Speech Analysis and Synthesis with a Computationally Efficient Adaptive Harmonic Model

Veronica Morfi, Gilles Degottex, and Athanasios Mouchtaris

***Abstract*—Harmonic models have to be both precise and fast in order to represent the speech signal adequately and be able to process large amount of data in a reasonable amount of time. For these purposes, the full-band adaptive harmonic model (aHM) used by the adaptive iterative refinement (AIR) algorithm has been proposed in order to accurately model the perceived characteristics of a speech signal. Even though aHM-AIR is precise, it lacks the computational efficiency that would make its use convenient for large databases. The least squares (LS) solution used in the original aHM-AIR accounts for most of the computational load. In a previous paper, we suggested a peak picking (PP) approach as a substitution to the LS solution. In order to integrate the adaptivity scheme of aHM in the PP approach, an adaptive discrete Fourier transform (aDFT), whose frequency basis can fully follow the variations of the f_0 curve, was also proposed. In this paper, we complete the previous publication by evaluating the above methods for the whole analysis process of a speech signal. Evaluations have shown an average time reduction by four times using PP and aDFT compared to the LS solution. Additionally, based on formal listening tests, when using PP and aDFT, the quality of the re-synthesis is preserved compared to the original LS-based approach.**

***Index Terms*—Fundamental frequency, harmonic model, peak picking (PP), speech analysis/synthesis, voice model.**

I. Introduction

HARMONIC models (HM) aim to represent the speech signal with a set of parameters such as frequencies, amplitudes and phases. These models can be used for speech modeling [1], speech coding and synthesis [2], [3], voice transformation [4], speech enhancement [5] for hearing aids [6]. The parameters computed can be used to build higher-level representations [7] (e.g. spectral envelopes) or to estimate glottal source characteristics [8]. For this purpose, the accuracy of the parameters is a key issue. Furthermore, a representation that can produce sounds with sufficient perceived quality is of high importance for applications in synthesis, which need robust and precise estimates of f_0. There are plenty of real-time applications that need this high-quality synthesis, such as text-to-speech applications, analysis and synthesis techniques for quiet environments, etc. Additionally, speech signal analysis for voice production studies require a precision, that is higher than what can be perceived. Finally, even for offline computations, researchers need to test multiple ideas and parameters, various methods and large databases in a convenient time frame, hence, computationally efficient algorithms are preferred.

Harmonic models are initially designed for representation of the deterministic part of the speech. In order to model the non-deterministic part of speech, these models, usually, either employ a random component [9] or represent the voiced speech spectrum by using multiple bands [10], [11]. Alternatively, simpler models have also been suggested in which the spectrum is split into two bands separated by the so-called maximum voiced frequency [12]. The lower and higher bands are used for the deterministic and the non-deterministic components, respectively. A reliable estimation of the voicing frequency limit is critical for all multi-band models, in order to avoid artifacts and provide a sufficient perceived quality of the synthesized sound. However, the need of a frequency limit is questionable. From the point of view of the voice production, there is no reason to abruptly low-pass the deterministic component of the voice, since the voiced source is made of glottal pulses that are fundamentally wideband signals whose amplitude spectrum is known to decrease smoothly [13], [14]. For this reason a full-band model called the adaptive Harmonic Model (aHM) has been suggested which estimates frequency components up to Nyquist [15]. A detailed explanation of aHM can be found in Section II.

In voiced segments, the speech signal is usually assumed to be stationary in a small analysis window ($\approx$ 3 pitch periods). This hypothesis is fairly acceptable at low frequencies, because the variations of the fundamental frequency, f_0, of the glottal source are negligible compared to the stationary basis of the usual frequency analysis tools (e.g. DFT). However, the variations of f_0 are proportional to the harmonic number. The non-stationarity of the voiced signal is, therefore, highly increased as frequencies increase, making the validity of the stationarity hypothesis questionable for mid and high frequencies up to Nyquist. To alleviate this issue of modeling non-stationarities, the Fan Chirp

Manuscript received March 11, 2015; revised June 29, 2015; accepted July 06, 2015. Date of publication July 20, 2015; date of current version July 27, 2015. This work was supported in part by the European Union's Horizon 2020 research and innovation programme under the Marie Skodowska-Curie Grant 644283. The work of G. Degottex was supported by the Foundation for Research and Technology-Hellas (FORTH), Heraklion, Greece. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Zhen-Hua Ling.

The authors are with the Computer Science Department, University of Crete, Heraklion, Crete 70013, Greece, and also with the Institute of Computer Science, Foundation for Research and Technology-Hellas (FORTH-ICS), Heraklion, Crete 70013, Greece (e-mail: veronicamorfi@gmail.com; mouchtar@ieee.org; degottex@csd.uoc.gr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2015.2458580

Transform (FChT), which uses a chirp related frequency basis (i.e. linear frequency trajectories) adapted to the input signal, has been suggested in [16]. For sinusoidal models, the adaptive Quasi-Harmonic Model (aQHM), a quasi-harmonic representation of the speech spectrum that does not rely on a chirp frequency basis, has also been suggested in [17]. Instead of limiting the frequency tracks to linear time evolution, as in FChT, aQHM relies on a more flexible frequency model. The frequency basis is adapted to the f_0 curve estimated from the speech signal. Thus, the adapted frequency basis can follow any non-linear variations of the frequency basis of the underlying signal. However, a proper estimation of the sinusoidal parameters can be obtained only if the input components of the frequency basis built from the f_0 curve are in a reasonable interval around the actual frequencies. Therefore, tracking the harmonic structure up to Nyquist can be easily compromised since any error on the f_0 curve is multiplied by the harmonic number.

In [15], a strict harmonic model using adaptivity and full-band representation, referred to as the adaptive Harmonic Model (aHM), was presented. The aHM is a full-band model which uses the adaptive scheme of the adaptive Quasi-Harmonic Model (aQHM) in order to represent a speech signal. Also, an iterative algorithm, called Adaptive Iterative Refinement (AIR), has been proposed in regard to the potential error in f_0 leading to erroneous localization of sinusoidal components. The AIR algorithm begins with the lower frequency components, where the error of f_0 is considered to be small, and iteratively increases the number of harmonics considered up to the Nyquist frequency by successive refinement of the f_0 curve at each iteration step. The Least Squares (LS) solution was used for the computation of the sinusoidal parameters of the harmonic model. However, even though aHM-AIR allows for a robust estimation of the harmonic components, the computational load of the LS solution does not allow processing of large databases in a reasonable amount of time, which is a serious drawback. This issue was addressed in [18], by replacing the LS solution with a Peak Picking (PP) approach [19]. In order to integrate the adaptivity scheme of the aHM to the PP approach, the adaptive Discrete Fourier Transform (aDFT) was also proposed in [18]. In contrast to the constant basis of the DFT, the frequency basis of the aDFT is fully adapted to the input f_0 curve of the signal, as the aHM basis is adapted to the signal. In [18], we used this approach for the refinement of the f_0 curve. In the present article, this approach is, also, used in order to estimate the sinusoidal parameters used for the synthesis. Consequently, this article expands the research done in [15] and presents a detailed study on the reduction of the computational load of aHM-AIR, by proposing and evaluating several approaches for performing this task for both aforementioned steps of the analysis process, using either aDFT or FChT and PP in the previously proposed aHM approach [18]. We carried a new evaluation procedure in order to assess the accuracy of the model parameters, using synthetic signals in order to properly evaluate the advantage of these parameters before building higher-level models (e.g. spectral envelope). Then, the Signal-to-Reconstruction Error Ratio (SRER) is computed for both voiced and unvoiced segments. Finally, the results of a listening test and the Perceptual Evaluation of Speech Quality (PESQ) are presented.

In the rest of the paper, Section II describes the necessary mathematical background for aHM, Section III presents the adaptive Discrete Fourier Transform and Section IV then provides all of the technical details of the AIR algorithm for aHM with the Peak Picking approach replacing the LS solution. The evaluation follows in Section V with the necessary discussions and conclusions in Section VI.

II. Adaptive Harmonic Model (AHM)

The main difference between the Harmonic Model (HM) and the adaptive Harmonic Model (aHM) is that the first uses random noise components (i.e. HNM [9]) or multiple bands in order to represent the non-deterministic part of speech while aHM is a full-band model that uses the adaptive scheme of aQHM. Given the speech waveform $s(t)$, it is first assumed that the values of its fundamental frequency curve f_0 are known a priori, though a potential error on this curve is taken into consideration in this work. Then, the following aHM model of $s(t)$ is used in a single window of 3 pitch periods:

$$x(t) = \sum_{h=1}^{H} a_h(t) \cdot e^{jh\phi_0(t)} \tag{1}$$

where $a_h(t)$ is a complex function of time representing both the amplitude and the instantaneous phase of the hth harmonic and $\phi_0(t)$ is a real function defined by the integral of $f_0(t)$:

$$\phi_0(t) = \frac{2\pi}{f_s} \int_0^t f_0(\tau) d\tau \tag{2}$$

where the time reference $t = 0$ is the center of the window, and f_s is the sampling frequency. According to the adaptive scheme proposed in [20], $a_h(t)$ and $f_0(t)$ are obtained by linear and spline interpolation of anchor values a_h^i and estimated f_0^i at specific instants t_i, respectively. The estimation of these anchor values is described in the rest of the current section. Therefore, aHM will provide estimates of these parameters, which are sufficient for the complete representation of the speech signal. A sequence on anchor time instants, t_i is created during the aHM analysis using the provided $f_0(t)$ curve. These anchors derive from:

$$t_{i+1} = f_0(t_i)^{-1} + t_i \tag{3}$$

where $t_0 = 0$. Even in the unvoiced segments, where the estimated f_0 does not hold a particular meaning, its value is used to generate these time instants. However, the number of anchors has to be properly chosen, since too many anchors may overfit the signal and represent variations that are not related to a deterministic component in voiced segments. A behavior like that has no true meaning for statistical modeling and may even cause the voice characteristics to be difficult to control in voice transformation. On the other hand, underfitting the signal with too few anchors should also be avoided. For speech, it can safely be assumed that the frequency modulation is related to a change of pulse duration and not to any modulation inside a single pulse. Hence, one anchor per period should suffice and in this article, a pitch synchronous analysis in which the distance between anchors respects an input f_0 curve, is assumed.

For the aHM parameter estimation with the presence of potential errors in the f_0 curve, the frequency correction mecha-

nism of aQHM is used [20]. Within a single window, this model is similar to:

$$x(t) = \sum_{h=1}^{H} (a_h + t b_h) e^{jh\phi_0(t)} \tag{4}$$

where $\phi_0(t)$ is still defined by Eq. (2) and a_h, b_h are complex values that are constant in the window, in contrast to $a_h(t)$ in Eq. (1). To estimate a_h and b_h the following squared error is minimized by discrete sampling between the windowed speech segment $s[n]$ and its model $x[n]$

$$\epsilon = \sum_{n=0}^{N-1} (s[n] - x[n])^2 \tag{5}$$

where N is the number of samples in the analysis window. Moreover, the model parameters are estimated via the LS solution, given the samples of the input signal in a vector s:

$$\begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} = (E^H W^H W E)^{-1} E^H W^H W s \tag{6}$$

where W is the diagonal matrix containing the window values in the diagonal, s is the input signal vector and $E = [E_0, E_1]$ is the adapted frequency basis, which have elements given by

$$E_0 = (E_0)_{n,h} = e^{jh\phi_0(t_n)} \tag{7}$$

$$E_1 = (E_1)_{n,h} = t_n (E_0)_{n,h} = t_n e^{jh\phi_0(t_n)} \tag{8}$$

III. Adaptive Discrete Fourier Transform (aDFT)

In order to increase the computational speed, the replacement of the LS solution (Eq. (6)) with a Peak Picking approach was suggested in [18], in order to estimate the above aHM parameters in a more efficient way.

The core of this faster approach lies in the adaptive Discrete Fourier Transform (aDFT) [18]. In order to properly describe the aDFT and emphasize the importance of adaptivity for the AIR algorithm, a comparison between the DFT, FChT and aDFT is first presented in this section. In Fig. 1, the frequency basis for the three transformations mentioned above for a single analysis window and the respective spectrogram, are made visible. The results obtained by these three methods for a longer time period are depicted in Fig. 2.

In order to compare all three transforms, we need to start with the frequency basis of the DFT. For a windowed signal $x[n]$ of length N, the DFT is defined as

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{k}{N}\phi[n]} \tag{9}$$

where N represents the DFT length, $k = 0, 1, \ldots, N-1$ (first row of Fig. 1). In the DFT, there is the assumption of stationarity in the analyzed signal, since the frequency basis $\phi[n]$ used to compute the DFT is constant inside the analysis window:

$$\phi[n] = n \tag{10}$$

with time derivative:

$$\phi'[n] = 1 \tag{11}$$

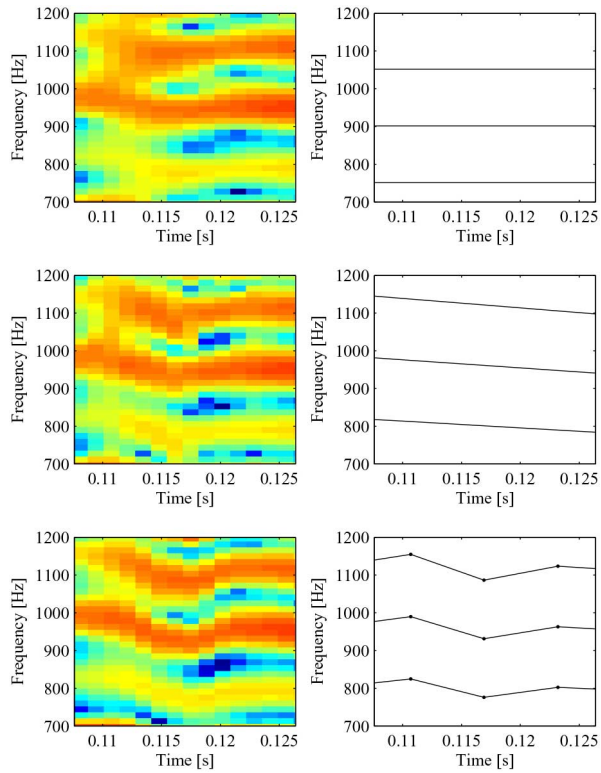

Fig. 1. Three different transforms and their respective frequency bases for a single analysis window. First row depicts the spectrogram and frequency basis of the central frame of DFT, second row of FChT and third row of aDFT.

However, in speech signals, this assumption of stationarity is valid only when the variations of the fundamental frequency, f_0, are negligible compared to the stationary basis of the DFT. Moreover, the variations of the harmonics are proportional to those of f_0 multiplied by the harmonic number. Hence, as frequencies increase so does the non-stationarity of the voiced signal, making the validity of the stationarity hypothesis questionable for mid and high frequencies. The first row of Fig. 2 presents the DFT spectrogram. One can see that the frequency content is highly blurred around 2.5 kHz.

To alleviate this issue, the Fan Chirp Transform (FChT) has been proposed in [16]. In this method, a chirp related frequency basis (i.e. linear frequency trajectories) is used, with its slope adjusted to the average slope of the f_0 curve in the analysis window. For a windowed signal $x[n]$ of length N, the FChT is defined as

$$X_a[k] = \sum_{n=0}^{N-1} x[n] \xi^*(n, k, a) \tag{12}$$

where N also stands for the FChT length, $k = 0, 1, \ldots N-1$, * denotes the complex conjugate and $\xi(n, k, a)$ is the frequency basis of the FChT defined as

$$\xi(n, k, a) = \sqrt{|\phi_a'[n]|} e^{-j2\pi \frac{k}{N}\phi_a[n]}, \tag{13}$$

where $\phi_a[n]$ rules the time dependence of the frequency basis exponent

$$\phi_a[n] = \left(n + \frac{1}{2} a n^2\right) \tag{14}$$

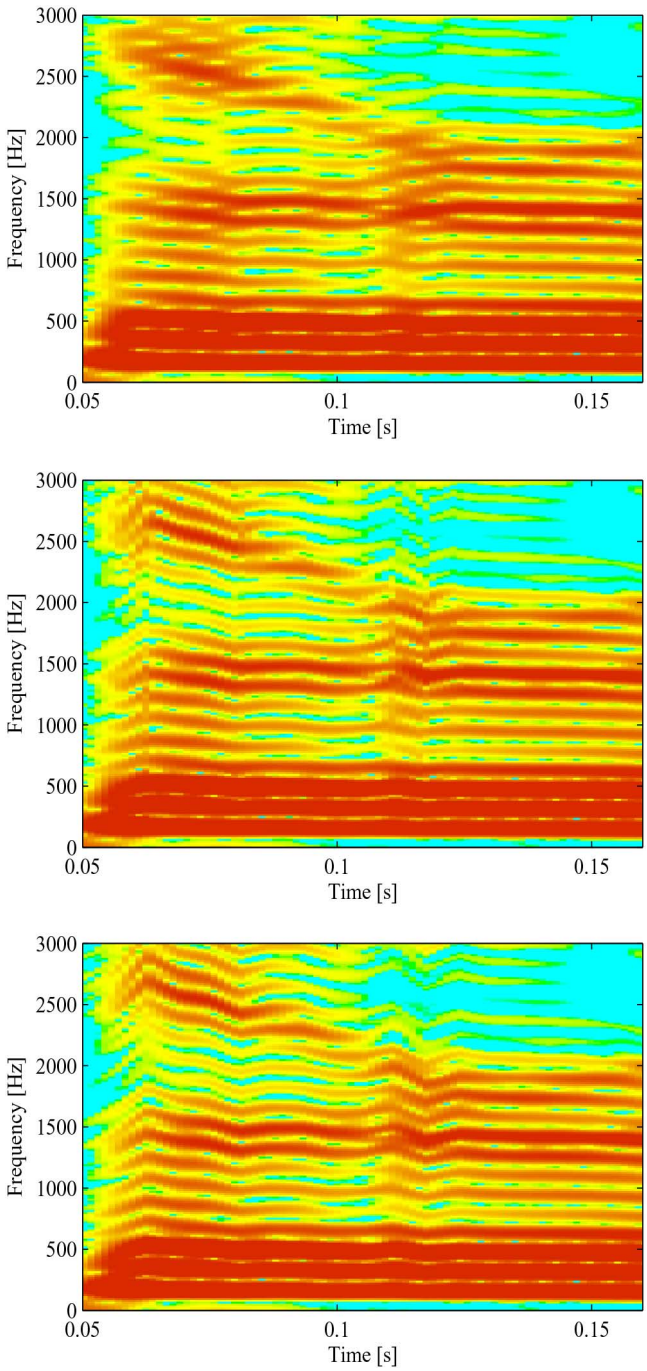

Fig. 2. Spectrograms produced by DFT, FChT and aDFT depicted in the first, second and third row, respectively.

whose time derivative is:

$$\phi_a'(n) = (1 + an) \tag{15}$$

where the parameter a is the chirp rate of the f_0 slope (second row of Fig. 1). It can be observed that with the linearly adapted frequency basis of FChT, the harmonics become clearer compared to the ones produced by the DFT. In the second row of Fig. 2, one can notice that even around 2.5 kHz the harmonics can be easily traced, especially compared to the results of the DFT spectrogram shown in the first row. Hence, there is a regularity in the frequency content even in mid/high frequencies when the FChT is used, which was not visible by using the DFT. Still, even though the FChT basis adapts to the frequency modulations better than the DFT, the frequency basis is constrained to linear trajectories only.

In [18], in order to better follow the non-linear variations of f_0, the adaptive Discrete Fourier Transform (aDFT), based on the adaptivity scheme of aQHM [17] and aHM [15], was proposed. The frequency basis used for the aDFT follows completely the f_0 curve variations. Since the tracking of real sinusoids needs only the positive frequencies, the following representation is limited to the positive part of the aDFT. For a windowed signal $x[n]$ of length N, the aDFT of the positive frequencies, is defined as

$$X[k] = \sum_{n=0}^{N/2} x[n] e^{-j2\pi \frac{k}{N} \phi_0[n]} \tag{16}$$

where N, also, refers to the aDFT length, $k = 0, 1, \ldots, N/2$ and $\phi_0[n]$ is the "fundamental phase" of the frequency basis, whose values are obtained from the discrete sampling of the continuous real function, $\phi_0(t)$, defined by the normalized integral of the fundamental frequency $f_0(t)$:

$$\phi_0(t) = \int_0^t \frac{f_0(\tau)}{f_0(0)} d\tau \tag{17}$$

where the time reference $t = 0$ is the center of the window and f_s denotes the sampling frequency. In (17), $f_0(0)$ normalizes the frequency basis so that in the center of the window, where $t = 0$, it corresponds to that of the DFT as shown through the time derivative:

$$\phi_0{}'(t) = \frac{f_0(t)}{f_0(0)} \tag{18}$$

According to the adaptivity scheme, $f_0(t)$ is obtained by linear interpolation of the anchor values f_0^i at specific instants t_i. The third row of Fig. 1, shows an example of the results of aDFT and its respective frequency basis of the central frame. It can be noticed that the frequency basis of aDFT compared to the other two methods is fully adapted on the variations of the f_0 curve, hence, it can produce more accurate time-frequency representation.

As mentioned above, in the second row of Fig. 2 the harmonics around 2.5 kHz can be more easily traced compared to the ones in the first row. This creates a regularity in the frequency content even in mid/high frequencies when FChT is used. In the third row of Fig. 2, where the aDFT is used, this regularity can be noticed even more in mid/high frequencies.

In Fig. 2 some discontinuities can be observed, especially in the first and second row, between time instants. However, aHM always forces harmonicity, as it is a purely harmonic model, even if there is none in the signal. Thus, possible discontinuities are always smoothed during the synthesis. Additionally, when there is an erroneous f_0 curve as an input, if the error is less than an octave, the spectrogram should be correct because the frequency basis is normalized by the central f_0 in aDFT.

IV. Parameter Estimation Method

In this section, the method for estimating the aHM parameters up to Nyquist is described, namely the Adaptive Iterative Refinement (AIR) algorithm which uses the proposed Peak

TABLE I
METHODS USED FOR BOTH STEPS OF THE ANALYSIS PROCESS

	Analysis Process Steps	
Method Name	**Refinement of f_0**	**Sinusoidal Parameters Estimation**
LS-LS	LS	LS
aDFT-aDFT	aDFT	aDFT
aDFT-LS	aDFT	LS
FChT-FChT	FChT	FChT
FChT-LS	FChT	LS

Picking (PP) approach on the adaptive Discrete Fourier Transform (aDFT) [18]. The global structure of the original AIR algorithm has been already described in [15] and is basically kept the same. In general, the AIR algorithm is used to refine the incorrect localization of sinusoidal components due to the potential error in f_0, in order to allow a robust estimation of harmonic components up to the Nyquist frequency. In the present article, three different estimation methods (based on LS, FChT, aDFT) were used for the AIR algorithm and the refinement of the f_0 curve. These three methods were, then, used in the last analysis step for the estimation of the final sinusoidal parameters used during the re-synthesis process. In the rest of this paper, the analysis process will be separated in the two aforementioned steps and, for clarity purposes, they will be referred to as the *Refinement of* f_0 step and the *Final Sinusoidal Parameters Estimation* step. Combining these methods (LS, FChT, aDFT) for the two different steps of the analysis process results in the five methods of Table I that will be later on discussed in the Evaluation Section V, one of them having the desired results for both the computational load and the re-synthesis quality.

The AIR algorithm is used for the *Refinement of* f_0 step of the analysis process. The basic idea of this algorithm is to begin by modeling the lower harmonics, where the error in the f_0 measurements can easily be corrected. Then, a refinement of the f_0 trajectory is evaluated based on the refinements of the f_0^i values for each frame and the harmonic order of the model is iteratively increased until the Nyquist frequency is reached. In the original version of the AIR algorithm [15], the refinement of the f_0 trajectory was computed by using the Least Squares (LS) solution, while in this paper, instead of the LS solution a Peak Picking approach is used. Every other aspect of the AIR algorithm was kept the same. A full description of the AIR algorithm and a more detailed explanation of the methods used follows.

During analysis, a parametrization of the speech signal at time instants t_i takes place, as mentioned in Section II. Using a rough estimate of the input f_0 curve, a sequence of instants t_i is first created, with distance of one pitch period between each of them. A Blackman window of 3 local pitch periods is then applied to the speech signal centered around each t_i, with the aDFT length (N) being defined as twice this window's length in order to make the main lobes appear in the aDFT. Consequently, voices with high pitch (e.g. female voices) will need a smaller aDFT length than voices with low pitch (e.g. male voices).

The AIR algorithm works first for each time instant t_i separately, estimating the value of the f_0 at that time instant, namely the f_0^i, where the original estimate of the f_0 curve is provided by the SWIPEP [21] algorithm. At the end of each iteration, the f_0 curve is updated by all the refined values. The algorithm begins at a low harmonic level, $H_i = 8$, for each time instant, meaning that only harmonics up to the 8th one will be taken into account for the refinement of the f_0 curve during the first iteration. For each iteration, the corrected $\hat{f}_0^i$ is estimated for each time instant t_i from the Peak Picking on the aDFT computed from the segment created around that time instant. For the computation of $\hat{f}_0^i$, the harmonic peaks, f_h^i, computed by PP, where h corresponds to the harmonic number, are taken into account. More specifically, the value of $\hat{f}_0^i$ derives from the median of those harmonic peaks, divided by each peak's harmonic number. It was assumed that some peaks are representing noisy components. Thus, some peaks might be unreliable and the median value is an efficient way to discard outliers in the computation of the mean.

$$\hat{f}_0^i = median(f_h^i/h) \tag{19}$$

At the end of each iteration, all f_0^i values are replaced by the new $\hat{f}_0^i$. Before the next iteration begins, H_i is updated for each time instant t_i, as in the original AIR algorithm [15]. Eventually, this process is repeated for all frames until the Nyquist frequency is reached for all of them. Algorithm 1 describes this analysis procedure:

Algorithm 1 AIR for aHM using Peak Picking

```
Create a sequence of time instants t_i according to f_0(t)
Initiate each f_0^i = f_0(t_i)
Initiate each H_i = 8
while ∃i such as f_0^i · H_i < f_s/2
  for each i for which f_0^i · H_i < f_s/2
    Create a segment of 3 periods around t_i
    Compute the aDFT of the segment
    Pick the harmonic peaks f_h^i up to H_i from aDFT
    Compute f̂_0^i = median(f_h^i/h)
    if f̂_0^i · H_i < f_s/2
      Compute f_corr^i = f̂_0^i − f_0^i
      Update H_i = ⌊0.5N_i/|f_corr^i|⌋
    end if
  end for
  Set f_0^i = f̂_0^i ∀i
end while
```

In Algorithm 1, f_{corr}^i is the correction of f_0^i estimated in each iteration (i.e. $f_{corr}^i = \hat{f}_0^i - f_0^i$) and N_i is the aDFT length of frame i. The updated value of H_i has as upper limit the Nyquist frequency.

A brief comparison with the previous version of aHM-AIR [15] can clarify the ways in which this new version (i.e. Algorithm 1) should be more computationally efficient. Originally, in the algorithm proposed in [15], in every iteration for each time

anchor t_i, where the frequency still hadn't reach the Nyquist frequency, the LS solution was used for the minimization of Eq. (5) in order to compute the a_k and b_k parameters of aQHM (i.e. Eq. (6)). On the other hand, in this article, with the substitution of the LS solution with a Peak Picking method, this computationally heavy estimation becomes unnecessary. In Algorithm 1, instead of computing the aHM parameters in each iteration, the f_0 refinement for each time instant t_i, namely $\hat{f}_0^i$, is computed via Peak Picking in an aDFT transform and Eq. (19). This substitution reduces the computational load, making the new version of the AIR algorithm more efficient timewise, as shown in Section V-A.

Taking into account that the main reason behind the replacement of the LS solution with Peak Picking and aDFT approach is to improve the computational load of the aHM-AIR method while preserving the quality of the re-synthesis, a few more modifications were made. The following subsection, IV-A, presents all these modifications used during the aHM-AIR to reduce the computational load, describes a faster version of the process, using a limited-aDFT method and explains how the use of this function affects our proposed Peak Picking scheme. Furthermore, a more detailed description of our proposed Peak Picking scheme and the techniques used for the unvoiced segments follows in IV-B and IV-C, respectively.

A. Reduction of Computational Load Using Limited-aDFT

When the aHM-AIR method begins, the harmonic level is set for each time instant t_i, at a low count. For the next iterations, this level is always limited until the Nyquist frequency is reached. The core of the limited-aDFT idea is that only the part of the aDFT containing the necessary harmonics needs to be calculated. Hence, the computation of any bins above the current harmonic level H_i can be avoided. This optimization cannot be done using the LS solution, because the corrections computed from aQHM are not meaningful when not applied for the full band, up to the Nyquist frequency.

Another improvement regarding the method's complexity is based on the fact that the f_0^i values refined in each iteration, eventually converge. It can be noted that the frequency basis remains almost the same for the low frequencies, as the harmonic level, H_i, increases. Hence, the aDFT in low frequencies is very similar between iterations and the correction of the frequency basis for lower frequencies becomes more and more negligible. Thus, it can be assumed that below a specific extent of correction for each f_0^i, the peaks estimated during the previous iteration would remain almost the same in the lower frequencies, so they can be maintained for all following iterations. In order to implement this idea in the proposed method, a threshold, B_i, in the frequency bins of the aDFT, needs to be decided upon. The use of following relation is suggested:

$$B_i = \left\lfloor \frac{tol \cdot f_0^i \cdot N_i}{|f_{corr}^i| \cdot f_s} \right\rfloor \tag{20}$$

where f_0^i is the frequency at the time instant t_i, N_i is the aDFT length for frame i, f_{corr}^i is the correction of f_0^i computed from the previous iteration (i.e. $f_{corr}^i = \hat{f}_0^i - f_0^i$). A tolerance factor of 10% of the f_0^i value (i.e. $tol = 0.1 f_0^i$) was chosen, which provided an important reduction of the computational time without altering drastically the results. This tolerance factor of 10% roughly means that 10% of the previously computed lower peaks, depending on the correction f_{corr}^i made during this step, can be kept the same in the next aDFT. Hence, it is assumed that refining the new values of these lower peaks in the next iteration will have a negligible influence in the computation of the new $\hat{f}_0^i$ value.

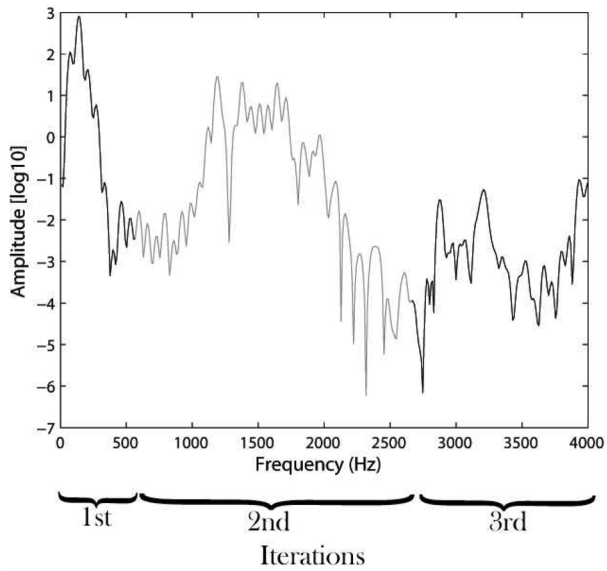

Fig. 3. Illustration of how the limited-aDFT idea works through iterations. Each part of the aDFT is computed in a different iteration, marked at the bottom of the figure.

Utilizing B_i, the bins of the aDFT below this threshold would be kept the same for the following iterations, thus, the aDFT is only computed for the rest of the bins and still only until the upper harmonic limit H_i. Hence, B_i can be considered as the lower harmonic limit for the aDFT computation. Fig. 3 shows an example of what the aDFT for a single window looks like in the third iteration. It can be observed that the lower bins of the aDFT were obtained during the two previous iterations. Each time there was an upper limit H_i and a lower limit B_i considered during the computations. Hence, only the part of aDFT containing harmonics between these two limits was estimated.

It becomes apparent that, by using the limited-aDFT, thus, keeping part of the aDFT intact from iteration to iteration, the harmonic peaks inside that part, also, remain the same. This has an interesting affect on the PP approach. The Peak Picking method can adapt to keep the harmonic peaks obtained from the frequency bins below the threshold and only compute the peaks in the rest of the frequencies. Later on, both the peaks from the previous iterations and the ones obtained in the current one are used for the computation of $\hat{f}_0^i$, that will replace f_0^i at the end of each iteration.

B. The Proposed Peak Picking Scheme

In this section a detailed explanation of our proposed Peak Picking scheme follows. Algorithm 2 describes our Peak Picking scheme and the conditions used to determine whether a peak obtained by the aDFT representation can be considered harmonic or not, and which harmonic count it corresponds to. Hence, which aDFT peaks will be taken into account in

the refinement of the fundamental frequency of a window i (Eq. (19)).

Algorithm 2 Peak Picking for an analysis window i

```
Find all peaks
for each harmonic count h = 1 : H_i
    Search for closest peak to h · f_0^i
    if (peak's distance from h · f_0^i) > f_0^i/2
        Dismiss
    else if peak is previously identified as harmonic
        Dismiss and find second closest peak to h · f_0^i
        if (peak's distance from h · f_0^i) > f_0^i/2
    Dismiss
    else if peak is previously identified as harmonic
            Dismiss
        else
            Identify peak as harmonic of count h: f_h^i
        end if
    else
        Identify peak as harmonic of count h: f_h^i
    end if
end for
```

The harmonic peaks are defined as integer multiples of the fundamental frequency of a window i, f_0^i, with the harmonic order h ($h = 1, 2, \ldots, H_i$). In order to determine which of the aDFT peaks have to be considered in the $\hat{f}_0^i$ computation, the minimum distance between each harmonic peak and the peaks measured in the aDFT is computed, thus, finding the closest peak to that harmonic peak similarly to [22]. However since not all frames have a strict harmonic structure (e.g. high frequencies, unvoiced frames, etc.), in order for Peak Picking to find the better fitted harmonic structure of the window, a few more restrictions were placed for the harmonic identification, as shown in Algorithm 2. At the end of the proposed Peak Picking scheme a set of harmonics, f_h^i, is produced for each analysis window. This set of harmonics is, then, used in the estimation of the refinement of the fundamental frequency of each window following Eq. (19). In future works, this part of the method could be improved by using peak classification criteria as suggested in [23].

The results of this method produce a resynthesis that has almost identical perceived quality compared to the one given by the LS solution for the *Refinement of* f_0 step, but its robustness is based on the assumption that the input f_0 curve is fairly correct. That is not the case when there is a substantial amount of noise in the curve. More precisely, in the first iteration the harmonic base derives from the input f_0 curve which, as mentioned above, could have some noise. During the first iteration of the PP method only the first harmonic ($h = 1$) is obtained, namely f_1^i. Based on the input frequency basis, PP will look for f_1^i around the frequency value $1 \cdot f_0^i$. Then, in the following iteration, PP will search for the second harmonic f_2^i around the frequency value $2 \cdot f_0^i$ and so forth. Consequently, at the end of PP all the harmonics collected will be almost multiples of the frequency basis, f_0^i, hence its error will be carried in all the following estimations, too, which may lead to skipping harmonics and wrongly recognizing others. In order to solve this problem and make the method more robust, instead of computing the refinement, $\hat{f}_0^i$, at the end of the PP method, $\hat{f}_0^i$ is evaluated whenever a new harmonic peak is identified, following (Eq. (19)). After the first couple of harmonic peaks are identified, the value used for the first harmonic base changes (i.e. $f_0^i = \hat{f}_0^i = \text{median}(f_h^i/h)$). Additionally, with every new harmonic peak identified by our proposed Peak Picking scheme, the harmonic base is recomputed. Hence, the influence of noise in the f_0^i value lessens through iterations, resulting to a more precise estimation of the rest of the harmonics, and finally, of the fundamental frequency of the window. This is based on the fact that not all harmonics are an exact multiple of the harmonic base, hence with each recomputation of the harmonic base its value will converge to the actual one. However, as a drawback, the algorithm becomes a little slower but the results become more robust.

C. Unvoiced Segments

In unvoiced segments, no harmonic structure exists, hence using a harmonic model in those parts becomes questionable. However, as it has been shown in [15], it is possible to use aHM for both voiced and unvoiced segments, thus providing a uniform representation across time which does not need any voicing decision. Nevertheless, often, while using the suggested PP approach in unvoiced segments, either substantial deviations from the input f_0 curve occurred or the $\hat{f}_0^i$ value computed for an unvoiced segment ended up not converging. This is caused by the lack of harmonic structure in addition to the low harmonic level used during the first steps (e.g. $H_i = 8$ for the first iterating step), which prevent convergence of the $\hat{f}_0^i$ values. However, it was observed that using a higher harmonic level this was not the case, even for unvoiced segments.

The original estimate of the f_0 curve for the unvoiced frames is provided by the SWIPEP [21] algorithm, as it was the case with the voiced frames. Ideally, while dealing with unvoiced frames, an estimator should favour low frequencies, so that there is enough frequency resolution for representing the noise. In this paper, the estimator considers a higher harmonic count than the original $H_i = 8$ in the unvoiced frames, thus, it doesn't favour the lower frequencies, but it tries to fit the most harmonic structure it can find closer to the initial f_0 curve values. We suggest to discard $\hat{f}_0^i$ values with any substantial deviations from the previous f_0^i values of each time instant t_i. Additionally, when a value is discarded, a forced increase of the harmonic level, before the next iteration, is used. In the current implementation, a deviation threshold of 8% from f_0^i is used to decide whether or not each ${f_0}^i$ will be discarded. It was observed, after experimentation, that any ${f_0}^i$ value that surpassed the 8% threshold either ended up converging in a extremely erroneous value or did not converge at all. In the case of a discard, the forced increase of the harmonic level takes place according to the following equation:

$$H_i' = |\hat{f}_0^i - f_0^i| \cdot H_i \tag{21}$$

This allows to force the harmonic level for the next iteration high enough that even the unvoiced frames will have enough pseudo-harmonic peaks to converge towards a stable f_0^i value.

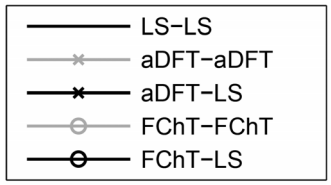

Fig. 4. Line styles for all methods shown in Fig. 5, Fig. 6 and Table I. The first term in all line styles denotes the method used for the *Refinement of* f_0 step and the second one denotes the method used for the *Final Sinusoidal Parameters Estimation* step.

TABLE II
AVERAGE TIME REDUCTION RATIOS FOR THE *REFINEMENT OF* f_0 STEP

Methods	Male Voices	Female Voices	All
$\frac{FChT}{LS}$	0.11	0.15	0.13
$\frac{aDFT}{LS}$	0.23	0.28	0.25

TABLE III
AVERAGE TIME REDUCTION RATIOS FOR THE *FINAL SINUSOIDAL PARAMETERS ESTIMATION* STEP

Methods	Male Voices	Female Voices	All
$\frac{FChT}{LS}$	1.98	2.23	2.10
$\frac{aDFT}{LS}$	3.16	3.38	3.27

V. EVALUATION

For the following evaluations, three different implementations of aHM-AIR were taken into account, namely the AIR algorithm can use either the LS solution [15], our proposed Peak Picking approach using FChT or our proposed Peak Picking approach using aDFT, for the *Refinement of* f_0 step of the analysis process. From this refined f_0 curve, the sinusoidal parameters of the harmonic model are, then, evaluated in the last step of the analysis process, namely the *Final Sinusoidal Parameters Estimation* step of analysis. For this step, all three methods mentioned above were, again, applied. This led to the comparison of the 5 methods from Table I, depicted by the line styles of Fig. 4 in the following evaluations.

All the evaluations, except for the parameter estimation error evaluation, were applied on a small database of 32 utterances (16 male and 16 female, originating from 16 different languages, between 2s and 4s length, with sampling frequency varying between 16 kHz and 44 kHz). The different phonemes and origins of these languages are assumed sufficient to provide a voice variability for supporting the validity of the results. For FChT, the chirp-factor a for each time instant t_i, was estimated based on the slope factor of the linear interpolation of the two neighboring f_0 values, f_0^{i-1} and f_0^{i+1}, around t_i.

A. Computational Time

For each method, the running time has been measured for each recording and the time reduction ratios, with respect to the LS-based method (Table II and Table III), were averaged among all sentences. Table II presents the ratios for the *Refinement of* f_0 step of the analysis process. While, Table III displays the ratios for the *Final Sinusoidal Parameters Estimation* step of the analysis, where the parameters are estimated by all three methods.

On Table II, it can be noticed that, on average, when using FChT, aHM-AIR becomes 7.69 (i.e. $\frac{LS}{FChT} = \frac{1}{0.13} \approx 7.69$) times faster, while, with aDFT, it becomes 4 (i.e. $\frac{LS}{aDFT} = \frac{1}{0.25} = 4$) times faster compared to the LS solution approach. Among the used sentences, the maximum ratio of time improvement observed was 21.67 for FChT and 7.67 for aDFT compared to the LS solution. The reason why there is such a difference between the improvement caused between FChT and aDFT is due to the fact that the frequency basis for FChT is less flexible than for aDFT and the slope parameters of FChT converge quicker than the actual f_0 values. On the one hand, the aDFT keeps on changing as long as the f_0 values change. Thus, if the f_0 values change from one iteration to the next, the frequency basis of the aDFT will also be different, hence, the peak picking will find different peaks and the next f_0 correction will be proportional to these changes. On the other hand, for FChT, even though the f_0 values can change between two refinement iterations, the slope can be extremely similar, since many different sets of f_0 values have the exact same linear regression. Thus, the FChT may not change, and as a consequence the peaks remain the same and the f_0 correction can be almost zero for the next step. Thus, one can, indeed, expect a faster convergence with FChT than with aDFT.

For the *Final Sinusoidal Parameters Estimation* step, all three methods were also used. By studying Table III it can be observed that using the LS solution is faster than using either FChT or aDFT in the *Final Sinusoidal Parameters Estimation* step. This is mainly due to the fact that in this step of the analysis process, both FChT and aDFT are computed for each frame up to the maximum harmonic level (i.e. Nyquist), while during the *Refinement of* f_0 step of analysis only parts of them are computed in each iteration, as discussed in Section IV-A. Moreover, the final parameter estimation using LS is fairly faster than using LS during the refinement iterations. Indeed, during the final estimation there is no need for the correction factors obtained using the aQHM model (Eq. (4)). Thus, computing only the aHM terms makes the use of the LS solution more computationally efficient than before. In conclusion, the approaches using transforms are, according to our experiments, not faster than the LS solution for the *Final Sinusoidal Parameters Estimation* step. Table III shows that in this step, on average, LS is 2.10 times faster than FChT and 3.27 times faster than aDFT.

B. Parameters Estimation Error

The purpose of studying the parameter estimation error is to evaluate the precision of the estimated parameters in terms of a sinusoidal representation, compared to aHM-AIR using the LS solution. In the following tests, the estimated frequency, amplitude and phase values are compared to ground truth values of synthetic signals. A synthetic signal, which is as close as possible to a natural speech signal, is obtained by using a Liljencrants-Fant glottal model [13] to synthesize the glottal source. To obtain a realistic vocal tract filter, a digital simulator is used [24] that allows production of 13 different voiced phonemes, including nasalized sounds.

The synthetic signal is obtained as:

$$s(t) = 2\Re\Big(\sum_{k\in\mathbb{R}^+} G^{f_0(t)}(kf_0(t)) \cdot C(kf_0(t)) \cdot e^{jk\phi_0(t)}\Big) \quad (22)$$

where $G^{f_0(t)}(f)$ is the spectrum of the Liljencrants-Fant model, $C(f)$ is the vocal tract filter representing a random phoneme among the 13 covering the vocalic triangle, and $\phi_0(t)$ follows (2), except that, here, $t = 0$ corresponds to the beginning of the signal. The pulse shape of the glottal model is controlled by a random parameter $Rd \in [0.3; 2.7]$ as in [13] and its period is defined by $f_0(t)$.

The following test evaluates the robustness of the different methods when the initial f_0 curve has errors which should be alleviated by the AIR algorithm. In the following tests, the original $f_0(t)$, in (22), is synthesized by using 5 anchors per second with random values in [80;400] Hz. A zero-mean Gaussian noise with various STandard-Deviation (STD) is, then, added to this curve which results to the input f_0 curve to the methods. In Fig. 5 and Fig. 6, the estimation error of the sinusoidal parameters is plotted as a function of the STD of this additive f_0 error. Using a sampling frequency of 44.1 kHz, 320 test samples of 500 ms duration each are generated. The samples are analyzed at regular intervals of 5 ms and the differences between the estimated parameters computed by each method and the reference parameters are determined. Fig. 5 and Fig. 6 show the mean and the STD (using a base-10 logarithmic scale) of the estimation error, in the first three and the last three rows, respectively. The phase error was computed by the wrapped difference between the unwrapped real and estimated values of the phase for these synthetic signals. For all figures, the same line style convention is followed, which is shown in Fig. 4. In the line style names, the first method mentioned denotes the method used for the *Refinement of* f_0 and the second one is the method used for the *Final Sinusoidal Parameters Estimation* step, as shown in Table I. The mean and the STD values were computed through the median and the interquartile range, respectively, to avoid the influence of outliers.

Overall, it can be observed that the results produced by the five different methods used in Fig. 5 and Fig. 6 are, in some cases, very similar. Thus, arises the question of whether or not the difference between the different systems is significant. In order to better understand their difference, the 95% confidence intervals were computed for each method for both mean and STD, prior to the parameter estimation error. The intervals were computed by using 464,870 and 2,073,504 samples for frequencies below 4 kHz and above 4 kHz, respectively. The width of these intervals was approximately 0.1 Hz, 0.01 dB and 0.003 rad for the mean error of frequencies, amplitudes and phases, and 0.0015 Hz, 0.0015 dB and 0.0015 rad (base-10 logarithmic scale) for the STD error, respectively. Additionally, in most cases, there was no overlap between the different methods and even when there was, it occurred for intervals of a very small width. From all the above we can conclude that the difference between the curves shown in Fig. 5 and Fig. 6 are relevant.

1) Refinement of f_0: Full Adaptivity vs. Linear Adaptivity (LS-LS vs. aDFT-LS vs. FChT-LS) : In Fig. 5, the results of the parameter estimation error for aHM-AIR, when the LS solution is replaced by a Peak Picking method in the *Refinement of* f_0, are shown. These values are obtained by the *Final Sinusoidal Parameters Estimation* performed by the LS solution. In the last three rows, the differences between the three methods can be observed more clearly. In the frequency error, row four,

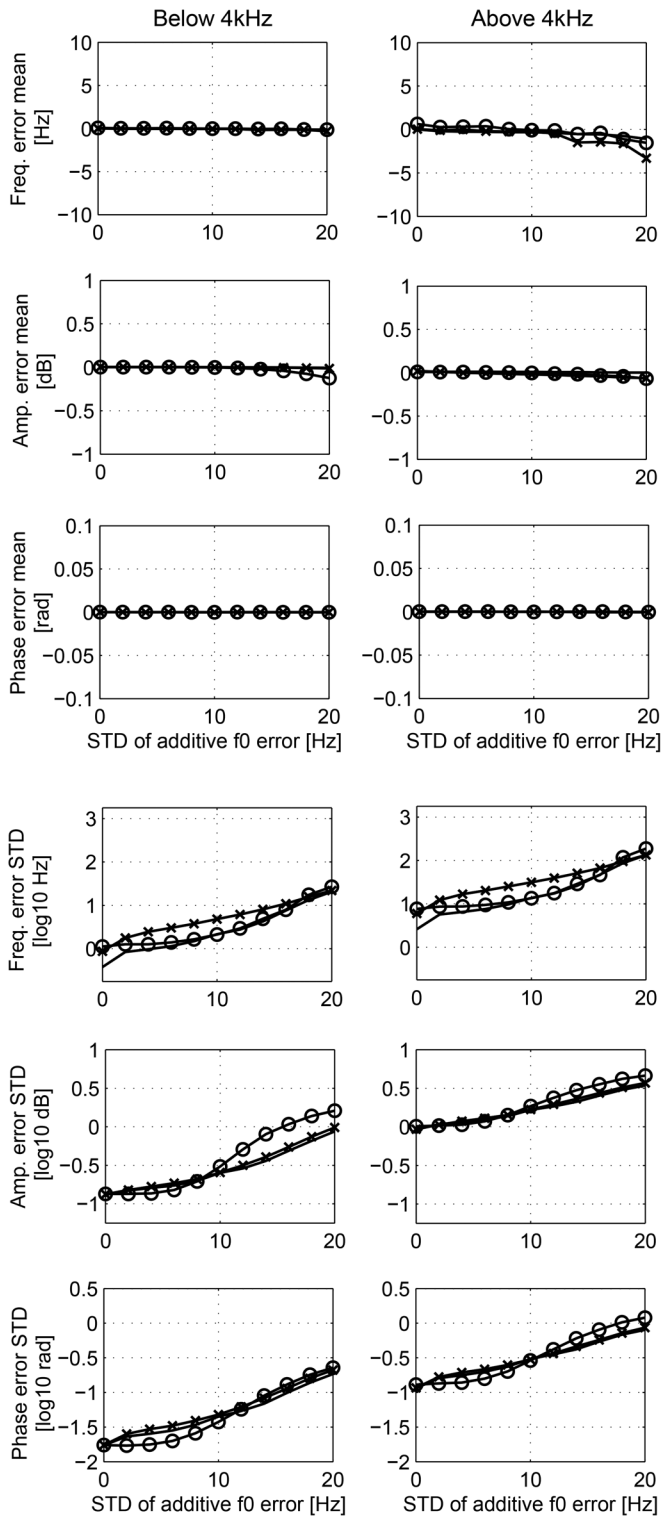

Fig. 5. Error of sinusoidal parameters with respect to a potential error on the f_0 curve provided to the analysis methods, comparing full adaptivity with linear adaptivity during the f_0 refinement steps.

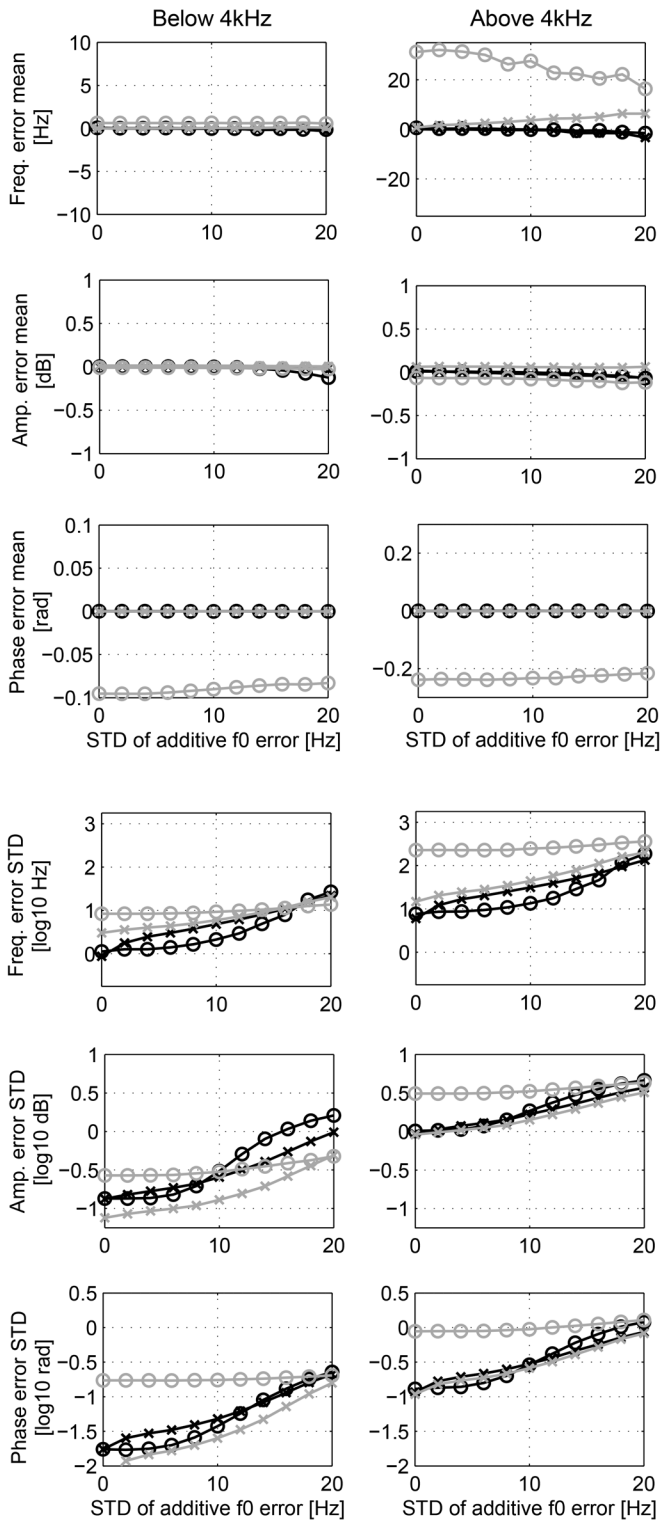

Fig. 6. Error of sinusoidal parameters with respect to a potential error on the f_0 curve provided to the analysis methods, comparing LS solution with Peak Picking in the last analysis step.

it can be observed that for the lower additive noise LS-LS has a smaller STD than the other two methods and as the noise increases LS-LS becomes indistinguishable from FChT-LS until 18 Hz of additive noise STD is reached. FChT-LS begins with the same STD as aDFT-LS but, as the additive noise increases, its results are similar to the ones produced by LS-LS, while aDFT-LS has a slightly higher STD than the other two methods, below 18 Hz STD of noise. However, still for the same row, for the higher values of additive noise (above 18 Hz STD), LS-LS and aDFT-LS have a smaller STD than FChT-LS. Finally, in the amplitude and phase errors, rows five and six respectively, it can be observed that FChT-LS has better results for the lower values of noise, while the results of aDFT-LS and LS-LS become better than those of FChT-LS as the noise increases. These two methods, aDFT-LS and LS-LS, have very similar results to each other. The behavior of FChT-LS in the higher values of the additive error can be contributed to its linear frequency basis. The more additive noise there is in the input f_0 curve, the harder it becomes for FChT-LS to find linear trajectories that can follow the adaptations of the f_0 values. On the other hand, this is not the case for LS-LS and aDFT-LS that are fully adaptive.

2) Final Sinusoidal Parameters Estimation: LS Solution vs. Peak Picking (aDFT-aDFT vs. aDFT-LS vs. FChT-FChT vs. FChT-LS): The results shown in Fig. 6 can be studied in order to better understand the influence of the method used in the *Final Sinusoidal Parameters Estimation* step of the analysis process. For this test, either aDFT or FChT was used for the *Refinement of* f_0 step, while all three methods (LS, aDFT, FChT) were combined with them, as shown in Table I, for the *Final Sinusoidal Parameters Estimation* step. It can be observed that, when using a Peak Picking method in the *Final Sinusoidal Parameters Estimation* step instead of the LS solution, the results of the parameter estimation are not always the best. In the first row, displaying the frequency mean error, it can be noticed that, in high frequencies, both aDFT-aDFT and FChT-FChT present an erroneous behavior, especially the latter with a mean error over 20 Hz in most of the cases. Another great deviation for FChT-FChT from the results of the rest of the methods can be observed in the phase error in third row. There, both in low and high frequencies, FChT-FChT demonstrates a highly erroneous behavior, having the highest error estimated in all four methods. In concern to the STD of the parameters estimation error, aDFT-aDFT has either almost the same or better results than aDFT-LS, while FChT-FChT experiences some further difficulties. Namely, in the fourth row, the STD of the frequency error is almost the same for aDFT-aDFT and aDFT-LS while FChT-FChT has the worst results out of all four of them. In the fifth row, the amplitude error of aDFT-aDFT is the smallest one. Finally, in the last row, the phase error of aDFT-aDFT is the smallest out of all four methods in low frequencies and almost the same as aDFT-LS in higher frequencies. The good results produced by aDFT-aDFT are due to the PP which always catches the summit of the peaks, whereas LS can miss the peaks leading to higher amplitude and phase errors.

C. Signal-to-Reconstruction Error Ratio (SRER)

The segmental Signal-to-Reconstruction Error Ratio (SRER) between the recorded utterances and their models was computed using equation 23 in order to evaluate the global reconstruction accuracy of the suggested methods. The SRER between an original signal $s(t)$ and its reconstruction $\hat{s}(t)$ can be written as

$$SRER = 20\log_{10}\left(\frac{\sigma_{s(t)}}{\sigma_{(s(t)-\hat{s}(t))}}\right) \tag{23}$$

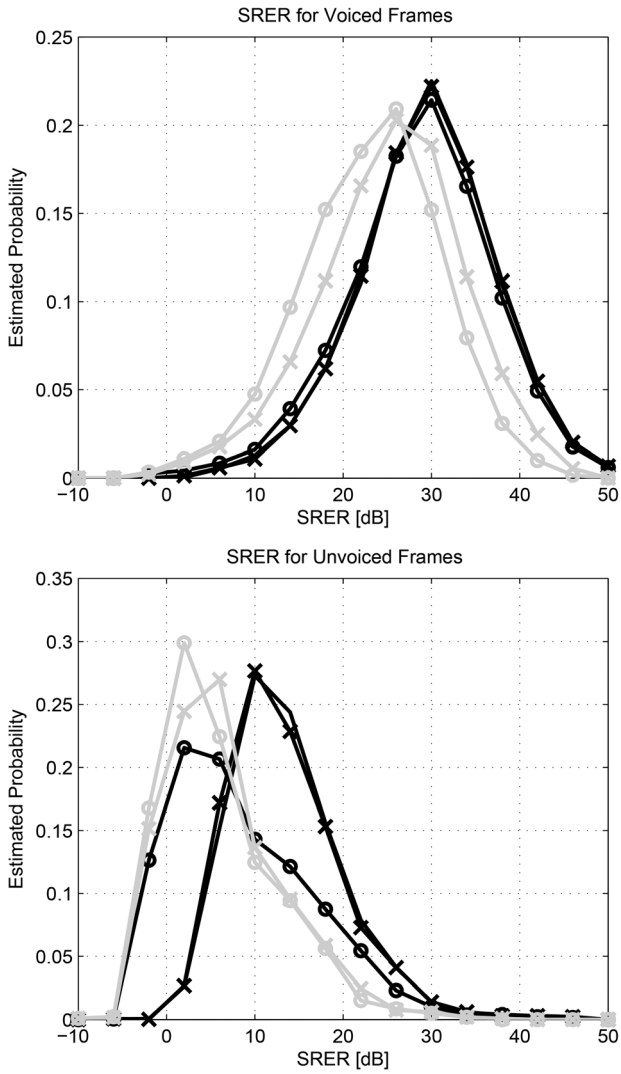

Fig. 7. Estimation of the full-band SRER distributions for voiced and unvoiced frames.

where $\sigma_{s(t}$ denotes the standard deviation of a signal $s(t)$. It can also be observed that the results of SRER are converted into decibel (dB). The higher the result of the above equation the better similarity $\hat{s}(t)$ has to the original signal $s(t)$.

A sliding window of 10 ms with 50% overlap was used. In order to evaluate both the impact of the AIR algorithm, which refines the fundamental frequency, and the best method to compute the final sinusoidal parameters used for the synthesis, all five previously mentioned methods are compared. The SRER was computed using the full-band of the recordings and its distribution of the voiced and unvoiced segments is shown on the top and bottom plots of Fig. 7, respectively. The sole 32 sentences were sufficient to obtain more than 10,000 values for each distribution.

It can be observed that the distributions of all three methods using the LS solution in the *Final Sinusoidal Parameters Estimation* step are very similar to each other for voiced segments. This means that the reconstruction quality is not degraded by the refinement method and, as was shown in Section V-A, the computation load has a considerable decrease with Peak Picking on aDFT. On the other hand, both FChT-FChT and aDFT-aDFT present some issues with both voiced and unvoiced frames which can be explained by the higher frequency errors when not using the LS solution. It is interesting to notice the behavior of FChT-LS in the unvoiced segments, where a smaller SRER is observed compared to the other two methods using LS in the *Final Sinusoidal Parameters Estimation* step. This is due to the fact that the frequency basis in FChT is constrained to linear trajectories and does not have the flexibility of the fully adaptive basis of aDFT.

D. Perceived Quality: Listening Test and PESQ

In this part of the evaluations, the perceived quality of the reconstructed signals using the five methods was evaluated subjectively and objectively, using listening tests and the PESQ method [25] respectively.

1) Subjective Perceptual Evaluation: The purpose of this listening test is to evaluate the methods which are used for both steps of the analysis process (i.e. *Refinement of* f_0 and *Final Sinusoidal Parameters Estimation*). In order to do so, the same 32 utterances of 16 different languages as in 5.2 were used. Listeners were asked to evaluate the quality of sound files compared to an original recording using a web interface. Among the six files they had to rate, five of them were synthesized with LS-LS, aDFT-aDFT, aDFT-LS, FChT-FChT and FChT-LS, while the sixth file was the original recording, which was added to the comparison set in order to check the consistency of the answers. In this test, each listener was asked to grade only 3 languages randomly selected from the 16 languages. Each language was represented by one male and one female voice, hence, each listener evaluated the resynthesis of $(6 \text{ recordings}) \times (\text{the } 6 \text{ different methods})$. The following grading scale of quality was used: (5)Excellent,(4)Good, (3)Fair, (2)Poor and (1)Bad. In order to optimize the results of the listening test, the listeners were asked what device they used to listen to the signals, and only the answers from listeners who used headphones or earphones were kept. Moreover, answers by listeners who did not rate the original recordings systematically between 4 and 5 were discarded, considering that they did not understand the instructions or they were not focused enough. After all the above answers were discarded, 24 remained. Since the sounds to evaluate were selected randomly, the number of occurrences of each sound was not uniform. In order to remove any possible bias, the mean and confidence intervals of the results were normalized according to the number of occurrence of each sound. Fig. 8 shows the results of this listening test.

According to Fig. 8, it can be noticed that only three methods have a global score under 4, aDFT-aDFT, FChT-FChT and FChT-LS. This is caused by the fact that all three methods cannot adapt adequately enough to the unvoiced parts of a signal, as shown in Fig. 7, hence creating artifacts in the resynthesis. On the other hand, aDFT-LS and LS-LS have very similar overall scores, very close to the results of the Original signal.

2) Objective Perceptual Evaluation of Speech Quality Using PESQ: It is expected that, since the results of SRER for the LS-LS and aDFT-LS methods are very similar, an objective measure of perception would give the same results. In order

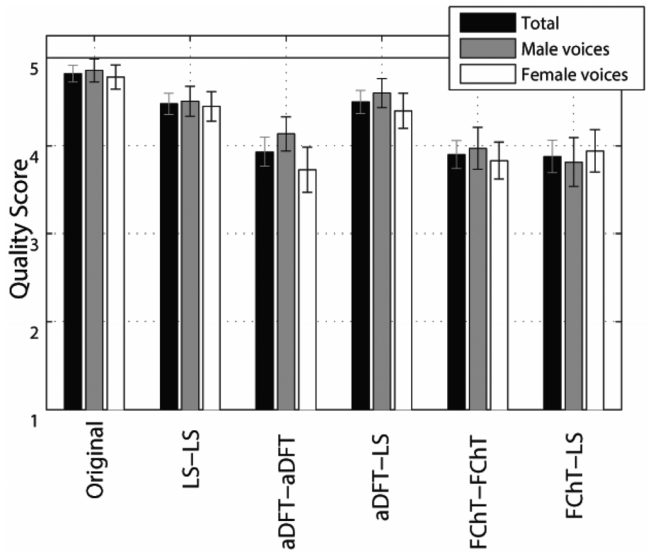

Fig. 8. Quality evaluation of the resynthesis quality by 24 listeners using 32 utterances of 16 different languages, with 95% confidence intervals.

TABLE IV
PESQ SCORES ASSESSING THE OVERALL QUALITY OF THE RE-SYNTHESIZED SIGNALS OF THE METHODS COMPARED TO THE ORIGINAL SIGNAL

PESQ Ratings (up to 4.5)	
LS -LS	4.18
aDFT - aDFT	3.92
aDFT - LS	4.15
FChT - FChT	3.73
FChT - LS	3.82

to verify this, the PESQ method [25] is used to assess the perceived quality of the reconstructed signals compared to the originals. Table IV presents the PESQ scores for the five methods of Table I, using the same database as in the previous tests. Due to the fact that the sampling frequency for the signals in the database varied from 16 kHz to 44 kHz, a re-sampling of all signals to 16 kHz was performed in order for the PESQ measurement to be used. The results show that the LS solution has the best PESQ score with aDFT-LS being a close second. On the other hand, FChT-LS and FChT-FChT have the worst results of them all, with aDFT-aDFT being a little better than them, as is expected from the SRER and listening test results.

VI. CONCLUSIONS

Taking advantage of the good perceived quality provided by aHM-AIR, a Peak Picking approach was suggested in a previous publication to replace the LS solution for the f_0 refinement, in order to reduce the computational time of the AIR algorithm. In this article we extend the previous proposal with a comprehensive study of the behavior of our suggested Peak Picking approach for the whole analysis process of a speech signal. Two different transforms were used for Peak Picking, namely FChT and aDFT. Evaluations have shown that by performing this substitution, the computational load of the AIR algorithm decreases, in average, by a factor of 7.69 and 4, for FChT and aDFT respectively. Moreover, using synthetic signals, the accuracy and precision of the parameter estimation of all versions of aHM-AIR was evaluated showing that the results of aDFT-LS are almost as robust as those of the original method, LS-LS, while all other methods experienced problems. Also, a listening test was carried out in order to assess the subjective perceived quality provided by the suggested analysis/synthesis procedure. According to this listening test, the resynthesis of aHM-AIR using Peak Picking and aDFT for the f_0 refinement and LS for the final sinusoidal estimation (aDFT-LS), has globally the same high quality as aHM-AIR using the LS solution, which is also confirmed by an objective measurement. Therefore, a Peak Picking approach can indeed replace the original LS solution approach of aHM-AIR, while reducing the computational load by four times and preserving the high quality.

REFERENCES

[1] Y. Stylianou, *Modeling speech based on harmonic plus noise models*. Berlin/Heidelberg, Germany: Springer, 2005, pp. 244–260.
[2] L. Almeida and J. Tribolet, "Harmonic coding: A low bit-rate, good-quality speech coding technique," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '82)*, 1982, pp. 1664–1667.
[3] Y. Stylianou, "Applying the harmonic plus noise model in concatenative speech synthesis," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 1, pp. 21–29, Jan. 2001.
[4] G. Kafentzis, G. Degottex, O. Rosec, and Y. Stylianou, "Time-scale modifications based on a full-band adaptive harmonic model," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Vancouver, BC, Canada, Aug. 2013, pp. 8193–8197.
[5] J. Jensen and J. Hansen, "Speech enhancement using a constrained iterative sinusoidal model," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 7, pp. 731–740, Oct. 2001.
[6] Y. Hu and P. C. Loizou, "On the importance of preserving the harmonics and neighboring partials prior to vocoder processing: Implications for cochlear implants," *J. Acoust. Soc. Amer.*, vol. 127, no. 1, p. 427434, 2010.
[7] M. Campedel-Oudot, O. Cappe, and E. Moulines, "Estimation of the spectral envelope of voiced sounds using a penalized likelihood approach," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 5, pp. 469–481, Jul. 2001.
[8] G. Degottex, A. Roebel, and X. Rodet, "Phase minimization for glottal model estimation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 5, pp. 1080–1090, Jul. 2011.
[9] Y. Stylianou, "Harmonic plus noise models for speech, combined with statistical methods, for speech and speaker modification," Ph.D dissertation, E.N.S.T, Paris, France, 1996.
[10] D. W. Griffin and J. S. Lim, "Multiband excitation vocoder," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 8, pp. 1223–1235, Oct. 1988.
[11] X. Serra, "A system for sound analysis, transformation, synthesis based on a deterministic plus stochastic decomposition," Ph.D. dissertation, Stanford Univ., Stanford, CA, USA, 1989.
[12] S. Kim and M. Hahn, "Two-band excitation for HMM-based speech synthesis," *IEICE Trans. Inf. Syst.*, pp. 378–381, 2007.
[13] G. Fant, "The LF-model revisited. transformations and frequency domain analysis," *STL-QPSR*, vol. 36, pp. 119–156, 1995.
[14] B. Doval, C. D'Alessandro, and N. Henrich, "The spectrum of glottal flow model," *Acta Acustica*, vol. 92, pp. 1009–1025, 2006.
[15] G. Degottex and Y. Stylianou, "Analysis and synthesis of speech using an adaptive full-band harmonic model," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 10, pp. 2085–2095, Oct. 2013.
[16] M. Kepesi and L. Weruaga, "Adaptive chirp-based time-frequency analysis of speech," *Speech Commun.*, vol. 48, pp. 474–492, 2006.
[17] Y. Pantazis, G. Tzedakis, O. Rosec, and Y. Stylianou, "Analysis/synthesis of speech based on an adaptive quasi-harmonic plus noise model," in *Proc. IEEE ICASSP*, Dallas, TX, USA, Mar. 2010.
[18] V. Morfi, G. Degottex, and A. Mouchtaris, "A computationally efficient refinement of the fundamental frequency estimate for the adaptive harmonic model," in *Proc. IEEE ICASSP*, Florence, Italy, May 2014.

[19] R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 4, pp. 744–754, Aug. 1986.
[20] Y. Pantazis, O. Rosec, and Y. Stylianou, "Adaptive AM-FM signal decomposition with application to speech analysis," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 2, pp. 290–300, Feb. 2011.
[21] A. Camacho, "Swipe: A sawtooth waveform inspired pitch estimator for speech and music," Ph.D. dissertation, Univ. of Florida, Gainesville, FL, USA, 2007.
[22] D. B. Paul, "The spectral envelope estimation vocoder," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 4, pp. 786–794, Aug. 1981.
[23] M. Zivanovic, A. Roebel, and X. Rodet, "Adaptive threshold determination for spectral peak classification," *Comput. Music J.*, vol. 32, no. 2, pp. 57–67, 2008.
[24] S. Maeda, "A digital simulation method of the vocal-tract system," *Speech Commun.*, vol. 1, pp. 199–229, 1982.
[25] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP '01)*, 2001., vol. 2, pp. 749–752.

Veronica Morfi received the Diploma degree in computer science from the University of Crete, Greece, in 2012. She is currently an M.Sc. student in the Computer Science Department at the University of Crete. Her research interests include speech and audio processing, pattern recognition, machine intelligence, statistical signal processing.

Gilles Degottex received the Diploma degree in computer science from University of Neuchâtel (UniNE), Switzerland. After a one-year specialization at École Polytechnique Fédérale de Lausanne (EPFL), Switzerland, he obtained his Ph.D. degree in 2010 at the Institut de Recherche et Coordination Acoustique/Musique (Ircam), Université Pierre et Marie Curie (UPMC), Paris, France. He is currently holding a postdoctoral position regarding voice modeling at University of Crete and FORTH, Heraklion, Greece. His research interests include voice source features, wide and narrow band voice models for speech synthesis.

Athanasios Mouchtaris (S'02–M'04) received the Diploma degree in electrical engineering from Aristotle University of Thessaloniki, in 1997 and the M.S. and Ph.D. degrees from the University of Southern California (USC), Los Angeles, in 1999 and 2003, respectively. Since 2004, he has been affiliated with the Computer Science Department of the University of Crete, where currently he is an Associate Professor, and with the Institute of Computer Science of the Foundation for Research and Technology Hellas (FORTH-ICS), Heraklion, Crete, as an Affiliated Researcher. From 2003 to 2004, he was a Postdoctoral Researcher in the Electrical and Systems Engineering Department of the University of Pennsylvania, Philadelphia. His research interests include audio and speech signal processing with emphasis on sound localization, acoustic sensor networks, microphone arrays, immersive audio environments, spatial and multichannel audio, parametric audio/speech modeling, sparse representations and compressed sensing for audio/speech signals, voice conversion, speaker identification, and speech enhancement. He has authored or co-authored more than 80 publications in various journal and conference proceedings in these areas. He has received grants from the European Commission and the Greek General Secretariat of Research and Technology. Dr. Mouchtaris is a member of IEEE.