SPEECH COMPRESSION BASED ON EXACT MODELING AND STRUCTURED TOTAL LEAST NORM OPTIMIZATION

Philippe Lemmerling, Ioannis Dologlou, and Sabine Van Huffel

ESAT Laboratory, Department of Electrical Engineering, Katholieke Universiteit Leuven, Kardinaal Mercierlaan 94, 3001 Leuven, Belgium
philippe.lemmerling@esat.kuleuven.ac.be

ABSTRACT

We present a new speech coding algorithm, based on an all-pole model of the vocal tract. Whereas current Auto Regressive (AR) based modeling techniques (e.g. CELP, LPC-10) minimize a prediction error, which is considered to be the input to the all-pole model, our approach determines the closest (in L_2 norm) signal, which exactly satisfies an all-pole model. Each frame is then encoded by storing the parameters of the complex damped exponentials deduced from the all-pole model and its initial conditions. Decoding is performed by adding the complex damped exponentials based on the transmitted parameters.
The new algorithm is demonstrated on a speech signal. The quality is compared with that of a standard coding algorithm at comparable compression ratios, by using the segmental Signal-to-Noise Ratio (SNR).

1. INTRODUCTION

This paper presents a new method for speech coding. It belongs to the class of vocoders which use an all-pole model for modeling the vocal tract. The resulting minimum phase model is sufficient for preserving the exact magnitude spectrum, whereas phase information is lost [6]. Most Linear Predictive Coding (LPC) based techniques make the additional assumption that the input to the Auto Regressive (AR) model is white noise, represented by the vector e. If we represent the speech signal by a vector s and assume a model of order L, the modeling of the ith frame of the speech signal can be recasted as the following optimization problem:

$$\begin{aligned} &\min_{a(l),l=1,\dots,L} \sum_{j=1+(i-1)N}^{iN} (e(j))^2 \text{ where} \\ &s(k) - e(k) = \sum_{l=1}^{L} a(l)s(k-l), \\ &k = 1 + (i-1)N + L, \dots, iN \end{aligned} \tag{1}$$

where N equals the number of samples per frame, $a(l), l = 1, \dots, L$ are the so-called prediction coefficients. Note that we adopt a Matlab-like notation, where $v(i)$ indicates the ith element of vector v, and $v(i:j)$ represents the subvector of v, starting at the ith element and ending at the jth element of vector v.
A closer look at (1) reveals that the problem is in fact a Least Squares (LS) problem. This is the basic scheme used by well-known LPC based algorithms such as LPC-10 [13] or CELP [4] (in practice however, the prediction coefficients are not determined by solving (1), but by using an equivalent autocorrelation method). At the receiver side, the speech is synthesized using the all-pole model based on the transmitted model parameters. In the case of a voiced frame, the input to the filter will be a periodic pulse with the transmitted pitch frequency, while in the unvoiced case the input is white noise. In the case of CELP the excitation is chosen out of a series of standardized noise-like sequencies in order to obtain the best synthesis.
Our new approach is still based on the all-pole model but instead of solving (1), we solve the following problem for the ith frame:

$$\begin{aligned} &\min_{\substack{\Delta s(j),j=1+(i-1)N,\dots,iN,\\ a(l),l=1,\dots,L}} \sum_{j=1+(i-1)N}^{iN} (\Delta s(j))^2 \\ &\text{such that } s(k)+\Delta s(k) = \sum_{l=1}^{L} a(l)(s(k-l)+\Delta s(k-l)), \\ &k = L+1+(i-1)N, \dots, iN. \end{aligned} \tag{2}$$

Philippe Lemmerling is a Research Assistant with the I.W.T. (Flemish Institute for Scientific and Technological Research in Industry). S. Van Huffel is a Research Associate with the F.W.O. (Fund for Scientific Research - Flanders). This paper presents research results of the Belgian Programme on Interuniversity Poles of Attraction (IUAP P4-02 and P4-24), initiated by the Belgian State, Prime Minister's Office - Federal Office for Scientific, Technical and Cultural Affairs and of a Concerted Research Action (GOA) project of the Flemish Government, entitled "Model-based Information Processing Systems". The scientific responsibility is assumed by its authors.

So instead of minimizing a prediction error, as in (1), we determine for each sample $s(k)$ a correction $\Delta s(k)$, such that the corrected signal $s(k) + \Delta s(k)$ exactly satisfies an AR model, with the correction as small as possible in L_2 norm. In the following section we describe the vocoder based on our new approach, by developing the kernel algorithm. The third section presents numerical results and a comparison with standard methods, using a speech signal. We discuss the quality performance and the efficiency of the new approach. We conclude with a summary and some further research.

2. DESCRIPTION OF THE VOCODER

As already mentioned in the introduction, the kernel problem of our new approach can be formulated as in (2). It is easy to recast this optimization problem in a matrix framework:

$$\min_{\substack{\Delta s(j), j=1+(i-1)N,\dots,iN,\\ a(l), l=1,\dots,L}} \sum_{j=1+(i-1)N}^{iN} (\Delta s(j))^2 \quad (3)$$
$$\text{such that} (S + \Delta S)a = b + \Delta b.$$

If we use the convention that the vector $s(1+(i-1)N : iN)$ can be read from the first row and the first column of the Hankel matrix $[b \;\; S]$ (and the same convention for $\Delta s(1 + (i-1)N : iN)$ and $[\Delta b \;\; \Delta S]$), by starting in the upper right corner and ending in the lower left corner, the matrices S, ΔS and the vectors b, Δb and a can readily be determined by comparing (2) with (3). Observe that both the matrices $[b \;\; S]$ and $[\Delta b \;\; \Delta S]$ have a Hankel structure.
Problem (3) is an extension of the LS approach in (1). First of all (3) allows also corrections on the left hand side of the equations in (1) and secondly, the error matrix $[\Delta b \;\; \Delta S]$ is forced to have the same Hankel structure as the original data matrix $[b \;\; S]$. In fact, (3) can be seen as the structured extension of the Total Least Squares (TLS) approach [15].

Problem formulation (3) has been the subject of many papers in recent years and is known under different names such as Structured Total Least Squares (STLS) problem [7], Structured Total Least Norm (STLN) problem [11][14] or also Constrained Total Least Squares (CTLS) problem [1][2]. As explained in [9] and [10] all these different approaches are equivalent. In our application we will pursue the STLN approach, since at this time, it is the computationally most effective way to tackle problem (3). It is not our goal to give an extensive description of the STLN algorithm, used here and outlined in [14] as algorithm STLNB. The differences between our implementation and the algorithm described in [14] can be summarized as follows:

- we replace the weighting method in Step 2(a) [14] of algorithm STLNB by the equivalent equality constrained LS problem.
- the stop criterion is either based on the norm of $\|\hat{r}\|_2 \equiv \|(S+\Delta S)a-(b+\Delta b)\|_2$ or the number of iterations is kept fixed, as explained further on.

We will now briefly describe some aspects of the STLN algorithm and the related difficulties.
As can be seen from (3), we have to deal with a quadratic objective function and nonlinear equality constraints (the nonlinearity resides in the term $\Delta S a$). This problem is solved by an iterative algorithm. In each iteration the nonlinear equality constraints are linearized around the current iteration point and an equality constrained LS problem is solved. Since we solve a nonlinear optimization problem, the use of good starting values is of utmost importance for convergence within a reasonable amount of time. A method which yields very good starting values in this respect is HTLS [16]. This is a suboptimal (it does not give the closest fit) subspace based harmonic retrieval method, that approximates the signal s by a sum of L complex damped exponentials. Straightforward calculations based on the parameters of these exponentials yield the initial a and $[\Delta b \;\; \Delta S]$. After applying STLN with the previously mentioned initial values, we could encode each frame by storing the vector a and the first L values of $s + \Delta s$ for that particular frame. Since this procedure will lead to large reconstruction errors at the receiver side, we apply HTLS to the obtained data matrix $[b + \Delta b \;\; S + \Delta S]$. Since the corrected data $s + \Delta s$ is rank deficient and real, HTLS gives an exact fit and the resulting $2L$ parameters of the complex damped exponentials can be used for encoding. The vocoder analysis and synthesis algorithms, applied to the ith frame, can thus be summarized as follows:

Vocoder Analysis Algorithm

Input: ith frame of the speech signal: $s(k)$, $k = 1 + (i-1)N, \dots iN$, with N the number of samples per frame, L the order of the AR filter
Output: $f_k, d_k, a_k, p_k, k = 1, \dots, L/2$, representing the frequencies, dampings, amplitudes and phases of the complex damped exponentials, satisfying $\sum_{k=1}^{L/2} c_k z_k^j + c_k \overline{z}_k^j = s(j) + \Delta s(j), j = 1 + (i-1)N, \dots, iN$.

Step 1: Initialize $\Delta s(j), j = 1 + (i-1)N, \dots, iN$ and $a(l), l = 1, \dots, L$ with the result of HTLS applied to $s(1 + (i-1)N : iN)$.
Step 2: Solve STLN problem (3)
Step 3: Apply HTLS to $s(1 + (i-1)N : iN) + \Delta s(1 + (i-1)N : iN)$, to extract $f_k, d_k, a_k, p_k, k = 1, \dots, L/2$

Vocoder Synthesis Algorithm

Input: $f_k, d_k, a_k, p_k, k = 1, \dots, L/2$, representing the frequencies, dampings, amplitudes and phases of the complex

damped exponentials.
Output: $s(1+(i-1)N:iN)+\Delta s(1+(i-1)N:iN)$, the rank-deficient speech signal that lies closest to $s(1+(i-1)N:iN)$ in L_2 norm.

Step 1: $s(j)+\Delta s(j) \leftarrow \sum_{k=1}^{L/2} c_k z_k^j + c_k \overline{z}_k^j$, $j = 1+(i-1)N, \ldots, iN$.

With $c_k = a_k e^{(\sqrt{-1}p_k)}$, $z_k = e^{(2\sqrt{-1}\pi f_k + d_k)\Delta t}$, Δt being the sampling interval and $\overline{x}$ indicating the complex conjugate of x. We note that, as described in [5], it is possible to merge Step 2 and Step 3 of the vocoder analysis algorithm. This increases the efficiency by replacing the SVD of Step 3 by a QR factorization. However, the overall effect on the number of computations would be marginal, since the contribution of Step 3 in the total amount of work is minor compared to the iterative Step 2. The quantization of the parameters will be discussed in the following section.

3. EXPERIMENTATION-TESTING

In this section we compare the exact AR modeling approach to the CELP standard algorithm, applied to a speech signal sampled at 8kHz, using 8 bits per sample. It contains 14749 samples (approximately 2 seconds of speech) and is a phonetically balanced French sentence, uttered by a male speaker. The sentence is an enumeration of geographical places:

Paris, Bordeaux, Le Mans, Saint-Leu, Léon, Loudun

which has the following phonetic transcription (according to the International Phonetic Association's rules [8]):

paʁi, bɔʁdo, ləmɑ̃, sɛ̃ lø, leɔ̃, ludœ̃

For the CELP algorithm, we used a Fortran implementation of the Federal Standard 1016 4800 bps CELP vocoder [4]. For the exact AR modeling approach we use the vocoder algorithm described in section 2. We set the frame length N to 301, the model order L to 12 and fix the number of iterations of STLN to 10. We quantize the parameters obtained in step 3 of the vocoder analysis algorithm as follows: 12 bits per frequency, 7 bits per damping, 7 bits per amplitude and 6 bits per phase. We remark that this is a very simple bit allocation scheme and could easily be improved by using more sophisticated schemes. These settings lead to the following compression ratios: $\frac{8bits/sample*8000(samples/sec)}{4800(bits/sec)} \approx 13.33$ for CELP and $\frac{301(samples/frame)*8(bits/sample)}{24(parameters/frame)*8(bits/parameter)} \approx 12.54$ for STLN. Both STLN and HTLS are coded in Matlab. In order to avoid inadmissable long computation times, we implemented the LAPACK subroutine DGGLSE [3] as a MEX-file for the kernel routine of STLN (the above mentioned equality constrained LS).

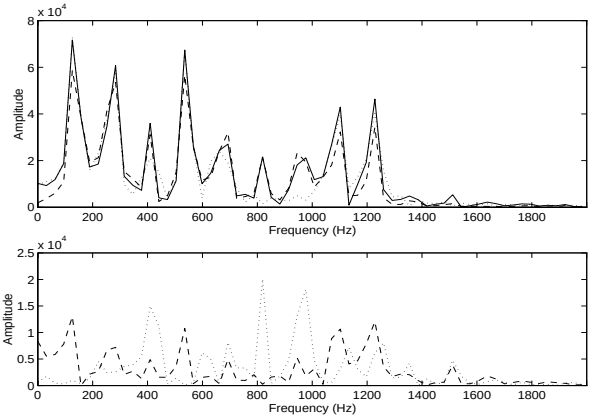

Figure 1: *The upper part of the figure shows the FFT magnitude spectrum of $s(500:755)$ (full line), the FFT magnitude spectrum of the CELP result (dashed line) and the FFT magnitude spectrum of the STLN result (dotted line). The lower part of this figure shows the absolute value of the difference between the FFT magnitude spectrum of the original signal and the FFT magnitude spectrum of the CELP result (dashed line), together with the absolute value of the difference between the FFT magnitude spectrum of the original signal and the FFT magnitude spectrum of the STLN result (dotted line).*

To assess the quality of the compressed speech, we use the following segmental SNR definition:

$$SNR_{seg} \equiv 10log_{10}\frac{1}{F}\sum_{j=1}^{F}\frac{\sum_{i=1}^{p}(s_j(i))^2}{\sum_{i=1}^{p}(s_j(i)-\hat{s}_j(i))^2}, \quad (4)$$

where F represents the number of frames, p is the frame length used for averaging, $s_j = s(1+(j-1)p:jp)$, $\hat{s}_j = \hat{s}(1+(j-1)p:jp)$ and $\hat{s}$ represents the synthesized signal. Here p is chosen equal to 60 but the result is rather insensitive with respect to p. For the CELP result , this gives a $SNR_{seg} = 12.8dB$. This value results from a comparison between the highpass filtered input and the non-postfiltered output (standard CELP applies at the end an adaptive postfilter routine to reduce perceptual coder noise). An upperbound for the quality of the STLN approach, is obtained when no quantization of the parameters occurs. This yields a segmental SNR of $17.5dB$. With the simple bit allocation scheme described above, we still obtain $SNR_{seg} = 16.4dB$. The upper part of figure 1 shows the magnitude spectrum of the FFT of $s(500:755)$ (full line, this corresponds to the "a" in "Paris") , the corresponding CELP result (dashed line) and the corresponding STLN result (dotted line). The lower part of figure 1 shows the absolute value of the difference between the FFT magnitude spectrum of $s(500:755)$ and the FFT magnitude spectrum of the corresponding CELP result (dashed line), together

with the absolute value of the difference between the FFT magnitude spectrum of $s(500:755)$ and the FFT magnitude spectrum of the corresponding STLN result (dotted line). At a sampling frequency of 8 kHz, the highest frequency on the x-axis should be 4 kHz, but we only show that part of the spectrum where the magnitude differs considerably from 0. We see, especially from the lower part of the figure that the 5 largest peaks are better fitted by the STLN result than by the CELP result, which illustrates the better quality of the signal poles obtained by STLN.

There is a drawback associated to our new approach, which is its computational load. Since the kernel problem is an equality constrained LS problem, we have per iteration approximately $4mn^2$ flops, with $m \times n$ the dimensions of S. However, it is possible to speed up the kernel problem, by exploiting the Hankel structure (as indicated for the Toeplitz structure in [12]). It is worth noting that starting values play an important role in the overall performance of STLN. It has been found that HTLS provides a satisfactory initialization, however further research is under way to find simpler alternatives.

4. CONCLUSIONS

In this paper we propose a new type of vocoder. Like many vocoders, it is based on an all-pole model of the vocal tract. In contrast to other LPC based methods, we derive a perturbed signal which exactly satisfies an all-pole model. This is what we call an exact AR modeling. As a result, we only have to transmit the model parameters and the initial values for each frame. Alternatively, we can transmit the corresponding frequencies, dampings, phases and amplitudes, which is numerically more stable but computationally more complex.

Results show that the segmental SNR of our approach is substantially higher than that of CELP, for similar compression ratios. The drawback of the approach is of course its computational load. The latter prevents the algorithm from being applied in real-time applications. However, the development of fast STLN algorithms will bring this vocoder closer to real-time implementation. For off-line applications (e.g. storage of sound files at a server), the simple reconstruction algorithm outweighs the computational expensive coding.

5. REFERENCES

[1] T. J. Abatzoglou and J. M. Mendel, *Constrained Total Least Squares*, IEEE ICASSP, Dallas, 1987, pp. 1485-1488.

[2] T. J. Abatzoglou, J. M. Mendel and G. A. Harada, *The Constrained Total Least Squares Technique and its Applications to Harmonic Superresolution*, IEEE Trans. on S.P., 39 (1991), pp. 1070-1086.

[3] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen (1995), LAPACK Users' Guide (SIAM, Philadelphia), pp. 201-202.

[4] Campbell, Joseph P. Jr., Thomas E. Tremain and Vanoy C. Welch (1991), "The Federal Standard 1016 4800 bps CELP Voice Coder", Digital Signal Processing, Academic Press, Vol. 1, No. 3, pp. 145-155.

[5] Chen H., Van Huffel S., Van Ormondt D., "Application of the Structured Total Least Norm technique in spectral estimation", in Proc. of the 8th European Signal Processing Conference (EUSIPCO'96), Trieste, Italy, 10 -13 September 1996, pp. 706-709.

[6] John R. Deller Jr., John G. Proakis, and John H. L. Hansen (1993), Discrete-Time Processing of Speech Signals (MacMillan Publishing Company, Englewood Cliffs, NJ), pp. 266-273 .

[7] B. De Moor, *Total least squares for affinely structured matrices and the noisy realization problem*, IEEE Trans. on S.P., 42 (1994), pp. 3004-3113.

[8] P. Ladefoged (1975), *A Course in Phonetics* (Harcourt Brace Jovanovich, New York).

[9] P. Lemmerling, B. De Moor, and S. Van Huffel, *On the equivalence of constrained total least squares and structured total least squares*, IEEE Trans. on SP, SP-44 (1996), no. 11, pp. 2908-2910.

[10] P. Lemmerling, S. Van Huffel, and B. De Moor (1997), "Structured total least squares problems: formulations, algorithms and applications", in: S. Van Huffel ,ed., Recent advances in total least squares techniques and errors-in-variables modeling (SIAM, Philadelphia), pp. 215-223.

[11] J.B. Rosen, H. Park, and J. Glick, *Total least norm formulation and solution for structured problems*, SIAM Journal on Matrix Anal. Appl., 1996, vol. 17, no. 1, pp. 110-126.

[12] J. B. Rosen, Haesun Park, John Glick, *Total Least Norm Formulation and Solution for Structured Problems*, Supercomputer Institute Research Report UMSI 93/223, Univ. of Minnesota, November, 1993, revised July, 1994.

[13] Thomas E. Tremain (April 1982), "The Government Standard Linear Predictive Coding Algorithm: LPC-10", Speech Technology Magazine, p. 40-49.

[14] S. Van Huffel, H. Park, and J. Ben Rosen, *Formulation and Solution of Structured Total Least Norm Problems for Parameter Estimation*, IEEE Trans. on Signal Processing, SP-44 (1996), no. 10, pp. 2464-2474.

[15] S. Van Huffel and J. Vandewalle (1991), *The total least squares problem : computational aspects and analysis* (SIAM, Philadelphia).

[16] S. Van Huffel, C. Decanniere, H. Chen, P. Van Hecke (1994), "Algorithm for Time-Domain NMR Data Fitting Based on Total Least Squares", J. Magn. Reson., A110, pp. 228-237.