# DIRECT ESTIMATION OF FREQUENCY FROM MDCT-ENCODED FILES

*S. Merdjani and L. Daudet*

Laboratoire d'Acoustique Musicale - Université Paris 6
11 rue de Lourmel 75015 Paris - France
daudet@lam.jussieu.fr

## ABSTRACT

The Modified Discrete Cosine Transform (MDCT) is a broadly-used transform for audio coding, since it allows an orthogonal time-frequency transform without blocking effects. In this article, we show that the MDCT can also be used as an analysis tool. This is illustrated by extracting the frequency of a pure sine wave with some simple combinations of MDCT coefficients. We studied the performance of this estimation in ideal (noiseless) conditions, as well as the influence of additive noise (white noise / quantization noise). This forms the basis of a low-level feature extraction directly in the compressed domain.

## 1. INTRODUCTION

For indexation purposes, it is necessary to retrieve low-level features about the signal, such as some information about the frequency and amplitude of the tonal components in the signal. A number of techniques have been proposed [1], usually based on the Short Time Fourier Transform of the signal [2].

It is interesting to note that nowadays, an increasing number of files in sound databases are stored in some compressed form, for instance using MPEG-1 layer III ("MP3") or MPEG-2 AAC coding standards. It would be desirable to have some analysis procedure that act directly on the compressed file, rather than performing a traditional analysis on the signal after decompression. Indeed, our work shows that this is theoretically possible, since the encoding process already uses some kind of time-frequency transform (see figure (1).

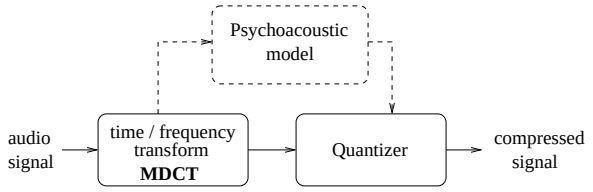


Figure 1: Block diagram of subband audio codecs.

In this paper, we give a proof of concept for a specific class of time-frequency transform, the Modified Discrete Cosine Transform (MDCT). This tool is actually employed in the majority of state-of-the art audio coders, such as MPEG-1 layer 3 ("MP3"), MPEG-2 AAC and Windows Media Audio. In a previous paper [3], we have shown that it is possible to compute explicitly the MDCT of a pure sinusoid. Here, we show that the inverse problem can also be performed explicitly, at least in the ideal noiseless case: from the set of MDCT coefficients it is possible to retrieve the frequency, amplitude and phase of the encoded sine wave (figure (2).

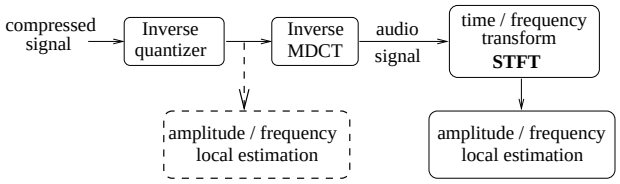


Figure 2: The standard method for extracting amplitude / frequency information is to uncompress the file, and process it through a STFT-based analyzer. The proposed method performs the analysis directly in the (quantized) MDCT domain.

The paper is constructed as follows. In section 2, we derive the forward and inverse problems for the analysis of pure sines with the MDCT. We then compare, in section 3, the frequency estimates with a number of standard methods, in the ideal noiseless case as well as in more realistic situations, such as the presence of noise. We specifically investigate these performances in the presence of quantization noise, which is present in the signal after a lossy compression scheme. Finally, we conclude of possible extensions of these techniques for a low-complexity analysis algorithm in the transform domain, that can be used for indexation purposes.

## 2. THEORETICAL CONCEPTS

### 2.1. MDCT analysis of pure tones

MDCT basis functions are defined as:

$$g_{p,k}[n] = g_p[n]\sqrt{\frac{2}{L}}\cos\left[\frac{\pi}{L}\left(k+\frac{1}{2}\right)\left(n+\frac{1}{2}\right)\right] \qquad (1)$$

where the windows $g_p[n] = \sin\left[\frac{\pi}{2L}\left(n - pL + \frac{L+1}{2}\right)\right]$, for $n = -\frac{L}{2}\ldots\left(\frac{3L}{2}-1\right)$ and $g_p[n] = 0$ elsewhere, are translated sinusoidal windows (half-length $L$). MDCT coefficients of a function $x$ are computed as $d_{p,k} = \langle x, g_{p,k}\rangle$, with the canonical scalar product in $\ell^2(\mathbb{Z})$.

Let us consider the MDCT of a pure sine wave : $x[n] = A\sin((f\pi/L)n + \phi)$, with $0 \leq f < L$. Let us denote $k_0 = \lfloor f \rfloor$ and $\varepsilon = f - k_0$ the integer and fractional parts of the normalized frequency $f$, respectively.

If we neglect aliasing terms from negative frequencies, i.e. far from the edges $f = 0$ or $f = L$, these coefficients can be explicitly

computed [3] :

$$d_{k,p} \simeq -\frac{\sqrt{2L}}{2\pi} \frac{A \sin(\pi f)}{(f-k)(f-k-1)} \cos\left[\frac{\pi}{2}(k-k_0)+\psi\right] \quad (2)$$

where $\psi = \frac{\pi}{2}k_0 + \frac{\pi(L-1)}{2L}f + \phi - \frac{\pi}{4}$.

Hence, we see that the MDCT of a pure sine is a combination of two terms: a 4-periodic cosine modulation and an amplitude term related to the Fourier transform of the window.

### 2.2. Inverse problem

Let us now focus on the inverse problem: from a set of MDCT coefficients of an unknown sine wave, how can we estimate the frequency, amplitude and phase of this wave ?

**Identification of $k_0$**

Finding the frequency bin $k_0$ corresponding to the integer part of the frequency is not as straightforward as in the Short-Time Fourier Transform (STFT) case, since the MDCT coefficients have a strong time (i.e. phase) dependency through the 4-periodic cosine modulation. Indeed, in a given window, the coefficient with maximum amplitude will not always be in bin $k_0$ but, depending on the phase, can be in bins $k_0 - 1$ or $k_0 + 1$.

Here, we will use a "regularized" version of the set of MDCT coefficients, constructed as follows:

$$\mathcal{S}_{k,p} = \left(d_{k,p}^2 + (d_{k+1,p} - d_{k-1,p})^2\right)^{\frac{1}{2}} \quad (3)$$

for $k = 0 \ldots (L-1)$.

This so-called "$\mathcal{S}$-spectrum" has the interesting property of being maximum, in a given window $p$, at the frequency bin $k_0$, regardless of the phase:

$$\forall \phi, \forall f \quad \max_k \mathcal{S}_k = \mathcal{S}_{k_0} \qquad \forall k = 0 \ldots (N-1)$$

This result is exact [3] if we neglect the aliasing terms as in eqn. (2), and numerical simulations showed that it still holds with the actual MDCT coefficients for $k_0 \in [3, 1021]$ with a typical window half-size $L = 1024$, i.e. (at 44.1 kHz sampling rate) in the frequency range 43 Hz – 22 kHz.

It is also worthwhile to mention that this regularized $\mathcal{S}$-spectrum is a good approximation of the FFT-spectrum [3].

**Identification of $\varepsilon$**

Let us define $\alpha$ as :

$$\alpha = -\frac{d_{k_0-1}}{d_{k_0+1}} \quad (4)$$

As $0 \leq \epsilon < 1$, it appears that

$$\alpha = \frac{(\epsilon-1)(\epsilon-2)}{\epsilon(\epsilon+1)} \geq 0 \quad (5)$$

Solving eq. 5 for $\epsilon$ as a function of $\alpha$ is a second-order equation that leads to:

$$\epsilon = \frac{3+\alpha-\sqrt{\alpha^2+14\alpha+1}}{2(1-\alpha)} \qquad \text{for } \alpha \neq 1 \quad (6)$$

which can be extended to $\alpha = 1 \leftrightarrow \epsilon = 1/2$.

In some cases, depending on the phase, the coefficients $d_{k_0+1}$ and $d_{k_0-1}$ are both very small, and therefore the computation of $\alpha$ leads to spurious results. In such cases it may be preferable to use the the coefficients $d_{k_0+2}$ and $d_{k_0-2}$ instead (remember that if $d_{k_0+1}$ and $d_{k_0-1}$ are small, then $d_{k_0+2}$ and $d_{k_0-2}$ are large, and vice-versa) :

$$\beta = \frac{d_{k_0-2}}{d_{k_0+2}} \quad (7)$$

As $0 \leq \epsilon < 1$, it appears that

$$\beta = \frac{(\epsilon-3)(\epsilon-2)}{(\epsilon+1)(\epsilon+2)} \geq 0 \quad (8)$$

and again $\epsilon$ can be deduced from $\beta$ as:

$$\epsilon = \frac{5+3\beta-\sqrt{\beta^2+62\beta+1}}{2(1-\beta)} \qquad \text{for } \beta \neq 1 \quad (9)$$

which can be extended to $\beta = 1 \leftrightarrow \epsilon = 1/2$.

Decision to use either formula 6 or 9 is based on the value of the ratio $\lambda = |d_{k_0}|/\mathcal{S}_{k_0}$. One always has $0 \leq \lambda \leq 1$. If $\lambda$ is very close to 1 then the main contribution to $\mathcal{S}_{k_0}$ comes from $d_{k_0}$ and very little from $d_{k_0 \pm 1}$, therefore equation 9 should be used; otherwise $d_{k_0 \pm 1}$ are not small and formula 6 should be used. Numerical simulations have shown that the critical value $\lambda_0 = .9685$ is close to optimality.

Finally, the procedure for the estimation of $\epsilon$ is as follows:

$$\begin{cases} \text{if } \lambda < \lambda_0 \text{ then} & \text{compute } \alpha \text{ from eq. 4} \\ & \text{and deduce } \epsilon \text{ from eq. 6} \\ \text{otherwise} & \text{compute } \beta \text{ from eq. 7} \\ & \text{and deduce } \epsilon \text{ from eq. 9} \end{cases} \quad (10)$$

**Extraction of the phase $\phi$ and amplitude $A$**

The modified phase $\psi$ is estimated from :

$$\frac{d_{k_0-1}}{d_{k_0}} = \frac{\epsilon+1}{\epsilon-1} \tan\psi \quad (11)$$

The phase $\phi$ of the sinusoid is easily deduced from the definition of $\psi$, given $k_0$ and $\epsilon$.

Similarly, one can deduce the amplitude $A$ from

$$(\epsilon-1)^2\left[\epsilon^2 {d_{k_0}}^2 + (\epsilon-2)^2 {d_{k_0+1}}^2\right] = A^2 \frac{L}{2\pi^2} \sin^2 \pi\epsilon \quad (12)$$

## 3. RESULTS

### 3.1. Noiseless case

In the ideal noiseless case, we have computed estimations of the fractional frequency $\epsilon$ using the procedure (10), and the amplitude $A$ using equ. (12). We have taken 1024 sine waves with integer fundamental frequency $k_0$ ranging from 1 to 1024 and a random fractional frequency$\epsilon$, and we have proceeded the estimation in 126 consecutive windows. The maximum error, for all windows, as a function of the frequency, is plotted on figure 3.

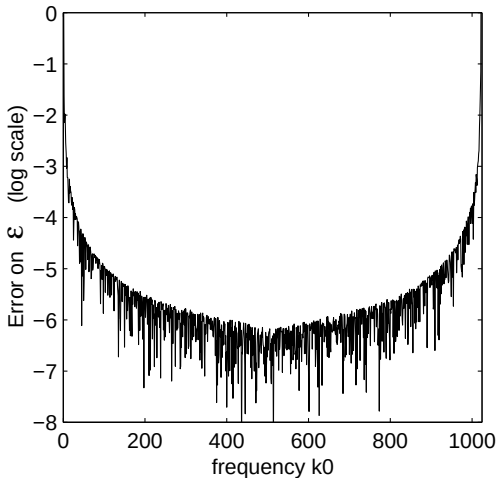


Figure 3: *Value of the maximum (across 126 consecutive windows) of the error in the estimation of the fractional frequency $\epsilon$.*

For a large range of frequencies, the error in the frequency estimate compares favorably with previous techniques. In [4], results based of the Odd-DFT, a complex version of the MDCT, are reported with an accuracy of 1 % of the bin size). Here, results are better in the range $k_0 \in [5 - 1008]$, and much better in most cases (up to $10^{-6}$ for average frequencies). For frequencies $k_0$ too close to 0 or the $L$, it is not possible to neglect the aliasing terms anymore, and the approximation (2) is not valid.

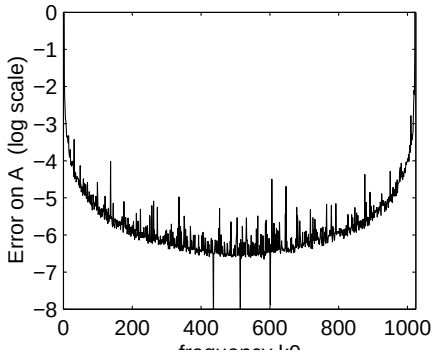


Figure 4: *Value of the maximum (across 126 consecutive windows) of the error in the estimation of the amplitude $A$.*

We have also investigated the estimation of the amplitude $A$ of the analyzed sine wave. Corresponding plots are shown on figure 4. Notice the large errors peaks due to a bad estimation of $\epsilon$ upon which the estimation of $A$ is made (according to eqn. 12).

As a basis for comparison with other methods based on the Short-Time Fourier Transform (STFT), we have compared our results with the results presented in [2]. The six methods are : (1) plain FFT (2) parabolic interpolation (3) triangle algorithm (4) spectral reassignment (5) derivative algorithm (6) phase vocoder (based on [5]). Note that all six methods are based on the FFT, and are therefore invariant through phase shifts.

Comparisons are made on 2090 sinusoids with random frequencies (with an exponential law between 215 and 4321 Hz), and frequency errors are measured in percentage of halftones. Results (mean error, variance and max error) are presented in Table 1. The frequency estimation based on the MDCT has similar performances as the best FFT-based method, the so-called triangle interpolation: its mean and variance are slightly lower, but the maximum error is slightly higher.

| Frequency estimates of pure sines | | | |
|---|---|---|---|
| Method | $\mu$ | $\sigma$ | max |
| (1) plain FFT | 28.11 | 28.12 | 149.76 |
| (2) parabolic interp. | 0.054 | 0.118 | 1.096 |
| (3) triangle algo. | 0.006 | 0.016 | 0.136 |
| (4) spectral reass. | 0.048 | 0.115 | 1.097 |
| (5) derivative algo. | 0.048 | 0.115 | 1.100 |
| (6) phase vocoder | 0.048 | 0.115 | 1.099 |
| **MDCT** | **0.005** | **0.013** | **0.167** |

Table 1: Frequency error in percentage of halftones. Results for methods 1 to 6 are extracted from [2].

| Amplitude estimates of pure sines | | | |
|---|---|---|---|
| Method | $\mu$ | $\sigma$ | max |
| (1) plain FFT | 0.470 | 0.428 | 1.424 |
| (2) parabolic interp. | 0.001 | 0.001 | 0.008 |
| (3) triangle algo. | 0.000 | 0.000 | 0.001 |
| (4) spectral reass. | 0.001 | 0.002 | 0.017 |
| (5) derivative algo. | 0.001 | 0.002 | 0.017 |
| (6) phase vocoder | 0.001 | 0.002 | 0.017 |
| **MDCT** | $\mathbf{3.8 \times 10^{-4}}$ | $\mathbf{6.5 \times 10^{-4}}$ | **0.007** |

Table 2: Amplitude error in dB. Results for methods 1 to 6 are extracted from [2].

As for the frequency estimation (Table 2), the MDCT method outperforms at least 5 out of 6 FFT-based methods. Unfortunately, results in [2] are not given with sufficient precision for the triangle interpolation method, therefore a precise comparison is not possible.

As a conclusion, for pure sinusoidal signals, the estimation method presented in section 2 performs equally well as the best FFT-based methods for the local estimation of frequency and amplitude.

### 3.2. Influence of noise

We also investigated the influence of noise in the precision of these estimates. More specifically, we have focused our studies on the influence of white noise and quantization noise. In the case of quantization *applied to the MDCT coefficients*, the noise is correlated with the MDCT signal, and estimates can be biased when the quantization is performed on a low number of bits. For the sake of simplicity we have only considered uniform quantization, although some MDCT-based codecs such as AAC use non-uniform quantization (but in this case exact comparisons are difficult since this quantization does depend on the implementation).

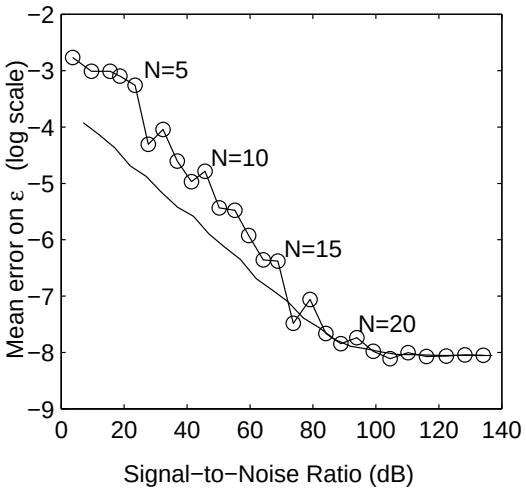


Figure 5: Value of the mean (across 254 consecutive windows) of the frequency error in the presence of white noise (full line) and uniform quantization of the MDCT coefficients ($\circ$). Numbers $N$ indicate the number of bits for the uniform quantization.

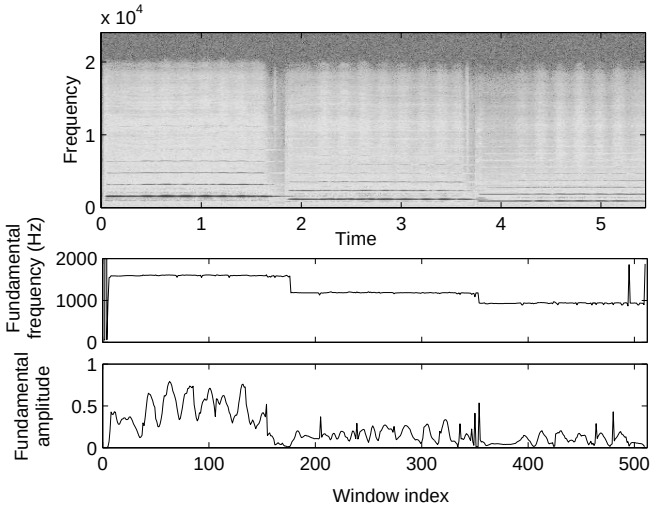


Figure 6: Frequency and amplitude estimation of the fundamental of a flute sound.

Figure 5 shows the variations of the mean error in the estimation of the frequency, as a function of the noise level, for a given sine wave. In the case of white noise, the logarithm of the error is a regularly decreasing function of the signal-to-noise ratio, until it reaches the asymptotic value of the noiseless case (in this case the error caused by the noise is much lower than the error due to the approximations in the computation of the equation (2) (aliasing terms neglected, linearization of $\sin$ terms). In the case of quantization noise (round marks in figure 5), the error curve has more fluctuations, but it almost consistently above the white noise error at the same SNR.

We have applied this technique on a real acoustic sound, the recording of an arpeggio played on a flute. Results are presented in figure 6. The algorithm has correctly identified the fundamental frequency, and the slow fluctuations of the amplitude (vibrato) are well measured. Note that so far this algorithm only identifies one peak in the MDCT domain, so it only works for monophonic signals, with a fundamental that has a stronger energy than the higher harmonics.

## 4. CONCLUSION : A BASIS FOR FEATURE EXTRACTION IN THE TRANSFORM DOMAIN

In this paper, we have investigated the possibility of extracting local frequency / amplitude information of a tonal signal from its MDCT coefficients. Explicit formulas prove the relevance of this approach in the noiseless case. We have also shown that the accuracy of these frequency estimates, as well as the robustness to noise, is comparable to the one obtained by Short-Time Fourier Transform-based methods, such as the phase vocoder.

Future work will focus on two aspects:

- improvement of the existing algorithm. In particular, it would be desirable to improve the robustness of the amplitude estimation with respect to noise. This is possible - in theory - since equation 12 only depends on two MDCT coefficients ; taking more coefficients into account would average the noise effects.
- work on complex spectra. At the moment the algorithm takes into account only one (possibly time-varying) tone. With a local peak picking algorithm in the $\mathcal{S}$-spectrum it may be possible to detect the more prominent partials. This can be the basis of a high-precision pitch estimation, or a way to compute the inharmonicity factor of piano tones, for instance.

Just another frequency estimator then ? Well, in a way, yes, but this one can actually be useful ! Since many signals nowadays are stored in a compressed form, such as in the MP3 or MPEG-2 AAC, this scheme is indeed the basis of a low-complexity feature extraction from already encoded filed, that can be used for audio indexing.

## 5. REFERENCES

[1] B. Picinbono, "On instantaneous amplitude and phase of signals," *IEEE Trans. Signal Processing*, vol. 45, pp. 552–560, 1997.

[2] F. Keiler and S. Marchand, "Survey on extraction of sinusoids in stationary sounds," in *Proceedings 5th Int. Conf. on Digital Audio Effects (DAFx-02), Hamburg*, 2002.

[3] L. Daudet and M. Sandler, "MDCT analysis of sinusoids: exact results and applications to coding artifacts reduction," *Submitted to IEEE Trans. Speech and Audio Proc.*

[4] A. Ferreira, "Accurate estimation in the ODFT domain of the frequency, phase and magnitude of stationary sinusoids," in *Proc. WASPAA, New Paltz*, oct. 2001.

[5] M. Puckette and J. Brown, "Accuracy of frequency estimates using the phase vocoder," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 2, 1998.