Institute of Phonetic Sciences,
University of Amsterdam,
Proceedings 17 (1993), 97-110.

ACCURATE SHORT-TERM ANALYSIS OF THE FUNDAMENTAL FREQUENCY AND THE HARMONICS-TO-NOISE RATIO OF A SAMPLED SOUND

Paul Boersma

Abstract

We present a straightforward and robust algorithm for periodicity detection, working in the lag (autocorrelation) domain. When it is tested for periodic signals and for signals with additive noise or jitter, it proves to be several orders of magnitude more accurate than the methods commonly used for speech analysis. This makes our method capable of measuring harmonics-to-noise ratios in the lag domain with an accuracy and reliability much greater than that of any of the usual frequency-domain methods.

By definition, the best candidate for the acoustic pitch period of a sound can be found from the position of the maximum of the autocorrelation function of the sound, while the degree of periodicity (the harmonics-to-noise ratio) of the sound can be found from the relative height of this maximum.

However, *sampling* and *windowing* cause problems in accurately determining the position and height of the maximum. These problems have led to inaccurate time-domain and cepstral methods for pitch detection, and to the exclusive use of frequency-domain methods for the determination of the harmonics-to-noise ratio.

In this paper, I will tackle these problems. Table 1 shows the specifications of the resulting algorithm for two spectrally maximally different kinds of periodic sounds: a sine wave and a periodic pulse train; other periodic sounds give results between these.

Table 1. The accuracy of the algorithm for a sampled sine wave and for a correctly sampled periodic pulse train, as a function of the number of periods that fit in the duration of a Hanning window. These results are valid for pitch frequencies up to 80% of the Nyquist frequency. These results were measured for a sampling frequency of 10 kHz and window lengths of 40 ms (for pitch) and 80 ms (for HNR), but generalize to other sampling frequencies and window lengths (see section 5).

Periods per window	Pitch determination error $\Delta F/F$		Resolution of determination of harmonics-to-noise ratio	
	sine wave	pulse train	sine wave	pulse train
> 3	$< 5\cdot10^{-4}$	$< 5\cdot10^{-5}$	> 27 dB	> 12 dB
> 6	$< 3\cdot10^{-5}$	$< 5\cdot10^{-6}$	> 40 dB	> 29 dB
> 12	$< 4\cdot10^{-7}$	$< 2\cdot10^{-7}$	> 55 dB	> 44 dB
> 24	$< 2\cdot10^{-8}$	$< 2\cdot10^{-8}$	> 72 dB	> 58 dB

1 Autocorrelation and periodicity

For a time signal $x(t)$ that is *stationary* (i.e., its statistics are constant), the *autocorrelation* $r_x(\tau)$ as a function of the *lag* τ is defined as

$$r_x(\tau) \equiv \int x(t)\, x(t+\tau)\, dt \tag{1}$$

This function has a global maximum for $\tau = 0$. If there are also global maxima outside 0, the signal is called *periodic* and there exists a lag T_0, called the *period*, so that all these maxima are placed at the lags nT_0, for every integer n, with $r_x(nT_0) = r_x(0)$. The *fundamental frequency* F_0 of this periodic signal is defined as $F_0 = 1/T_0$. If there are no global maxima outside 0, there can still be local maxima. If the highest of these is at a lag τ_{max}, and if its height $r_x(\tau_{max})$ is large enough, the signal is said to have a periodic part, and its *harmonic strength* R_0 is a number between 0 and 1, equal to the local maximum $r'_x(\tau_{max})$ of the *normalized autocorrelation*

$$r'_x(\tau) \equiv \frac{r_x(\tau)}{r_x(0)} \tag{2}$$

We could make such a signal $x(t)$ by taking a periodic signal $H(t)$ with a period T_0 and adding a noise $N(t)$ to it. We can infer from equation (1) that if these two parts are uncorrelated, the autocorrelation of the total signal equals the sum of the autocorrelations of its parts. For zero lag, we have $r_x(0) = r_H(0) + r_N(0)$, and if the noise is white (i.e., if it does not correlate with itself), we find a local maximum at a lag $\tau_{max} = T_0$ with a height $r_x(\tau_{max}) = r_H(T_0) = r_H(0)$. Because the autocorrelation of a signal at zero lag equals the power in the signal, the normalized autocorrelation at τ_{max} represents the relative power of the periodic (or *harmonic*) component of the signal, and its complement represents the relative power of the noise component:

$$r'_x(\tau_{max}) = \frac{r_H(0)}{r_x(0)} \quad ; \quad 1 - r'_x(\tau_{max}) = \frac{r_N(0)}{r_x(0)} \tag{3}$$

This allows us to define the logarithmic *harmonics-to-noise ratio* (HNR) as

$$HNR\ (\text{in dB}) = 10 \cdot {}^{10}\log \frac{r'_x(\tau_{max})}{1 - r'_x(\tau_{max})} \tag{4}$$

This definition follows the same idea as the frequency-domain definitions used by most other authors, but yields much more accurate results thanks to the precision with which we can estimate $r_x(\tau)$. For perfectly periodic sounds, the HNR is infinite.

For non-stationary (i.e., dynamically changing) signals, the *short-term* autocorrelation at a time t is estimated from a short windowed segment of the signal centred around t. This gives estimates $F_0(t)$ for the local fundamental frequency and $R_0(t)$ for the local harmonic strength. If we want these estimates to have a meaning at all, they should be as close as possible to the quantities derived from equation (1), if we perform a short-term analysis on a stationary signal. Sections 2 and 3 show how to cope with the windowing and sampling problems that arise. Section 4 presents the complete algorithm. Sections 5, 6, and 7 investigate the performance of the algorithm for three kinds of stationary signals: periodic signals without perturbations, with additive noise, and with jitter.

2 Windowing and the lag domain

Candidates for the fundamental frequency of a continuous signal $x(t)$ at a time t_{mid} can be found from the local maxima of the autocorrelation of a short segment of the sound centred around t_{mid}. In figure 1, we summarize the algorithm for the speech-like signal $x(t) = (1 + 0.3 \sin 2\pi\, 140\, t) \sin 2\pi\, 280\, t$, which has a fundamental frequency of 140 Hz and a strong 'formant' at 280 Hz. The algorithm runs as follows:

Step 1. We take from the signal $x(t)$ a piece with duration T (the *window length*, 24 ms in figure 1), centred around t_{mid} (12 ms in figure 1). We subtract from this piece its mean μ_x and multiply the result by a *window function* $w(t)$, so that we get the *windowed signal*

$$a(t) = \left(x(t_{mid} - \tfrac{1}{2}T + t) - \mu_x\right) w(t) \tag{5}$$

The window function $w(t)$ is symmetric around $t = \frac{1}{2}T$ and zero everywhere outside the time interval [0, T]. Our choice is the *sine-squared* or *Hanning window*, given by

$$w(t) = \tfrac{1}{2} - \tfrac{1}{2} \cos \frac{2\pi t}{T} \tag{6}$$

We will see how the Hanning window compares to several other window shapes.

Step 2. The normalized autocorrelation $r_a(\tau)$ (we suppress the primes from now on) of the windowed signal is a symmetric function of the lag τ:

$$r_a(\tau) = r_a(-\tau) = \frac{\int_0^{T-\tau} a(t)\, a(t+\tau)\, dt}{\int_0^T a^2(t)\, dt} \tag{7}$$

In the example of figure 1, we can see that the highest of these maxima is at a lag that corresponds to the first formant (3.57 ms), whereas we would like it to be at a lag that corresponds to the F_0 (7.14 ms). For this reason, Hess (1992) deems the autocorrelation method "rather sensitive to strong formants". Moreover, the skewing of the autocorrelation function makes the estimate of the lag of the peak too low, and therefore the pitch estimate too high (e.g., for 3 periods of a sine wave in a Hanning window, the difference is 6%). One method commonly used to overcome the first problem, is to filter away all frequencies above 900 Hz (Rabiner, 1977), which should kill all formants except the first, and estimate the pitch from the second maximum. This is not a very robust method, because we often run into higher formants below 900 Hz and fundamental frequencies above 900 Hz. Other methods to lose the formant include centre clipping, spectral flattening, and so on. Such ad-hoc measures render the method speech- and speaker-dependent. All these patches to the autocorrelation method are unnecessary, for there is a simple remedy:

Step 3. We compute the normalized autocorrelation $r_w(\tau)$ of the window in a way exactly analogous to equation (7). The normalized autocorrelation of a Hanning window is

$$r_w(\tau) = \left(1 - \frac{|\tau|}{T}\right)\left(\frac{2}{3} + \frac{1}{3}\cos\frac{2\pi\tau}{T}\right) + \frac{1}{2\pi}\sin\frac{2\pi|\tau|}{T} \tag{8}$$

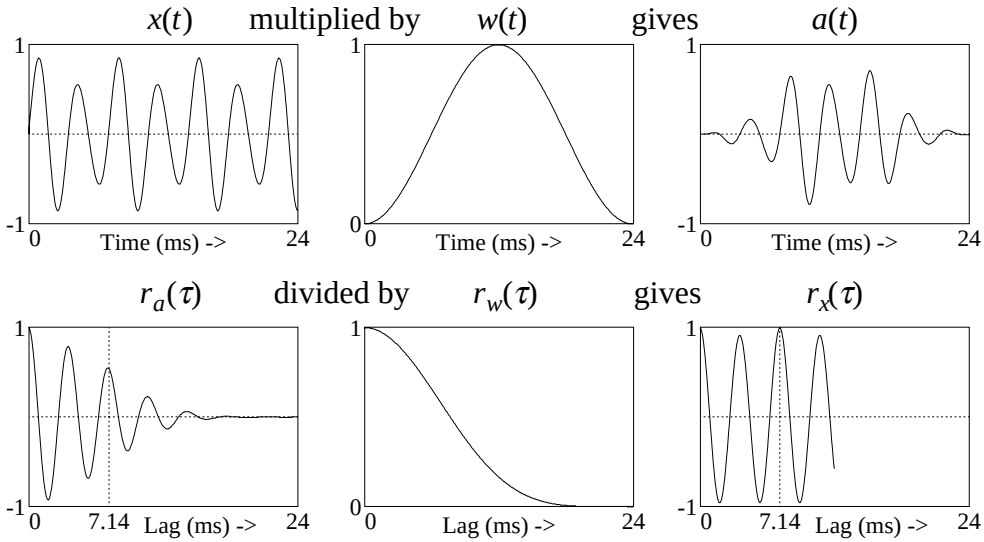

Fig. 1. How to window a sound segment, and how to estimate the autocorrelation of a sound segment from the autocorrelation of its windowed version. The estimated autocorrelation $r_x(\tau)$ is not shown for lags longer than half the window length, because it becomes less reliable there for signals with few periods per window.

To estimate the autocorrelation $r_x(\tau)$ of the original signal segment, we divide the autocorrelation $r_a(\tau)$ of the windowed signal by the autocorrelation $r_w(\tau)$ of the window:

$$r_x(\tau) \approx \frac{r_a(\tau)}{r_w(\tau)} \tag{9}$$

This estimation can easily be seen to be exact for the constant signal $x(t) = 1$ (without subtracting the mean, of course); for periodic signals, it brings the autocorrelation peaks very near to 1 (see figure 1). The need for this correction seems to have gone by unnoticed in the literature; e.g., Rabiner (1977) states that "no matter which window is selected, the effect of the window is to taper the autocorrelation function smoothly to 0 as the autocorrelation index increases". With equation (9), this is no longer true.

The accuracy of the algorithm is determined by the reliability of the estimation (9), which depends directly on the shape of the window. For instance, for a periodic pulse train, which is defined as

$$x(t) = \sum_{n=-\infty}^{+\infty} \delta\left(t - t_0 - nT_0\right) \tag{10}$$

where T_0 is the period and t_0/T (with $0 \le t_0 < T_0$) represents the phase of the pulse train in the window, our estimate for the relevant peak of the autocorrelation is

$$r_x\left(T_0\right) = \frac{\sum_n w\left(t_0 + nT_0\right)\, w\left(t_0 + (n+1)T_0\right)}{r_w\left(T_0\right) \sum_n w^2\left(t_0 + nT_0\right)} \tag{11}$$

This depends on the phase t_0/T. If the window is symmetric and the pulse train is symmetric around the middle of the window, the derivatives with respect to t_0 of both

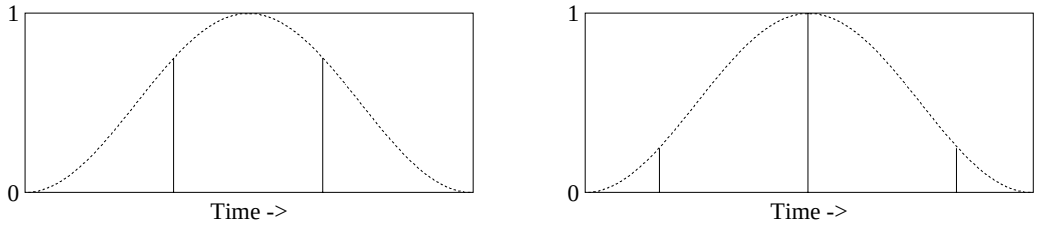

Fig. 2. Example of a windowed signal showing the two phases of a pulse train that yield extrema in the HNR estimation of the autocorrelation peak at a lag that equals the period.

the numerator and the denominator are 0; the extrema of $r_x(T_0)$ as a function of t_0 are thus found for the two phases exemplified in figure 2 for 3.0 periods per window. If such an extremum is greater than 1, it must be reflected through 1 to give a mathematically possible value of the autocorrelation, e.g., an initial estimate of 1.01 must be converted to 1/1.01 before computing its final HNR estimate, which is 20 dB.

Figure 3 shows the worst-case HNR values for a perfectly periodic pulse train, calculated with equation (11) for a Hanning window, and for the rectangular window

$$w(t) = 1 \quad ; \quad r_w(\tau) = 1 - \frac{|\tau|}{T} \tag{12}$$

and for the Welch window

$$w(t) = \sin\frac{\pi t}{T} \quad ; \quad r_w(\tau) = \left(1 - \frac{|\tau|}{T}\right)\cos\frac{\pi\tau}{T} + \frac{1}{\pi}\sin\frac{\pi|\tau|}{T} \tag{13}$$

as well as for the Hamming window

$$w(t) = 0.54 - 0.46\cos\frac{2\pi t}{T}$$
$$r_w(\tau) = \frac{\left(1 - \frac{|\tau|}{T}\right)\left(0.2916 + 0.1058\cos\frac{2\pi\tau}{T}\right) + 0.3910\frac{1}{2\pi}\sin\frac{2\pi|\tau|}{T}}{0.3974} \tag{14}$$

As we can see from figure 3, the Hanning window performs much better than the other three window shapes. Furthermore, the Hanning window is the ‘narrowest’ of the four window shapes, which makes it the least vulnerable of the four to rapidly changing sounds. That makes two reasons for forgetting about the other three.

In our implementation, the autocorrelations of the windowed signal and the window are numerically computed by Fast Fourier Transform. This is possible thanks to the fact that the autocorrelation can be obtained by first computing the Fourier transform of the windowed signal, which gives in the frequency domain

$$\tilde{a}(\omega) = \int a(t)\, e^{-i\omega t} dt \tag{15}$$

and then computing the inverse Fourier transform of the *power density* $|\tilde{a}(\omega)|^2$, which brings us to the lag domain

$$r_a(\tau) = \int |\tilde{a}(\omega)|^2 e^{i\omega\tau} \frac{d\omega}{2\pi} \tag{16}$$

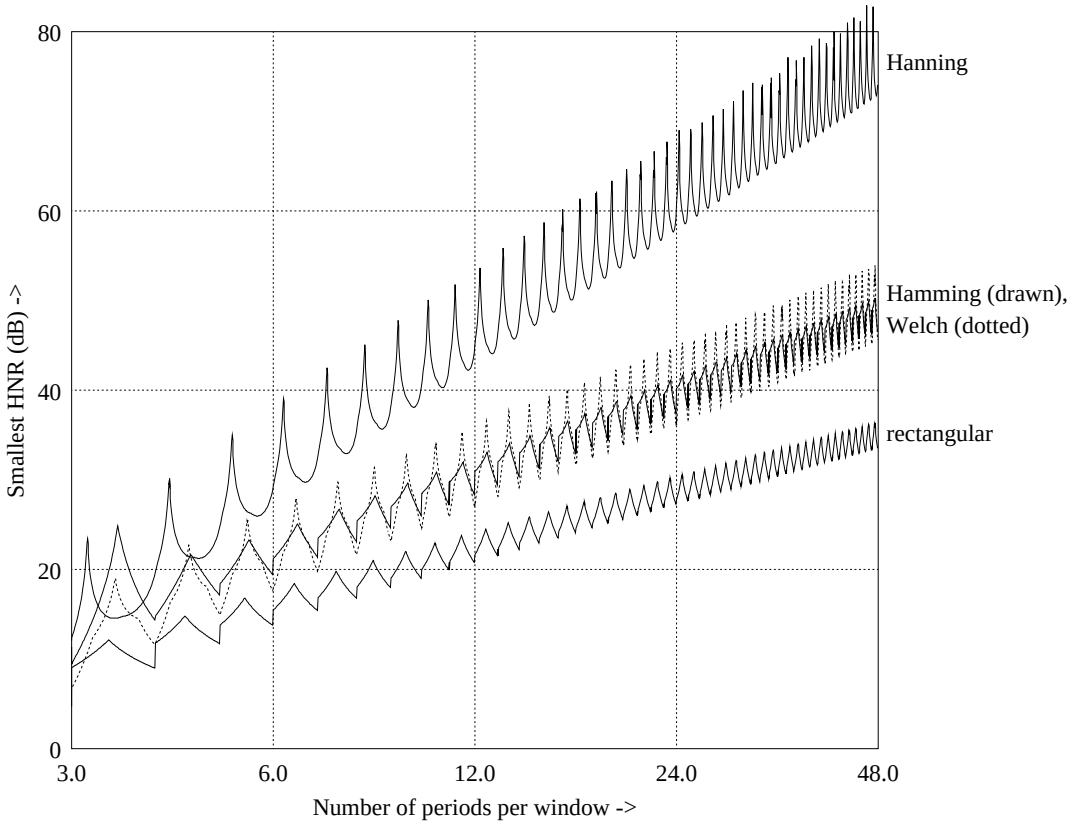

Fig. 3. The sensitivity of several window shapes to the phase of a pulse train. Every point on a curve represents the worse of the two HNR values belonging to the phases in fig. 2.

This procedure allows us to try two other functions in the lag domain, besides the autocorrelation. The first of these functions is what we will call the 'zero-phased' windowed signal, which is the sum of all the Fourier components of $a(t)$, reduced to cosines with a starting phase of 0. This function is obtained by computing in the frequency domain the *absolute value* $|\tilde{a}(\omega)|$ instead of the power density. This conserves the relative amplitudes of the components of the windowed signal, which is nice because it gives the formants the peaks that they deserve. The second function is known as the *cepstrum* (Noll, 1967) and is obtained by computing in the frequency domain the logarithm of the power:

$$\log\left(1 + c|\tilde{a}(\omega)|^2\right) \tag{17}$$

for large enough $c > 0$. The cepstral pitch-detection tactic was very common in the days that equation (9) was unknown, because it was the only one of the three methods that could raise the second peak of $r_a(\tau)$ (see figure 1) above the first peak. However, for both the zero-phased signal and the cepstrum, the addition of noise strongly suppresses all peaks relatively to the one at zero lag to a degree that depends on the frequency distribution of the noise. This makes these two methods unsuitable both for voiced-unvoiced decisions and for determination of the harmonics-to-noise ratio in the lag domain. Also, the pitch estimates are less accurate by several orders of magnitude as compared to the autocorrelation method. With equation (9) at our disposal, the advantages of these two alternative methods have vanished.

3 Sampling and the lag domain

Consider a continuous time signal $x(t)$ that contains no frequencies above a certain frequency f_{max}. We can sample this signal at regular intervals $\Delta t \le 1/(2 f_{max})$ so that we know only the values x_n at equally spaced times t_n:

$$x_n = x(t_n) \quad ; \quad t_n = t_0 + n\Delta t \tag{18}$$

We lose no data in this sampling, because we can reconstruct the original signal as

$$x(t) = \sum_{n=-\infty}^{+\infty} x_n \frac{\sin \pi(t - t_n)/\Delta t}{\pi(t - t_n)/\Delta t} \tag{19}$$

The autocorrelation computed from the sampled signal is also a sampled function:

$$r_n = r(n\Delta\tau)$$

There is a local maximum in the autocorrelation between $(m-1)\Delta\tau$ and $(m+1)\Delta\tau$ if

$$r_m > r_{m-1} \quad \text{and} \quad r_m > r_{m+1} \tag{20}$$

A first crude estimate of the pitch period would be $\tau_{max} \approx m\Delta\tau$, but this is not very accurate: with a sampling frequency of 10 kHz and $\Delta\tau = \Delta t$, the pitch resolution for fundamental frequencies near 300 Hz is 9 Hz (which is the case for most time-domain pitch-detection algorithms); moreover, the height of the autocorrelation peak (r_m) can be as low as $2/\pi = 0.636$ for correctly sampled pulse trains (i.e., filtered with a phase-preserving low-pass filter at the Nyquist frequency prior to sampling), which renders HNR determination impossible and introduces octave errors in the determination of the fundamental period. We can improve this by parabolic interpolation around $m\Delta\tau$:

$$\tau_{max} \approx \Delta\tau\left(m + \frac{\frac{1}{2}(r_{m+1} - r_{m-1})}{2r_m - r_{m-1} - r_{m+1}}\right) \quad ; \quad r_{max} \approx r_m + \frac{(r_{m+1} - r_{m-1})^2}{8(2r_m - r_{m-1} - r_{m+1})} \tag{21}$$

However, though the error in the estimated period reduces to less than 0.1 sample, the height of the relevant autocorrelation peak can still be as low as $7/(3\pi) = 0.743$.

Now for the solution. We should use a 'sin x / x' interpolation, like the one in equation (19), in the lag domain (we do a simple upsampling in the frequency domain, so that $\Delta\tau = \Delta t/2$). As we cannot do the infinite sum, we interpolate over a finite number of samples N to the left and to the right, using a Hanning window again to taper the interpolation to zero at the edges:

$$\begin{aligned} r(\tau) \approx & \sum_{n=1}^{N} r_{n_r - n} \frac{\sin \pi(\varphi_l + n - 1)}{\pi(\varphi_l + n - 1)}\left(\tfrac{1}{2} + \tfrac{1}{2}\cos\frac{\pi(\varphi_l + n - 1)}{\varphi_l + N}\right) + \\ & \sum_{n=1}^{N} r_{n_l + n} \frac{\sin \pi(\varphi_r + n - 1)}{\pi(\varphi_r + n - 1)}\left(\tfrac{1}{2} + \tfrac{1}{2}\cos\frac{\pi(\varphi_r + n - 1)}{\varphi_r + N}\right) \end{aligned} \tag{22}$$

where $n_l \equiv$ largest integer $\le \dfrac{\tau}{\Delta\tau}$; $n_r \equiv n_l + 1$; $\varphi_l \equiv \dfrac{\tau}{\Delta\tau} - n_l$; $\varphi_r \equiv 1 - \varphi_l$

In our implementation, N is the smaller of 500 and the largest number for which $(n_l + N)\Delta\tau$ is smaller than half the window length. This is because the estimation of the autocorrelation is not reliable for lags greater than half the window length, if there are few periods per window (see figure 1). Note that the interpolation can involve autocorrelation values for negative lags.

The places and heights of the maxima of equation (22) can be determined with great precision (they are looked for between $(m-1)\Delta\tau$ and $(m+1)\Delta\tau$). We can show this with long windows, where the windowing effects have gone, but the sampling effects remain. E.g., with a 40-ms window, any signal with a frequency of exactly 3777 Hz, sampled at 10 kHz, will be consistently measured as having a fundamental frequency of 3777.00000 ± 0.00001 Hz (accuracy 10^{-8} sample in the lag domain, N=394) and a first autocorrelation peak between 0.99999999 and 1. The measured HNR (80-ms window) is 94.0 ± 0.1 dB. This looks like a real improvement.

4 Algorithm

A summary of the complete 9-parameter algorithm, as it is implemented into the speech analysis and synthesis program *praat*, is given here:

Step 1. Preprocessing: to remove the sidelobe of the Fourier transform of the Hanning window for signal components near the Nyquist frequency, we perform a soft upsampling as follows: do an FFT on the whole signal; filter by multiplication in the frequency domain linearly to zero from 95% of the Nyquist frequency to 100% of the Nyquist frequency; do an inverse FFT of order one higher than the first FFT.

Step 2. Compute the global absolute peak value of the signal (see step 3.3).

Step 3. Because our method is a short-term analysis method, the analysis is performed for a number of small segments (*frames*) that are taken from the signal in steps given by the *TimeStep* parameter (default is 0.01 seconds). For every frame, we look for at most *MaximumNumberOfCandidatesPerFrame* (default is 4) lag-height pairs that are good candidates for the periodicity of this frame. This number includes the *unvoiced* candidate, which is always present. The following steps are taken for each frame:

Step 3.1. Take a segment from the signal. The length of this segment (the window length) is determined by the *MinimumPitch* parameter, which stands for the lowest fundamental frequency that you want to detect. The window should be just long enough to contain three periods (for pitch detection) or six periods (for HNR measurements) of *MinimumPitch*. E.g. if *MinimumPitch* is 75 Hz, the window length is 40 ms for pitch detection and 80 ms for HNR measurements.

Step 3.2. Subtract the local average.

Step 3.3. The first candidate is the unvoiced candidate, which is always present. The strength of this candidate is computed with two soft threshold parameters. E.g., if *VoicingThreshold* is 0.4 and *SilenceThreshold* is 0.05, this frame bears a good chance of being analyzed as voiceless (in step 4) if there are no autocorrelation peaks above approximately 0.4 or if the local absolute peak value is less than approximately 0.05 times the global absolute peak value, which was computed in step 2.

Step 3.4. Multiply by the window function (equation 5).

Step 3.5. Append half a window length of zeroes (because we need autocorrelation values up to half a window length for interpolation).

Step 3.6. Append zeroes until the number of samples is a power of two.

Step 3.7. Perform a Fast Fourier Transform (discrete version of equation 15), e.g., with the algorithm `realft` from Press et al. (1989).

Step 3.8. Square the samples in the frequency domain.

Step 3.9. Perform a Fast Fourier Transform (discrete version of equation 16). This gives a sampled version of $r_a(\tau)$.

Step 3.10. Divide by the autocorrelation of the window, which was computed once with steps 3.5 through 3.9 (equation 9). This gives a sampled version of $r_x(\tau)$.

Step 3.11. Find the places and heights of the maxima of the continuous version of $r_x(\tau)$, which is given by equation 22, e.g., with the algorithm `brent` from Press et al. (1989). The only places considered for the maxima are those that yield a pitch between *MinimumPitch* and *MaximumPitch*. The *MaximumPitch* parameter should be between *MinimumPitch* and the Nyquist frequency. The only candidates that are remembered, are the unvoiced candidate, which has a *local strength* equal to

$$R \equiv VoicingThreshold + \max\left(0,\ 2 - \frac{(local\ absolute\ peak)/(global\ absolute\ peak)}{SilenceThreshold/(1 + VoicingThreshold)}\right) \quad (23)$$

and the voiced candidates with the highest (*MaximumNumberOfCandidatesPerFrame* minus 1) values of the local strength

$$R \equiv r(\tau_{max}) - OctaveCost \cdot {}^2\log(MinimumPitch \cdot \tau_{max}) \quad (24)$$

The *OctaveCost* parameter favours higher fundamental frequencies. One of the reasons for the existence of this parameter is that for a perfectly periodic signal all the peaks are equally high and we should choose the one with the lowest lag. Other reasons for this parameter are unwanted local downward octave jumps caused by additive noise (section 6). Finally, an important use of this parameter lies in the difference between the acoustic fundamental frequency and the perceived pitch. For instance, the harmonically amplitude-modulated signal with modulation depth d_{mod}

$$x(t) = (1 + d_{mod} \sin 2\pi Ft) \sin 4\pi Ft \quad (25)$$

has an acoustic fundamental frequency of F, whereas its perceived pitch is $2F$ for modulation depths smaller than 20 or 30 percent. Figure 1 shows such a signal, with a modulation depth of 30%. If we want the algorithm's criterion to be at 20% (in order to fit pitch perception), we should set the *OctaveCost* parameter to $(0.2)^2 = 0.04$; if we want it to be low (in order to detect vocal-fold periodicity), say 5%, we should set it to $(0.05)^2 = 0.0025$. The default value is 0.01, corresponding to a criterion of 10%.

After performing step 2 for every frame, we are left with a number of frequency-strength pairs (F_{ni}, R_{ni}), where the index n runs from 1 to the number of frames, and i is between 1 and the number of candidates in each frame. The *locally* best candidate in each frame is the one with the highest R. But as we can have several approximately equally strong candidates in any frame, we can launch on these pairs the *global path finder*, the aim of which is to minimize the number of incidental voiced-unvoiced decisions and large frequency jumps:

Step 4. For every frame n, p_n is a number between 1 and the number of candidates for that frame. The values $\{p_n \mid 1 \le n \le \text{number of frames}\}$ define a *path* through the candidates: $\{(F_{np_n}, R_{np_n}) \mid 1 \le n \le \text{number of frames}\}$. With every possible path we associate a *cost*

$$cost(\{p_n\}) = \sum_{n=2}^{numberOfFrames} transitionCost(F_{n-1,p_{n-1}}, F_{np_n}) - \sum_{n=1}^{numberOfFrames} R_{np_n} \quad (26)$$

where the *transitionCost* function is defined by (F = 0 means unvoiced)

$$transitionCost(F_1, F_2) = \begin{cases} 0 & \text{if } F_1 = 0 \text{ and } F_2 = 0 \\ VoicedUnvoicedCost & \text{if } F_1 = 0 \text{ xor } F_2 = 0 \\ OctaveJumpCost \cdot \left| {}^2\log \frac{F_1}{F_2} \right| & \text{if } F_1 \neq 0 \text{ and } F_2 \neq 0 \end{cases} \quad (27)$$

where the *VoicedUnvoicedCost* and *OctaveJumpCost* parameters could both be 0.2. The globally best path is the path with the lowest cost. This path might contain some candidates that are locally second-choice. We can find the cheapest path with the aid of dynamic programming, e.g., using the Viterbi algorithm described for Hidden Markov Models by Van Alphen & Van Bergem (1989).

For stationary signals, the global path finder can easily remove all local octave errors, even if they comprise as many as 40% of all the locally best candidates (section 6 presents an example). This is because the correct candidates will be almost as strong as the incorrectly chosen candidates. For most dynamically changing signals, the global path finder can still cope easily with 10% local octave errors.

For many measurements in this article, we turn the path finder off by setting the *VoicedUnvoicedCost* and *OctaveJumpCost* parameters to zero; in this way, the algorithm selects the locally best candidate for each frame.

For HNR measurements, the path finder is turned off, and the *OctaveCost* and *VoicingThreshold* parameters are zero, too; *MaximumPitch* equals the Nyquist frequency; only the *TimeStep*, *MinimumPitch*, and *SilenceThreshold* parameters are relevant for HNR measurements.

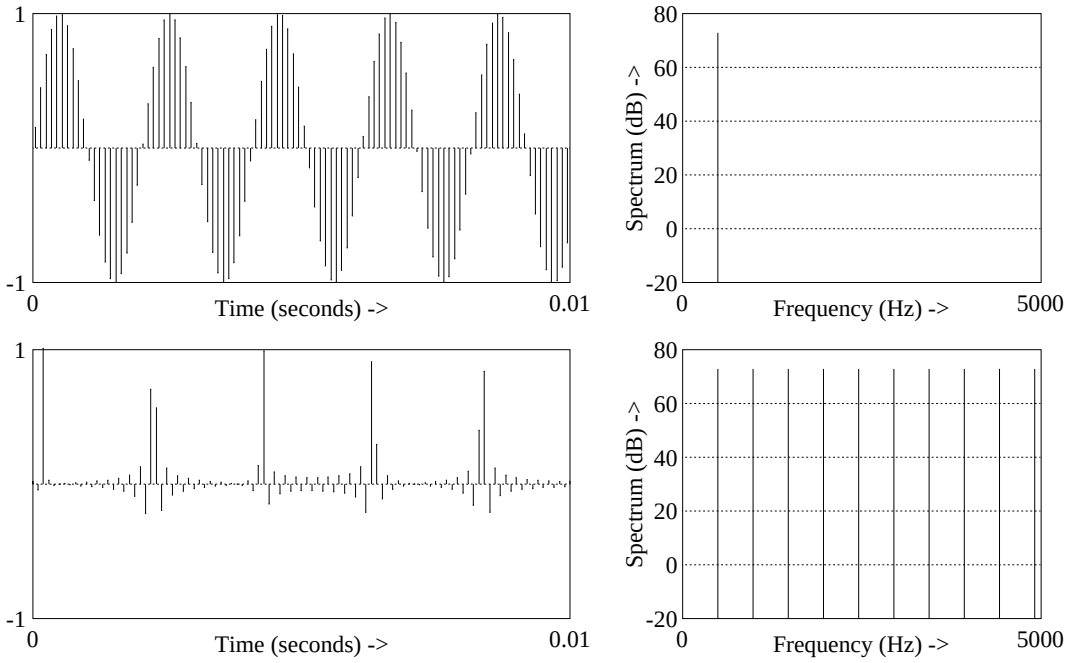

Fig. 4. At the left: two periodic signals, sampled at 10 kHz: a sine wave and a pulse train, which was squarely low-pass filtered at 5000 Hz (acausal, phase-preserving filter). Both have a fundamental frequency of 490 Hz. At the right: their spectra.

5 Accuracy in measuring perfectly periodic signals

The formula for a sampled perfect sine wave with frequency F is

$$x_n = \sin 2\pi F t_n \tag{28}$$

and the formula for a correctly sampled pulse train (squarely low-pass filtered at the Nyquist frequency) with period T is

$$x_n = \sum_{m=-\infty}^{+\infty} \frac{\sin \pi F_s (t_n - mT)}{\pi F_s (t_n - mT)} \tag{29}$$

These two functions form spectrally maximally different periodic signals. Figure 4 shows examples of these signals, together with their spectra. The spectrum of the sine wave is maximally narrow, that of the pulse train is maximally wide.

Table 1 (page 97) shows our algorithm's accuracy in determining pitch and HNR. We see from table 1 that for pitch detection there should be at least three periods in a window. The value of 27 dB appearing in table 1 for a sine wave with the worst phase (symmetric in window) and the worst period (one third of a window), means that the autocorrelation peak can be as low as 0.995, which means that the signal

$$x_n = (1 + d_{mod} \sin 2\pi F t_n) \sin 4\pi F t_n \tag{30}$$

(see also figure 1), whose fundamental frequency is F, can be locally ambiguous for F near *MinimumPitch*, if the modulation depth d_{mod} is less than $\sqrt{1–0.995}$ = 7%. The critical modulation depth, at which there are 10% local octave errors (detection of $2F$ as the best candidate), is 5%, for the lowest F (equal to *MinimumPitch*). Note that the global path finder will not have any trouble removing these octave errors.

We also see from table 1 that for HNR measurements, there should be at least 6.0 periods per window. The values measured for the HNR of a pulse train are the same as those predicted by theory for continuous signals, as plotted in figure 3. This suggests that the windowing effects have the larger part of the influence on HNR measurement inaccuracy, and that the sampling effects have been effectively cancelled by equation (22).

For very short windows (less than 20 samples in the time domain, *MinimumPitch* greater than 30% of the Nyquist frequency), the HNR values for pulse trains do not deteriorate, but those for sine waves approach the values for pulse trains; the relative pitch determination error rises to 10^{-4}.

The problems with short-term HNR measurements in the frequency domain, are the sidelobes of the harmonics and the sidelobes of the Fourier transform of the window: they occur throughout the spectrum. Pitch-synchronous algorithms try to cope with the first problem, but they require prior accurate knowledge of the period (Cox et al., 1989; Yumoto et al., 1982). Using fixed window lengths in the frequency domain requires windows to be long: the shortest window used by Klingholz (1987) spans 12 periods, De Krom (1993) needs 8.2 periods; with the shortest window, both have a HNR resolution of apx. 30 dB for synthetic vowels, as opposed to our 37 dB with only 6 periods (48 dB for 8.2 periods, 52 dB for 12 periods). In the autocorrelation domain, the only sidelobe that could stir trouble, is the one that causes aliasing for frequency components near the Nyquist frequency; that one is easily filtered out. This is the cause of the superior results with the present method.

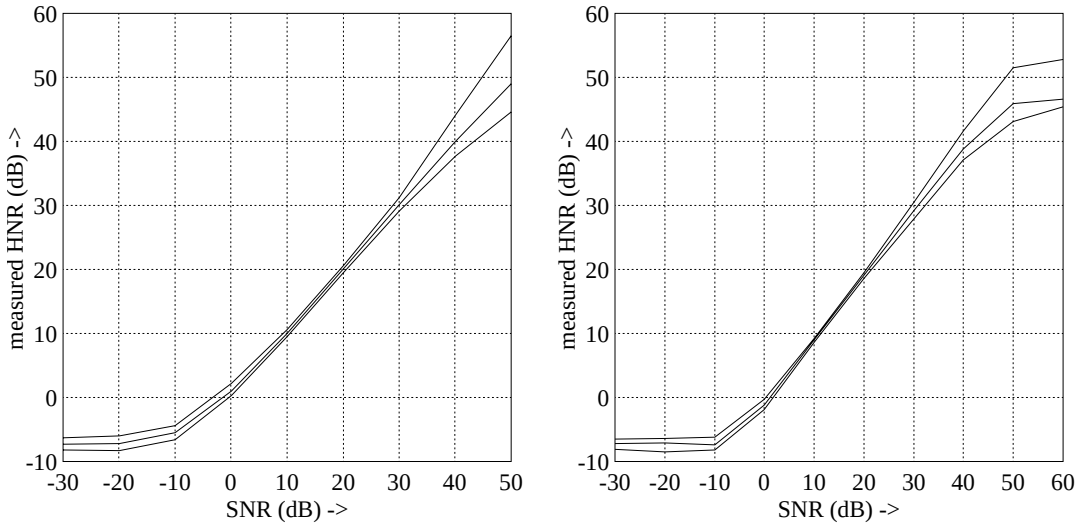

Fig. 5. Measured HNR values for a sine wave (left) and a pulse train (right) sampled at 10 kHz, both with a periodicity of 103 Hz, with additive noise. The figures show the 10%, median, and 90% curves. The window length was 80 ms.

6 Sensitivity to additive noise

The formula for a sampled sound consisting of a sine wave with frequency F and additive 'white' noise (squarely low-pass filtered at the Nyquist frequency) is

$$x_n = \sqrt{2}\,\sin 2\pi F t_n + 10^{-SNR/20}\,\mathbf{z}_n \tag{31}$$

where *SNR* is the signal-to-noise ratio, expressed in dB, and $\mathbf{z}_n$ is a sequence of real numbers that are independently drawn from a Gaussian distribution with zero mean and unit variance. The formula for a sampled sound consisting of a correctly sampled pulse train (squarely low-pass filtered at the Nyquist frequency) with period T and additive 'white' noise is

$$x_n = \sqrt{\frac{F_s / F}{1 - F / F_s}} \sum_{m=-\infty}^{+\infty} \frac{\sin \pi F_s (t_n - mT)}{\pi F_s (t_n - mT)} + 10^{-SNR/20}\,\mathbf{z}_n \tag{32}$$

Adding noise obscures the underlying fundamental frequency. For example, additive noise with a SNR of 20 dB gives the following results for sounds with an underlying F_0 of 103 Hz, sampled at 10 kHz, and analyzed with a *MinimumPitch* of 75 Hz (40-ms window): the relative pitch 'error' (measured as the worse of the 10% and 90% points of the distribution of the measured pitch) rises to 0.7% for a sine wave, and to 0.007% for a pulse train (these 'errors' are not failures of the algorithm: they are signal properties). Gross pitch determination 'errors' (more than 10% off) are only found for negative signal-to-noise ratios (more noise than signal).

For a sine wave with a frequency of 206 Hz and a window length of 40 ms (*MinimumPitch* is 75 Hz), with noise added at a SNR of 20 dB, there are 40% local octave 'errors' (a detected pitch of 103 Hz; these are not failures of the algorithm, either: the 103 Hz is locally in the signal) if the *OctaveCost* parameter is 0.001. The global path finder leaves 0% octave 'errors'. However, we cannot expect this good behaviour for dynamically changing signals. There remain 10% local octave 'errors' if the *OctaveCost* parameter is raised to 0.003. This gives a critical modulation depth of $\sqrt{0.003}$ = 5%. Thus, if we want to detect reliably the pitch of noisy signals, we

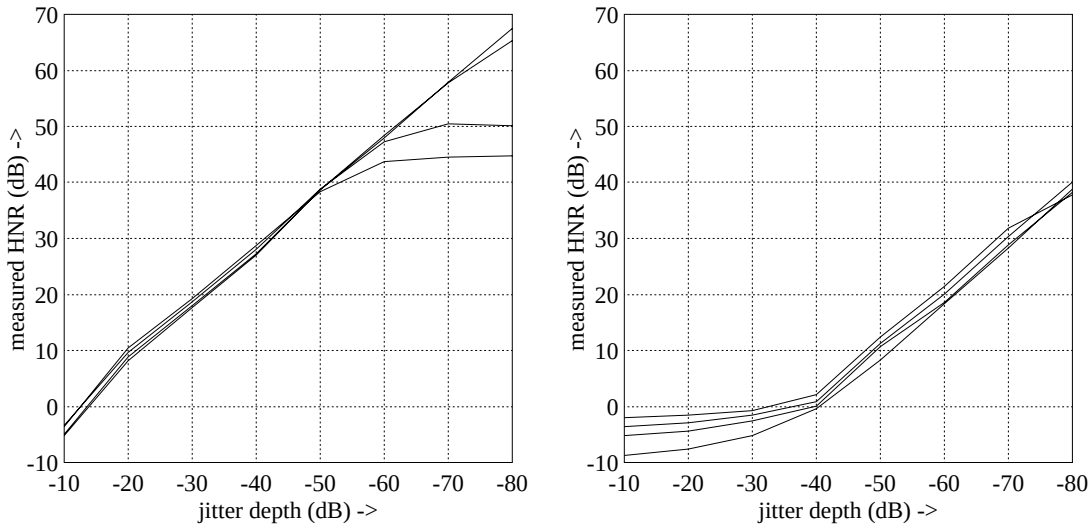

Fig. 6. Measured HNR values for a sine wave (left) and a pulse train (right), sampled at 10 kHz, with fundamental frequencies that vary randomly around 103 Hz. The curves are shown for window lengths of 60, 80, 150, and 600 milliseconds. The short windows give slightly larger HNR values than the long windows, except where the HNR measurements for short windows level off for very low jitter depths.

should not expect to see the difference between the fundamental frequency and a first formant whose relative amplitude is higher than 95% (we must note here that the zero-phased signal would raise this number to 97%).

We see from figure 5 that the measured HNR values are within a few dB from the underlying SNR values (between 0 and 40 dB). These results are better than those found in the literature so far. For instance, in De Krom (1993), the measured HNR values for additive noise depend to a large degree on the number of periods per window: for a SNR of 40 dB (in the glottal source, before a linear filter), the averaged HNR varies from 27 dB for 8.2 periods per window (102.4 ms / 80 Hz) to 46 dB for 121 periods per window (409.6 ms / 296 Hz), and the slope of HNR as a function of SNR is 0.7. With our algorithm, the median HNR, for a SNR of 40 dB, varies from 39.0 dB for 8.2 periods per window (103 Hz / 80 ms) to 40.0 dB for large windows, and the slope is near to the theoretical value of 1.

7 Sensitivity to random frequency modulations (jitter)

A jittered pulse train with average period T_{av} has its events at the times

$$T_n = T_{n-1} + T_{av}\left(1 + jitterDepth \cdot \mathbf{z}_n\right) \tag{33}$$

A jittered sine wave with average frequency F_{av} involves a randomly walking phase:

$$x_n = \sin\left(2\pi F_{av} n\Delta t + \varphi_n\right) \quad ; \quad \varphi_n = \varphi_{n-1} + 2\pi \cdot jitterDepth \cdot \sqrt{F_{av}\Delta t} \cdot \mathbf{z}_n \tag{34}$$

For sine waves, the harmonics-to-noise ratio is much less sensitive to *jitterDepth* than for pulse trains. This is shown in figure 6 (jitter depth in dB is $20 \cdot {}^{10}\log jitterDepth$). The slope of the HNR as a function of the logarithmic jitter depth is apx. –0.95, which is closer to the theoretical value of 1 than De Krom's (1993) value of –0.66.

8 Conclusion

Our measurements of the places and the heights of the peaks in the lag domain are several orders of magnitude more accurate than those of the usual pitch-detection algorithms. Due to its complete lack of local decision moments, the algorithm is very straightforward, flexible and robust: it works equally well for low pitches (the author's creaky voice at 16 Hz, alveolar trill at 23.4 Hz, and bilabial trill at 26.0 Hz), middle pitches (female speaker at 200 Hz), and high pitches (soprano at 1200 Hz, a two-year-old child yelling [i] at 1800 Hz). The only 'new' tricks are two mathematically justified tactics: the division by the autocorrelation of the window (equation 9), and the '$\sin x / x$' interpolation in the lag domain (equation 22).

In measuring harmonics-to-noise ratios, the present algorithm is not only much simpler, but also much more accurate, more reproducible, less dependent on period and window length, and more resistant to rapidly changing sounds, compared to other algorithms found in the literature.

Postscript

After finishing this article, we discovered that the 'Gaussian' window, which is zero outside the interval $\left[-\frac{1}{2}T, \frac{3}{2}T\right]$, and $(\exp(-12(t/T - \frac{1}{2})^2) - e^{-12}) / (1 - e^{-12})$ inside, produces much better results than the Hanning window defined on [0,T], though its effective length is approximately the same. The worst pitch determination error (table 1) falls from $5 \cdot 10^{-4}$ to 10^{-6}; the worst measurable HNR for a pulse train (at 6 effective periods per window) rises from 29 dB to 58 dB, and for a sine wave it rises from 40 to 75 dB. In figure 3, the theoretical value for a *real* Gaussian window at 6 effective periods per window would be 58 dB (30 dB at 4.5 periods per window, and 170 dB at 10 periods per window). In figures 5 and 6, the curves would not level off at 45 dB, but at 60 and 80 dB instead. In practice, a window only 4.5 periods long guarantees a minimum HNR of 37 dB for vowel-like periodic signals (65 dB for 6 periods). This means that we can improve again on the analysis of running speech.

References

Alphen, P. van, & Bergem, D.R. van (1989): "Markov models and their application in speech recognition", *Proceedings of the Institute of Phonetic Sciences of the University of Amsterdam* **13**: 1-26.

Cox, N.B., Ito, N.R., & Morrison, M.D. (1989): "Technical considerations in computation of spectral harmonics-to-noise ratios for sustained vowels", *J. Speech and Hearing Research* **32**: 203-218.

Hess, W.J. (1992): "Pitch and voicing determination", in S. Furui & M.M. Sondhi (eds.): *Advances in Speech Signal Processing*, Marcel Dekker, New York, pp. 3-48.

Klingholz, F. (1987): "The measurement of the signal-to-noise ratio (SNR) in continuous speech", *Speech Communication* **6**: 15-26.

Krom, G. de (1993): "A cepstrum-based technique for determining a harmonics-to-noise ratio in speech signals", *Journal of Speech and Hearing Research* **36**: 254-265.

Noll, A.M. (1967): "Cepstrum pitch determination", *J. Acoust. Soc. America* **41**: 293-309.

Press, W.H., Flannery, B.P., Teukolsky, S.A., & Vetterling, W.T. (1989): *Numerical Recipes*, Cambridge University Press.

Rabiner, L.R. (1977): "On the use of autocorrelation analysis for pitch detection", *IEEE Transactions on Acoustics, Speech, and Signal Processing ASSP*-**25**: 24-33.

Yumoto, E., Gould, W.J., & Baer, T. (1982): "Harmonics-to-noise ratio as an index of the degree of hoarseness", *Journal of the Acoustical Society of America* **71**: 1544-1550.