Proceedings of the Institute of Acoustics

PITCH SYNCHRONISATION FOR FREQUENCY DOMAIN ANALYSIS

A J H Simons

Department of Computer Science, University of Sheffield,
PO Box 600, Mappin Street, Sheffield, S1 4DU.

ABSTRACT

One of the main impediments to 'natural' data reduction in frequency-time is the tendency of parametric models to impose their prejudices on the speech signal. Pitch synchronous Fourier analysis is one way of seeking to combat the tendency of fixed-frame methods to overlook or smear important frequency information; however this presupposes an ability to detect glottal pulses accurately. We consider dynamic programming based techniques for estimating glottal cycles and periodic smoothness (after Ney 1982, 1983) and present an adaptation of Allerhand's algorithm (Allerhand, 1987) for establishing the location and size of the analysis window.

HIGH QUALITY FREQUENCY DOMAIN ANALYSIS

Extracting robust and sensitive information in frequency-time remains a major goal for automatic speech recognition. In the interests of robustness, many information extraction techniques rely on quite large sampling intervals, use large analysis windows or else use smoothing functions to extract major envelope features from finely sampled data. This conflicts with the interests of sensitivity, where quite small sampling intervals and analysis windows are required to capture important transient events.

The scale problem is sometimes addressed in a framework for progressively filtering [1] or averaging [2] finely sampled data in a hierarchical representation; however at fine scales of analysis further complications are introduced where a short analysis window captures an unrepresentative set of samples. This means that smoothing and merging adjacent frames may not be a desirable strategy where the variation from frame to frame is due more to the offset of the analysis window within the local speech event, such as a glottal cycle, than to an actual difference between acoustic equivalence classes.

Similarly, parametric methods impose their prejudices upon analysis results. A recent assessment of techniques for formant extraction [3] reported limits on the capabilities of LPC root-finding, due to problems with missing poles, or generally ill-fitting pole-zero models; and the instability of the method in noise, especially when using short windows. Fourier based methods were seen as a realistic alternative, opening the way for spectral peak analysis techniques including generalised centroids [4] and the group delay function [5]. To obtain high-quality information in the frequency domain, it was judged desirable to develop further, on the one hand, pitch-synchronous Fourier analysis methods and, on the other hand, group delay based peak enhancement methods.

PITCH SYNCHRONOUS FOURIER ANALYSIS

As an alternative to parametric, or fixed-frame analysis, self-adapting methods have greater value, since they are sensitive to scale and as a result window their data with some prior

expectation about the scale at which important phonetic detail will emerge. We shall consider pitch-synchronous Fourier analysis from this viewpoint.

In order to capture the envelope due to the vocal tract transfer function during voiced speech, the frequency resolution W = 1/NT (Hz) should be set to exactly one pitch period. Any smaller than this under-samples the waveform in time, leading to a succession of rapidly changing spectra (seen as vertical striations in conventional broadband spectrograms). Any larger than this over-samples the waveform in time, leading to the resolution of pitch harmonics in the frequency domain, which can distort estimates of envelope peaks. As well as adapting the local analysis frame size to the inter-period distance, it is also critical to synchronise the position of the window over the glottal pulse, so that the energy maximum is contained within the main lobe of the windowing function (typically Hamming).

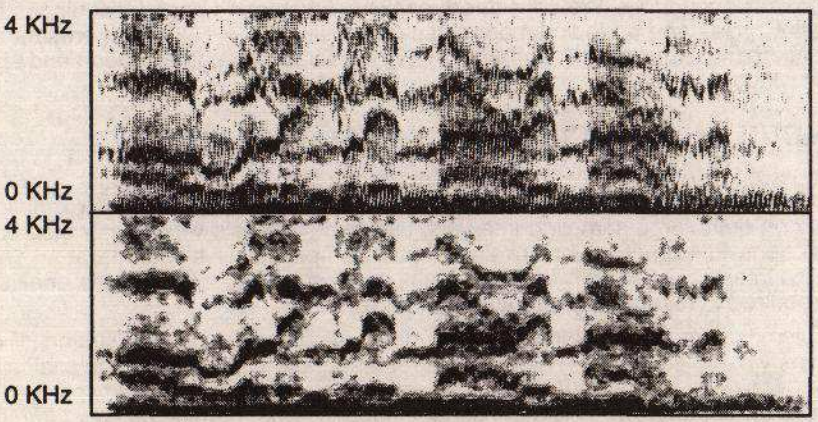

Figure 1: Short time and Gaussian smoothed Fourier transforms

In this way, glottal pulses may be analysed exactly in the time domain and pitch harmonics in the frequency domain. Not only does this better satisfy the Fourier assumption of quasi-periodicity, but in practice results in a smoother spectrogram on both time and frequency axes, achieving maximum data reduction while preserving salient features ([6], p95). Alternative methods for extracting the spectral envelope are not as accurate. If a short analysis window is chosen (for fine time resolution), any attempt to smooth the uneven spectral sequence results in a marked loss of information (Figure 1 and [6], p104). Homomorphic filtering [7], a process which deconvolves the source spectrum from the vocal tract transfer function, might seem a feasible alternative until you consider that the long window required to resolve the pitch peak in the cepstrum mitigates against fine time resolution (Figure 2).

The application of the pitch synchronous method involves calculating the location and separation of all glottal pulses during voiced speech. For these portions, the windowing function samples exactly P points where $F_0 = 1/PT$. This can be done by setting the signal

buffer size to 2^k where $2^k >= P_{max}$ and $2^{k-1} < P_{max}$; filling the buffer with the next P points starting from the zero-crossing prior to the glottal pulse; and padding the remainder with zeros. The Hamming window is centred over P/2 and shifted by P points after each calculation. At the boundaries of voiced speech, P is constrained in an arithmetic progression to default eventually to a shorter, fixed frame size for voiceless speech in order to capture transient events, such as bursts. The time-axis of the resulting spectrogram is of course non-linear and this can be rectified by interpolating a time-normalised spectrogram from recorded inter-frame distances.

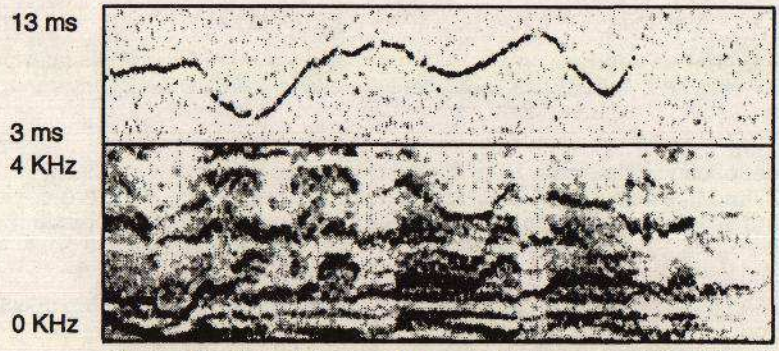

Figure 2: Running cepstrum and homomorphic filtered spectrum

GLOTTAL PULSE ESTIMATION

Pitch-synchronous Fourier analysis is highly dependent on an ability to estimate accurately the location of glottal pulses in the time waveform. A thorough introduction to the state of the art in pitch analysis is given by Hermes [8] who correctly classifies apart techniques for (physical) period estimation and (perceptual) pitch estimation. We are interested in the former. Many classic time-domain methods for F_0 estimation have used low-pass filtering, autocorrelation [9] or peak picking methods, of which [10] was noted for its ability to make parallel estimates from different peak-to-peak measurements. Because of their reliance on local constraints, these methods produced period estimates that were subject to a sporadic halving or doubling. Examples of waveforms that can easily mislead simple period estimators are presented by Hermes [8], p15-16, where creaky voice (vocal fry) is shown to lead to quasi-regular subharmonic vibrations, or where the coincidence of F_1 with the second harmonic causes this to dominate the time waveform.

Hermes suggests that a more sophisticated period determination algorithm should produce a set of ranked candidates (instead of a single estimate) on the basis of local constraints and then find an optimum path through these by dynamic programming, such that local periodicity jumps are prohibited. In fact, dynamic programming can be applied to the period determination task in several ways. Ney [11] uses a time-warping function instead of autocorrelation to determine most likely period intervals; and Ney [12] uses

dynamic programming as a post-processing technique to obtain interpolated measures of periodicity from the noisy output of a conventional period detector. Here, we are interested more in the precise location of glottal epochs than in smoothed or interpolated measures of periodicity. Dynamic programming can also be used to enforce global periodicity smoothness during the estimation of glottal epochs. Allerhand [6] does this by including a periodicity smoothness constraint in simple peak-to-peak measurements for his local cost functions. This algorithm falls into a class which attempt to enforce sequence constraints globally in order to avoid excursions into local optima during the estimation of glottal epochs; these are significantly more resistant to the pitch-halving and doubling effect.

A BASIC PEAK-PICKING ALGORITHM

Our basic algorithm is derived from Allerhand [6], p92-105, whose inter-peak measurements we adopt, but whose local cost functions and dynamic programming regime we vary in several experiments.

A waveform peak (or dip) is defined as the highest (lowest) point between two zero crossings. A set of candidate peaks S_c is selected from the time waveform such that $S_c = \{s(z,m) \mid z>Z, m>M\}$, where $s(z,m)$ in S_c is a waveform peak (or dip) with zero-crossing separation z and absolute magnitude m. The values of Z and M are set such that $|S_c|$ is a minimum with $S_g <= S_c$, where S_g is the set of actual glottal peaks we are seeking to discover. We set Z = 0.5ms and M = 1 in our experiments.

The task of finding the globally most likely sequence of glottal peaks in a stretch of voiced speech is posed as a dynamic programming problem, in which peak sequences are represented by a graph constructed over the set of candidate peaks. The search is then constrained to find the least cost peak sequence which is defined as that containing the most prominent candidate peaks with the smoothest amplitude and period variations.

Let $s_1, \ldots s_i, s_j, s_k, \ldots s_N$ be the sequence of candidate peaks S_c over a voiced stretch. Let M_i be the magnitude of peak s_i and T_{ij} be the period between peaks s_i and s_j. For any sequence of three peaks (or three dips) s_i, s_j, s_k, the local cost can be broken down into three basic parts (Allerhand combines the first two):

Peak Prominence: $c1(i) = 1 - (M_i - M_{min})/(M_{max} - M_{min})$

where M_{min}, M_{max} are the minimum and maximum magnitude calculated over the whole signal; this is a linear function into the interval [0,1] which reaches its minimum for the peak with the largest magnitude in S_c.

Magnitude Variation: $c2(i,j) = (M_i - M_j)/(M_i + M_j)$

Period Variation: $c3(i,j,k) = (T_{ij} - T_{jk})/(T_{ij} + T_{jk})$

are non-linear functions into the interval [0,1] which reach a minimum for two peaks/periods of identical magnitude. When peaks/periods are large, their relative difference needs to be correspondingly large to have an appreciable effect on cost; the denominator in these terms compensates in a somewhat ad-hoc way for the greater variation expected between peaks/periods of larger magnitude.

Boundary conditions are placed upon plausible peak sequences to allow early pruning of impossible sequences. These are incorporated into the appropriate cost functions (above), such that if a condition is violated, the cost tends to infinity. The conditions are:

$sgn(M_i) = sgn(M_j)$; {two peaks or two dips}

$T_{min} <= T_{ij} <= T_{max}$; {plausible period length}

$|M_i - M_j| <= M_{maxvar}$; {plausible magnitude variation}

$|T_{ij} - T_{jk}| <= T_{maxvar}$; {plausible period variation}

Allerhand's dynamic programming algorithm is designed to work on voiced stretches of speech, which are pre-selected ([6], p97) on the basis of at least three peaks (p100) existing which presumably satisfy the path constraints above. In this way, he is able to construct the dynamic programming solution (using our formulation of the cost functions) as:

$$C(i,N) = \min_{i<j<N} \{ c1(i) + c2(i,j) + \min_{j<k<=N} \{ c1(j) + c2(j,k) + c3(i,j,k) + C(j,N) \} \}$$

$0<i<N-1$;

$C(N-1,N) = 0$;

and because the first and Nth candidate peaks are not guaranteed to be glottal pulses, Allerhand calculates the minimum C(m,n) for m = {1,2,3} and n = {N-2,N-1,N}.

VARIATIONS ON THE PEAK-PICKING ALGORITHM

In our approach, we redesign the algorithm to work over speech which contains both voiced and unvoiced stretches, allowing the DP to determine the presence of periodicity. This means that isolated peaks or unattached pairs of peaks must be allowed to contribute some high cost, but not an infinite one as in Allerhand's formulation, to the peak sequence paths constructed by the algorithm. This is done by assigning the maximum penalties for all those cost-components for which no measurements can yet be made.

Each successive peak is potentially the start of a new periodic stretch. Cost c1 is calculated and the maximum penalties for c2 and c3 are added. Likewise for all pairs of peaks, costs c1 and c2 are calculated and the penalty for c3 added. For all triples, c1, c2 and c3 are calculated. In this way, paths corresponding to a complete periodic sequence are always guaranteed to have a lower cost than broken periodic sequences, while allowing the latter to arise naturally if no other low-cost solution can be found.

A second difference in our approach is that we assign explicit weights to each component cost function. By examining the DP formula, it is evident that c1 and c2 each contribute twice for every occurrence of c3, which means that the periodicity constraint only contributes 1/5th to the cost in the long term. We have, by experimentation on a small dataset (32 mainly voiced utterances), discovered that an improved performance is gained by giving equal weighting to the periodicity constraint and all the other magnitude constraints combined.

A final observation concerns the appositeness of the DP formula literally as presented. Taking the usual procedural interpretation of the formula, recursive results are assumed to have been pre-computed and stored in a table. When calculating the global minimum cost for some path reaching from peak s_i to s_N, the algorithm considers all neighbouring peaks s_j for which the global minimum cost paths to s_N have already been calculated. Let us denote each of these paths by the sequence s_j, s_k ... s_N and note that each

penultimate peak s_k' has already been determined by optimal criteria with respect to each s_j' and the other peaks on the path s_j' ... s_N. The calculation for the current peak sequence s_i ... s_N involves minimising over all neighbouring s_j and each of these calculations involves minimising over all neighbouring s_k. The s_k in this calculation is not affected by the recursive calculation for each sj' ... s_N; but rather is selected for the fact that it minimises local costs with respect to s_i and s_j. The main recursion in the formula ensures that $s_j=s_j'$ always, but it is not necessarily true that $s_k=s_k'$, except by coincidence (see Figure 3).

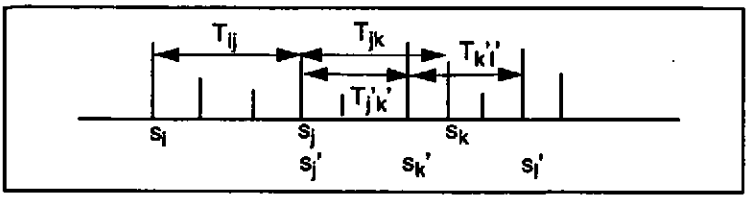

Figure 3: Non-coincidence of peaks s_k and s_k' where $s_j'...s_N$ are peaks on the globally optimum path from s_j.

This apparently has the following consequences. When constructing a sequence of peaks, the algorithm judges local magnitude and periodicity smoothness with respect to the previous peak in the sequence and some penultimate peak which may or may not be included in that sequence. We find this counter-intuitive; and instead we suggest the alternative view that all costs might be calculated with respect to peaks that are already on the path under consideration. Mathematically, this means that we would assume k is bound inside the recursive call C(j,N) such that $s_k=s_k'$ always; and computationally this would remove the need for the inner loop minimising over k (since s_k is determined by s_k').

Of course, we might be wrong; so to test this out we ran four different versions of the DP algorithm. Version 1 attempts to mimic Allerhand's cost weightings using the original DP algorithm; version 2 uses our cost weightings and the original DP algorithm; version 3 is a variant of version 2 where the minimum global cost for each peak sequence s_i ... s_N is calculated with respect to all previous paths s_j ... s_N and some future peak s_h - the rationale behind this was to centre peak s_i in the calculation of local periodicity smoothness; and finally version 4 is our simpler, modified DP algorithm with our cost weightings. Some early unsmoothed results are illustrated in (Figure 4) and can be compared with the clear period track visible in the running cepstrum in (Figure 2).

What seems most remarkable here is that the algorithm as originally presented mathematically seems to fail absolutely to deliver. We can only assume that Allerhand, by including the recursive term C(j,N) inside the minimisation over k, accepts implicitly that k is also bound here, since the results he reports are similar to our version 4. Although we feel we may not yet have discovered the best values for the global parameters governing the path constraints, it seems clear that our simpler algorithm outperforms all the other versions we tested on the 32-utterance dataset. The small, localised perturbations in periodicity (Figure 4, version 4) correspond in the time waveform to pitch jitter; this is also

represented in the cepstrum (Figure 2) by small gaps or discontinuities in the local estimate of periodicity, which although not so prominent visually are nonetheless there. The jitter is due in most cases to reduced airflow effects, resulting from the articulation of nasals and semivowels (see the time-aligned spectrum below the cepstrum in Figure 2), leading to glottalisation in some instances.

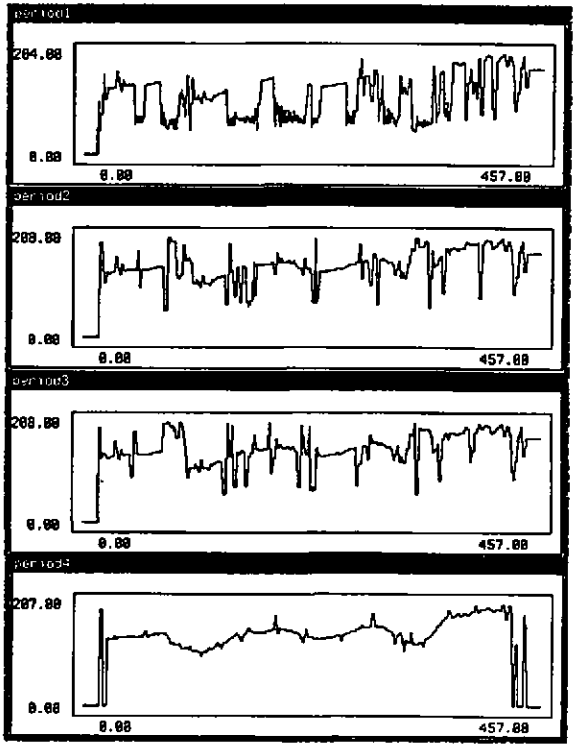

Figure 4: Four variations on a peak-picking algorithm (see main text)

We feel therefore that our simple time-domain technique for determining glottal epochs is successful; although we still have some reservations over our tentative explanation as to why the original interpretation of the DP mathematical formula fails to deliver. We would be grateful for any alternative explanations for this; and suggestions for improvements to our method. We also wish to acknowledge Mike Allerhand for discussing his approach with us in some detail; and thank Jim Hieronymus and Mark Huckvale for further suggestions.

FUTURE WORK

Currently we are engaged in constructing a pitch-synchronous Fourier transform package based around this period determination algorithm. Regarding the latter, a whole variety of more sophisticated algorithms suggest themselves. Dynamic programming is only optimal with respect to the local cost functions chosen; we mentioned earlier the sensitivity of this class of algorithms to the weightings of different cost components. It might prove possible to develop a more robust version based on autocorrelation (which exploits all data points rather than selected peaks to determine local periodicity) provided that some localised power calculation could help isolate the glottal pulses exactly. An interesting variant would be to use Ney's time-warping approach to discover local candidate period estimates and then use global sequence constraints to find a path through these alternatives. It might also prove possible to map from period estimates in the running cepstrum, which seems to provide a reliable gold-standard for smoothed local periodicity, back to glottal pulses in the time waveform.

REFERENCES

[1] A P Witkin (1983), 'Scale-space filtering', Proc. IJCAI, 1019-1022.

[2] J Glass (1988), 'Finding acoustic regularities in speech: applications to phonetic recognition', PhD Dissertation, MIT Dept. Elec. Eng. and Comp. Sci., Cambridge MA.

[3] Panel on Formant Analysis (1992), ESCA Workshop on Speech Signal Representations, University of Sheffield, 7-9 April.

[4] A S Crowe (1987), 'Globally optimising formant tracker using generalised centroids', Electron. Lett. 23, 1019-1020.

[5] B Yegnanarayana, H A Murthy and V R Ramachandran (1990), 'Speech enhancement using group-delay functions', Proc. ICLSP, Kobe Japan, 301-304.

[6] M H Allerhand (1987), *Knowledge Based Speech Pattern Recognition*, Kogan Page.

[7] L R Rabiner and R W Schafer (1978), Digital Processing of Speech Signals, Prentice Hall.

[8] D J Hermes (1993), 'Pitch analysis', Chapter 1 in: *Visual Representations of Speech Signals*, eds. M P Cooke, S W Beet and M D Crawford, John Wiley, 1-24.

[9] J Dubnowski, R Shafer and L Rabiner (1976), 'Real-time digital hardware pitch detector', IEEE Trans. ASSP 24, 1.

[10] B Gold and L R Rabiner (1969), 'Parallel processing techniques for estimating pitch periods of speech in the time domain', J. Acoust. Soc. Am. 46, 442-449.

[11] H Ney (1982), 'A time-warping approach to fundamental period estimation', IEEE Trans. Syst., Man., Cybern. 12, 383-388.

[12] H Ney (1983), 'Dynamic programming algorithm for optimal estimation of speech parameter contours', IEEE Trans. Syst., Man., Cybern. 13, 208-214.