A nonlinear oscillator model with physically meaningful parameters for robust pitch detection in noise

Irina Gorodnitsky

Department of Cognitive Science, University of California, San Diego, 9500 Gilman Dr., La Jolla, CA 92093-1505, USA

(Dated: Preprint, to appear in 2009)

A nonlinear oscillator model is developed for robust estimation of the fundamental frequency of acoustic signals. The oscillator is formulated as a projection of dynamic information (i.e., the signal derivative) onto a manifold in a state-space, and then implemented as a parametric estimation problem requiring no reference to nonlinear theory. The method combines properties of classical nonlinear oscillators with several distinctive features, including extremely low order models, physically meaningful model parameters, and high robustness to noise. The later two properties are examined in the paper. The relationship of the parameters to frequency is derived first. Then, the robustness to noise is investigated using the TIMIT database with three noise types added to create low signal-to-noise (SNR) conditions. The noise types include stationary white noise, nonstationary speech babble, and strongly deterministic M109 tank noise with power concentrated in the same range as the fundamental frequency in speech. The method is shown to be robust at SNR as low as -5 dB and to perform considerably better than the popular ESPS algorithm in extreme noise (0 dB SNR and below). While its performance deteriorates at -10 dB SNR, the method could still be useful in white and babble noise for applications such as voice activity detection.

PACS numbers: 43.72.Ar, 43.60.Ac, 43.60.Wy, 43.72.Dv

I. INTRODUCTION

Analysis of acoustic signals in noisy and nonstationary environments remains very challenging. Significant effort has been made in recent years to increase robustness of speech analysis methods, but many traditional techniques have inherent limitations that are difficult to overcome. This has motivated interest in development of new, physically motivated methods that are completely different from traditional techniques. In particular, the physical phenomena involved in speech production have been argued to lend themselves naturally to nonlinear modeling approaches, e.g. (Kubin, 1995).

In this domain, nonlinear predictors (Kantz, 1997), of which nonlinear oscillators (NOs) (Kubin et al., 2005) is a special case, were proposed over a decade ago for applications that include coding, synthesis, and modeling (Kubin, 1995). The basic premise of this approach, which we refer to as the ‘classical NO’, is to estimate adaptively a function (model) that predicts future data from the time-delay embedded data vector. Improvement in prediction gains achieved with nonlinear estimators over linear estimators were reported in (Kubin et al., 2005). Another major attraction of NOs is their potential for providing very efficient signal coding, since oscillators require only trivial or no excitation sequence at all.

Unfortunately, the theoretical promises of this NO method have not yet been realized. A multitude of methods for representing the prediction function have been explored, including function approximations using a truncated Volterra series with quadratic filters or trainable nonlinear mapping functions, e.g. a radial basis functions (RBFs)-based neural network (Kubin et al., 2005). To date, the most successful synthesis behavior has been reported for NOs realized using RBF networks (Kubin et al., 2005). A ‘state-of-the-art’ implementation described in (Kubin et al., 2005) incorporates Bayesian Regularization of an RBF network aimed at improving prediction stability and pruning of the initial set of basis functions from several hundreds to 200 or fewer bases. Such numerical strategies have lead to a significant increase in the number of signals that can be re-synthesized stably with this NO. However, the ability of this NO to model full speech signals remains poor (Kubin et al., 2005). Stable re-synthesis was reported for only 18% of the vowels (pure extended vowels, not vowels extracted from natural speech). The vowels on which this method is less likely to succeed, i.e. /a/, /e/, and /i/, have a ‘complicated’ structure in state space, as compared to the vowels /o/ and /u/, which have a ‘simple’ state-space structure. The complexity in this case refers to the ability of an oscillator to ‘unfold’ the signal trajectory in a state space of a given dimension. For an oscillator to unambiguously identify the signal structure in the state space, one must choose an appropriate embedding dimension and embedding delay. Practical considerations, however, preclude estimating these embedding variables for each speech segment. Instead, a fixed embedding delay was used that was experimentally determined to be sufficient for vowels and voiced signals in (Kubin et al., 2005). The embedding dimension N=4 was also fixed, and was chosen based on it being sufficient for certain stationary vowel signals. Unfortunately, such an implementation does not provide consistent results for voiced signals in general, including the vowels with ‘complex’ structures. To enable re-synthesis of such vowels, a low-pass filter has to be employed to remove structures that are due to the influence of the vocal tract on a glottis source signal. Having to employ a low-pass filter is not ideal, since it removes key speaker-related characteristics. Moreover, even with this filter, only a 56% success rate for stable re-synthesis of vowels was obtained (Kubin et al., 2005), which cannot be considered suf-
cient for wide practical use. Processing of non-vowel signals with this NO appears to be even less successful.

In addition to the problem with fixed-point stability, classical NOs suffer from at least two other serious limitations that can be seen as inherent in the overall approach. The first is the large parameter space, on the order of several hundreds of parameters, that is needed for successful prediction. The advanced pruning of the basis done in (Kubin et al., 2005) still leads to a space containing 100-200 parameters. This tends to negate at a great degree the coding efficiency gained by using only trivial excitation sequences in NOs. The other drawback is the lack of any obvious relationship between the classical NO model parameters and the physical properties of speech signals. This undermines the original inspiration for the movement toward the physically motivated models.

A notably different state-space method was proposed in the context of pitch detection in (Terez, 2002). The method does not attempt to model signal structure. Rather it uses a variant of the nearest neighbors procedure as a way to observe the repetitive cycles in a signal. This methodology exploits the basic property of state-space representations that reveals nearly repetitive structure in data as a set of closely spaced trajectories. The method in (Terez, 2002) uses a two-delay model (embedding dimension of 3) and a constant delay, chosen to be 12 for the 16 KHz sampling rate. While a number of claims were made in (Terez, 2002), including superior robustness, they were not supported. Considerable doubts remain about the efficiency of the method and how the fixed parameter set demonstrated for one vowel in (Terez, 2002) would work for speech overall.

The current paper presents a novel nonlinear oscillator model for speech analysis whose properties have the potential to remedy some of the drawbacks of the existing approaches. One novel feature of the method is that it models the dynamics (first derivative) of the signal rather than the data themselves. The model structure is simply a sum of two time-delayed data sequences. Estimating this model amounts to adaptively estimating the two unknown delays. This novel, and by comparison to the classical NO, extremely simple framework was motivated from several perspectives. Several unrelated sources suggest that inclusion of dynamic features improves performance metrics as compared with methods that rely on static features. For speech, inclusion of time-differentiated spectral features (derived from cepstral coefficients) is known to improve noise robustness in speech and speaker recognition, e.g. (Yang et al., 2007). Yang et al. (2007) showed that this property holds for a wide range of noise levels and different noise types. More generally, studies of time-delay models have found that representations of signal derivatives via time-delay functions are highly robust to noise (Bünn et al., 1997). A second important property revealed in (Bünn et al., 1997) concerned simple scalar models with as few as one time-delay. The models were found to provide a unique low-dimensional projection of the data dynamics, with no restriction placed on the dimensionality of the dynamical system being described. Yet a third interesting observation in (Bünn et al., 1997) was that such low dimensional projection revealed nonlocal correlations in data.

The three properties observed in (Bünn et al., 1997) are potentially very beneficial for speech processing. To emphasize the singular importance of the second property, consider that Takens’ method of delays (Takens, 1981) requires the embedding dimension to be at least double the dimension of the dynamical system in order to avoid ambiguities in the representation. In the discussion of classical NOs above, we borrowed the term ‘complicated’ structure from Kubin et al. (2005) to qualitatively describe signals that are not identified well by the oscillator in the low-dimensional embedding of N=4. One immediate inference that can be drawn from the observation in (Bünn et al., 1997) is that if we model the speech derivative, a simple one-delay equation can unambiguously model any structure regardless of its ‘complexity’. The third observation in (Bünn et al., 1997), that the parameters reveal nonlocal correlations, suggests that they may have a physical meaning. Finally, the observed robustness raises the possibility that these models can be practical for analysis of acoustic signals in noisy and nonstationary environments. Unlike the method in (Terez, 2002), this approach results in an actual oscillator that can be adapted for a variety of uses, including coding, synthesis, and modeling.

Motivated by these prospects, the current paper presents an exploratory study of the derivative-based NO in the context of speech processing. Understanding a new representational framework requires systematic studies of all properties, something that a single paper cannot provide. Instead, the paper focuses on two key aspects of the derived NO, the relation of its parameters to frequency and its robustness to high noise and to various statistical characteristics of noise. The pitch detection problem provides a vital practical application for investigating these properties.

A few theoretical results exist for derivative-based state-space models. Gorodnitsky (2007) showed that one-delay models describe an output of any (up to infinite-dimensional) system of uncoupled linear oscillators, damped and undamped. This result shows that any signal expressed in the Fourier domain as a superposition of sinusoidal components can alternately be modeled (via its derivative) by a weighted one-delay term. In addition, transient signals that are expressed as a superposition of damped and undamped sinusoidal components can similarly be modeled compactly by a one-delay term. While the result in (Gorodnitsky, 2007) constrained the observations in (Bünn et al., 1997) regarding the structure of high-dimensional dynamics that can be modeled by a one-delay model, these constraints are extremely loose. Many canonical data classes one expects to encounter in engineering are described exactly by low-order derivative models.

One immediate outcome of the result in (Gorodnitsky, 2007) is that for many signals the choice of models becomes trivial. Since the model in these cases is exact, errors due to model misspecification are eliminated. Model misspecification can weigh critically on robustness and this suggests a possible factor behind the robustness of this approach. Another property worth noting is an inherent efficiency in the expression of transients, which suggests certain advantages in modeling co-articulation and transitions between voiced and unvoiced speech. Efficient modeling of transients could also be useful in coding and synthesis of expressive speech, for example laughter, which can be modeled as damped harmonics (Sundaram and Narayanan, 2007). The ability to efficiently code and transmit such nonlinguistic information is important for improving the perceptual quality of reproduced speech.

The current paper builds on the theory presented in (Gorod-
The current realization of the algorithm for estimating the ID is denoted by the one-delay derivative-based model. The delay values can be interpreted as encoding timing information, so that a sequence of uniformly sampled observations in the intervals between the timing events. Due to this timing-based coding principle, the representation domain of the time-delay derivative models is termed the Interval Domain (ID) and the models will be referred to here as the ID models. The following notation is adopted throughout the paper. Continuous time-series are denoted by \( x(t) \), while bold letters indicate discrete vector quantities. An individual data sample taken at the current time \( n \) is denoted by \( x(n) \). A sample at an earlier time \( n - \tau, \tau \in \mathbb{T} \), is denoted by \( x(n - \tau) \). In this notation, a sequence of uniformly sampled observations is written as \( x(n) = \{x(n), x(n - \tau), \ldots \} \). A time-delay reconstructed state vector as defined in (Takens, 1981) is

\[
s(n) = \{x(n), x(n - \tau), ..., x(n - M\tau)\}.
\]  

II. THE METHOD

A. ID model: general theory

The following notation is adopted throughout the paper. Continuous time-series are denoted by \( x(t) \), while bold letters indicate discrete vector quantities. An individual data sample taken at the current time \( n \) is denoted by \( x(n) \). A sample at an earlier time \( n - \tau, \tau \in \mathbb{T} \), is denoted by \( x(n - \tau) \). In this notation, a sequence of uniformly sampled observations is written as \( x(n) = \{x(n), x(n - \tau), \ldots \} \). A time-delay reconstructed state vector as defined in (Takens, 1981) is

\[
s(n) = \{x(n), x(n - \tau), ..., x(n - M\tau)\}.
\]  

Conventionally, the elements \( x(n - k\tau) \) in Eq. 1 are assumed to be uniformly spaced. This means, \( \tau \) is a constant, called the normalized embedding delay, and the state vector \( s(n) \) is said to have uniform delays. The classical NO developed in (Kubin, 1995) is defined in this notation as \( x(n) = F(s(n - 1)) \), where \( F(\cdot) \) is a general non-linear function that is estimated from the data.

The alternative time-delay derivative model proposed here is a discretized form of a scalar linear time-delay differential equation. The basic component of the model is a time-delayed vector of data \( x(n - \tau) = \{x(n - \tau), x(n - \tau - 1), x(n - \tau - 2), \ldots, x(n - \tau - l)\} \). To simplify notation, we denote \( x_r = x(n - \tau) \). In this notation, the time-delay derivative model takes the form

\[
\dot{x}(n) = \sum_{i=0}^{K} a_i x_{\tau_i},
\]  

where \( \tau_i \) indicate different and hence, no longer uniform, delay values. Note that \( x_r \) represents time-evolution of the \( (n - \tau) \)th component of the reconstructed state vector \( s(n) \) in Eq. 1. Thus, the classical and the ID oscillator models differ in their formulation, in addition to their difference in utilizing the dynamic versus static features of data. For the pitch detection application, the model with \( K=2 \) is chosen, leading to

\[
\dot{x} = a_1 x_{\tau_1} + a_2 x_{\tau_2}.
\]  

The model estimation task reduces to solving in the Least Squares (LS) sense for the four unknowns in (3).

Gorodnitsky (2007) analyzed two of the simplest, one-delay ID models:

\[
\dot{x} = ax_r \tag{4}
\]

\[
x = a_0 x + a_1 x_r \tag{5}
\]

She showed that the solution spaces of the continuous time-delay equations that correspond to these models contain any waveform that can be written as a sum of sustained and damped oscillations. The general solution form is

\[
x(t) = B_0 e^{\tau_0 t} + B_1 e^{\tau_1 t} + \sum_{k=2}^{\infty} e^{r_k t} (C_k \sin(w_k t + v_k)), \tag{6}
\]

with \( C_k \neq 0 \), one or both coefficients \( B_{0,1} \) assuming non-zero values on the interval defined by the parameters \( \tau \) and \( a \), and \( r_i \) designating the real part of roots of the characteristic equation of (4) (for details see (Gorodnitsky, 2007)).

Eq. (6) implies that models (4) and (5) are appropriate for describing sampled output of a (stable) system containing an arbitrary number of uncoupled linear oscillators. As a side note, Eq. (6) also admits up to two growing exponentials, hence, the models can describe signals generated by certain unstable systems. The results in (Gorodnitsky, 2007) can be extended in a straightforward way to show that the solution set described in Eq. (6) forms part of the solution set of any time-delay equation if their discretized form is represented by Eq. (2). This means that for the purpose of providing a good fit to oscillatory signals, the choice of the model is inconsequential. Having an ‘exact’ model eliminates error due to model mis-specification, one of the two main sources of error in parameter estimation. It should be noted that efficient encoding of
multiple frequencies is somewhat opposite of efficient spectral factorization. The compact ID models obscure spectral information other than \( F_0 \). Specifically, while ID models with more than two delays may be defined by the overall spectral content of a signal, the relationship between the two sets of parameters is not straightforward.

Kadtke (1995) arrived at a model similar to Eq. (2), but containing nonlinear terms, by considering the problem of classifying dynamical systems from noisy scalar observations in state space. Using a truncated Volterra series to realize the model, Kadtke (1995) reasoned that a few leading terms of this series, sans the constant term, would be sufficient for the purposes of classification. A typical model choice in (Kadtke, 1995, and the follow-up papers) contains a sum of two linear delay terms and a product of these two terms. It is clear from the numerical experiments shown in the referenced papers, however, that model delays were not adaptively estimated. Only the weights of the delay terms were estimated adaptively and then used for classification. Model delays, on the other hand, were assumed constant for the entire data vector and estimated rather approximately. In fact, it was reasoned in the papers that identical delay values should be used to model distinct dynamical systems to facilitate the classification process. It should be stressed that except in special cases of stationary signals when the chosen delays happen to match the optimal values in (2), this procedure is not equivalent to the method proposed here. In particular, the main property of the low-order ID models, namely exact description of oscillatory systems of arbitrary dimension, does not carry over to the procedure used in (Kadtke, 1995, and the follow-up papers). This could explain why the papers employed higher complexity models with a nonlinear term in analysis of even basic oscillatory functions. Another key property not realized in the earlier works is the physical relevance of the delays in models such as Eq. 3, which reveal nonlocal correlations in the data. This property is discussed in detail in the next subsection.

B. \( F_0 \) estimation

Continuous speech is highly nonstationary and consists of sequences of oscillatory states, typically abruptly, by non-oscillatory noise-like states. The oscillatory parts are not uniform - some are sustained, but others are transient, for example the parts corresponding to co-articulations. In order to apply ID modeling to pitch detection we first need to determine which ID model from the set in Eq. (2) is best suited for modeling \( F_0 \) of signals composed of sustained and transient oscillations. The second task is to find the analytical expression that relates parameters of the selected model and \( F_0 \).

We first consider the simplest model (4). Based on Eq. (6), this model can be applied to the voiced parts of speech that are composed of sums of sustained and transient harmonics. Let’s consider the relationship between the delay \( \tau \) in (4) and the frequency \( f = w/2\pi \) of a pure tone, which we write as a sustained sinusoid \( x(t) = A \sin(\omega t + \phi) \). Substituting the sinusoid into Eq. (4) we get

\[
\frac{\omega}{a_1} \cos(\omega t + \phi) = \sin(w(t - \tau) + \phi). \tag{7}
\]

The only possible choice for which this equality holds is:

\[
\tau_n = \frac{(2n - 1)\pi}{2\omega}, \quad n \in I^+, \tag{8}
\]

with the corresponding coefficient \( a = -\omega \) for odd \( n \) and \( a = \omega \) otherwise. Hence, the multiple values of \( \tau_n \) identify periodic structure of the sinusoid and are inversely related to the frequency harmonics. One caveat here is that \( \tau_n \) are integer valued. Hence, the actual correspondence between \( \tau \) and \( w \) is \( \tau_n = int\{\frac{(2n-1)\pi}{2\omega}\}, n \in I^+, \) where \( int\{\cdot\} \) designates the closest integer to the quantity \{\cdot\}. The smallest value \( \tau = \frac{\pi}{4\omega} = \frac{1}{4\tau} \), corresponds to a quarter wavelength and is inversely proportional to four times the fundamental frequency or the 4th harmonic. For consistency, we refer to \( \tau_f = \frac{1}{4\tau} \) as the fundamental delay. Note that unlike \( \tau \), the amplitude of the coefficient \( |a| = \omega \) unambiguously identified the angular frequency in the pure tone case.

Next, we consider the relationship between \( \tau \) and frequency in the case of an exponentially damped or growing sinusoid \( x(t) = A e^{\lambda t} \). By expanding \( \lambda = r + jw \) and substituting the resulting waveform into Eq. (4), we obtain the following transcendental equation

\[
(r^2 - \omega^2) \cos(w\tau) = aw^2. \tag{9}
\]

Finding \( \omega \) involves solving this transcendental equation, which is not straightforward. Moreover, similarly nontrivial relationship between \( \tau \) and \( F_0 \) can be shown to exist when the signal contains more than one oscillatory component (omitted in the interest of space). For these reasons, one-delay models (4) and (5) are impractical for detecting pitch of speech. This conclusion was confirmed through numerical experiments which found that \( F_0 \) can be estimated correctly in approximately 75% of the voiced speech frames when we use \( \tau_f \) from model (4) and calculate \( F_0 = 1/\tau_f \). Another potential consideration that weighs against one-delay models is that they would not capture nonlinear behavior in data, for example quadratic phase coupling, which has been hypothesized to exist in voiced speech (McLaughlin et al., 1994).

The next step up in complexity from Eq. (4) is the two-delay model in Eq. (3). This model was verified numerically to provide readily extractable \( F_0 \) information for any linear or nonlinear signal that contains distinct periodic or ‘close to’ periodic structure. The model was also found to provide a good representation for voiced speech. Good in the present context refers to the LS residual not decreasing with further increases in model order. Models with more than two delays lead to an incrementally smaller LS residual due to the ability of these delays to better approximate the non-integer value of \( F_0 \).

To understand how \( F_0 \) is encoded by the parameters of model (3), let’s consider the numerical form of this equation used in the IDEA implementation where we compute the derivative at each point in \( \dot{x}(n) \) using the central difference formula \( \dot{x}(n) \approx (x_{n+1} - x_{n-1})/2 \). With this, the model (3) becomes

\[
\frac{x_{n+1} - x_{n-1}}{2} = a_1 x_{\tau_1} + a_2 x_{\tau_2}. \tag{10}
\]

The two sides are equal when \( a_1 = -a_2 = 0.5, x_{n+1} = x_{\tau_2} (\tau_2 = -1), \) and \( x_{n-1} = x_{\tau_1} (\tau_1 = 1) \). Note that from
here on, we will assume the convention $\tau_1 > \tau_2$. The given solution is purely theoretical since ID search space contains only past values of data. For a strictly periodic sequence, which is invariant under a shift in time that is equal to its period, a suitable substitute for $x_{n+1}$ is $x_{n+1} = x_{n+1 - f_s/F_0}$, where $f_s$ denotes the sampling rate and the subscript $n+1 - f_s/F_0$ indicates a shift by a single period $f_s/F_0$ from the time point $n+1$. Therefore, one of the solutions to Eq. (10) is given by $a_1 = -a_2 = -0.5$, $\tau_1 = 1$, $\tau_2 = f_s/F_0 - 1$. Numerical experiments confirm that IDEA, in fact, selects this particular solution when a signal is strictly periodic. As discussed below, IDEA also limits the search space to avoid selecting $\tau_2$ corresponding to $F_0$ harmonics. $F_0$ in this case is found as

$$F_0 = f_s/\left(\tau_2 + 1\right). \quad (11)$$

Note that the approach accommodates various waveform morphologies within a periodic cycle, including nonlinear features. Clearly, other solutions to Eq. (10) exist. In particular, if a signal is also $f_s/2F_0$-anti peri odic, where $f_s/2F_0$ is half of the period, then $x_{n+1} = -x_{n+1 - f_s/2F_0}$. A set of delays, $\tau_1 = 1$, $\tau_2 = f_s/2F_0 - 1$, provides an alternate model in this case. The corresponding estimate $F_0 = f_s/\left(\tau_2 + 1\right)$ results in a pitch halving error. Moreover, for phonemes that are nearly stationary over multiple periods, $\tau_2$ values corresponding to the $F_0$ harmonics can give rise to a similar LS residual, hence choosing one of these $\tau_2$ values would result in a harmonic error. Harmonic and sub-harmonic errors present a well known problem in pitch detection. To minimize occurrences of these errors, pitch detection algorithms (PDAs) use a bounded search range for $F_0$. We adopt a similar approach here, which will be discussed in Subsection II.D. However, we also note that numerical experiments with model (3) using continuous speech samples drawn from several databases did not generate halving errors. The reason for this robustness is not entirely clear. Asymmetry in small scale features in otherwise oscillatory voiced speech may help robustness of ID models in this respect.

Another well known problem that is encountered with some popular PDAs, such as the autocorrelation (AC) method, is sensitivity to amplitude changes in the signal over time. The changes cause period-to-period dissimilarity which can result in “too high” and “too low” errors. The coefficients $a_{1,2}$ in the ID model (3) can compensate for certain amplitude changes, for example, gradually damped or growing oscillations, making the ID model immune to this problem. We find numerically that in such cases, parameter values can deviate from $1$ to $10$ points from the nominal solution derived above for steady periodic signals. Overall, for noise-free, voiced speech signals, we find that model (3) returns $\tau_1 = 1$ in $\sim 82\%$ of the frames and $1 < \tau_1 < 5$ in $\sim 95\%$ of the frames (the percentages may vary slightly depending on the frame length). $\tau_2$ values correspondingly shift by a few points from the nominal solution $\tau_2 = f_s/F_0 - 1$. Overall, the difference between the two delays $\tau_2 - \tau_1$ in these cases still closely approximates the period length in the signal (minus 2 time points). Hence, the more general formula for fundamental frequency estimation that accommodates transient harmonics becomes $F_0 = f_s/\left(\tau_2 - \tau_1 + 2\right)$.  

C. Cost function optimized with the two-delay ID model

Solving in the LS sense for parameters of Eq. (10) minimizes the residual

$$e(\tau_1, \tau_2) = \sum_{j=-W}^{n} \left( a_1 x_j - \tau_1 + a_2 x_{j-\tau_2} - x_{j+1} - x_{j+1} \right)^2 = \|a_1 x_\tau_1 + a_2 x_\tau_2\|^2 . \quad (12)$$

where $\Delta x_\tau$ denotes the direction vector defined by the data derivative. Note that throughout the paper, all cost functions will be represented using the causal notation employed in Eq. (12).

From the derivation in the previous subsection, when $x$ is a purely periodic sequence, model (3) returns $\tau_1 = 1$, $a_1 = 0.5$. The LS residual in this case simplifies to

$$e(\tau) = \|x_n - a_\tau x\|^2 . \quad (13)$$

This cost function is analogous to the difference function $d(\tau)$ that forms the basis for the YIN algorithm (Cheveigne and Kawahara, 2002):

$$d(\tau) = \|x_n - x_\tau\|^2 . \quad (14)$$

The entire YIN method involves five steps (presented in six steps in (Cheveigne and Kawahara, 2002), but Step 1 is simply a discussion of the AC function (Rabiner and Schafer, 1978)). In Step 2 the AC function is replaced with $d(\tau)$. This is shown to reduce the occurrence of harmonic and sub-harmonic errors in the $F_0$ estimate. Steps 3-6 aim at further reducing these errors. Since the multi-step process in (Cheveigne and Kawahara, 2002) results in modifications to $d(\tau)$, it is difficult to draw a precise comparison between YIN and IDEA even for purely periodic signals. The most notable improvement in performance of YIN over AC, however, according to Cheveigne and Kawahara (2002), was due to Step 2, that is, the move from the AC function to $d(\tau)$. The second notable improvement was due to Step 4, which implements a threshold for selecting the solution aimed at minimizing subharmonic errors. However, since model (3) appears to be robust to subharmonic errors, the modifications in Step 4 move YIN closer to IDEA in their robustness properties. Hence, we conclude that for the $\sim 80\%$ of the voiced speech frames in which the cost being optimized with model (3) is effectively $d(\tau)$, the two methods draw on a very similar optimization criterion.

The question remains about the other $\sim 20\%$ of the frames where $e(\tau_1, \tau_2) \neq d(\tau)$. The addition of the direction vector $e(\tau_1, \tau_2)$ significantly alters the cost being optimized. To support this statement, let’s first consider the AC function

$$r(\tau) = \sum_{j=-W}^{n} x_j x_{j-\tau} = \|x_n\| \|x_\tau\| \cos(\theta_n, \tau) , \quad (15)$$

where $\cos(\theta_n, \tau)$ represents the angle between the vectors $x_n$ and $x_\tau$. The difference function, expressed in similar form, becomes

$$d(\tau) = \|x_n\|^2 + \|x_\tau\|^2 - 2\|x_n\| \|x_\tau\| \cos(\theta_n, \tau) . \quad (16)$$

The only essential difference between $r(\tau)$ and $d(\tau)$, besides the reversal of maxima to minima, is the contribution of the

Nonlinear oscillator for robust pitch detection 5
energy term \(\|x_n\|^2\), which can vary with \(\tau\) when a signal is not purely periodic or stationary. By comparison, the cost optimized by model (3) is

\[
e(\tau_1, \tau_2) = \|a_1 x_{\tau_1}\|^2 + \|a_2 x_{\tau_2}\|^2 + \|\Delta x_n\|^2 + 2|a_1 x_{\tau_1}|\|a_2 x_{\tau_2}\| |\cos(\theta_{\tau_1}, \Delta x_n)| - \|a_1 x_{\tau_1}\| \|\Delta x_n\| |\cos(\theta_{\tau_1}, \Delta x_n)| \cos(\theta_{\tau_2}, \Delta x_n) + 1)
\]

This cost involves considerably more degrees of freedom than either \(r(\tau)\) or \(d(\tau)\). The fact that \(e(\tau_1, \tau_2) \neq d(\tau)\) in over 20\% of the voiced speech frames indicates that we cannot simplify the parameter estimation problem by assuming \(x_{\tau_i} \approx x(n - 1)\), which would reduce our method to optimizing the \(d(\tau)\) function of YIN. Moreover, such a simplified scheme would not have the flexibility of model (3) in the presence of noise. Noise was found to cause much larger deviations from the nominal values in both \(\tau_i\) than the deviations observed in noise-free cases. Hence, to take advantage of the ID robustness, it appears that a model incorporating both delays, \(\tau_1\) and \(\tau_2\), is essential.

Another relevant comparison can be drawn between IDEA and nearest neighbor approach. A method for pitch estimation based on computing time separation of nearest neighbors in state space was proposed in (Terez, 2002). The time separation values are computed for neighborhoods of different radii and delay vector sequences \(x_{\tau_i}\) with \(\tau_j\) ranging over the entire sample sequence. The results are accumulated in a normalized periodicity histogram, which exhibits peaks at \(F_0\), and its harmonics and subharmonics. The time separation between nearest neighbor points computed for a range of time-delayed sequences is actually the difference function of YIN, i.e., \(\|x_n - x_{\tau_i}\|^2\). Thus the nearest neighbor method can be considered as a recasting of the \(d(\tau)\) optimization cost of YIN in a state space paradigm. This is not to say that the methods are the same. Clearly, being able to base the cost measure on points distributed within some neighborhood provides a robust way of dealing with aperiodic features arising from perturbations around the exact period. Apart from such obvious flexibility afforded by the state space framework, however, the decision mechanisms in the nearest neighbor method are constrained by the distance measure analogous to the one underlying YIN.

The complexity of \(e(\tau_1, \tau_2)\) relative to the costs optimized in the AC, YIN, and nearest neighbor methods does not necessarily imply that model (3) provides a more robust PDA. The term robustness is used broadly in this field and can refer to performance in noise-free conditions, i.e., fine and gross errors in \(F_0\) estimation, or to robustness to noise. Numerical evaluations show that model (3) provides moderate resolution in a noise-free environment. As in the case of time-domain PDAs, such as AC and YIN, its accuracy is limited by the delays being integer valued, with the precision dependent on the sampling frequency. Harmonic and subharmonic errors appear to be negligible in our implementation. However, IDEA produces occasional errors in \(F_0\) estimates in noise-free environments that are unrelated to harmonics or subharmonics. The rate of the occurrence of this error appears to be \(\sim 1 - 2\%\). The origin of the error is not immediately clear and could be due to the particular numerical implementation of the method that is adopted here. Tracing the source of this error would require an extensive study beyond the scope of the current presentation. Thus, for the purposes of this paper, we accept the present level of noise-free performance. The question that is of interest here is how performance is maintained in severe noise conditions. As shown in the Section III, IDEA maintains robust performance even in negative SNR, where higher precision techniques are known to fail. Such robustness appears to be a key property of this time-delay derivative method.

D. Implementation Issues

A linear algebraic approach to solving Eq. (3) is presented. Alternate approaches, such as global searches, could also be used. A significant advantage of the linear algebraic approach, as opposed to global searches, is that regularization can be built into the cost function to reduce the effect of noise. This can appreciably enhance estimator robustness in noise.

The sparse optimization framework (Gorodnitsky and Rao, 1997) is proposed here to solve Eq. (3). In this framework, we define an overcomplete dictionary of bases formed from past observations, namely vectors \(\{x_n\}, 1 \leq i < L\). \(L\) defines the search range as discussed below. Let \(D = [\ldots x_{\tau_i}, \ldots]\) denote the matrix of dictionary bases. The parameter estimation problem can then be written as

\[
\text{minimize } \text{card}\{a\}, \quad \text{subject to } Da = x, \quad (19)
\]

where \(\text{card}\{\cdot\}\) denotes the cardinality of a vector, that is the number of its nonzero elements. The nonzero entries in the solution \(a\) correspond to the most effective minimal set of bases which combine linearly to match \(x\). The selected bases provide us the delays \(\tau_i\) and the nonzero entries of \(a\) correspond to the coefficient values \(a_i\) in (3).

The sparse optimization framework has one weakness: the methods are free to select the number of bases that make up the solution and this, in turn, dictates the model order being estimated. In the case of ID estimation, a sparse optimization algorithm could select for certain speech signals a model like one-delay Eq. (4) in place of model (3). This variability is undesirable and we would lose the straightforward relationship between the delays and \(F_0\) with models other than (3). The problem can be circumvented by extending the above optimization framework to multiple frames of data and by choosing an algorithm whose convergence depends on initialization.

One computationally simple algorithm which fits the initialization criteria is FOCUSS (Gorodnitsky and Rao, 1997) which can be extended to multiple frames. To solve Eq. (3) using the Multiple Frame extension of FOCUSS (MFF), we use consecutive measurement vectors, namely the derivatives of three data sequences, to form the columns of the measurement matrix: \(\tilde{X} = [\tilde{x}(n) \tilde{x}(n - 1) \tilde{x}(n - 2)]\). The solution vector \(a\) in (19) becomes a matrix \(A = [a_1 a_2 a_3]\) and the problem is formulated as

\[
\text{minimize } \text{card}\{q\}, \quad q_i = \sum_{i=1}^{3} a_i^l, \quad \text{subject to } DA = \tilde{X}, \quad (20)
\]

where \(a_i^l\) denotes the \(i\)th entry of vector \(a_i\). The solution is found using the interactive weighted minimum norm algorithm described in (Gorodnitsky and Rao, 1997) with the weight matrix at each iterative step computed as \(W = \text{diag}(q)\). The given implementation typically yields a solution within two iterations.
PDAs typically restrict the search range in order to limit occurrence of harmonic errors. Limiting the search space here would be advantageous for two reasons, reducing harmonic errors and reducing the ‘ill-posed’ nature of our algebraic problem, which is reflected in the number of columns in the matrix $D$. We assume here a search range of 50–400 Hz, which is commonly used in PDAs. This range translates into 40–320 delays per period for a sampling rate of $f_s = 16 KHz$.

Since the delays in model (3) are assumed to be in the neighborhood of the nominal values $\tau_1 = 1$, $\tau_2 = f_s/F_0 - 1$, we can restrict the search range for $\tau_1$ to the ‘current’ cycle and the search range for $\tau_2$ to the end of the current cycle and the beginning of cycle immediately preceding it. The ‘current’ cycle refers to one that includes $x(n)$ as its last point. To do this, we use two sets of subdictionaries in matrix $D$, each subdictionary spanning the putative range of values of one delay. A key consideration here is to allow sufficient room for variation in $\tau_i$ due to noise. The following selection was experimentally determined to work well for this purpose. The first subdictionary is chosen to span the range $1 < \tau_1 < 64$, which corresponds to the last fifth of the longest possible current period. The second subdictionary is chosen to span the range $32 < \tau_2 < 384$. This range was obtained by expanding by a fifth the theoretical boundaries of the cycle that precedes the current one. Note that subdictionaries overlap in this case. The resulting columns in the dictionary $D$ therefore provide continuous coverage for delay values from 1 to 384 and the total number of columns is 384. The number of rows in $D$ is equivalent to the length of the vector $x$, that is, the length of the analysis frame. In our implementation, frame length is 20 msec, which is equivalent to 320 points.

The final part of the discussion is the initialization $x_0$. The role of $x_0$ is to preferentially weight the columns of $D$, thus biasing the algorithm to choose the solution in the neighborhood of the weighted bases. In the case of the two-delay model, the vector $x_0$ has two distinct maxima. The first peak value in $x_0$ preferentially weights the basis corresponding to $\tau_1 = 1$. The second peak is spread across multiple entries of $x_0$, preferentially weighting bases which correspond to $40 < \tau_2 < 320$. It should be noted that neither the absolute nor the relative values of the peaks with respect to the rest of the $x_0$ entries are important, and so they can be chosen fairly loosely, as long as falloff from both peaks is gradual. This implementation provides robust pitch determination for voiced speech. For the unvoiced parts of speech, $\tau_{1,2}$ often assume values that are on the edge of the allowable range. This plus a distinct rise in misfit error can be used as indicators of unvoiced speech.

III. EXPERIMENT

Pitch detection is a multi-step procedure. $F_0$ estimates are first computed over time. The second step is a voicing decision designed to retain only $F_0$ for voiced speech and remove the rest. The second step is necessary for arriving at meaningful pitch results but it is also very sensitive to errors arising in candidate $F_0$ due to data noise. To remedy this, postprocessing, called pitch tracking, is commonly employed to filter and select among the noise-distorted $F_0$ candidates. Original pitch tracking methods utilized estimates from earlier frames (Griffin and Lim, 1988), while modern methods utilize global methods based on dynamic programming (Tabrikian et al., 2006).

The IDEA algorithm addresses the first step in the pitch determination process described above, that is it computes an $F_0$ candidate over an analysis frame. Cepstrum and autocorrelation are the most popular methods for this step (Hess, 1983), but are known to perform poorly in high noise. The sinusoidal and harmonic models have drawn considerable interest recently as alternatives to the standard techniques (Ito and Yano, 2007; Tabrikian et al., 2006). In particular, the harmonic model combined with the state-of-the-art Maximum A-posteriori Probability (MAP) pitch tracking method, was shown to be robust at SNR as low as -15 dB in the presence of white, stationary noise (Tabrikian et al., 2006). However, in colored, nonstationary babble noise, the performance level was sustained only at SNR above 0 dB.

Most modern PDA research, e.g. (Ito and Yano, 2007; Tabrikian et al., 2006; Cheveigne and Kawahara, 2002), focuses on accuracy in reproducing ‘true’ laryngograph-derived pitch. The two standard measures used in evaluating PDA performance are: ‘fine pitch error’, which is the root mean squared error (RMSE) and Gross Error Rate (GER), which account for larger than 20% deviations from the laryngograph-derived pitch estimates. The goals here quite different. Here we are interested in performance in very high-noise, where the error rate is expected to be high, yet the result may still be useful for certain applications. To illustrate this point, consider the case where an estimate in each frame deviates by the exact same gross error from the ‘ground truth’ pitch value. Both error rates would indicate a poor estimate. An estimate however, that replicates the true pitch contour with a constant off-set is useful in any application that relies on the change in pitch rather than the exact values. This result can be used for evaluating prosodic variations, voice activity detection, and even audio content searches that use pitch variation rather than absolute values.

A. Evaluation Framework

In a more realistic scenario than the one above, high noise will result in some noisy $F_0$ estimates for voiced speech being misclassified as unvoiced and omitted from the output. The pitch curves will lose their original continuity but the rate and the pattern of this degradation will be different for the different methods. Error rates do not account well for such omissions. Also, depending on the nature of the application, an omission can amount to a more severe error than the inclusion of a particular erroneous estimate. If a method produces clusters of continuous pitch estimates in close proximity to each other rather than scattered throughout the utterance, they may still be useful in analysis of prosody or voice activity detection.

In order to illustrate the frame-to-frame continuity in estimated values, how the continuous clusters of estimates are distributed, how close the estimates are to the noise-free reference, and how the various noise statistics affect these results, we use a visual presentation. Pitch estimated from noisy signals are plotted along with a noise-free benchmark estimate for each noise condition. The presentation helps us develop a
conceptual understanding of the usefulness of the results obtained.

Since the standard error rates are not informative for our purpose, there is little reason to hinge our evaluation to laryngograph-derived ‘ground truth’. A reasonably accurate estimate obtained for noise-free speech would provide us a sufficient benchmark for calibrating how the ID model is affected by various properties of noise. For this reason, a popular ESPS (Secret and Doddington, 1983) algorithm was chosen to provide the noise-free reference. (The pitch halving errors were fixed in the reference as described below). ESPS uses normalized cross-correlation for candidate $F_0$ generation and pitch tracking based on dynamic programming. A standardized implementation used here is available from SJlander (2005). While other standardized programs are available for $F_0$ calculation, most are not PDAs, that is they do not incorporate pitch tracking. Dynamic programming in ESPS is superior to the pitch tracking we employ with IDEA, but it does not appear to impact the outcome of the evaluation, which focuses on the voiced parts of speech. We also chose to use the TIMIT database (Garofolo et al., 1993) for similar reasons. It is widely utilized for testing purposes and its lack of laryngograph-derived reference pitch is not critical for our purposes.

The final consideration in our evaluation is the choice of pitch tracking method that we apply to postprocess IDEA output. As discussed earlier, some type of pitch tracking is imperative for obtaining meaningful results in low SNR environment. However, it would be premature to simply assume that a state-of-the-art pitch tracking method, such as MAP in (Tabrikian et al., 2006), provides the optimal choice for our purpose. There is little reason to hinge our evaluation to a large range of speakers. (The pitch halving errors were fixed in the reference as described below). ESPS was chosen to provide the noise-free reference. (The pitch halving errors were fixed in the reference as described below). ESPS uses normalized cross-correlation for candidate $F_0$ generation and pitch tracking based on dynamic programming. A standardized implementation used here is available from SJlander (2005). While other standardized programs are available for $F_0$ calculation, most are not PDAs, that is they do not incorporate pitch tracking. Dynamic programming in ESPS is superior to the pitch tracking we employ with IDEA, but it does not appear to impact the outcome of the evaluation, which focuses on the voiced parts of speech. We also chose to use the TIMIT database (Garofolo et al., 1993) for similar reasons. It is widely utilized for testing purposes and its lack of laryngograph-derived reference pitch is not critical for our purposes.

The final consideration in our evaluation is the choice of pitch tracking method that we apply to postprocess IDEA output. As discussed earlier, some type of pitch tracking is imperative for obtaining meaningful results in low SNR environment. However, it would be premature to simply assume that a state-of-the-art pitch tracking method, such as MAP in (Tabrikian et al., 2006), provides the optimal choice for IDEA. Since IDEA solutions themselves depend on previous frames, it is not clear which pitch tracking approach would best complement this property. Another consideration here is that while the optimal pitch tracking method would produce superior results, they would also obscure the evaluation of the intrinsic robustness of ID-based $F_0$ estimation. Since this paper concerns an evaluation of intrinsic robustness of IDEA, we chose a basic pitch tracking method that preserves as much as possible the candidate $F_0$ estimates in voiced sections. The pitch tracking method we use is based on filtered estimates from earlier frames and is closest to principle to the earlier type of methods as described in (Griffin and Lim, 1988). The results presented below show that our pitch tracker/voicing decision mechanism occasionally retains candidate $F_0$ values in unvoiced parts of data. As long as this is understood to be an artifact of the pitch tracker, which we expect to ultimately remedy with a more advanced algorithm, it does not impact our evaluation of the robustness of ID model parameters in the voiced parts of speech. The important consideration for us is that our postprocessing does not alter the values of the $F_0$ estimates in the voiced sections. Optimal postprocessing of ID model parameters is a sufficiently involved subject on its own to be a topic for a separate paper.

B. Experimental Details

A total of 42 utterances from the TIMIT database were evaluated. The results were consistent across the three tested speakers, one female, two male, and gender appears to be irrelevant to IDEA performance. Since we present the results visually, in the interest of compactness, only one sentence SX42 (‘Biblical scholars argue history’), spoken by male speaker MTAB0, is chosen for the presentation. The results shown are representative of those observed for all evaluated utterances.

The output of IDEA and ESPS was compared for SNRs ranging from -10dB to 0dB, using three distinct types of noise: 1) white, stationary noise, which provides a baseline performance; 2) babble noise, which is a colored noise with spectrum similar that of the speech signal, and 3) M109 tank noise, which is a deterministic low-frequency noise with power dominating that of speech in the critical $F_0$ range. White Gaussian noise was generated using the MATLAB ‘randn’ command. The babble and M109 tank noise samples were downloaded from the Signal Processing Information Base (SPIB, 1995).

Noisy speech was generated by adding scaled noise samples to the noise-free signal. All computations, with the exception of ESPS, were performed in MATLAB. The parameters of (3) were estimated using the extended FOCUS algorithm described in Section II. The derivatives were computed using a three-point numerical derivative approximation. ESPS estimates were obtained from the Tcl/Tk SNACK library audio analysis module for Linux platforms SJlander (2005). Algorithms were applied to 30 msec speech frames with 10 msec frame shift (frame length in SNACK). The 16 KHz sampling rate of the original recordings was used throughout. ID model estimation was restricted to a 50-400 Hz search range while the search range for ESPS was restricted to 60-400 Hz. Setting the lower bound to 60 Hz instead of 50Hz for ESPS was necessary to avoid pitch halving errors. The 50 Hz lower bound was kept for IDEA to create a more realistic testing environment that would be applicable to a large range of speakers.

1. Noise-free performance

The first experiment is designed to illustrate pitch detection capability with the ID model (3) under noise-free conditions. Fig. 1a presents the raw $F_0$ estimates obtained by running IDEA across the entire SX42 sentence spoken by MTAB0. The pitch contour computed with our pitch tracking method overlays the raw $F_0$ values. Fig 1b displays the pitch track from IDEA and ESPS for the same utterance. Note that ESPS sets to zero the frames deemed to be unvoiced. To facilitate the comparison between IDEA and ESPS, IDEA will also set to zero the unvoiced frames.

Fig. 1a illustrates the performance of our basic postprocessing routine in identifying and eliminating unvoiced sections from the utterance and at the same time not altering the values of $F_0$ estimates in the voiced sections. The further comparison with ESPS in Fig. 1b shows that this process works reasonably well for identifying voiced sections. Only a few points are retained from the unvoiced sections that should have been omitted. Overall, pitch tracks from IDEA and ESPS in Fig. 1b are in a very close agreement. Both methods produced some errors, however. One large disagreement between the methods is at the end of the utterance, past the 2 sec mark. ESPS, but not IDEA, deems this part to be unvoiced. Existence of voiced speech in that part can be verified by plotting or listening to the waveform. Further investigation revealed that the
error was due to pitch halving in a large number of frames, so the overall $F_0$ estimate appeared noisy and was set to zero. To avoid such problems, the search space for ESPS was limited to 60-400 Hz in subsequent experiments.

Two small errors appear in IDEA output. One is around the 0.5 sec mark. From Fig. 1a, the same discrepancy between the outputs of two methods is evident in the raw $F_0$ estimates. A similar problem occurs at the very beginning of the second voiced segment of the utterance. Output of occasional errors by IDEA was mentioned in Section II. At this point, we can only hypothesize about the cause of these errors. They may be due to the numerical implementation used here or to rapid transitions between either distinct oscillatory states or between oscillatory and non-oscillatory states, since parameters of ID models rely on past oscillatory cycles. The dependence of the model estimation process on the past oscillatory states is clearly a weakness of this method and would be expected to generate errors when modeling signals containing rapidly changing oscillatory states. A more advanced pitch tracking method would omit these $F_0$ estimates altogether. As discussed earlier, for now we accept this level of performance in noise-free conditions and consider instead how well this performance is maintained in high noise.

2. Performance in high noise

The spectra of the speech signal and the three noise types used in the evaluation are shown in Fig. 2. A stationarity measure is not formally presented, but the property is easily inferred from the figure. Babble noise is background speech involving 100 people speaking in a canteen where individual voices are only slightly distinguishable. From Fig. 2, the speech signal and babble are both globally nonstationary and the ranges of their spectral power distribution overlap almost completely. In many automatic recognition systems babble is considered the hardest noise to deal with, but this determination appears to be biased by the narrow set of noise types that have been considered. In particular, machine type noise has not been commonly considered in speech analysis. The M109 noise was acquired from a tank moving at 30 km/h. It is locally non-stationary (on the scale of 2-6 frames). While, babble and white noise have spectral profiles with considerable power in the harmonics above 500 Hz, the power in the tank noise spectrum is predominantly in the lower range, particularly below 200 Hz. Fig. 3 illustrates the distribution of power for equally scaled tank noise and speech (0 dB SNR).

The power of this largely deterministic noise dominates that of speech below 200 Hz, which is the $F_0$ search range, while a considerable amount of power in the speech signal is allocated to higher harmonics. This makes the effective SNR in the $F_0$ search range even smaller than what is reflected in the SNR computed over the entire frequency range.

Many machine-generated noises have power concentrated in the lower frequency range. Since time-delay models rely on the deterministic structure of data, the tank noise is expected to present a worst-case scenario for NO models. Our analysis shows this noise to be the most challenging of the three for both methods. Unexpectedly, we find that IDEA still deals with this noise better than ESPS.

Figs. 4-6 present a comparison between IDEA and ESPS estimates for the three types of noise with SNR ranging from -10 dB to 0 dB. The reference pitch in these examples was computed as an average of noise-free ESPS and IDEA pitch estimates, with points that deviated from the smooth curves removed before averaging. For frames for which this procedure resulted in eliminating one of the pitch estimates or for which only a single estimate was available in the first place, the value was estimated without the averaging. This resulted in the reference pitch inheriting the smoothness of the ESPS estimate in the 0-2 sec part of the utterance and nonzero pitch values from IDEA for the voiced speech after the 2 sec point.

Figs. 4-6 reveal that performance characteristics of the two methods are different even at 0 dB SNR. In the case of white noise (Fig. 4), IDEA identifies each voiced segment of the utterance, while ESPS identifies short sections in only two of the voiced segments. The difference cannot be simply attributed to a difference in the voicing decision mechanism of the two methods. IDEA estimates in the voiced sections that ESPS deemed unvoiced agree closely with the reference pitch. Hence these have been estimated and retained correctly. In only three places do a few isolated pitch values that should have been rejected as unvoiced appear in IDEA output. As commented before, we are not concerned with these errors for the purpose of the current evaluation. These pitch values are clearly far apart from the clusters of smoothly varying estimated values and thus could be identified and removed using this criterion.

IDEA performance degrades gradually through the two increases in noise level. For -5 dB SNR, IDEA identifies continuous pitch contours in the first, fourth, and fifth voiced sections of the utterance. By comparison, ESPS produces short pitch segments only in the first section. As in the case of 0 dB SNR, IDEA estimates in these three sections closely adhere to the reference pitch and, therefore, the performance discrepancy with ESPS in these parts is attributable to higher estimation accuracy of $F_0$ by IDEA rather than a looser voicing decision mechanism. At -5 dB SNR we also see emergence of incidences of gross errors in IDEA estimates for some of the voiced sections (second and third). The values in these sections are distributed randomly over a 30 Hz range, which should make their identification and removal straightforward.

At -10 dB SNR the proportion of correctly estimated $F_0$ values drops considerably. Two short smooth segments of correctly found pitch are present. Otherwise, nonzero estimates are distributed randomly and would expect to be omitted with a more advanced voicing decision mechanism. The few correctly identified pitch segments could still be useful, however, for example, for voice activity detection. ESPS, on the other hand, does not identify any voiced speech at -10 dB SNR.

Observations for babble and tank noise (Figs. 5 and 6) are similar to those above. Both methods appear to deal most successfully with babble noise (Fig. 5), which is composed of multiple waveforms that individually are very weak relative to the signal of interest. Performance of both methods degrades more gradually with the rise of babble noise power. Even at -10 dB SNR, IDEA pulls out three continuous pitch sections that agree with the reference pitch. Overall, the output is not "noisy", that is, the correct to wrong estimate ratio is greater in the case of babble noise than the other two cases.

Tank noise (Fig. 6) is the most challenging for both methods. Pitch segments estimated with IDEA are not as continu-
ous as in the case of the other two noises, even at 0 dB SNR. Estimates in the second part of the utterance (after the 1 sec mark) are scattered and would be expected to be deemed unvoiced. Thus, we consider performance in the voiced sections in the first half of the utterance. For 0 and -5 dB SNR, pitch values in this part were found to fall within 12% of the reference pitch, with exception of the clear outlier in the 0dB SNR case, which is still within 20% (the GER cut-off) of the reference value. The drop in performance between 0 dB and -5 dB SNR is also gradual in this part. There is a slight reduction in the number of voiced decisions at -5dB SNR, but the values that are retained are within the same close range of the reference as those in the 0 dB SNR case. At -10 dB SNR, there are two small stretches (three frames each) of gradually varying pitch values, but the rest of the returned values are so sporadic that the result is unlikely to be useful even for voice activity detection. Thus the breakdown point for this type of application occurs somewhere between -10 and -5 dB SNR for IDEA. For ESPS, this breakdown occurs between -5 and 0 dB SNR.

In summary, IDEA is shown to be more robust than ESPS in extremely noisy conditions (SNR ≤ 0dB) for statistically different noise types. It maintains its ability to estimate pitch in at least some sections of voiced speech in as low as -5dB SNR in all noise cases. At -10dB SNR levels, voiced/unvoiced determination becomes sporadic, but consistent pitch estimates are still found in some parts of the utterance in the case of white and babble noises. IDEA output even at these SNR levels could be useful in applications such as voice activity detection.

### IV. CONCLUSIONS

We have presented a nonlinear oscillator that has a very compact parameterization space and its parameters are physically meaningful. These properties distinguish this model from conventional nonlinear oscillators, which are parameterized over a large space and the model parameters lack any physical interpretation. The relationship of the oscillator parameters to fundamental frequency was presented and then used to formulate a pitch estimation algorithm based on the proposed model. Another important practical feature of the oscillator is that it is robust with respect to noise. The robustness in generating $F_0$ candidates for nonstationary speech signals was evaluated experimentally in three types of high noise conditions: white noise, babble produced by a crowd of one hundred people, and tank generated noise. The experiments were designed to test the limits of the algorithm rather than highlight the best possible performance. In particular, low frequency machine generated noise presented one of the worst scenarios for the oscillator model. Nevertheless, robustness was observed for the method presented for all tested noise cases through -5 dB SNR. Further, whereas a popular pitch detection algorithm, ESPS, did not detect voiced speech at -5 dB SNR levels, the IDEA algorithm was able to detect some snippets of voiced speech even at -10dB SNR for two noise cases, white and babble. The work presents an early exploration of an intriguing and potentially very practical framework. The properties identified here indicate ways this novel method may be useful, from immediate applications, such as robust pitch tracking, to longer term possibilities, in particular, exploiting the ID models for modeling, coding, and synthesis.


FIG. 1. ID and ESPS pitch estimates for sentence SX42, speaker MTAB0, in the noise free case. (a) Raw $F_0$ estimates obtained with the two-delay ID model and the corresponding pitch track computed with the pitch tracking method described in the paper. (b) Pitch tracks obtained with the ID model and ESPS.
FIG. 2. Waveforms and time-frequency decomposition of the speech signal and the three noise types.
FIG. 3. A close up view of power distribution in the frequency range below 400 Hz in the speech and the tank noise signals. The two signals have equal overall spectral power.

FIG. 4. A comparison of the 1D model and ESPS estimated pitch tracks in the presence of white background noise. (a) 0dB SNR. (b) -5dB SNR. (c) -10dB SNR.
FIG. 5. A comparison of the ID model and ESPS estimated pitch tracks in the presence of babble background noise. (a) 0dB SNR. (b) -5dB SNR. (c) -10dB SNR.

FIG. 6. A comparison of the ID model and ESPS estimated pitch tracks in the presence of M109 tank background noise. (a) 0dB SNR. (b) -5dB SNR. (c) -10dB SNR.