TR－A－0097

Fo estimation from mixed speech

Alain de Cheveigne

1990. 12.20

ATR視聴覚機構研究所

〒619-02 京都府相楽郡精華町乾谷　☎07749-5-1411

ATR Auditory and Visual Perception Research Laboratories

Inuidani,Sanpeidani, Seika-cho, Soraku-gun, Kyoto 619-02 Japan

Telephone: +81-7749-5-1411
Facsimile: +81-7749-5-1408
Telex: 5452-516 ATR J

F0 estimation from mixed speech

Introduction

Listeners can often follow and understand the speech of one speaker among many, even when binaural information is not available. This aspect of our ability to organize the sound environment has received renewed interest of late (Palmer 1990, Assman and Summerfield 1990, Meddis and Hewitt 1990).

Among the effects of interfering speech, one might expect *vowels* to be particularly susceptible to interference from concurrent vowels, because a vowel's identity depends on the steady-state shape of the spectrum, and this is strongly affected by the presence of a concurrent vowel. It appears however that synthetic vowels can be perceptually identified within concurrent pairs at levels far above chance, particularly if there is a difference in fundamental frequency (Assmann and Summerfield 1990). Various perception models and signal processing methods for monaural concurrent vowel separation have been proposed, most of which require at some stage that the f0 of both individual speakers be determined. However it is notoriously difficult to design algorithms capable of extracting f0 from real speech (Hess 1983) and such difficulties are likely to be compounded for mixed speech.

The work reported here addresses the problem of how to track the f0 of two or more simultaneous speakers, and proposes an algorithm for that purpose. Although I discuss the problem in speech engineering terms, another motivation of this research is to understand how *human* listeners separate sounds and organize their auditory environment. I start by reviewing previous methods of mixed speech f0 extraction and speech separation, and discuss in detail some differences between the approaches. Then I present the mixed speech f0 estimation algorithm, together with some experimental results that demonstrate its performance. Finally, I attempt to relate the algorithm to the issue of *sound organization* in hearing.

Appendix I. shows examples of extracted f0 tracks, and Appendix II discusses various implementation issues and perspectives for future development.

I. Review of mixed speech f0 extraction and separation.

1. The problem of F0 extraction from mixed speech

Many f0 estimation algorithms have been proposed for the speech of a single speaker (Hess 1983). These algorithms usually rely on regularities that occur either in the time or in the spectral domain (fig 1). For mixed speech however the problem is more complex: there is often no clear regularity in the time domain, and in the spectral domain it may be difficult to attribute spectral components to one or the other of the two speakers (fig 2). Indeed, it may seem that speech separation is a *prerequisite* for f0 determination. For the purpose of this review, the f0 estimation from mixed speech is discussed in the context of mixed speech separation, which is also its main application.

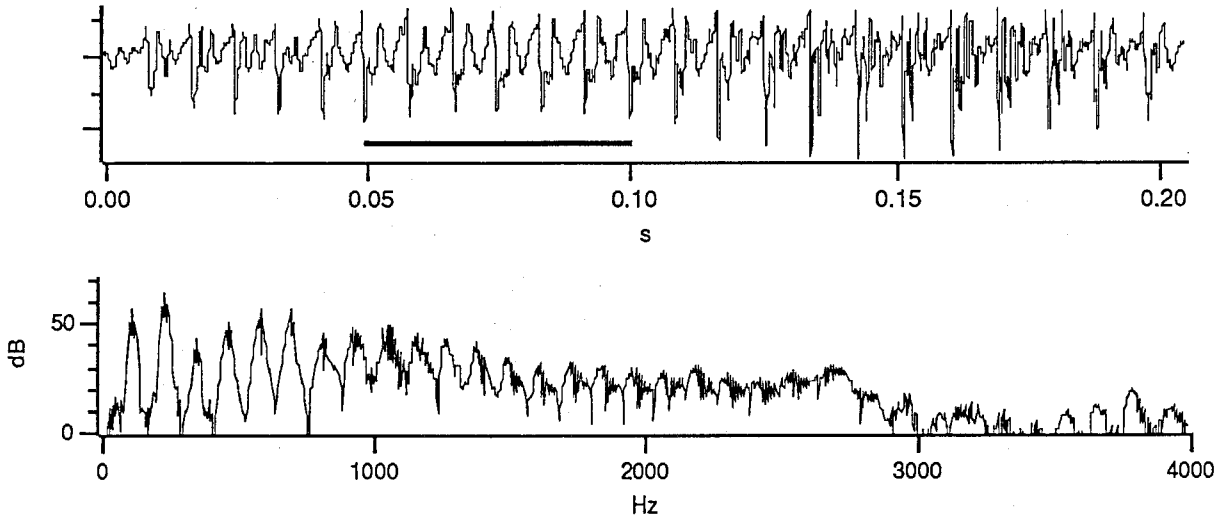


Fig. 1. Signal (top) and spectrum (bottom) of a portion of voiced speech. Spectrum was calculated using a 50 ms Hanning shaped window starting 50 ms into the speech (bar in upper plot).

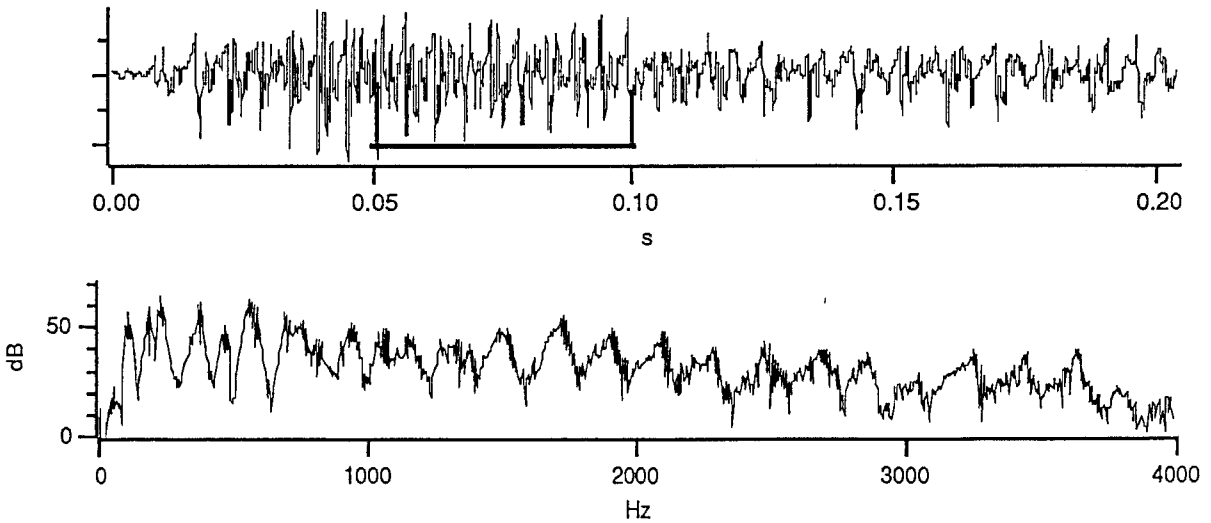


Fig. 2. Signal (top) and spectrum (bottom) of a portion of mixed speech (obtained by summing the speech token in Fig. 1 with another voiced token).

2. Review of previous solutions.

Frazier, Samsam, Braida and Oppenheim (1976) proposed a method by which the speech of one speaker can be enhanced relatively to a competing speaker or to noise by convolution in the time domain with a comb filter "tuned" to its fundamental period (as in Fig. 3). The estimate of this parameter was obtained from a glottal accelerometer attached to the speaker's throat, and the authors did not suggest how it might be extracted from the speech signal instead.

Parsons (1976) proposed a method by which the Fourier transforms of 51.2 ms windows of mixed speech are "dissected" into spectral peaks. The peaks are accumulated in a peak table and used to construct a Schroeder histogram (a histogram that counts all potential fundamentals of all spectral components) (Schroeder 1968) from which the f0 of a first (dominant) speaker is determined. The f0 of the second speaker is then obtained by subtracting from the spectrum the peaks belonging to the first speaker, and repeating the histogram calculation on the remaining peaks. The speech of each speaker is then re-synthesized by reverse Fourier transformation of its share of the spectrum.

Nagabuchi et al. (1979) proposed a related method, in which the f0 estimates were obtained from peaks in the autocorrelation function of the PARCOR residual. F0 estimation was done in two steps: in a first step the highest peak of the autocorrelation served to indicate the f0 of the dominant voice. This estimate served to determine the parameters of a frequency-domain comb filter that eliminated that voice, allowing the second f0 to be measured in turn.

The last two methods rely on a high-resolution spectral representation of the speech signal. One may wonder whether such a representation is available within the auditory system, and whether a model similar to the previous can serve to explain human speech separation performance. Assmann and Summerfield (1990) tested such a model using the output of a filter bank with characteristics derived from auditory masking. They found that it performed poorly, mainly because the spectral representation lacked the necessary resolution. In particular, the model had difficulty extracting the *f0* of concurrent vowels. They found better performance for a place-time model (time domain processing of filter bank outputs).

Weintraub (1985) proposed two versions of a place-time method for speech separation. Both were based on Lyon's (1982) model of cochlear filtering and nerve-firing coincidence analysis. In each version, the f0s of the two speakers was extracted from a pattern derived by pooling autocoincidence patterns from all filter bank channels. In the first version, speech streams were segregated by *selecting* channels dominated by a fundamental periodicity. Lyon (1983) had used a similar approach for segregating spatially distinct sources in a binaural model. In the second version, instead of being selected, channels were *split* by modeling the relative contributions of each stream to a channel.

Stubbs and Summerfield (1988) proposed a method based on cepstral filtering and evaluated it in comparison with an implementation of Parson's (1976) algorithm. F0 estimation in the former method was done by searching for two peaks in the cepstrum. In a more recent paper Stubbs and Summerfield (1990) discuss in detail the difficulties that arise when f0 tracks cross. For their simulations they used the f0 tracks obtained from the original *separate* speech channels to guide the double-voice f0 tracking algorithm.

Assmann and Summerfield (1990) described a place-time model similar to Weintraub's method. Both f0s were determined in a first step from the peaks of a pooled autocorrelation function, and then each speech stream was extracted by sampling the autocorrelation functions at lags corresponding to the stream's period. In a similar model proposed by Meddis and Hewitt (1990) f0s are also derived from a pooled autocorrelation function, but segregation is achieved instead by channel *selection,* as in the first version of Weintraub's system. An interesting feature of the model of Meddis and Hewitt is that vowel recognition is performed on the short-lag

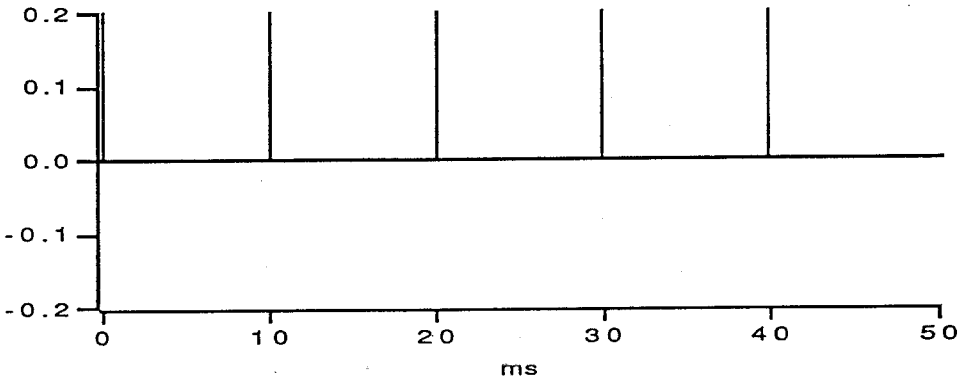
portion of the pooled autocorrelation functions of selected channels, rather than on a spectral pattern as with other models.

Palmer (1990) investigated vowel separation from a physiological point of view. He recorded the activity of a population of cochlear-nerve fibers in the guinea pig in response to vowel pairs with different fundamentals, and tested several models of vowel segregation using these data. All models involved f0 estimation as a first step. One class of models was based on the ALSR (Average Localized Synchrony Rate) pattern of Young and Sachs (1979). This pattern is essentially indistinguishable from that obtained by pooling the Fourier transforms of nerve fiber period histograms over the population of fibers. The pattern was submitted to a) a harmonic selection method similar to Parson's, b) a harmonic sieve method, and c) cepstral analysis. All three methods yielded correct estimates of the f0s. Palmer also tried two methods based on auto-correlation of individual channels: a) a histogram of the within-channel dominant autocorrelation peak positions, and b) a pooled autocorrelation function, similar to that used by Weintraub, Assmann and Summerfield, and Meddis and Hewitt. These last two methods were disappointing, mainly because they failed to correctly estimate both f0s.

It is interesting to point out some key differences between approaches:

• *Enhancement vs cancellation*

Frazier's (1976) method takes advantage of the periodicity of the target speech to *enhance* it: successive f0 periods are added together to average out noise or interfering speech. Parson's (1976) and other methods, on the other hand, use the periodicity of the interfering speaker to *cancel* it. The advantage of the enhancement approach is that it can enhance speech despite non-periodic interference. However it offers only a small improvement in signal-to-noise ratio unless one uses a comb filter with an impractically long impulse response (Fig. 3), whereas the cancellation approach in principle allows an infinite signal-to-noise ratio to be obtained with a very compact impulse response (Fig. 4). Stubbs and Summefield (1988, 1990) discuss in further detail the relative merits of these two approaches.



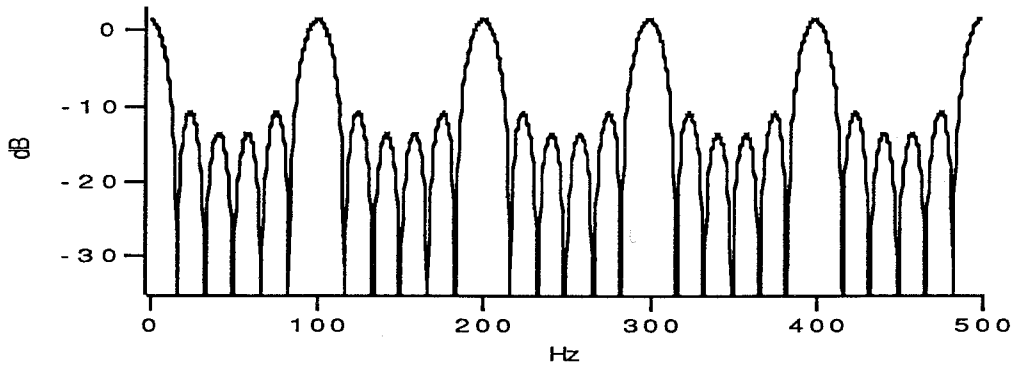


Fig. 3. Impulse response (top) and transfer function (bottom) of an enhancing comb filter. The interval between "teeth" is adjusted so that the peaks in the transfer function coincide with the harmonics of the signal to be enhanced. Unwanted components are typically attenuated by no more than 15 dB.

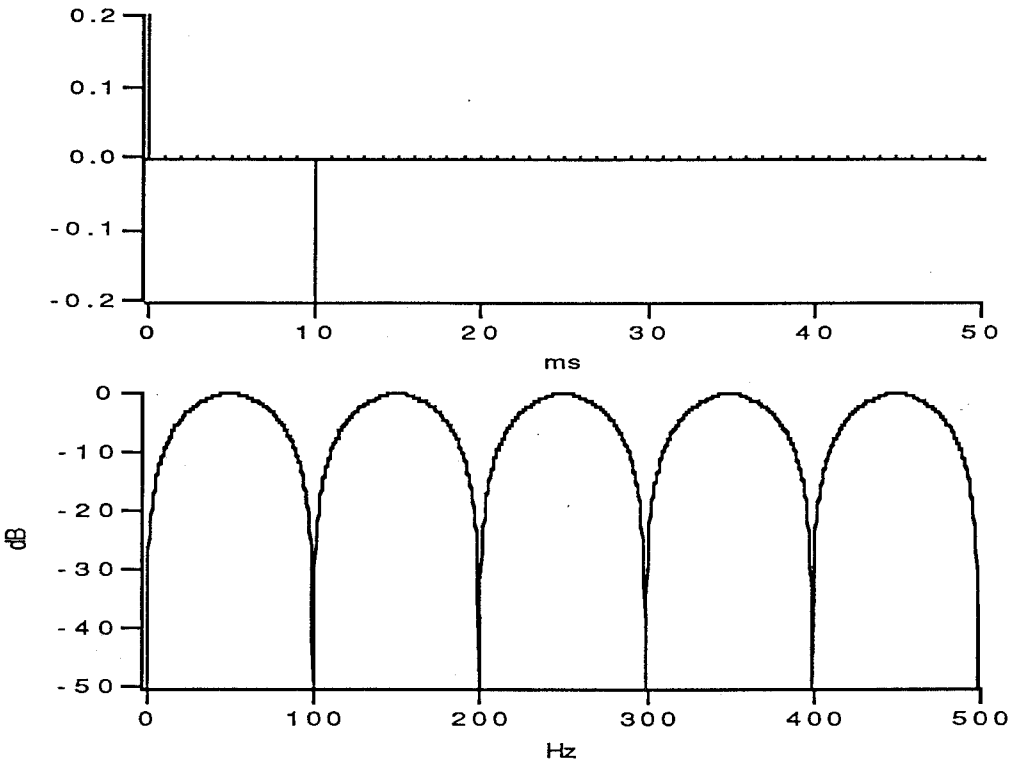


Fig. 4. Impulse response (top) and transfer function (bottom) of a cancelling comb filter. The interval between "teeth" is adjusted to cancel all harmonics of interfering speech. In principle cancellation can be complete.

• *F0 detection: before vs after separation*

Both Parson (1976) and Nagabuchi (1979) combine the speech separation algorithms with the f0 extraction process: an iterative algorithm is bootstrapped by an initial f0 estimate for a "dominant" speaker, and this estimate serves to design a filter that cancels that stream, thus allowing the f0 of the second speaker to be measured. On the other hand Weintraub, Meddis and others attempt to estimate both f0s from a pattern derived *before* any speaker separation. This latter approach has the advantage of simplicity of design in that well defined processing modules are connected in a bottom-up fashion. However, as anyone with experience in speech f0 extraction will agree, the extraction of *two* f0s from the same basic pattern is likely to be difficult when applied to real speech.

• *Separation process: split vs select*

Different methods use f0 information in different ways to split information between the two speech streams. In Weintraub's first system, as in the model of Meddis and Hewitt, f0 information is used to *select* filter channels according to their dominant periodicity (on the assumption that, since spectral envelopes are likely to differ, many filter channels will be dominated by components from a single speaker). Non-selected channels for a stream are set to zero. A similar approach was taken by Lyon (1983) for separating spatially distinct sources on the basis of cross-correlation. On the other hand the methods of Parson (1976) and Nagabuchi et al. (1979) attempt to share each spectral component between speech streams, rather than allot it to either. The same is true of Weintraub's (1985) second system. As frequency resolution improves, components are more easily isolated, so the results of the two approaches become similar. The advantage of selection over splitting is that there is no need to decide how a spectral component must be shared. The drawback is that the necessary spectral resolution requires stationarity over possibly unrealistically long windows.

There is a subtle difference between models concerning the way *inseparable* components are handled. These are components that are precise harmonics of both fundamentals. In Nagabuchi's method any harmonic common to both speakers will be *canceled*, leaving a hole in the spectrum of the target speaker. In spectrum or autocorrelation sampling methods, such as that of Assmann and Summerfield, any shared harmonic *remains undisturbed* in the spectra of both streams. In more sophisticated methods such as those of Parson or Weintraub (second version), common harmonics are *shared* more equitably. However Weintraub notes that a shared component can be the result of either additive or subtractive interaction, depending on phase, so there is a fundamental uncertainty as to how such a component should be split.

• *Engineering vs hearing theory*

Frazier, Parson, Nagabuchi and Weintraub approach speech separation from an engineering point of view. Assmann and Summerfields, Meddis and Hewitt, and Palmer are more interested in understanding hearing. The scope and methods are different in each case, and it would be foolish to apply criteria of one domain to the other. However from the engineering point of view it is difficult not to feel concern about the *robustness* of some models of auditory processing when applied to speech. The reason for this concern is that it often happens in speech processing that an algorithm performs well on idealized or synthetic speech, but fails catastrophically when applied to real speech.

One last point should be mentioned. Assmann and Summerfield (1990) found that concurrent vowel identification rates remain far above chance level *even when the fundamentals are the same*. This implies a mechanism that can operate without f0 information, possibly on the basis of spectral pattern template matching. Contrary to what the theme of the present work might imply, f0 detection is not indispensable for vowel separation. Assmann and Summerfield (1988) found further that, for a common fundamental frequency of 100Hz, performance depends also on the *phase* of voice periods of one voice relative to the other. This indicates that, whatever the mechanism may be, it has high temporal resolution and is capable of sampling the spectral pattern in time separately for each vowel. When both fundamentals are the same, mixed vowels sound like a single vowel "colored by another vowel" and identification rates (for both vowels correct) are about 55%. When fundamentals are different, two separate vowels can be heard and identification rates are about 75% for a 2 semitone f_0 difference (Assmann and Summerfield 1990).

In summary, mixed speech f0 estimation is a critical step in most methods and models of mixed speech separation.

II. The mixed speech f0 estimation algorithm

1. Periodicity

The algorithm models voiced speech as a periodic function. A function S of real variable t is periodic if there is a non-zero number T such that for all t:

$$S(t) = S(t+T).$$

The smallest positive T for which this holds true is called the period. This definition requires no particular assumption on the values of S (they could be for example neural firing patterns). However if S has values in an additive group (such as a vector space) then for all t:

$$S(t) - S(t+T) = 0$$

The left hand side is equivalent to applying a comb filter with impulse response $\delta(t) - \delta(t+T)$. For a periodic input of period T, the output of such a filter is zero. This idea is the basis of a f0 extraction method known as AMDF (Average Magnitude Difference Function, Ross et al. 1974). This method searches for a global minimum of the output magnitude of the comb filter averaged over a window:

$$AMDF(lag) = \int_{window} |S(t) - S(t + lag)| dt$$

For a purely periodic signal the AMDF shows a succession of zeroes, the first one (apart from the one at zero lag) indicating the period. For approximately periodic signals such as voiced speech the pattern shows dips instead of zeroes (fig. 5).

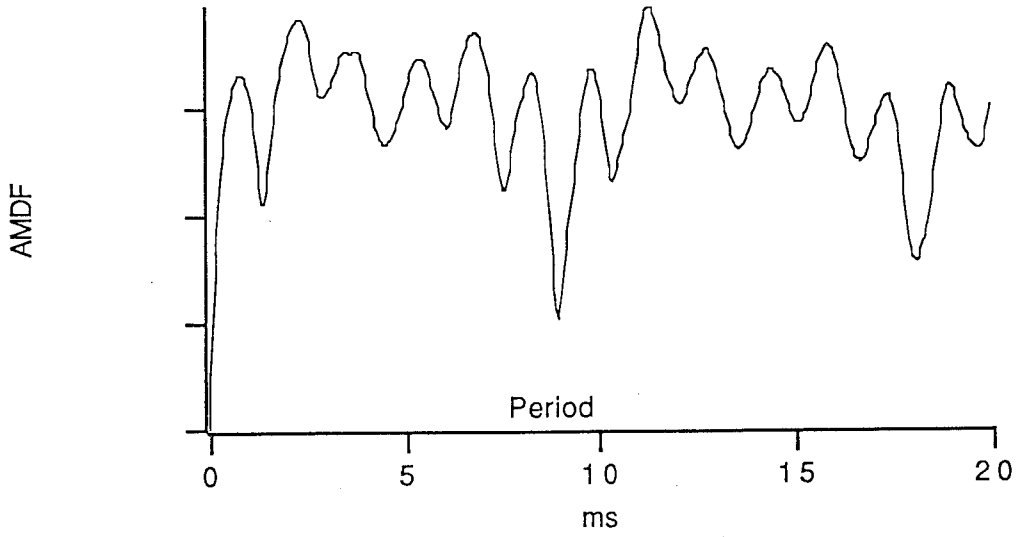


Fig. 5 Example of AMDF function for voiced speech.

2. Single voice f0 estimation algorithm

A version of the AMDF algorithm was used to obtain reference f0 estimates from the single voice channels before mixing, for evaluation of the mixed speech algorithm.

The AMDF for each lag is divided by the cumulative mean of AMDFs for shorter lags (the effect is to eliminate the zero at zero lag and attenuate spurious dips at short lags). The minimum of this function over a plausible range of lags indicates the period. This algorithm can easily and inappropriately lock on to a period multiple (subharmonic). To avoid this, a period minimum is arbitrarily required to be less than 0.9 times the value of the AMDF at 1/2 or 1/3 its lag. No other smoothing or error correction algorithm is used. The AMDF uses two windows, one fixed and the other sliding to the right (positive time). By convention, the period estimate produced is aligned with the *left hand side* of the fixed window. Period values are transformed to a base 2 logarithmic frequency scale (octaves re: 110 Hz).

The algorithm produces as an interesting by-product a value that can be interpreted as a *measure of periodicity*. This is defined as:

$$PM = \log_2\left(\frac{\text{mean(AMDF)}}{\text{AMDF(period)}}\right)$$

The periodicity measure PM indicates the depth of the dip in AMDF at the period, relative to its average value. The measure is large (2 to 6) during steady state voiced portions and small (0 or negative) at transitions and during unvoiced portions. The measure also gives an indication of the reliability of the f0 estimate produced by the algorithm.

3. Double Difference Function mixed speech f0 estimation algorithm

The algorithm is extremely simple. Suppose that mixed voiced speech is modeled as the sum of two periodic functions $S_A(t)$ and $S_B(t)$, of periods T_A and T_B:

$$S(t) = S_A(t) + S_B(t)$$

If this signal is fed to a time-domain comb filter of impulse response $\delta(t) - \delta(t + T_A)$, represented as the operator F_A, the first component will be cancelled and the output will be:

$$F_A[S](t) = F_A[S_B](t)$$

No contribution of S_A remains. $F_A[S_B]$ shares the same periodicity as S_B, so if we feed it to a second comb filter of impulse response $\delta(t) - \delta(t+T_B)$ the output will be zero:

$$F_B[F_A[S]](t) = F_A[F_B[S]](t) = 0$$

The output of the cascaded comb filters is null if and only if the lags match the periods. In a similar way to the AMDF algorithm that performs an exhaustive search of the lag parameter space of a single comb filter, the DDF algorithm performs an

exhaustive search of the parameter space of two cascaded comb filters. The lags that it finds correspond to the minimum of the function:

$$DDF(lagA, lagB) = \int_{window} |S(t) - S(t + lagA) - S(t + lagB) + S(t+lagA +lagB)| dt$$

As in the single voice case the algorithm can produce a periodicity measure as a by-product:

$$DPM = \log_2 \left(\frac{mean(DDF)}{DDF(period)} \right)$$

4. Experiments

The f0 tracks obtained with the DDF algorithm from mixed speech were compared with those obtained from the speech components before mixing using the the AMDF single voice f0 algorithm.

Test data (chosen at random from the ATR database) consisted of 3 Japanese sentences pronounced according to five different intonation patterns by two speakers, one male (known as MYI), and one female (known as FST), a total of 30 sentences. The speech data was sampled at 20 kHz, and low-pass filtered by convolution with a 1ms rectangular window (first notch at 1kHz). First, the single voice AMDF algorithm was run on all sentences using overlapping 20 ms (400 sample) rectangular windows, at 1.5 ms (30 sample) intervals. The allowable range of lags was set to correspond to a f0 range of 60 to 300 Hz for the male speaker and 100 to 600 Hz for the female speaker. Next, portions of speech data were selected based the *periodicity measure* produced at this step. The periodicity measure was scanned for contiguous runs with a value greater than an arbitrary threshold (1.4), and a length greater than 150ms. Using this information, nine 150 ms segments of speech signal were excised for each speaker. For each speaker the nine segments were paired and summed to obtain "mixed speech" (producing two sets of 32 pairs). The segments for both speakers were also paired (producing a set of 81 pairs). The motivation for selecting portions with a good periodicity as evidenced by the periodicity measure was: a) to ensure that the reference f0 tracks used for evaluation were reliable, and: b) to avoid compounding difficulties for the double-f0 algorithm. In the test data sentences, about 75% of all voiced portions (defined conservatively as any portion with PM > .5 for more than 30 ms) had a periodicity measure above this threshold.

The DDF mixed voice algorithm was then run on all mixed speech tokens, using a window size and analysis interval identical to those for the AMDF algorithm. The search range for each lag was limited to one octave (1:2 lag ratio), based on a priori knowledge of the correct range derived from the single voice f0 algorithm. The ranges were not necessarily the same from pair to pair, or for both lags. This search range limit was imposed to avoid difficulties in interpretation of subharmonic matches, and display of results.

Typical results are shown in figures 6-8. Others are shown in Appendix II together with waveform and periodicity measure plots.

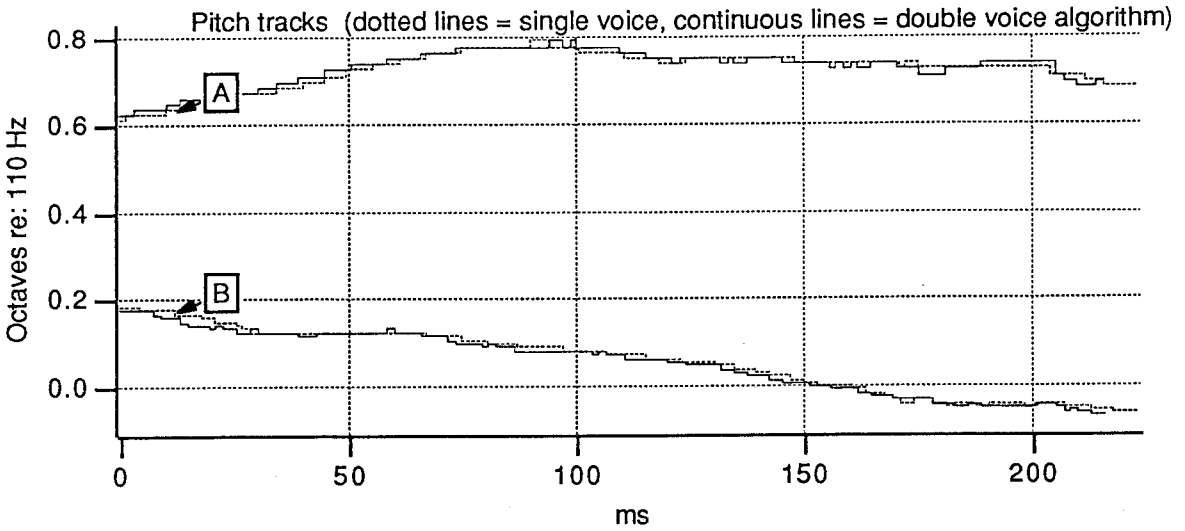


Fig 6. F0 values produced by the DDF double f0 algorithm, compared with those obtained by AMDF on the separate voices. Male speaker, tokens a0, a1.

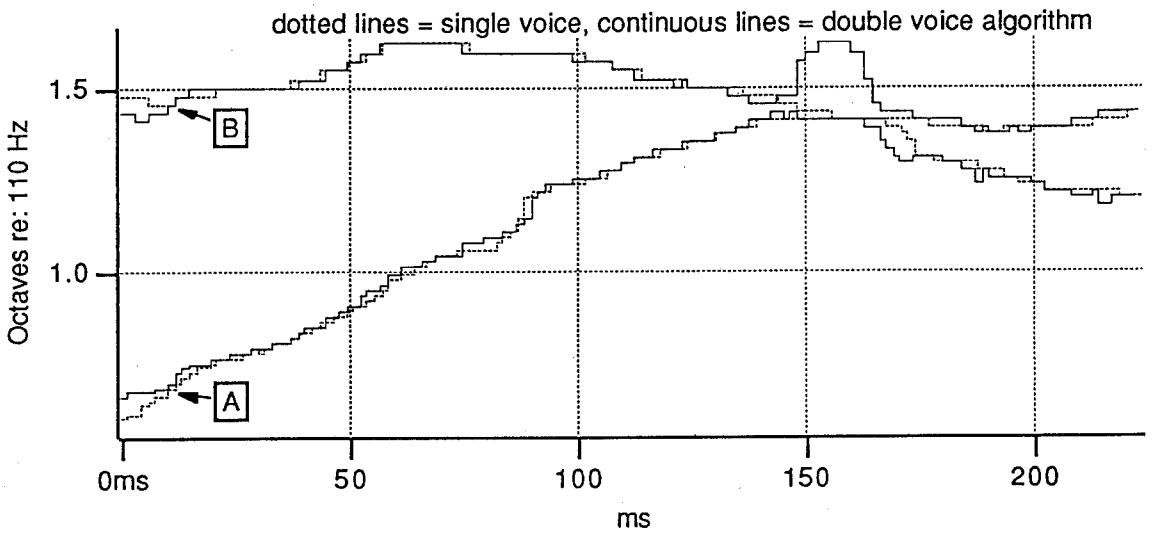


Fig 7. F0 values produced by the DDF double f0 algorithm, compared with those obtained by AMDF on the separate voices. Female speaker, tokens b2, b7.

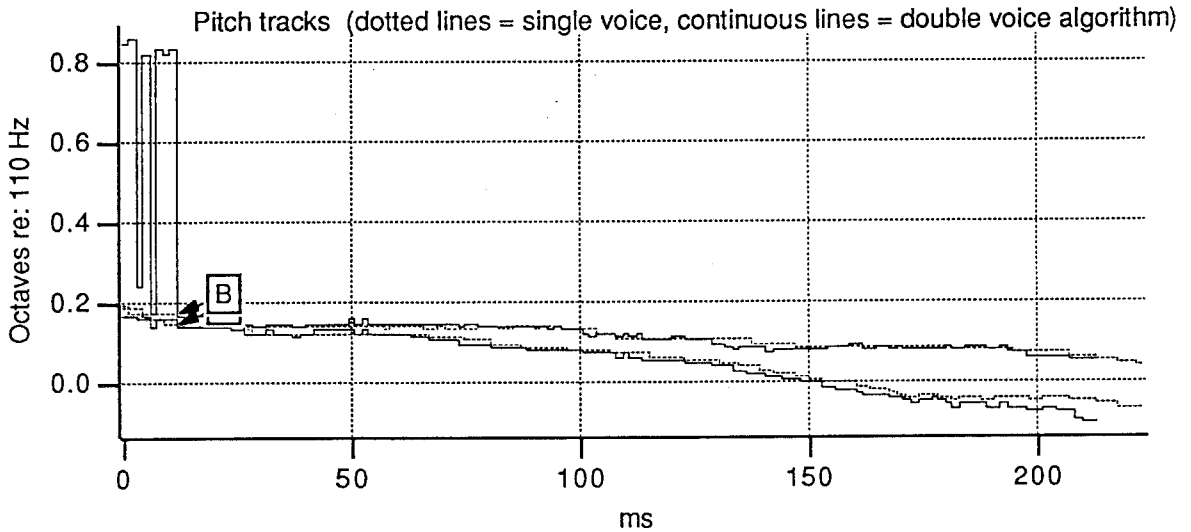


Fig 8. F0 values produced by the DDF double f0 algorithm, compared with those obtained by AMDF on the separate voices. Male speaker, tokens a0, a3.

The correspondence between tracks is remarkably good. The algorithm breaks down when both f0s are identical, but often still performs quite well even when the frequencies are close (fig. 8). Figure 9 shows the histogram of errors (deviation between a DDF f0 track and the closest AMDF f0 track) for male and female data (excluding the male-female pairs). As evident in fig. 10, half of all estimates made from mixed speech fall within 1% of an octave of a f0 estimate extracted from either speech component alone and some 90% within about 3% of an octave.

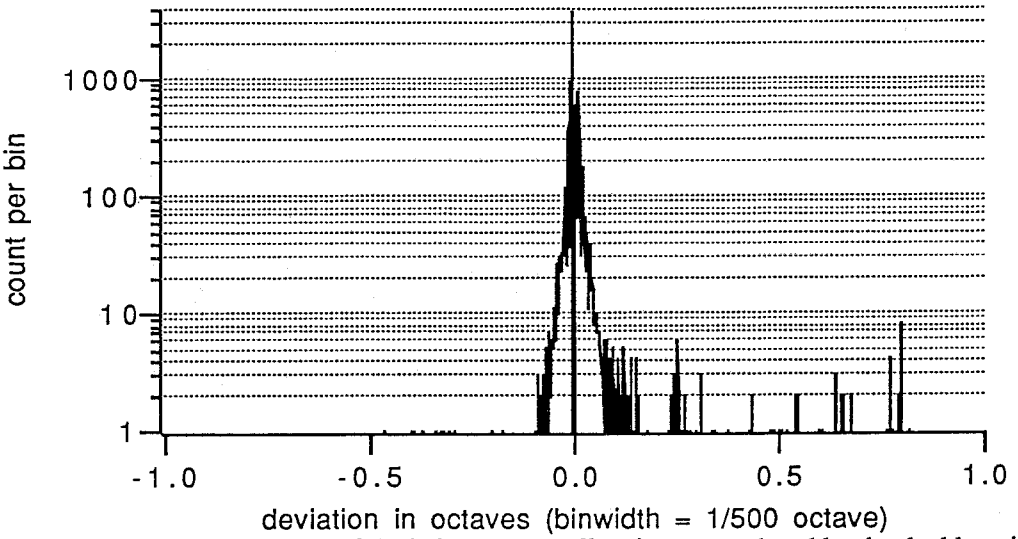


Fig 9. Histogram of deviation between f0 estimates produced by the double voice and single voice f0 algorithms. Note the log scale.

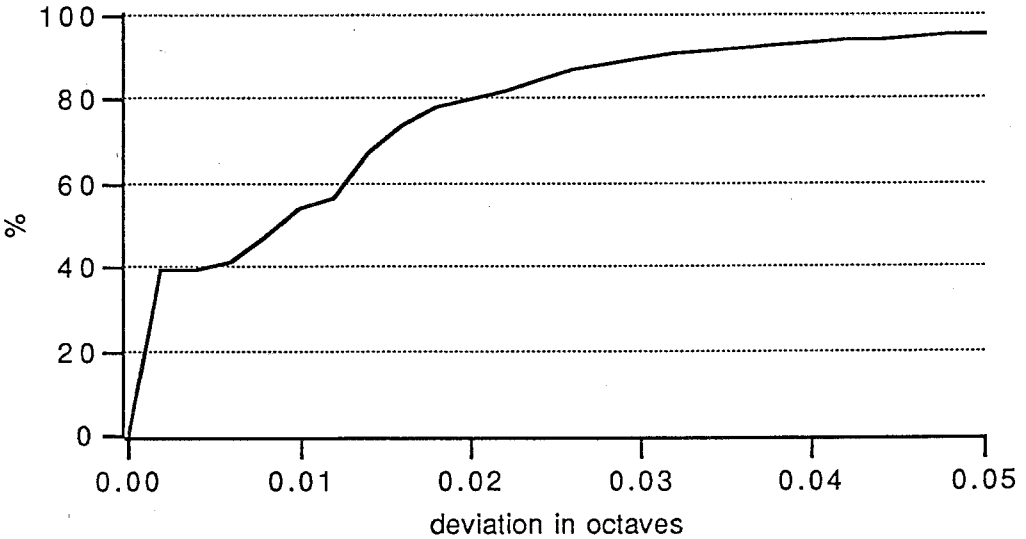


Fig 10. Plot showing the percentage of estimates that fall within a certain deviation. 50% of all DDF estimates are within 1% of an octave of an AMDF estimate.

In conclusion these results show that the algorithm is extremely accurate and reliable. One should keep in mind that these results were obtained on relatively "clean" data. In addition the one octave limit on each dimension of the search space, intended to avoid subharmonic errors, may have reduced the occurrence of other kinds of error. However as the algorithm makes as yet no use of continuity or other constraints, there is also much room for improvement.

The next section discusses various implementation issues and possible future improvements.

III. Relevance for hearing

The DDF mixed speech f0 estimation algorithm performs well. This does not automatically qualify it as a model of auditory perception. As noted above, the criteria that apply in speech engineering and in hearing theory are not the same, even when both seem to be studying the same object. To qualify as a model of auditory perception, a processing algorithm should a) address a phenomenon in auditory perception that needs explaining, b) be capable of predicting quantitative aspects of performance, and c) be physiologically realistic, i.e. it must be possible to imagine how the model might be "implemented" in physiological terms. The DDF algorithm is obviously relevant to the phenomenon of mixed speech segregation. However, it is perhaps also relevant in the wider context of concurrent sound organization.

1. Concurrent sound organization

a. The importance of sound organization.

Our physical environment usually contains a variety of sources that simultaneously produce sound. Some are of importance for survival, such as food or predators (or, in more recent times, speech, traffic or warning signals). Others may be thought of as interfering noise (wind, rustling of clothes, irrelevant signals or speech, etc.). These sources usually overlap intricately in time and frequency, but despite this fact, we are often capable of attending to one particular source as if it were alone.

It can be argued that sound organization is more important than other issues that have dominated hearing theory (pitch, loudness, timbre, etc.), both because of its importance for survival, and because sound organization logically *precedes* extraction of sound qualities, and thus is intricately associated with the processes involved. Plomp (1988) offered a classification of auditory phenomena, reproduced in part below:

	TOP
level	phenomenon
central	separation of sound streams perceptual grouping continuity in time co-modulation masking release pitch of complex tones profile analysis timbre perception
peripheral	lateral inhibition forward masking frequency discrimination amplitude discrimination phase effects combination tones masking patterns
	BOTTOM

In this classification, sound organization phenomena appear at the very top and correspond, in Plomp's view, to more central auditory processes.

b. Implicit framework

Much effort has gone into investigating the various stimulus factors that influence the segregation or integration of spectral components (Bregman 1990). For example it has been found that components tend to fuse into a single sound if they share

the same onset time and amplitude envelope shape, if they belong to the same harmonic series, and if they are coherently modulated in frequency. A partial that does not share these qualities with others will tend to segregate, and does usually not participate in determining the sound quality of the remaining stream.

Such a description of streaming in terms of integration or segregation of partials implicitly assumes that auditory processing occurs as a three-step process:

a) Sound is analyzed into physiological "elements",

b) Elements are grouped into streams,

c) Streams undergo perceptual processing (for example extraction of a perceptual quality such as pitch).

The grouping decision occurs of course within the auditory system, and concerns physiological elements rather than the acoustic partials. To speak nevertheless in terms of partials that "segregate" implies that the physiological elements are *isomorphic* to acoustic partials.

c. Four problems with this framework

One can question this view on four accounts:

1) We are not sure that physiological equivalents of partials exist. From psychophysical evidence it appears that at most about 5 to 7 individual partials can be "heard out" within a harmonic complex tone, even by the best of listeners and after much effort. The fact that most spectral components are not readily separable perceptually (and some not at all) suggests that they do not exist as separate physiological entities. It can be argued that they do exist at some peripheral stage but are *not accessible* for perception. However physiological experiments have also failed to confirm the existence of elements isomorphic to partials. Due to the limited frequency resolution of the ear, strong interaction between partials seems to be the rule.

2) The three-step process is in some cases circular. For example, one of the factors that governs segregation is harmonicity, which presumably depends on pitch. However pitch is only available after the grouping step. The grouping decision thus involves its own outcome.

3) Available models of sound quality perception such as pitch operate *directly* on the output of peripheral analysis (as far as they assume any physiological process or locus). There is no provision for a preliminary grouping step.

4) Plomp's view placed sound organization at the most *central* level, but the three-step process just outlined places it on the contrary at the most *peripheral* (or at least lowest) level.

A description in terms of segregation and integration of partials is of course useful to describe sound organization phenomena, but one may expect difficulties when relating this description to physiological mechanisms or models. Most signal processing methods that such models can be based on (such as those reviewed in the introduction) do assume a spectral decomposition.

d. Relevance of the DDF algorithm to these issues

The DDF mixed speech f0 extraction algorithm is of interest in this context because it performs an operation similar to streaming (extraction of pitch tracks belonging to separate voices). With respect to the the four issues mentioned above:

1) The DDF algorithm works in the time (or rather lag) domain and does not require analysis into partials,

2) The algorithm is bottom-up, and does not require knowledge of the outcome of a later stage,

3) Pitch and segregation occur in the same step, so there is no need to assume a preliminary segregation stage.

4) The algorithm performs this task at a low level.

e. "Streaming as cancelling" vs "streaming as grouping"

The classical view of sound organization is as an additive process: partials (or their physiological equivalents) are combined additively to form streams. Given the problems evoked above (especially #1), it may be more appropriate to view sound

organization as a *cancelling* process: in the presence of multiple auditory sources, the auditory system strives to create channels in which all but one source is cancelled. If one admits the existence of physiological equivalents of partials, cancelling is not very different from grouping. However if such equivalents do not exist, grouping per se is impossible, and the process must be seen as cancelling. The relevance of the DDF algorithm to this issue is that it shows, in one particular case, that sound organization can be performed without a decomposition into partials.

A cancelling process consists in principle of two parts: a *decision* process as to what should be cancelled, and the actual cancelling process. In the DDF algorithm the two parts are intimately mixed, though one can design a version in which they are separate (Appendix II-1-b). The cancelling is done in the time domain by comb filtering, and the decision is based on the immediate result of the comb filtering. One can however imagine more complex models in which cancelling is performed partly in the spectral domain, or on higher-level representations, and decision is performed on higher-level information.

One aspect of hearing that may confuse this issue is that in many cases the auditory system separates sounds with apparent ease (Bregman's "transparency of sound", Bregman 1990), including in some case partials. It is probable that the auditory system evolved to provide an approximation to this transparency as perfect as possible, because it sense from a pragmatic point of view. Or from another point of view, that of information processing theory, independent evaluation of separate sources allows optimal information integration (Massaro 1987).

Quantitative prediction of performance

It is customary to require that a model provide quantitative predictions as to performance. Models are refuted if they predict performance *poorer* than observed, *better* than observed, or that shows a different pattern of dependency on certain parameters. Such criteria are indeed required to keep the model population down, I will nevertheless argue for less emphasis on quantitative evaluation, and more on *qualitative* issues such as robustness when applied to real speech, or physiological realism.

a) A model that predicts performance poorer than observed can certainly be discarded. On the other hand there is no reason to refute a model that predicts performance *better* than observed (unless that model claims to accurately describe all aspects of processing). Poor performance can be due to other stages of processing, or to the physiological implementation of the model.

b) Quantitative evaluation can serve to choose between rival models that all adequately explain the observed performance. However, judging from the field of speech engineering, there are no methods of speech separation or sound organization that are satisfactory from a practical (qualitative) point of view. In this sense it is premature to attempt quantitative evaluation. The requirement of quantitative prediction often imposes simplifications that result in a model too weak to stand on *qualitative* grounds. In trying to implement processing models one often discovers structural weaknesses. Models should be examined for such weaknesses before quantitative evaluation is attempted.

c) Quantitative evaluation encourages a "black-box" view of performance that discourages interaction with physiology and engineering.

The DDF model appears to predict pitch tracking performance better than observed in human listeners. However any physiological implementation would have a hard time providing the linear representation of the speech signal that the computer implementation provides. This in itself would explain less good performance. The interesting issue is "can a physiological implementation perform well enough" rather than "does the computer model perform too well".

Possible physiological implementation of the DDF algorithm

A neural coincidence network

An advantage of the DDF algorithm, as a basis for a physiological model, is that it is fundamentally *parallel* rather than sequential, and so does not require complex control structures. One can imagine an implementation along the lines of Licklider's duplex or triplex models of pitch perception (Licklider 1956, 1959, 1962) (Fig. 11).

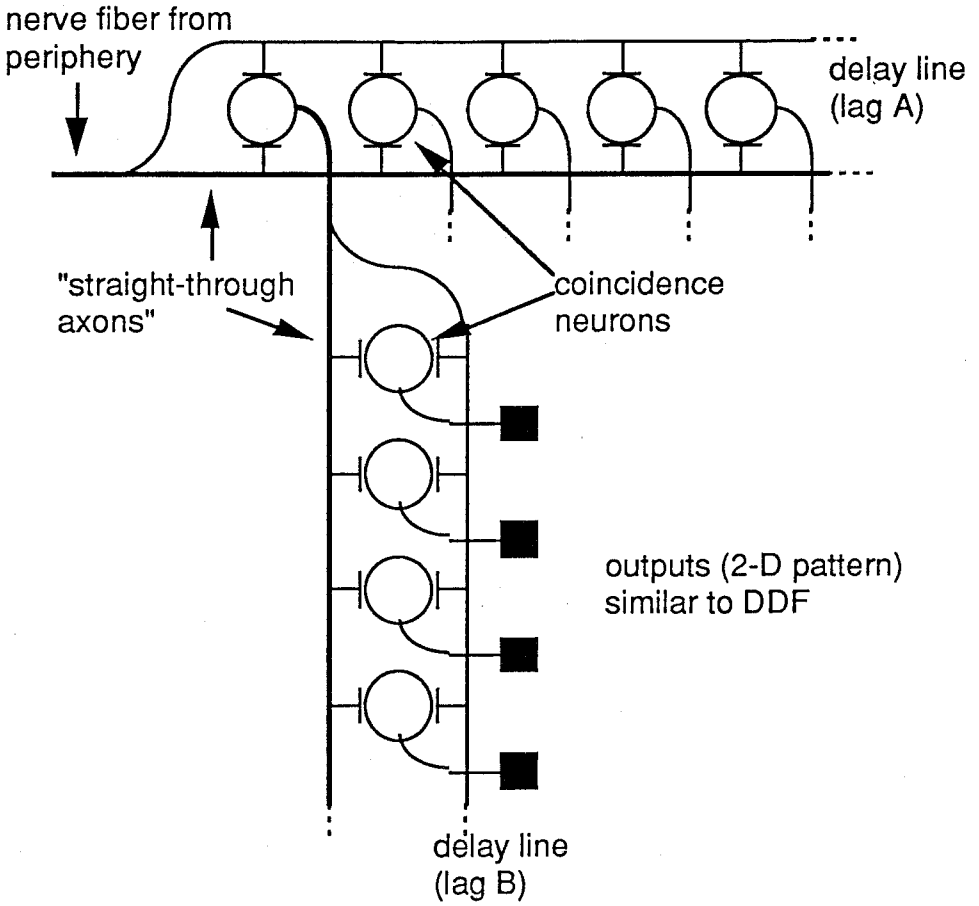


Fig. 11. A neural network implementation of the DDF algorithm along the lines of Licklider's duplex and triplex models of pitch perception.

The main difficulty is that the coincidence neurons in Licklider's model implement a form of coincidence summation similar to the *multiplication* step in autocorrelation, whereas the DDF algorithm calls for a *subtraction* operation. One solution might be to design a version of DDF based on autocorrelation (in the single voice case, AMDF and ACF pitch extraction methods are closely related, Ney 1982). However I have not yet succeeded in finding such an algorithm. Another solution would be to assume inhibitory interaction at the coincidence neurons.

Physiological plausibility of DDF

The DDF model can be broken down into four components: delay lines, cancellation mechanism, temporal averaging mechanism, decision mechanism.

• Delay lines can be provided by conduction delay along axons. This is a reasonable component: considerable evidence has been gathered of late in favor of the Jeffress model of sound localization (Carr and Konishi 1990, Carney and Yin 1989,

Chan, Yin and Musicant 1987, Konishi, et al. 1988, Jeffress 1948, Yin, Chan and Carney 1987, Yin and Chan 1988, Yin and Chan 1990) that assumes such delay lines.

• For the cancellation mechanism one can suggest inhibitory synapses that "gate" the nerve fiber discharge pattern along a fiber (or group of fibers). However, to my knowledge no such mechanism has yet been proposed or demonstrated physiologically. One argument in favor of a neural cancellation mechanism is that some sort of neural cancellation mechanism is convenient to explain binaural interaction (Durlach's "equalization and cancellation" theory). Another argument is that cancellation is so useful for sound organization that one would expect evolution to have provided some sort of approximation to it, possibly at the level of individual neural circuits.

• Temporal averaging (smoothing) of the output of the cancellation stage can be provided by simple integration of spike activity.

• The DDF model provides a "place" coding of both f0s (in terms of a null of activity in a two-dimensional pattern). It is similar to most other models of pitch or localization, and one can assume similar decision processes.

The most questionable aspect of the hypothesis of a physiological implementation of the DDF model has to do with *linearity*. The algorithm requires cancellation by subtraction. If the representation is not perfectly linear, perfect cancellation will not be possible. It is not certain that neural coding provides an approximation to a vector-space representation of acoustic information. Furthermore, even given a linear representation (for example in terms of firing rate within a group of fibers), it is not clear that inhibitory mechanisms could provide canceling, sufficiently accurate for the two successive cancelling steps required by the DDF algorithm.

One non-linear step that is well known is the half-wave rectification that occurs in cochlear transduction. In order to assess its possible effects, the DDF algorithm was run on the half-wave rectified mixed speech signals. The result for one mixed speech token is shown in fig 12. It is evident that the algorithm still functions, but less well.

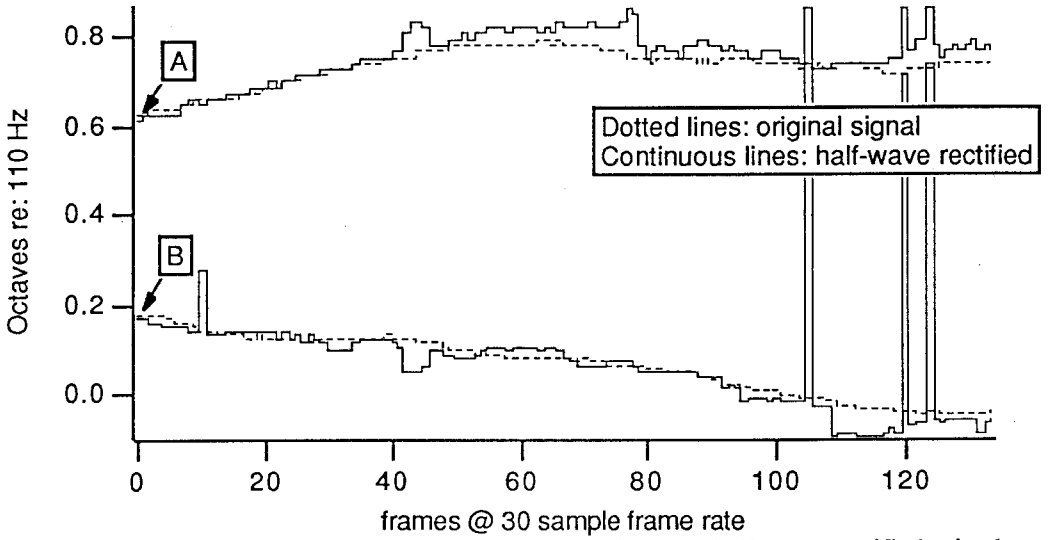


Fig 12. F0 tracks produced by DDF algorithm on half-wave rectified mixed speech (same tokens as in Fig. 6).

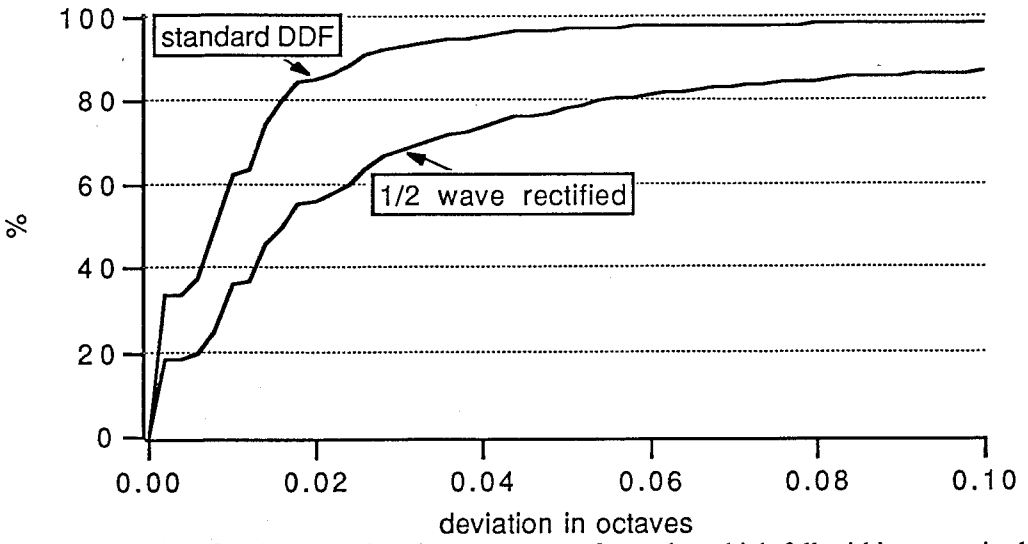


Fig. 13 Plots showing the percentage of samples which fall within a certain deviation, for DDF applied to normal and half-wave rectified speech signal.

It is possible that cochlear filtering enhances overall linearity in that it operates a linear "decomposition" of the sound signal before it reaches non-linear stages. For example, a phase shift of 180 degrees would allow both polarities of a component to be represented despite rectification.

This, and other aspects of a DDF-based physiological model (such as the cancellation stage based on inhibitory neural interaction) are best investigated by computer simulation.

Conclusion

The DDF algorithm provides a reliable and straightforward, if computationally expensive, method for estimating the f0s of two simultaneous speakers.

Acknowledgements

The author wishes to thank the CNRS (Centre National de la Recherche Scientifique) for leave of absence, and the ATR Auditory and Visual Perception Research Laboratories for their kind hospitality and support.

Bibliography

Assmann, P. and Q. Summerfield. (1988). Pitch-pulse asynchrony and the perceptual segregation of competing voices. Speech 88 (7th FASE). 2: 531-538.

Assmann, P. F. and S. Q. (1990). "Modeling the perception of concurrent vowels: vowels with different fundamental frequencies." JASA. 88: 680-697.

Bregman, A. S. (1990). Auditory scene analysis. Cambridge, Mass., MIT Press.

Carney, L. (1990). "Sensitivities of cells in anteroventral cochlear nucleus of cat to spatiotemporal discharge patterns across primary afferents." J. Neurophysiol. 64: 437-456.

Carney, L.H., and Yin, T.C.T. (1989). "Responses of low-frequency cells in the inferior colliculus to interaural time differences of clicks: excitatory and inhibitory components", J. Neurophysiol. 62, 144-161.

Carr, C. E. and M. Konishi. (1990). "A circuit for detection of interaural time differences in the brain stem of the barn owl." J. Neuroscience. 10: 3227-3246.

Chan, J.C.K., Yin, T.C., and Musicant, A.D. (1987). "Effects of interaural time delays of noise stimuli on low-frequency cells in the cat's inferior colliculus. II. Responses to band-pass filtered noises", J. Neurophysiol. 58, 543-561.

de Cheveigné, A. (1990) "Experiments in pitch extraction", ATR Interpreting Telephony Res. Labs. technical report, TR-I-0138, 37p.

Delgutte, B., Kiang, N.Y.-S. (1984a). "Speech coding in the auditory nerve: I. Vowel-like sounds", J. Acoust. Soc. Am. 75, 866-878.

Delgutte, B., and Kiang, N.Y.S. (1984). "Speech coding in the auditory nerve: V. Vowels in background noise", J. Acoust. Soc. Am. 75, 908-918.

Delgutte, B. (1984). "Speech coding in the auditory nerve: II. Processing schemes for vowel-like sounds", J. Acoust. Soc. Am. 75, 879-886.

Frazier, R. H., S. Samsam, L. D. Braida and A. V. Oppenheim. (1976). Enhancement of speech by adaptive filtering. IEEE ICASSP. 251-253.

Ghitza, O. (1988). "Temporal non-place information in the auditory-nerve firing patterns as a front-end for speech recognition in a noisy environment" Journal of Phonetics 16, 109-123.

Geisler, C.D., and Sinex, D.G. (1980) "Responses of primary auditory fibers to combined noise and tonal stimuli", HR 3, 317-334.

Hafter, E. R. and T. N. Buell. (1990). "Restarting the adapted binaural system." JASA. 88: 806-812.

Hermes, D.J. (1988). "Measurement of pitch by subharmonic summation", J. Acoust. Soc. Am. 83, 257-264.

Hess, W. (1983). Pitch determination of speech signals (Springer-Verlag, Berlin), pp 698.

Jeffress, L. A. (1948). "A place theory of sound localization." J. Comp. Physiol. Psychol. 41: 35-39.

Konishi, M., Takahashi, T.T., Wagner, H., Sullivan, W.E., Carr, C.E. (1988). "Neurophysiological and anatomical substrates of sound localization in the owl" in *Auditory function - neurobiological bases of hearing*, edited by G.E. Edelman, W.E. Gall, W.M. Cowan (Wiley, New York), 721-745.

Kuwada, S., Batra, R., Stanford, T.R. (1989) "Monaural and binaural response properties of neurons in the inferior colliculus of the rabbit: effects of sodium pentobarbital", J. Neurophysiol. 61, 269-282.

Langner, G. (1981). "Neuronal mechanisms for pitch analysis in the time domain", Exp. Brain Res. 44, 450-454.

Langner, G. (1983a). "Evidence for neuronal periodicity detection in the auditory system of the guinea fowl: implications for pitch analysis in the time domain", Exp. Brain Res. 52, 333-355.

Langner, G. (1983b). "Neuronal mechanisms for a periodicity analysis in the time domain", in *Hearing — Physiological bases and psychophysics*, edited by R. Klinke, R. Hartmann (Springer-Verlag, Berlin), 334-341.

Langner, G., and Schreiner, C.E. (1988). "Periodicity coding in the inferior colliculus of the cat. I. Neuronal mechanisms", J. Neurophysiol. 60, 1799-1822.

Licklider, J.C.R. (1956). "Auditory frequency analysis" in *Information theory*, edited by C. Cherry (Butterworth, London), 253-268.

Licklider, J.C.R. (1959). "Three auditory theories" in *Psychology, a study of a science*, edited by S. Koch (McGraw-Hill), vol. I, 41-144.

Licklider, J.C.R. (1962). "Periodicity pitch and related auditory process models", International Audiology 1, 11-36.

Lyon, R.F. (1983). "A computational model of binaural localization and separation", Proc. IEEE ICASSP-83, 1148-1151, reprinted in "*Natural computation*", edited by W. Richards (MIT Press, Cambridge Massachusettts), 319-327.

Lyon, R.F. (1984). "Computational models of neural auditory processing", IEEE ICASSP, 36.1.(1-4).

Massaro, D.W. (1987). "Speech perception by ear and by eye: a paradigm for psychological inquiry", Hillsdale NJ: Erlbaum.

McAdams, S. (1982). "Spectral fusion and the creation of auditory images", in "*Music, mind and brain*, edited by M. Clynes (Plenum Press), 279-298.

Meddis, R. and M. J. Hewitt. (1990). "Modelling the identification of concurrent vowels with different fundamental frequencies." Submitted for publication. :

Miller, M.I., and Sachs, M.B. (1984). "Representation of voice pitch in discharge patterns of auditory-nerve fibers", Hearing Research 14, 257-279.

Moore, B.C.J. (1982). *An introduction to the psychology of hearing* (Academic Press, London).

Nagabuchi, H., T. Kobayashi and H. Yamamoto. (1979). "Speech enhancement and suppression in mixed speech." , Trans. IECE **62**(10): 627-634 (in Japanese).

Ney , H. "A time-warping approach to fundamental period estimation", IEEE Trans. SMC 12, 383-388.

Palmer, A. R. (1988). The representation of concurrent vowels in the temporal discharge patterns of auditory nerve fibers. Basic issues in hearing. London, Academic Press.

Palmer, A. R. (1990). "The representation of the spectra and fundamental frequencies of steady-state single- and double-vowel sounds in the temporal discharge patterns of guinea pig cochlear-nerve fibers." J. Acoust. Soc. Am. **88**: 1412-1426.

Parsons, T. W. (1976). "Separation of speech from interfering speech by means of harmonic selection." JASA. **60**: 911-918.

van Noorden, L. (1982). "Two channel pitch perception", in *Music, mind, and brain*, edited by M. Clynes (Plenum Press, New York), 251-269.

Ross, M.J., Schaffer, H.L., Cohen, A. Freudberg, R., Manley, H.J. (1974) "Average Magnitude Difference Function pitch extractor", IEEE Trans. ASSP-22, 353-362.

Nordmark, J. (1963). "Some analogies between pitch and lateralization phenomena", J. Acoust. Soc. Am. 35, 1544-1547.

Schreiner, C.E., Langner, G. (1988a). "Coding of temporal patterns in the central auditory nervous system" in *Auditory function - neurobiological bases of hearing*, edited by G.E. Edelman, W.E. Gall, W.M. Cowan (Wiley, New York), 337-361.

Schreiner, C.E, Langner, G. (1988b). "Periodicity coding in the inferior colliculus of the cat. II. Topographical organization", J. Neurophysiol. 60, 1823-1840.

Schroeder, M. R. (1968). "Period histogram and product spectrum: new methods for fundamental frequency measurement." JASA. **34**: 829-834.

Seneff, S. (1985). *Pitch and spectral analysis of speech based on an auditory synchrony model*, Thesis, MIT tech. rep. 504.

Siebert, W.M. (1970). "Frequency discrimination in the auditory system: place or periodicity mechanisms", Proc. IEEE 58, 723-730.

Shamma, S. (1988). "The acoustic features of speech sounds in a model of auditory processing: vowels and voiceless fricatives", Journal of Phonetics 16, 77-91.

Stubbs, R. J. and Q. Summerfield. (1988). "Evaluation of two voice-separation algorithms using normal-hearing and hearing-impaired listeners." JASA. **84**: 1236-1249.

Stubbs, R. J. and Q. Summerfield. (1990). "Algorithms for separating the speech of interfering talkers: evaluations with voiced sentences, and normal-hearing and hearing-impaired listeners." JASA. **87**: 359-372.

Voigt, H.F., Sachs, M.B., Young, E.D. (1982). "Representation of whispered vowels in discharge patterns of auditory-nerve fibers", Hearing Research 8, 49-58.

Weintraub, M. (1985). "A theory and computational model of auditory monaural sound separation", Thesis, Stanford University, 158p.

Yin, T.C.T., Chan, J.C.K., and Carney, L.H. (1987) "Effects of interaural time delays of noise stimuli on low-frequency cells in the cat's inferior colliculus. III. Evidence for cross-correlation." J. Neurophysiol. 58, 562-583.

Yin, T.C.T., Chan, J.C.K. (1988). "Neural mechanisms underlying interaural time sensitivity to tones and noise" in *Auditory Function - Neurological bases of hearing*, edited by G.E. Edelman, W.E. Gall, W.M. Cowan (Wiley), 385-430.

Yin, T. C. T. and J. C. K. Chan. (1990). "Interaural time sensitivity in medial superior olive of cat." J. Neurophysiol. **64**: 465-488.

Young, E.D., and Sachs, M.B. (1979). "Representation of steady-state vowels in the temporal aspects of the discharge patterns of populations of auditory-nerve fibers", J. Acoust. Soc. Am. 66, 1381-1403.

Appendix I: Implementation details, future development

1. Speed of computation.

The double f0 algorithm uses a brute-force exhaustive search method that is computationally expensive. The algorithm is parallelizable and should run fast on a parallel machine, but on an ordinary machine it is sufficiently slow to impede experimentation. Two methods were investigated to reduce computation time.

a. linked list implementation

The algorithm was implemented using a linked list of split window arrays. The program maintains a linked list of two-dimensional arrays of *partial DDF* values. The partial values are obtained by calculation of the DDF over a short window (30 points, same as analysis increment). The partial DDF is then accumulated with previous values (kept in the linked list) to form the full-window DDF function used by the algorithm.

For highly overlapping windows such as we used, the linked list implementation provides a better than ten-fold speed-up. Using this implementation, if **n** and **m** are the number of lags to be searched in each dimension a minimum of **5nm** operations (integer sum, difference or test) per speech sample are required.

b. Two step iterative algorithm

Despite the linked-list implementation, exhaustive search of the 2-dimensional lag space is expensive. A faster search algorithm is the following:

a) keeping the first lag constant, vary the second searching for an initial period estimate (voice A),

b) using this estimate, comb filter the input speech,

c) keeping the second lag constant, vary the first lag to find a period estimate of voice B,

d) return to a) until both estimates are stable.

This iterative approach is similar to those of Parson (1976) and Nagabuchi et al. (1979) mentioned above, and a priori one might expect it to perform as well as an exhaustive search. This possibility was investigated using pairs of simple stimuli (sine, rectified sine, and 3:5:7 complexes). It appears that, for many frequency ratios, the algorithm gets "stuck", possibly because of insufficient sampling resolution. It is interesting to examine the details of this failure. This is illustrated in figure 14. The plots show the value found in step (c) as a function of the lag used in step (b). Successive plots show the progression of the algorithm, and the last plot shows how it gets stuck in a cycle.

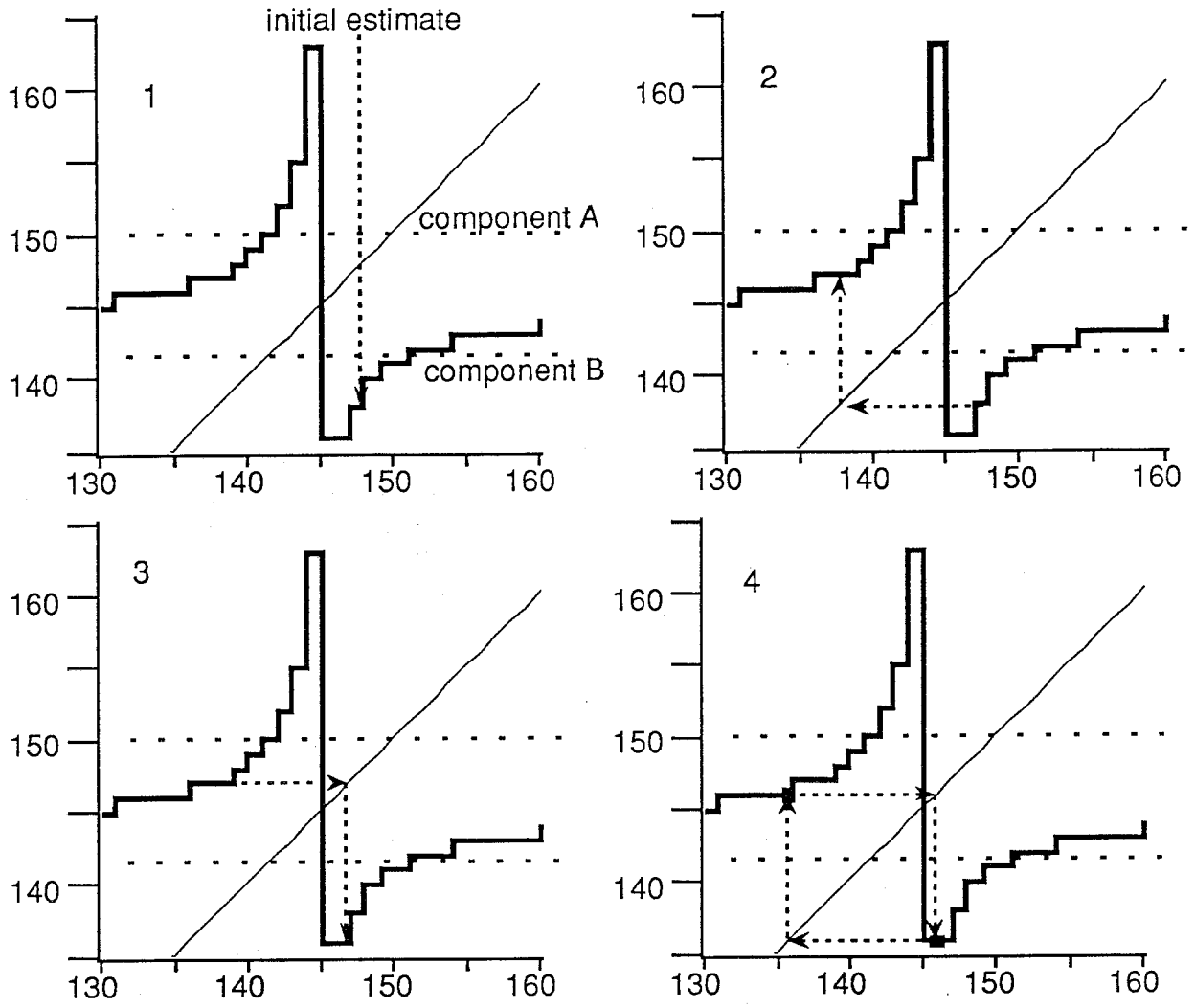


Fig. 14. Plots showing the progression of the iterative algorithm. Each plot shows the period estimate found in step (c) as a function of the lag used in step (b). The algorithm gets stuck in a cycle and produces incorrect estimates (squares) of the periods of components (horizontal dotted lines).

Even though the iterative algorithm fails to find the correct period pair, it can quickly find a region within which the exhaustive search can be carried out, so a hybrid algorithm should be both fast and accurate. This combined algorithm was not tested.

2. Estimating the number of voices

The DDF algorithm produces a pair of estimates whatever the input, but in mixed speech both speakers are not always talking at the same time. Although the possibility was not investigated, the algorithm can in principle give an estimate of the number of voices present. This estimate is obtained by comb-filtering the mixed speech into separate channels, and then applying the AMDF single-voice algorithm to each channel. According to the values of the periodicity measure found in this step and that found when applying AMDF to the mixed speech, relative to an appropriate threshold α:

a) if (PM(A) > α) and (PM(B) > α) and (PM(mixed) <α) decide that there are *two voices,*

b) if (PM(A) > α) and (PM(B) > α) and (PM(mixed) > α) decide that there is *one voice*,

c) if (PM(A) < α) and (PM(B) < α) and (PM(mixed) < α) decide that there are *no voices*(or else more than 2).

In other words, if the signal can be modeled adequately as a single periodic function, there is one voice. If that fails but the signal can modeled by two periodic functions, then there are two voices. If that fails, this indicates either that more than two voices are present, or that the signals were not periodic in the first place (no voices).

The algorithm depends critically on periodicity, and therefore is likely to run into difficulties for real speech. It is probably necessary to adjust thresholds adaptively and incorporate other sources of information (such as continuity constraints).

3. F0 tracking

The DDF algorithm produces an ordered pair of f0 estimates, but makes no attempt to track a given stream. This is no problem if the f0 tracks remain separate, but if they intersect the estimates will "swap streams". F0 tracking across intersections can be done in the following way:

a) apply DDF to mixed speech,

b) comb filter speech into two streams using period estimates from (a),

c) identify all potential intersection points of f0 tracks.

d) At each intersection there are two possible outcomes (tracks either cross or they don't): using the filtered streams, compare spectral or amplitude continuity across the intersection for both eventualities. Choose the tracks that give the best continuity. This algorithm is likely to run into trouble in cases where the f0 tracks don't intersect clearly (i.e. remain together for a long time).

4. Voice separation

The issue of voice separation is discussed in detail in many of the references given in the introduction, and so was voluntarily left outside the scope of this work. However, I will briefly investigate one of the issues: what do when a component of the mixed speech is an exact harmonic of both fundamentals. A similar question is of interest in hearing: how does such a component affect each stream in the case of segregation? (Bregman 1990)

There is no way to tell from the amplitude of such a component what the amplitudes of the original components were (apart from a lower bound on their sum), because of phase incertitude. Several courses of action are possible, the simplest being to cancel the component from both outputs, leaving a "hole" in their spectrum. The concern in this case is whether the spectral distortion affects intelligibility.

This question was investigated informally. Arbitrary f0 tracks from the ATR f0 data base (hand-corrected) were used to control the lag parameter of a comb filter applied to speech data (sentences). The filtering was gated on and off according to the presence or absence of voicing in the f0 track, and the transitions were smoothed with a 20 ms rised cosine function. The distortion is clearly audible, but does not at all affect intelligibility, suggesting that "holes" in the spectrum may be acceptable in practice, especially relatively to other sources of distortion that would appear if the method were applied to filter real mixed speech.

5. The cocktail party effect.

It has often been noted that pitch phenomena and binaural phenomena are related (Nordmark 1963). This is not surprising, since the major cue for source localization (interaural time difference or ITD), is similar in nature to a period: given an appropriate internal time lag, signals from both ears can be brought into correspondence; in a similar way a delayed periodic signal can be brought into correspondence with itself when the delay equals the period. From an engineering point of view, given signals from two microphones, each source can be separately canceled if the differential delays from the sound sources to the microphones are known.

To obtain this information, an algorithm similar to DDF can be applied in the case where one of the sources is periodic. For that purpose, the cascade of two monochannel comb filters is replaced by a combination of one monochannel comb filter (to cancel the periodic source) and one *bi-channel* comb filter (to cancel the other source). As for DDF, the algorithm searches a two dimensional lag parameter space looking for the minimum of the output signal.

More concretely, suppose sources A and B (B periodic), recorded from microphones 1 and 2 placed such that there is a different time delay for both sources. Given adequate time origins, the signal at microphone 1 is

$$S_1(t) = S_A(t) + S_B(t + \delta B)$$

and that at microphone 2:

$$S_2(t) = S_B(t) + S_A(t + \delta A)$$

The bi-channel comb filter calculates:

$$O(t) = S_1(t) - S_2(t+lagA)$$

If lagA is equal to - δA, source A is canceled:

$$O(t) = S_B(t + \delta B) - S_B(t - \delta A).$$

If $S_B(t)$ is periodic then so is O(t), so if O(t) is fed to a monochannel comb filter with lag parameter lagB equal to the period of source B, the output will be null. The algorithm searches the (lagA, lagB) space looking for such a null. If both sources are periodic, then the algorithm will find two such nulls, one for each source. Unfortunately I have found no way to solve the problem if *neither* source is periodic (unless some other criterion is used for deciding when an interfering source is canceled).

6. Improvements of the DDF algorithm

a. Amplitude variation control

The DDF algorithm assumes periodicity, and it is likely to fail when that assumption is not true. One source of aperiodicity is amplitude change. For single voice f0 estimation, amplitude change can be compensated by amplitude normalization of the speech signal prior to f0 estimation. This cannot work with mixed speech. One

possible solution is to do amplitude normalization after speech separation (in a a two-step iterative algorithm): once a stream has been separated, its amplitude and spectral course can be better model, so it can better be canceled, allowing measurement of the second f0. Another possible solution is based on an idea by Barry Vercoe (Media Lab, MIT). Instead of subtracting the delayed signal, one subtracts the *mean of delayed and advanced* signals. The impulse response of one stage of such a comb filter is:

$$\delta(t) - (\delta(t - lag) + \delta(t + lag))/2$$

Any linear change in amplitude is automatically compensated for. Two such comb filters can be cascaded and used within the DDF algorithm. The result of this modified algorithm is shown in Fig. 15.

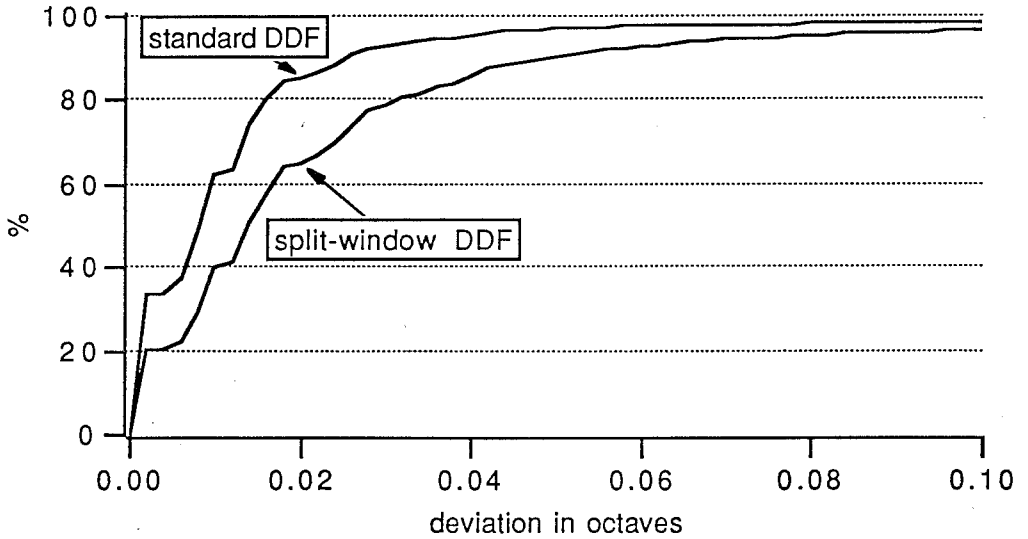


Fig. 15. Plots showing the percentage of estimates that differ less than a certain amount from reference values, for standard and modified DDF.

The estimates produced by the modified algorithm are on the whole less good than those obtained from the standard algorithm. It may be that the data set contained too few amplitude changes, so the modification had no opportunity of proving its worth. It may also be that the tracks registered less well (comparisons should have been made with a similarly modified AMDF). However the split-window scheme also proved disappointing for single voice speech extraction (de Cheveigné 1990). Amplitude change is a relatively minor factor of aperiodicity compared to other factors such as spectral change, and it seems that the assumption of periodicity over a two-period span makes the algorithm less robust.

b. Filter-bank pre-processing

In contrast to the methods of mixed voice separation reviewed in the introduction, the DDF algorithm does not require spectral analysis or filtering. However one might expect filtering to improve the signal-to-noise ration within certain channels, and this might allow more reliable f0 estimation of a weak voice. This is a subject for future investigation.

c. Non-linear summation

The DDF algorithm integrates the output of the comb filters over a window. This is to average out fluctuations, but the operation can also be seen as a way of integrating information from a number of individual measurements. From that perspective, a method closer to bayesian integration might be more appropriate than linear summation. For example, for a random signal it would be a rare occurrence for

the output of the filters to remain zero for a number of samples. Such an occurrence should strongly suggest periodicity, but is given little weight by linear summation. One way of manipulating the weight of occurrences such as this is to perform a non-linear transformation on the comb filter output values before integration.

Several non-linear transformations were investigated. To save time, they were applied to magnitudes summed over partial windows (see description of linked-list implementation above) instead of to individual magnitudes. Each window contains 14 partial windows 30 samples in length.

• Square

Squaring gives more emphasis to large local values that to smaller distributed values, which is the opposite of what we want. It however of interest to examine this case because of its relation to autocorrelation: replacing the sum of magnitudes by a sum of squares in the AMDF produces a function closely related to the autocorrelation function (Ney 1982):

$$\int_{-\infty}^{+\infty} (s(t) - s(t+T))^2 dt = \int_{-\infty}^{+\infty} s(t)^2 dt + \int_{-\infty}^{+\infty} s(t+T)^2 dt - 2\int_{-\infty}^{+\infty} (s(t)s(t+T))dt$$

$$= 2\,ACF(0) - 2\,ACF(T)$$

A peak in ACF corresponds to a dip in the squared difference function, and algorithms based on either information should give similar results.

The histogram of deviations of f0 estimates obtained using sum of squares is shown in Fig. 16. The skirts of the histogram show large errors that do not occur with standard DDF (compare with Fig. 9).

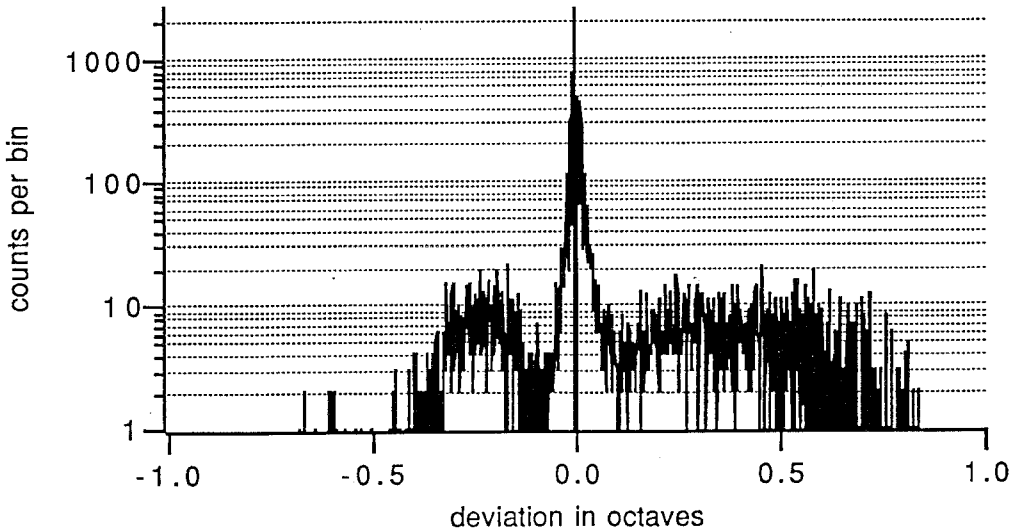


Fig. 16. Histogram of deviation between f0 estimates produced by the modified double voice and reference values. The modified algorithm used a minimum sum-of-squares criterion.

• Square root

Taking the square root should on the contrary improve performance. This was tried, but the result is practically indistinguishable from standard DDF. It is possible that the test data was too "clean" to allow any improvement.

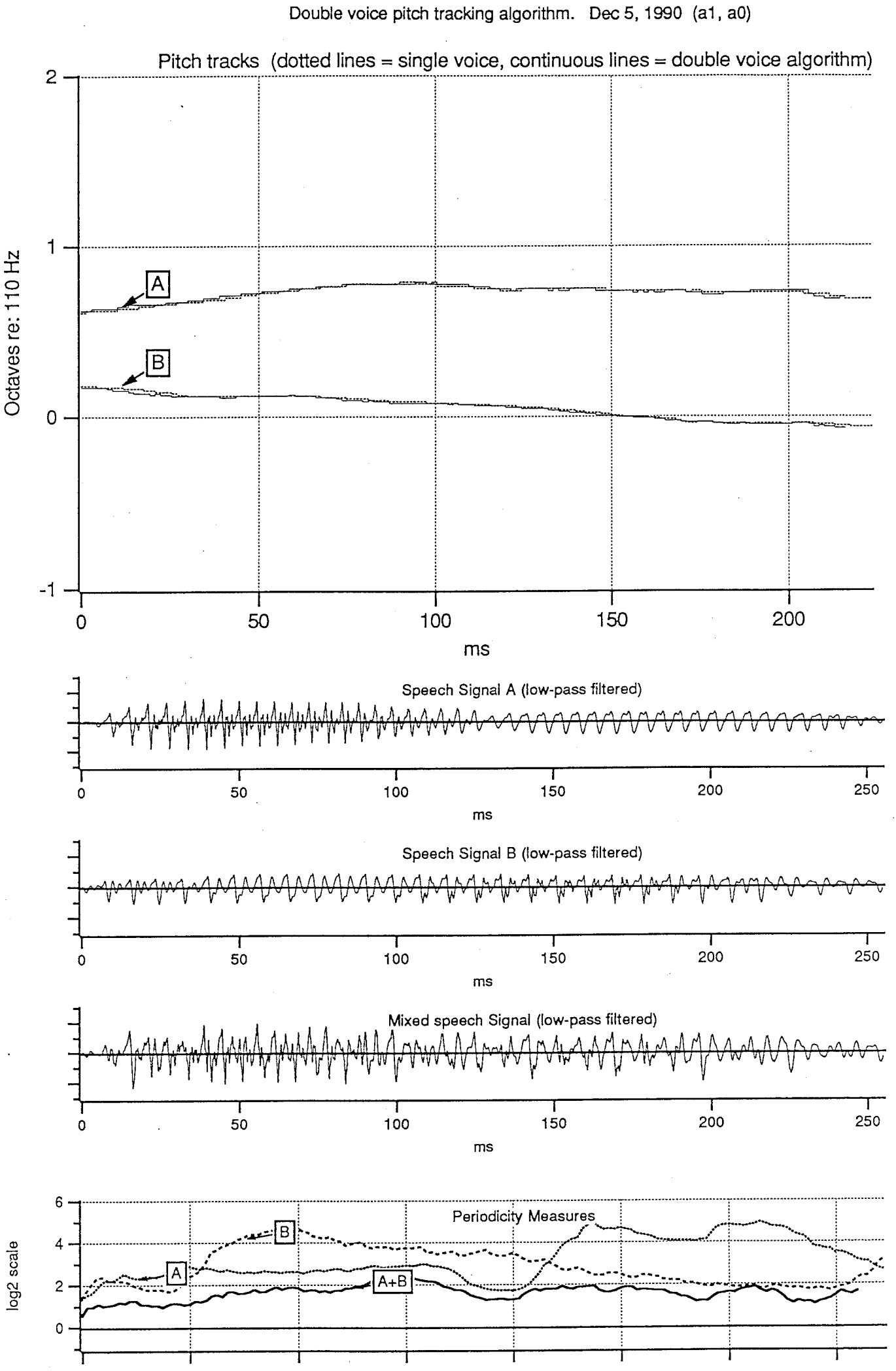
• Logarithm

Taking the logarithm of values before summation should have a similar effect as taking the square root. As in that case, performance is indistinguishable from standard DDF.

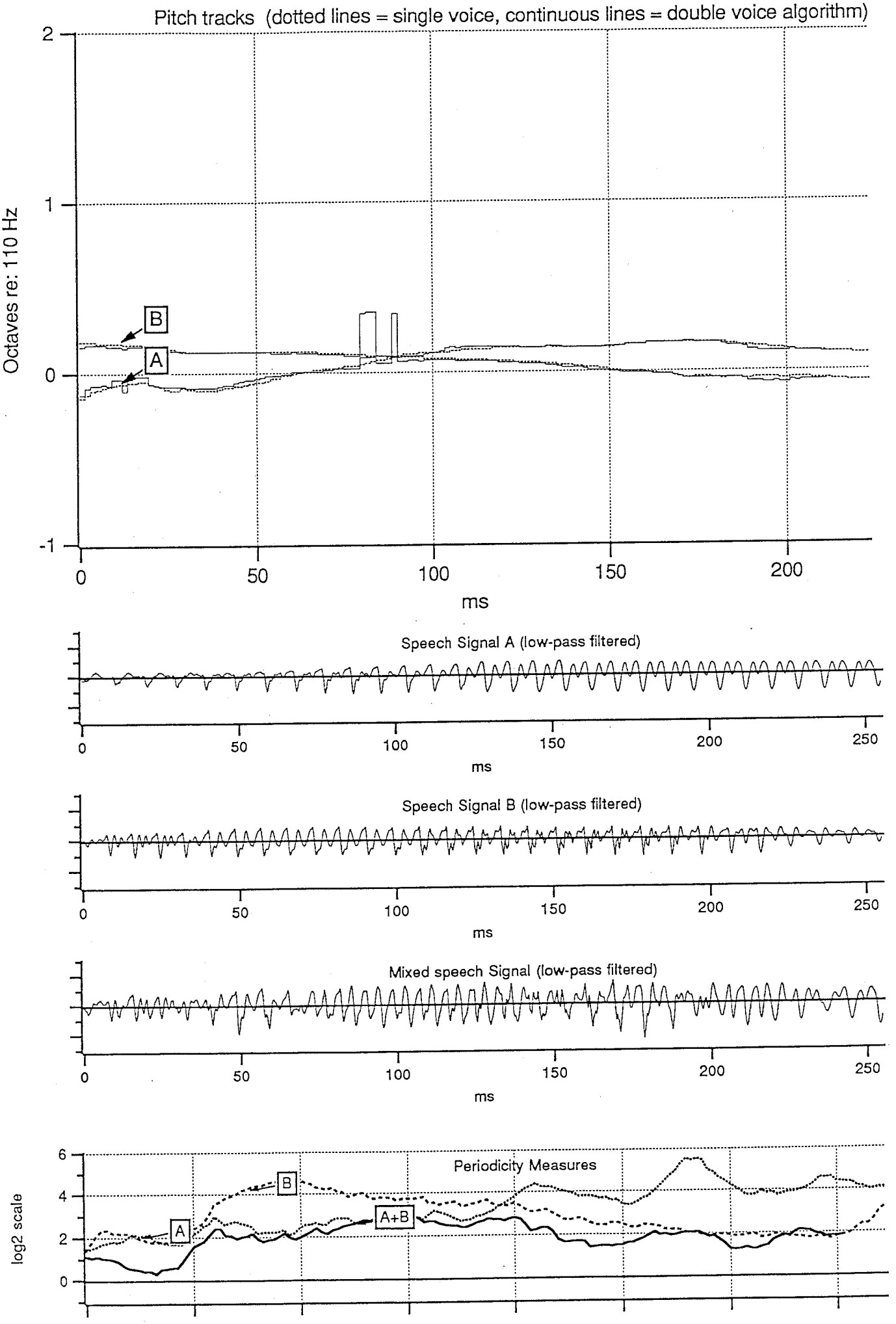
The issue of information integration is worth investigating in more detail.

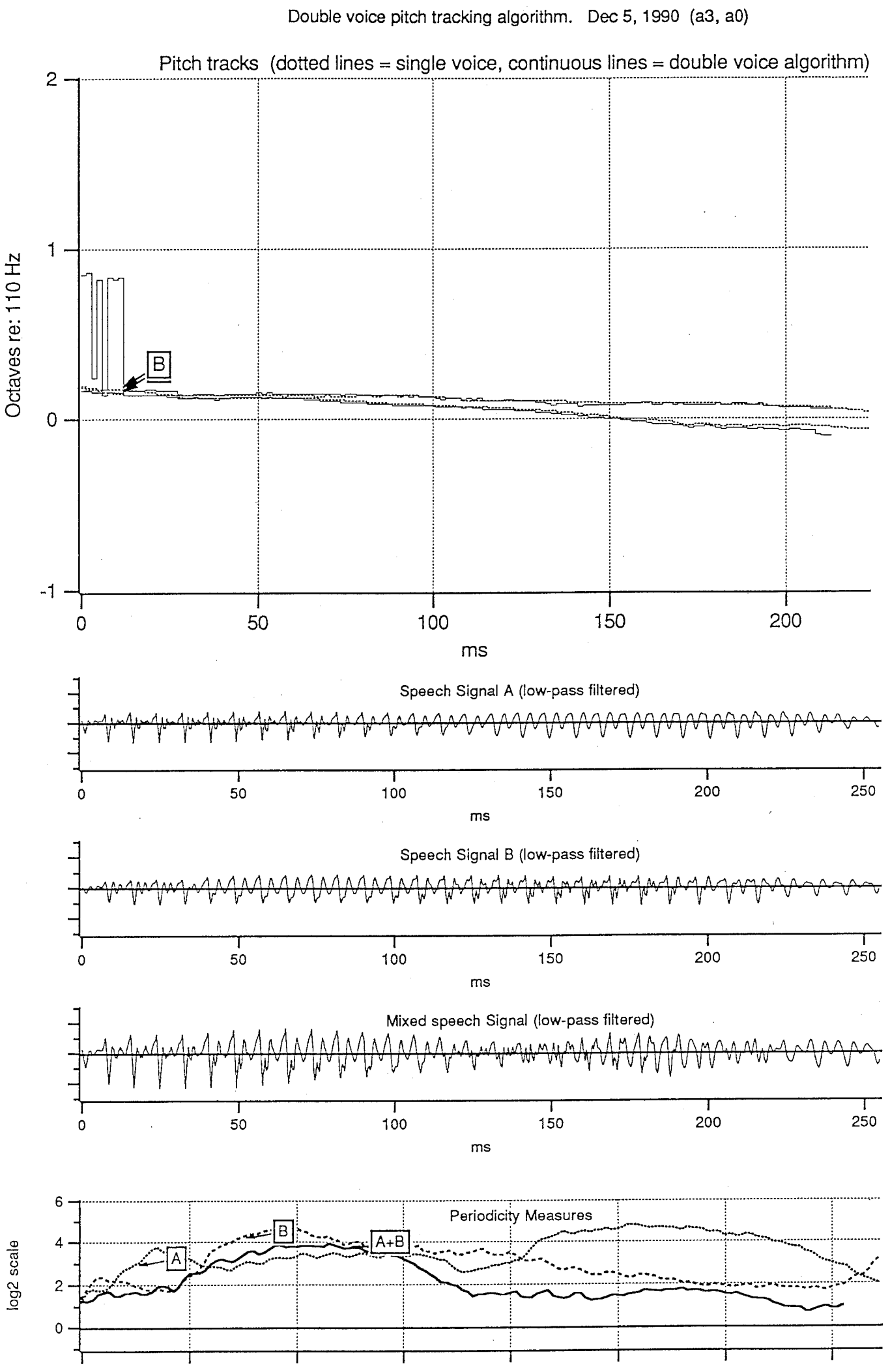
Appendix II: Examples of mixed speech f0 estimation.

The following pages show examples of the performance of the DDF mixed speech f0 estimation algorithm. Male speaker tokens are noted a0, a1, etc., female speaker tokens are noted b0, b1, etc..

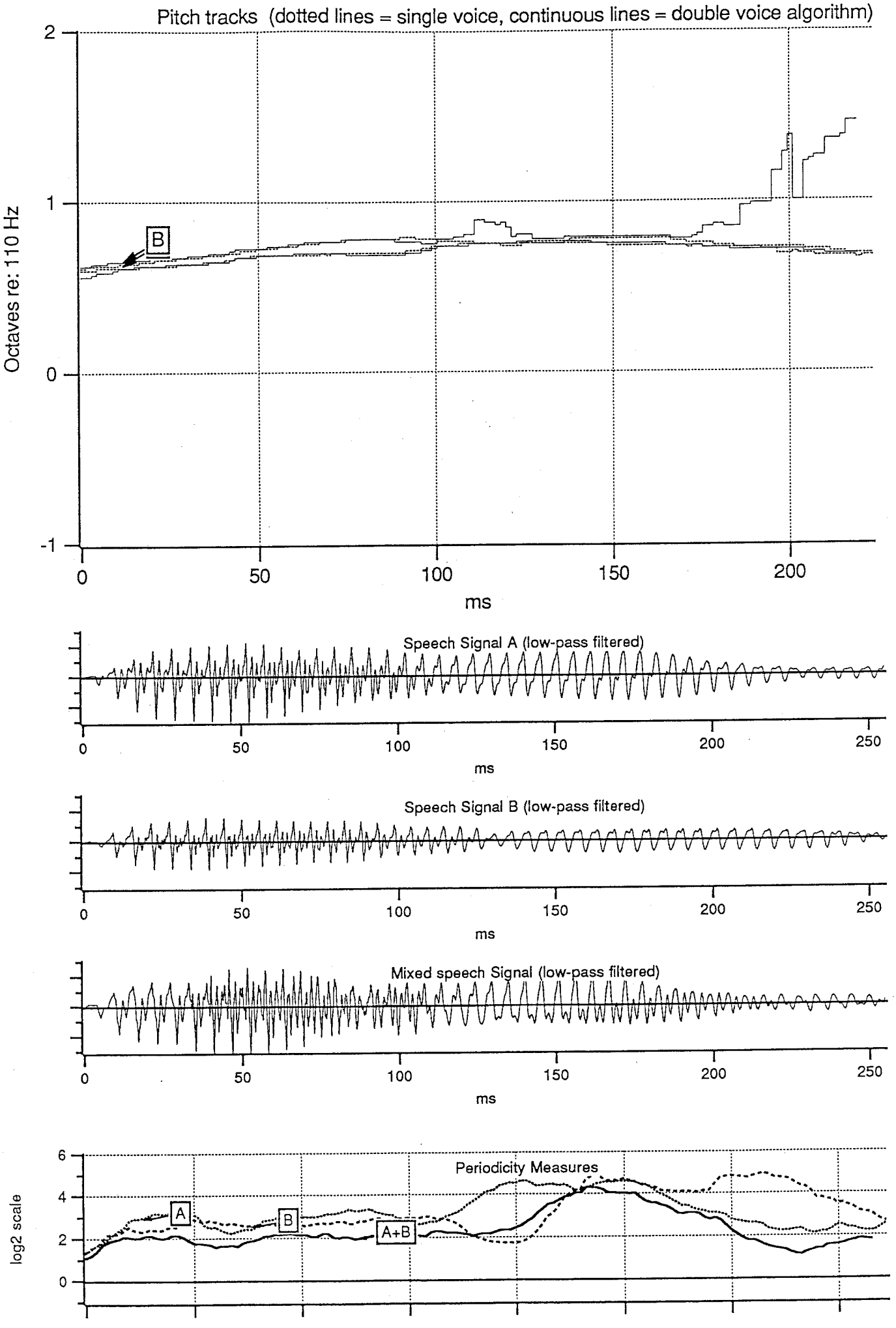

Double voice pitch tracking algorithm. Dec 5, 1990 (a1, a0)
Pitch tracks (dotted lines = single voice, continuous lines = double voice algorithm)
Octaves re: 110 Hz
2
1
0
-1
A
B
0
50
100
150
200
ms
Speech Signal A (low-pass filtered)
0
50
100
150
200
250
ms
Speech Signal B (low-pass filtered)
0
50
100
150
200
250
ms
Mixed speech Signal (low-pass filtered)
0
50
100
150
200
250
ms
log2 scale
6
4
2
0
Periodicity Measures
B
A
A+B

Double voice pitch tracking algorithm. Dec 5, 1990 (a2, a0)

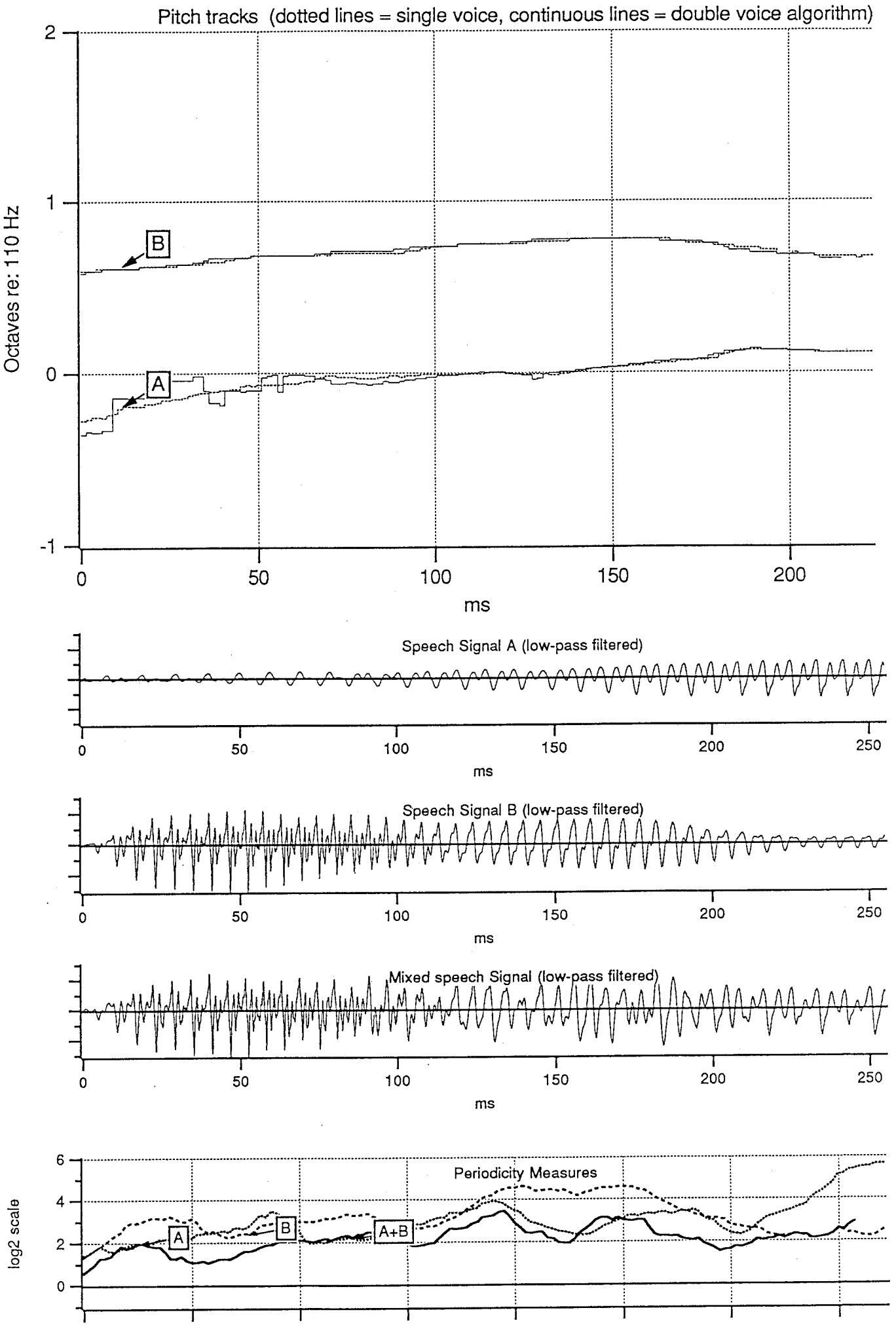


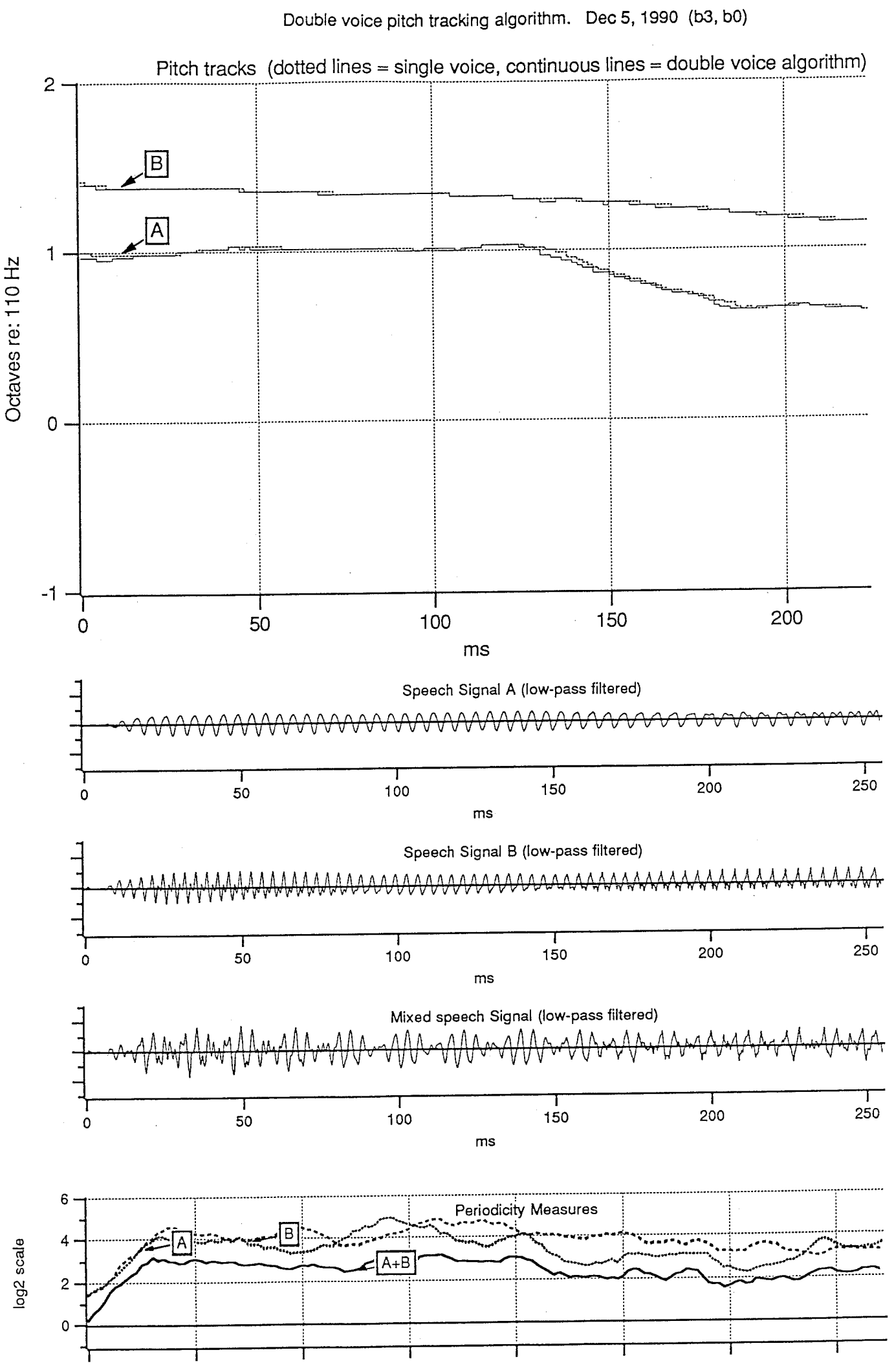

Double voice pitch tracking algorithm. Dec 5, 1990 (a3, a0)
Pitch tracks (dotted lines = single voice, continuous lines = double voice algorithm)
Octaves re: 110 Hz
2
1
0
-1
B
0
50
100
150
200
ms
Speech Signal A (low-pass filtered)
0
50
100
150
200
250
ms
Speech Signal B (low-pass filtered)
0
50
100
150
200
250
ms
Mixed speech Signal (low-pass filtered)
0
50
100
150
200
250
ms
log2 scale
6
4
2
0
Periodicity Measures
A
B
A+B

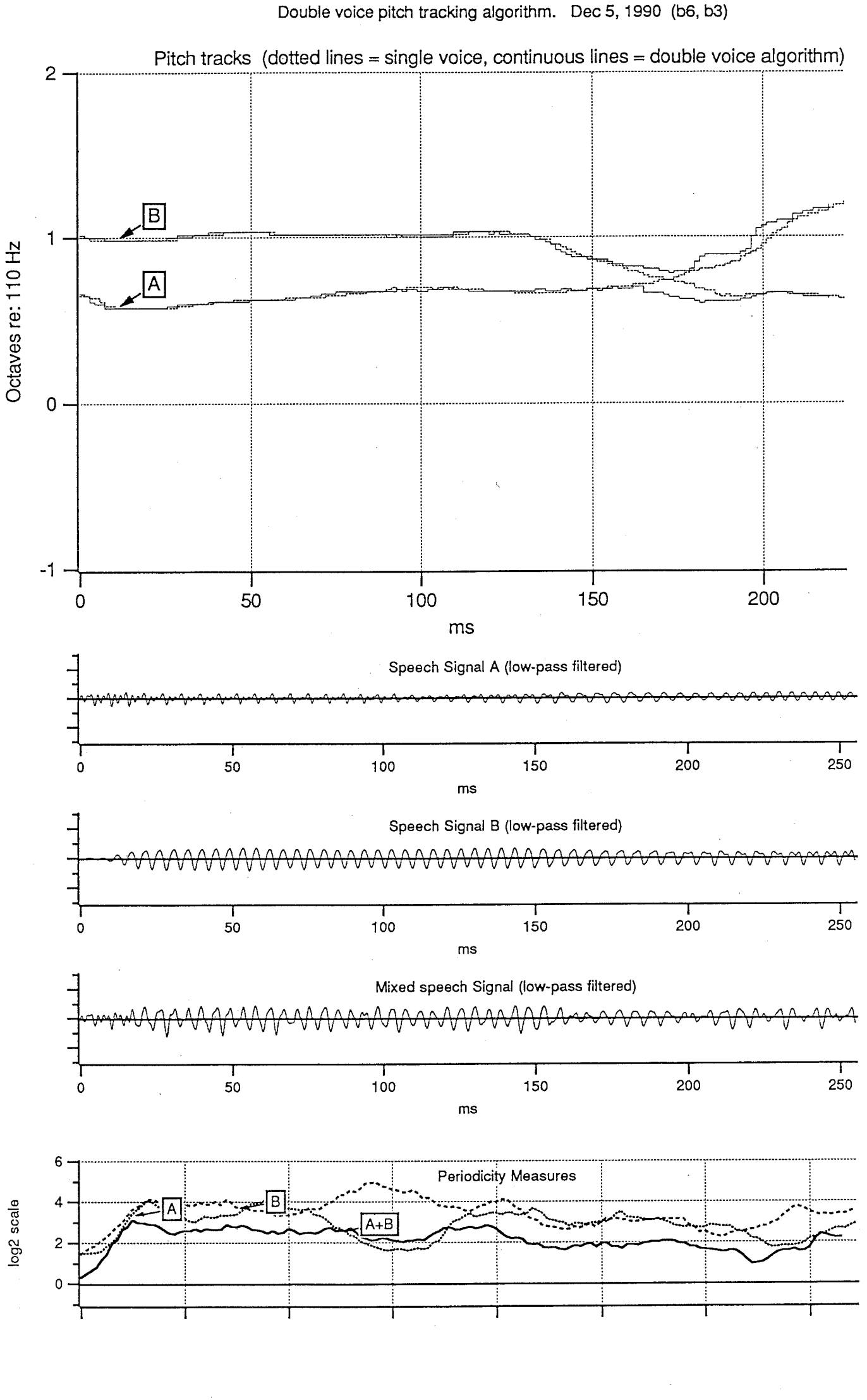
Double voice pitch tracking algorithm. Dec 6, 1990 (a4, a1)

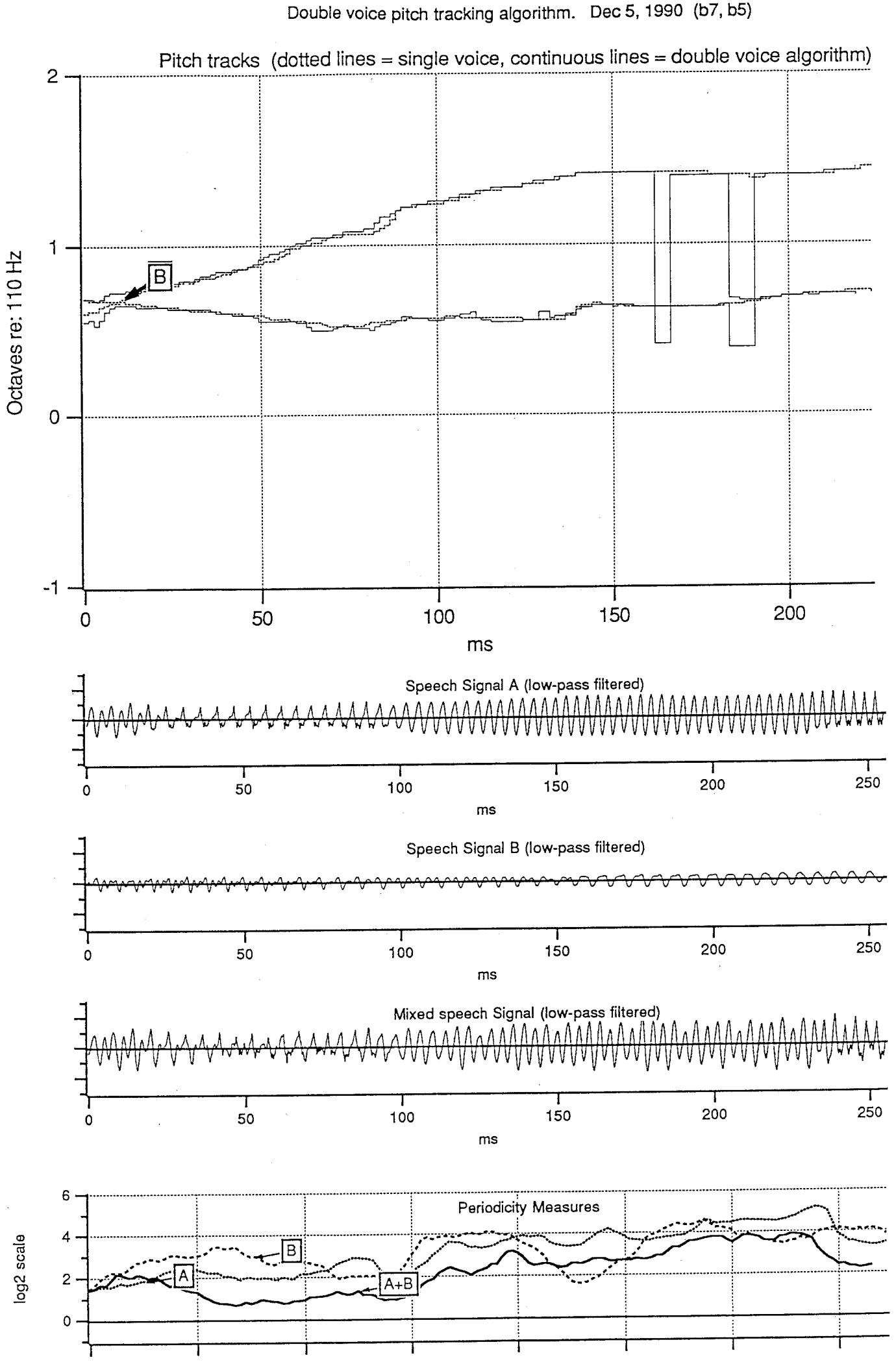


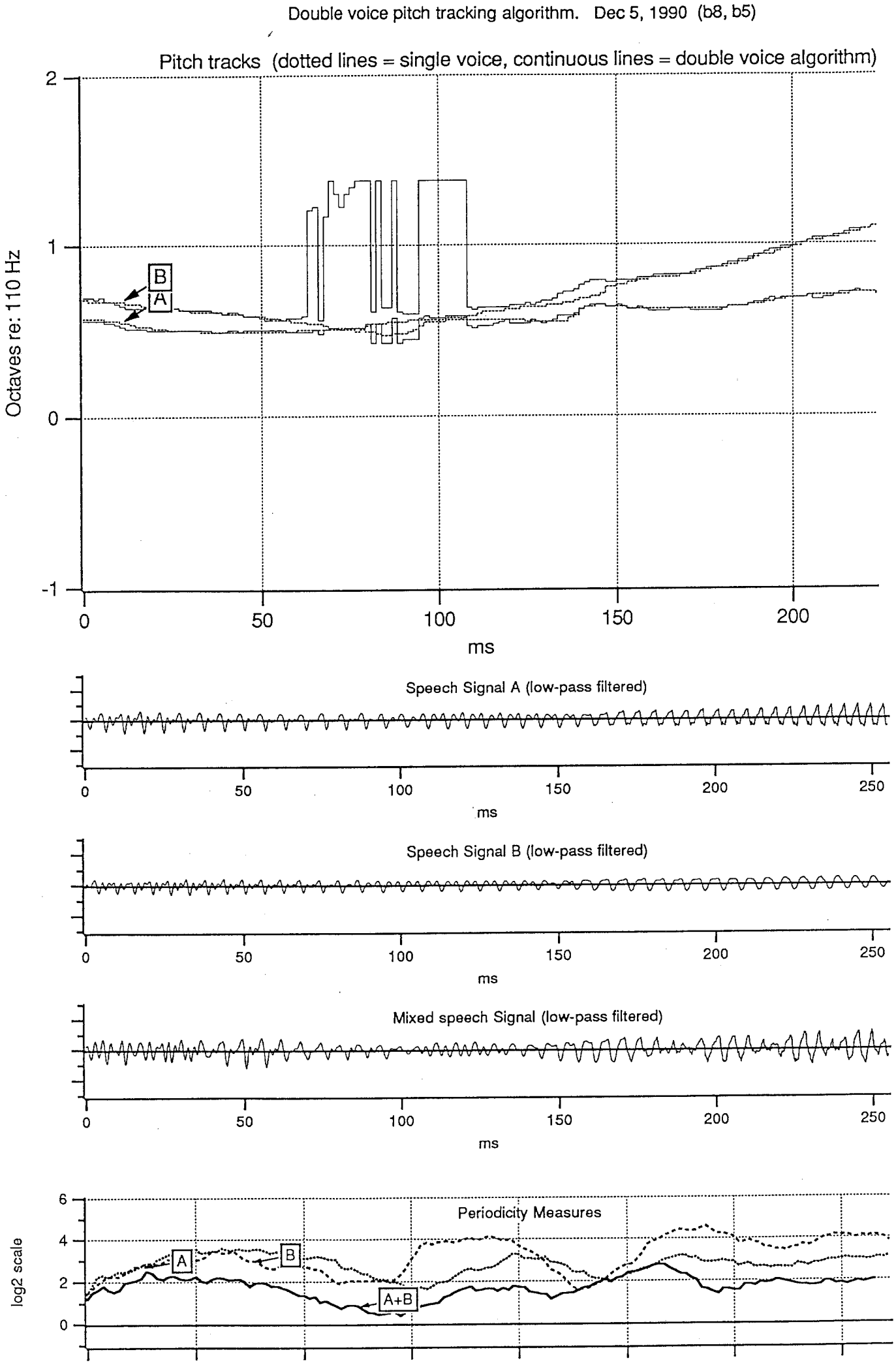
Double voice pitch tracking algorithm. Dec 6, 1990 (a6, a4)




Double voice pitch tracking algorithm. Dec 5, 1990 (b3, b0)
Pitch tracks (dotted lines = single voice, continuous lines = double voice algorithm)
Octaves re: 110 Hz
2
1
0
-1
B
A
0
50
100
150
200
ms
Speech Signal A (low-pass filtered)
0
50
100
150
200
250
ms
Speech Signal B (low-pass filtered)
0
50
100
150
200
250
ms
Mixed speech Signal (low-pass filtered)
0
50
100
150
200
250
ms
log2 scale
6
4
2
0
Periodicity Measures
A
B
A+B


Double voice pitch tracking algorithm. Dec 5, 1990 (b6, b3)
Pitch tracks (dotted lines = single voice, continuous lines = double voice algorithm)
Octaves re: 110 Hz
2
1
0
-1
B
A
0
50
100
150
200
ms
Speech Signal A (low-pass filtered)
0
50
100
150
200
250
ms
Speech Signal B (low-pass filtered)
0
50
100
150
200
250
ms
Mixed speech Signal (low-pass filtered)
0
50
100
150
200
250
ms
log2 scale
6
4
2
0
Periodicity Measures
A
B
A+B


Double voice pitch tracking algorithm. Dec 5, 1990 (b7, b5)
Pitch tracks (dotted lines = single voice, continuous lines = double voice algorithm)
Octaves re: 110 Hz
2
1
0
-1
B
0
50
100
150
200
ms
Speech Signal A (low-pass filtered)
0
50
100
150
200
250
ms
Speech Signal B (low-pass filtered)
0
50
100
150
200
250
ms
Mixed speech Signal (low-pass filtered)
0
50
100
150
200
250
ms
log2 scale
6
4
2
0
Periodicity Measures
B
A
A+B


Double voice pitch tracking algorithm. Dec 5, 1990 (b8, b5)
Pitch tracks (dotted lines = single voice, continuous lines = double voice algorithm)
Octaves re: 110 Hz
2
1
0
-1
B
A
0
50
100
150
200
ms
Speech Signal A (low-pass filtered)
0
50
100
150
200
250
ms
Speech Signal B (low-pass filtered)
0
50
100
150
200
250
ms
Mixed speech Signal (low-pass filtered)
0
50
100
150
200
250
ms
log2 scale
6
4
2
0
Periodicity Measures
A
B
A+B