, there is no preemption in the partial to note assignment.

3.2 Note validation

Each candidate's amplitude pattern is checked: only expectable patterns are accepted. Expectable patterns may come from some source modeling, but to achieve a source independent algorithm we have tested the performance of some more general criteria:

- For low pitched candidates (<450Hz) at least 3 of the first 5 harmonics shouldn't be weak (amplitude no less than 0.05 times that of the biggest harmonic) OR at least 2 of the 3 lower ordered harmonics shouldn't be weak and 2 additional harmonics should be active.
- If the note candidate is >600Hz, the fundamental should be active and at least have an amplitude of 0.1 times that of the biggest harmonic.
- Independent of pitch, there should be at least 2 odd numbered harmonics not weak OR the harmonics with order 2, 3 and 4 should be active.
- Independent of pitch, the sum of amplitudes for odd harmonics should not be less than 0.1 times the sum for the even harmonics.

If these criteria are met, a pitch is calculated for the note as:

$$\frac{\sum_{n=1}^{10}\left[arm(n)\cdot amp(n)\cdot freq(n)/n\right]}{\sum_{n=1}^{10}\left[arm(n)\cdot amp(n)\right]}$$

freq and *amp* (vectors of harmonic frequencies and amplitudes) enhance the role of the more prominent partials, and *arm* is a weighting factor that enhances that of lowered order harmonics.

The result is a collection of pitch candidates that typically shows note candidates clustered around discrete values. A single candidate for each cluster is accepted (the one with the largest active partial count, weighted by the partial order).

The results so far still keep octave related note candidates (fig. 5). If a candidate lacks any energy at the fundamental, and the rest of its meaningful partials are already justified by higher pitched notes, the candidate is discarded because it looks like being a cross-sub-product of higher notes.

Once this subharmonic rejection has been completed, we check for multiples of true notes: if a note candidate is multiple (to a ¼ tone precision) of another lower pitched note, it is discarded.

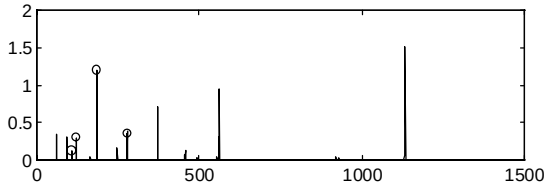

Figure 5. Selected notes (circles) for the example. Subharmonic products are correctly discarded. The 4 notes of the piano chord are detected. The woodwind note (1130 Hz) is missing because it is the 4th octave of one of the piano notes, and it is discarded.

We finally get a note map: a set of note labels through time.

3.3 Postprocessing

Up to now, the algorithm performs 'instantaneously' in the sense that it operates in a pure frame by frame manner. Results for a typical musical signal are still non acceptable, showing a large number of errors, mostly octave or suboctave jumps. But the note map can be easily classified by eye into the correct notes. This suggest some kind of postprocessing (in fact, hearing itself is just a real-time illusion, not true real-time).

The front-end has been tested by means of resynthesis, confirming the validity of the acoustic analysis. Then, the 'context' or 'history' of the signal should just be applied to partial grouping and/or note validation (including selection between octave related note candidates).

We keep a history of each alive note, as the set of values corresponding to that note during the last N time frames (N=10, <150 ms.). The note list obtained for the current frame is checked against this history: if the frequency value of a current note resembles that of some historical note, we update the historical note and delete that note from the current note list. Remaining current notes are 'new' notes, but it may happen that they are harmonically related to some historical note, and then it is better to keep the historical and not to validate the new one. We check the current notes to be similar to 2, 3, 1/2, 1/3 or ¼ times some historical note. If this is the case, historical note gets updated, and the current note is deleted from current note list.

Remaining notes in the current note list are included in the history, marked as not still alive: they need to build up some history before being accepted.

Any historical note that has not been updated yet, advances one time step with zero amplitude, signaling the lack of information about that note at that time step: lags can be later interpolated if the historical note wakes up again soon.

Now all the historical notes have been updated (some with null values) and the new notes included (they can begin to grow their own history), and its time to validate final notes.

If a note that is not still alive has already gained enough history (6 of its 10 time cells in history are not null) it should be accepted and included in the final note pool together with its history, filling the possible gaps in this initial time steps. A note that is not still alive and comes to a state where all its time cells are null is killed and deleted from history (without ever coming alive).

A note that is alive, but lacks enough history (less than 4 of its time cells are not null), must be killed: its currently stored history is saved to the note pool (again filling the gaps), and then the note disappears from history. A note that is alive and does not need to be killed saves (only) its current value to the final note pool.

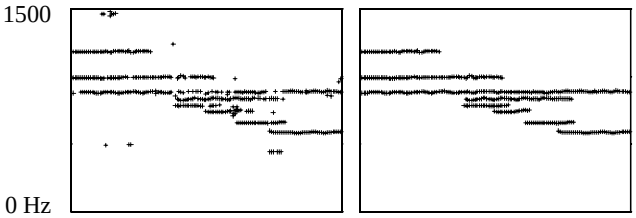

Figure 6. Note map before and after postprocessing; 2 second excerpt (3 flutes are playing, but reverberation makes up to 5 notes simultaneous at some time steps).

4. SUMMARY

A low computational cost multiscale sinusoidal model capable of extracting a meaningful set of partials representation for polyphonic signals has been presented. A method to sort this partials through time into note estimates manages to obtain an score-like representation for the original musical signal, with little non-causality.

5. REFERENCES

[1] Bregman. *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, London, 1990.

[2] G.J. Brown. *Computational Auditory Scene Analysis:a Representational Approach*. Ph.D. Thesis Sheffield, 1992.

[3] F.J. Casajús-Quirós and P.Fernández-Cid. *Real-time, loose-harmonic matching fundamental frequency estimation for musical signals*. ICASSP 1994 pp. 221-224.

[4] Choi. *RealTime Fundamental Frequency Estimation by Least-Square Fitting.* IEEE SAP 5 (2) pp 201-205. 1997.

[5] D.P.W. Ellis. *Prediction-driven computational auditory scene analysis*. Pd.D. Thesis MIT, 1996

[6] M. Slaney and R.F. Lyon. *A Perceptual Pitch Detector*. ICASSP 1990 pp. 357-360.