

DIXIEME COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

863



NICE du 20 au 24 MAI 1985

INPUT ESTIMATION WITH APPLICATION TO SPEECH CODING

Alf Isaksson and Mille Millnert

Div of Autom Control, Department of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden

RESUME

Dans ce papier une méthode pour l'estimation de l'entrée (ou déconvolution) est présentée.

La méthode est basée principalement sur l'utilisation d'une certaine paramétrisation du modèle du signal d'entrée. Pour utiliser cette méthode, nous devons être capable d'exprimer le signal d'entrée en fonction de quelques paramètres inconnues et du temps.

L'algorithme est conçu pour estimer, simultanément, les paramètres du signal d'entrée et ceux de la fonction de transfert du système. On se limite à l'étude des systèmes dont la fonction de transfert ne comportant que des pôles (c.à.d modèles ARX). La méthode peut être étendue pour considérer aussi les zéros de la fonction de transfert. Il est évident que ceci entraîne une augmentation de la charge numérique. L'algorithme est basé sur des méthodes numériques efficaces comme par exemple la factorisation QR utilisant les transformations de Householder. L'application d'un tel algorithme au codage de la parole est présentée. Il est à noter que la qualité du signal synthétisé de la parole, peut être nettement améliorée si un modèle plus détaillé est utilisé pour décrire, le modèle du mouvement des cordes vocal plutôt qu'un train d'impulsion. On montre aussi que la méthode envisagée peut être utilisée pour estimer les paramètres du système vocales et ceux du modèle du mouvement des cordes vocales simultanément.

SUMMARY

In this paper a method for input estimation or deconvolution is presented. The basis of the method is to use a parameterized model of the input signal. To use the method we should thus be able to express the input signal as a function of some unknown parameters and time. The algorithm simultaneously estimates the parameters of the input signal and the parameters of the system transfer function. The presentation here is restricted to transfer functions of all pole type, i.e. ARX-models. The method can be extended to handle zeros in the transfer function. The computational burden would however increase significantly. The algorithm uses efficient numerical methods, as for instance QR-factorization through Householder transformation.

The algorithm is in this paper applied to a problem in speech coding. It has been observed that the quality of synthesized speech can be improved, if a more detailed model than an impulse train is used for the pitch pulses, see Fant (1980). It is here shown how the method presented in this paper can be used to estimate the system parameters of the speech production and the parameters of the glottal pulse simultaneously.



1 INTRODUCTION

To estimate the input of a dynamical system is a problem of great practical interest. Applications can be found in such different areas as speech coding, see Hedelin (1984), Geoscience, Mendel (1979) or Control Theory, Ahlén (1984). In many situations not only the input is unknown, but also the system itself might be unknown.

In this paper a method for simultaneously estimating both the input and the parameters of a system model is presented. The method is applicable to systems with transfer functions of all-pole type. The structure of the input should also be known. We should thus be able to parametrize the input signal in some unknown constants, e.g. a sinusoid with unknown frequency and phase. To include zeros in the transfer function would only require minor changes in the algorithm. The computational burden would however increase significantly. The algorithm uses efficient methods like QR-factorization through Householder transformations for the numerical calculations. The numerical algebra part of the algorithm is modelled on the method presented in Golub et al (1973)

The algorithm is applied to a speech coding problems, which is described in the second part of this paper. The problem is to estimate unknown parameters in a model for the glottal pulses in voiced speech, see Markel and Gray (1976).

The paper is organized in the following way. In Section 2 the input estimation problem in general is discussed briefly. In Section 3 the basic steps of the algorithm are derived. In Section 4 the application to speech-coding is presented and in Section 5 some aspects on the current implementation are discussed.

2 INPUT ESTIMATION

The input estimation or deconvolution problem can loosely be stated as follows: Construct an estimate of the input to a system based on observations of the output signal. The situation is depicted in Figure 2.1

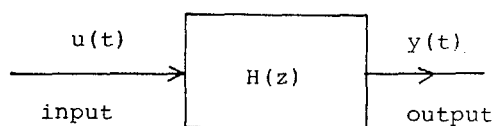


Figure 2.1 The input estimation problem. Estimate the input signal $u(t)$ based on the output signal $y(t)$.

In Figure 2.1 we assumed that the system is linear and for instance defined by its transfer function $H(z)$. Formally the solution to this problem is straightforward. If

$$y(t) = H(q^{-1})u(t), \quad (2.1)$$

we directly get that

$$u(t) = H^{-1}(q^{-1})y(t). \quad (2.2)$$

In (2.1) and (2.2) q^{-1} is the shift operator defined by

$$q^{-1}x(t) \triangleq x(t-1). \quad (2.3)$$

The estimate (2.2) suffers however from a serious drawback. If, as most often is the case, the signal

$y(t)$ contains noise a part of the estimate will also be filtered noise. The only way to improve the estimate (2.2) and reduce the influence of the noise is to have some information or assumptions of the nature of the input signal. This a priori information can be of various type. One can for instance assume that the input signal is a stationary white noise sequence. A more useful approach might be to assume that the signal we are looking for is a stochastic process with stationary characteristics. The input signal can then be modelled as white noise through a linear filter $G(q^{-1})$, see Figure 2.2.

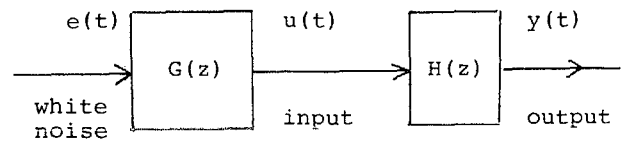


Figure 2.2 A model of the input with stationary characteristics.

This kind of problem is investigated in Ahlén (1984). If the characteristics of the unknown input $u(t)$ is known to be nonstationary some other kind of approach must be taken. We will here assume that a parametric model of the signal is available. A simple model of this type is

$$u(t) = A \sin(\omega t + \phi)$$

with unknown parameters A , ω and ϕ . A bit more complicated example is

$$u(t) = \begin{cases} y_1 + (y_2 - y_1) \frac{t - t_1}{t_2 - t_1} & t_1 \leq t < t_2 \\ y_2 + (y_3 - y_2) \sin^2\left(\frac{\pi}{2} \frac{t - t_2}{t_3 - t_2}\right) & t_2 \leq t < t_3 \\ y_4 + (y_3 - y_4) \cos\left(\frac{\pi}{2} \frac{t - t_3}{t_4 - t_3}\right) & t_3 \leq t \leq t_4 \end{cases} \quad (2.4)$$

with unknown parameters t_i , y_i , $i=1,2,3,4$. This is a model for the glottal wave suggested in Fant (1980). This model will be used in Section 4 in connection with the speech coding application. The input signal is thus known except for some unknown parameters, formally

$$u(t) = u(t, \alpha) \quad (2.5)$$

where α is a vector containing the unknown parameters.

In many situations not only the input signal is unknown but also the system $H(q^{-1})$ itself. This will of course further complicate the situation. In a situation where the input is assumed to be white noise, the coefficients of the filter $H(q^{-1})$ can be found by least-squares or LPC estimation, see Ljung and Söderström (1984) or see Markel and Gray (1976). In the case of stationary characteristics there is no hope of separating the properties of the input, i.e. the $G(q^{-1})$ filter, from the system $H(q^{-1})$. With a parametric model of a nonstationary signal the task is easier. One possible approach would then be the following.

1. First assume that the input is white noise and estimate $H(q^{-1})$ using LPC. Call this estimate $\hat{H}(q^{-1})$.
2. Perform an inverse filtering through $\hat{H}(q^{-1})$ i.e.



$$\varepsilon(t) = \hat{H}^{-1}(q^{-1})y(t) \quad (2.6)$$

3. Use some nonlinear optimization method to approximate $\varepsilon(t)$ with $u(t, \alpha)$. Formally solve

$$\hat{\alpha} = \arg \min_{\alpha} \sum_{t=0}^N (\varepsilon(t) - u(t, \alpha))^2 \quad (2.7)$$

and denote

$$\hat{u}(t) = u(t, \hat{\alpha}). \quad (2.8)$$

Then use $u(t, \hat{\alpha})$ as the input to improve the estimate of the filter $H(q^{-1})$. Make a new inverse filtering to obtain a new estimate of the input. Repeat the procedure until no further improvement can be observed. Conceptually the method alternates between optimizing over the parameters of the $H(q^{-1})$ filter and the α -parameters of the input. This method has been applied to the speech coding problem in Hedelin (1984). We will in the sequel show how the approximation over the θ and α parameters instead can be carried out simultaneously. We will here only consider $H(q^{-1})$ filters that are of all-pole type, i.e.

$$H(q^{-1}) = \frac{1}{A(q^{-1})} \quad (2.9a)$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_n q^{-n}. \quad (2.9b)$$

The input-output relationship corresponding to (2.9) is thus

$$y(t) + a_1 y(t-1) + \dots + a_n y(t-n) = u(t) + e(t) \quad (2.10)$$

In (2.10) we have introduced a noise term $e(t)$ to account for measurement noise, model uncertainties etc. In (2.9b) we will thus consider the a_i , $i=1, \dots, n$ parameters as unknown. Denote a vector containing these parameters with θ . Denote further a specific estimate of the θ vector with $\hat{\theta}$. To stress the dependence of the $H(q^{-1})$ filter on the a_i parameters we write

$$H(q^{-1}) = H(q^{-1}, \theta) \quad (2.11)$$

An estimate of the transfer function would accordingly be written as

$$\hat{H}(q^{-1}) = H(q^{-1}, \hat{\theta}). \quad (2.12)$$

Introduce further $\hat{\alpha}$ as an estimate of the α -vector and define $\hat{y}(t)$ as

$$\hat{y}(t, \hat{\theta}, \hat{\alpha}) = H(q^{-1}, \hat{\theta})u(t, \hat{\alpha}). \quad (2.13)$$

The input estimation problem is thus in this framework equivalent to estimating the θ and α vectors. If we use a quadratic goodness measure we can formally state the problem as

$$\min_{\theta, \alpha} \left\{ \sum_{t=1}^N (\hat{y}(t, \theta, \alpha) - y(t))^2 \right\} \quad (2.14)$$

The minimization in (2.14) consists of two parts

- Minimization over the θ -parameter which is a linear problem.
- Minimization over the α -parameter which in general is a nonlinear problem.

We will in the next section see how we can use the linearity in the θ part to obtain an algorithm.

3 FORMULATION OF THE ALGORITHM

In the previous section we saw how the input estimation problem could be stated as a minimization prob-

lem. We will now show how an efficient algorithm for solving this problem can be designed. It will prove to be convenient to rewrite the minimization problem (2.14) using matrix notation. Introduce therefore the regression vector $\varphi(t)$ defined by

$$\varphi(t) \triangleq \begin{pmatrix} -y(t-1) \\ -y(t-2) \\ \vdots \\ -y(t-n) \end{pmatrix} \quad (3.1)$$

Also recall the notation

$$\theta = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} \quad (3.2a)$$

The input-output relation (2.10) can then be written as

$$y(t) = \theta^T \varphi(t) + u(t) + e(t). \quad (3.2b)$$

Introduce further the vectors

$$Y_N \triangleq \begin{pmatrix} y(1) \\ \vdots \\ y(N) \end{pmatrix}, \quad U_N(\alpha) \triangleq \begin{pmatrix} u(1, \alpha) \\ \vdots \\ u(N, \alpha) \end{pmatrix} \quad (3.3)$$

The minimization criterion

$$V_N(\theta, \alpha) = \frac{1}{2} \sum_{t=1}^N (y(t) - \hat{y}(t, \theta, \alpha))^2 \quad (3.4)$$

can then be written as

$$V_N(\theta, \alpha) = \frac{1}{2} \|Y_N - \Phi_N \theta - U_N(\alpha)\|^2 \quad (3.5)$$

In (3.5) the norm is the standard Euclidian norm in R^N . The matrix Φ_N is defined as

$$\Phi_N \triangleq \begin{pmatrix} \varphi^T(1) \\ \vdots \\ \varphi^T(N) \end{pmatrix} \quad (3.6)$$

The plan of attack for solving the minimization problem

$$\min_{\theta, \alpha} V_N(\theta, \alpha) = \min_{\theta, \alpha} \frac{1}{2} \|Y_N - \Phi_N \theta - U_N(\alpha)\|^2 \quad (3.7)$$

is to take advantage of the linearity in the θ -parameters. It is well known that the problem

$$\min_{\theta} V_N(\theta, \alpha) = \min_{\theta} \frac{1}{2} \|Y_N - \Phi_N \theta - U_N(\alpha)\|^2 \quad (3.8)$$

can be given a closed form solution. A vector $\hat{\theta}$ with the property

$$\hat{\theta} = \arg \min_{\theta} V_N(\theta, \alpha) \quad (3.9)$$

is such that $\Phi_N \hat{\theta}$ is the projection of $Y_N - U_N(\alpha)$ on the column space of Φ_N . If one further among the vectors with the above property selects the one with the shortest length it can per definition be written as

$$\hat{\theta} = \Phi_N^+ (Y_N - U_N(\alpha)). \quad (3.10)$$

Φ_N^+ is the (Moore-Penrose) pseudo-inverse of Φ_N , see any textbook on least squares problems, for instance Dahlqvist and Björk (1974). There are many ways of calculating Φ_N^+ available. If the matrix Φ_N has full column rank i.e. $\text{rank } \Phi_N = n$ the pseudo inverse can for instance be calculated as

$$\Phi_N^+ = [\Phi_N^T \Phi_N]^{-1} \Phi_N^T \quad (3.11)$$



We can thus for each given α vector obtain a $\hat{\Theta} = \hat{\Theta}(\alpha)$ defined through

$$\hat{\Theta}(\alpha) = \arg \min_{\Theta} \frac{1}{2} \| Y_N - \Phi_N \Theta - U_N(\alpha) \|^2 = \Phi_N^+ \tilde{Y}_N(\alpha) \quad (3.12)$$

where

$$\tilde{Y}_N(\alpha) = Y_N - U_N(\alpha). \quad (3.13)$$

We can thus in (3.7) replace Θ with $\hat{\Theta}(\alpha)$ and just minimize over the α parameters. The dimension of the Θ vector is typically larger than the dimension of the α vector and we will thus considerably reduce the dimensionality of the problem. In a speech-coding application the dimension of the Θ vector is typically 8 or 10, see Markel and Gray (1976). By inserting (3.12) into (3.7) we get

$$V_N(\hat{\Theta}(\alpha), \alpha) = \frac{1}{2} \| \tilde{Y}_N(\alpha) - \Phi_N \Phi_N^+ \tilde{Y}_N(\alpha) \|^2 = \frac{1}{2} \| (I - \Phi_N \Phi_N^+) \tilde{Y}_N(\alpha) \|^2. \quad (3.14)$$

The matrix

$$I - \Phi_N \Phi_N^+$$

is the projection on the orthogonal complement to the column space of Φ_N and will in the sequel be denoted by Φ^- .

The problem

$$\min_{\alpha} V_N(\hat{\Theta}(\alpha), \alpha) \quad (3.15)$$

is a non-linear minimization problem and must be solved with some iterative technique like for instance quasi-Newton. What we have gained compared with the approach suggested in the previous section is that we do not have to solve for the Θ parameters more than once. Another advantage is that the minimization problem (3.15) can be solved using efficient and well understood numerical algebra methods.

To implement for instance a quasi-Newton search procedure we must besides the criterion function itself also calculate its gradient with respect to α . We must thus be able to calculate the derivatives

$$\begin{aligned} \frac{\partial V_N(\hat{\Theta}(\alpha), \alpha)}{\partial \alpha_i} &= \frac{\partial}{\partial \alpha_i} (\Phi^- \tilde{Y}_N(\alpha))^T (\Phi_N^- \tilde{Y}_N(\alpha)) = \\ &= -(\Phi_N^- \tilde{Y}_N(\alpha))^T \Phi_N^- \frac{\partial}{\partial \alpha_i} U_N(\alpha). \end{aligned} \quad (3.16)$$

To calculate both the criterion and its gradient is thus straightforward once we have a way of calculating Φ^- . An efficient way of doing this is to use some factorization method, e.g. QR factorization. It is well-known that an $m \times n$ matrix Φ with rank n can be given a so called QR factorization

$$\Phi = QR \quad (3.17)$$

In (3.17) Q is an orthogonal matrix i.e.

$$Q^T Q = I \quad (3.18)$$

and

$$R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix} \quad (3.19)$$

where R_1 is an $n \times n$ upper triangular, nonsingular matrix, see for instance Strang (1980). To calculate Φ^- is then straightforward since

$$\Phi^- = I - \Phi \Phi^+ = I - \Phi [\Phi^T \Phi]^{-1} \Phi^T = I - QR [R^T Q^T QR]^{-1} R^T Q^T$$

$$= Q \begin{pmatrix} 0 & 0 \\ 0 & I \end{pmatrix} Q^T. \quad (3.20)$$

Let us further decompose $Q^T \tilde{Y}_N(\alpha)$ into

$$\begin{aligned} h_1(\alpha) \\ h_2(\alpha) \end{aligned} \quad (3.21)$$

where $h_1(\alpha)$ is $n \times 1$ and $h_2(\alpha)$ $(m-n) \times 1$ matrices. We then have

$$\begin{aligned} V_N(\hat{\Theta}(\alpha), \alpha) &= (\Phi^- \tilde{Y}_N(\alpha))^T (\Phi^- \tilde{Y}_N(\alpha)) = \\ &= (Q^T \tilde{Y}_N(\alpha))^T \begin{pmatrix} 0 & 0 \\ 0 & I \end{pmatrix} Q^T \tilde{Y}_N(\alpha) = h_2^T(\alpha) h_2(\alpha) \end{aligned} \quad (3.22)$$

We also get

$$\begin{aligned} \frac{\partial}{\partial \alpha_i} V_N(\hat{\Theta}(\alpha), \alpha) &= (Q^T \tilde{Y}_N(\alpha))^T \begin{pmatrix} 0 & 0 \\ 0 & I \end{pmatrix} Q^T \frac{\partial}{\partial \alpha_i} U_N(\alpha) = \\ &= \Psi^T(\alpha) \frac{\partial}{\partial \alpha_i} U_N(\alpha) \end{aligned} \quad (3.23)$$

where $\Psi(\alpha)$ is defined as

$$\Psi(\alpha) = Q \begin{pmatrix} 0 \\ h_2(\alpha) \end{pmatrix}. \quad (3.24)$$

To calculate the criterion and its gradient is thus quite straightforward. The only nontrivial step is the QR-factorization.

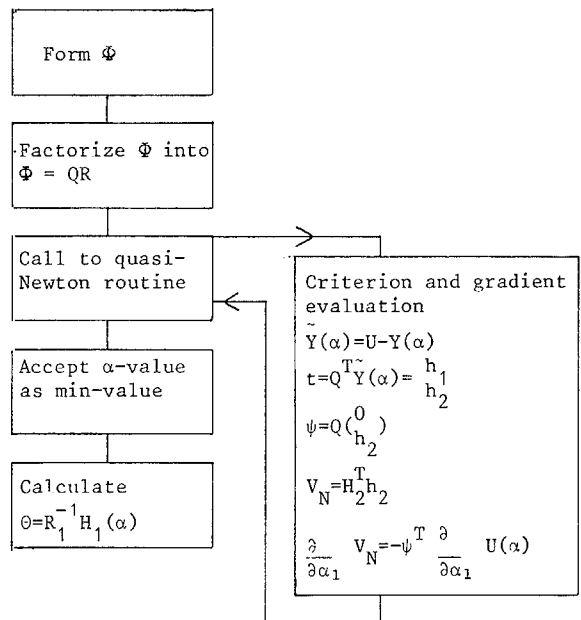


Figure 3.1 A flowchart showing the basic steps of a quasi-Newton search procedure.

There are many methods and software packages available for this task. In the applications described in the next section we have used Householder transformations and the FORTRAN subroutine package LINPACK, Dongarra et al (1982). In Figure 3.1 a flowchart of the basic steps of an implementation of a Quasi-Newton search procedure is given. We note from Figure 3.1 that the QR factorization only has to be performed once. Note also that the estimate of the Θ parameters is only formed explicitly after the minimization with respect to the α parameters is completed.

4 APPLICATION TO SPEECH CODING

In this section we will apply the optimization method presented in the previous section to a speech coding problem. The standard way to do speech coding is the so called Linear Predictive Coding (LPC), see e.g. Markel and Gray (1976). In LPC we use a linear production model for the pre-emphasized speech signal $y(t)$, i.e.

$$A(q)y(t) = e(t).$$

We define the filter $A(q)$ as

$$A(q) = 1 + \sum_{k=1}^n a_k q^{-k},$$

where q^{-1} is the backward shift operator. The order n typically takes the value 8 or 10. In case of voiced speech we assume $e(t)$ to be periodic impulses of zero length. However, this is of course an approximation of the real glottal pulse. A more exact model looks like the parametrization (2.4) depicted in Figure (4.1). See Fant (1980).

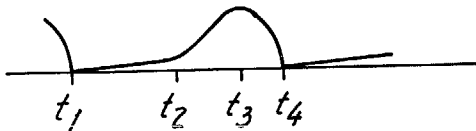


Figure 4.1 The glottal pulse.

It has been observed that an improvement in the synthesised speech is gained by using this more exact model, see Hedelin (1984). To utilize this parametrization we reformulate the LPC-model as

$$A(q)y(t) = u(t, \alpha) + e(t)$$

where $e(t)$ now is white noise. Our problem is then to estimate both $A(q)$ and $u(t, \alpha)$ using the sampled speech data $y(t)$. One solution to the problem, discussed in Section 3, was suggested by Hedelin (1984). If we instead apply the algorithm described in Section 3, we get an efficient method to simultaneously estimate $A(q)$ and α .

5 TEST RESULTS

The glottal LPC-vocoder described in Section 4 has been implemented and tested on recorded speech data. Below we present a typical example. The data originates from a recording of the vowel "i", like the one in the English word beach.

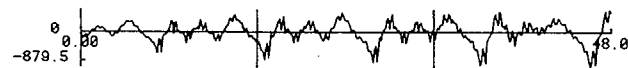


Figure 5.1a Sampled data.

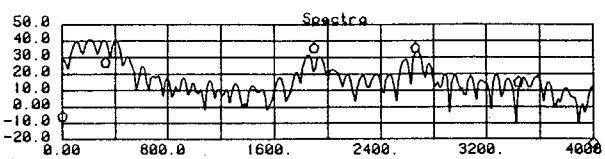


Figure 5.1b Spectrum of data and initial formants.

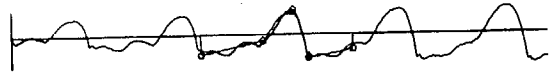


Figure 5.1c Initial residuals and glottal parameters.

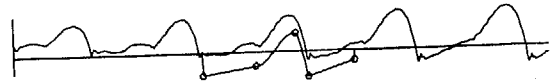


Figure 5.1d Optimized residuals and glottal parameters.

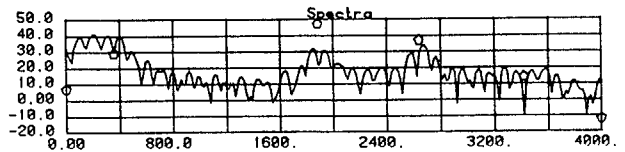


Figure 5.1e Spectrum of data and optimal formants.

We now use a decibel measure of the improvement compared to pure LPC. Define

$$\text{Improvement} = 10 \log(V_{LPC}/V_{OPT}),$$

where V_{LPC} and V_{OPT} are the accumulated loss functions of LPC and our method respectively. In Table 5.1 we present a sample of results from another recording of a whole sentence. Together with the improvement measure we also present the CPU-time used on a DEC-2060 computer to calculate the optimal estimate.

Frame no	Improvement (dB)	CPU-time (ms)
1	0.45	2560
4	0	1840
9	0	1692
12	4.00	5429
13	4.07	5719
17	4.49	4062
19	4.29	10562
20	1.97	6857
23	7.16	5450
25	1.93	5799

Table 5.1.

As we can see the algorithm gives a considerable improvement compared to the standard LPC algorithm. It can be observed that for a few formants the improvement is negligible. The explanation to this is that these frames do not correspond to global waves with the structure assumed by our parametrization, hence our method has nothing to offer. This leads to the question of how to handle frames that do not meet the model assumptions. This question is however beyond the scope of this paper.

Another interesting problem leading to the same question is the possible existence of local minima. We have tried to examine this, by plotting the goodness measure $V_N(\theta, \alpha)$ given by (3.4) as a function of the glottal parameters α . In Figure 5.2 we present one of these plots, where V_N is plotted against the closing time t_4 of the vocal folds and the corresponding amplitude y_4 .

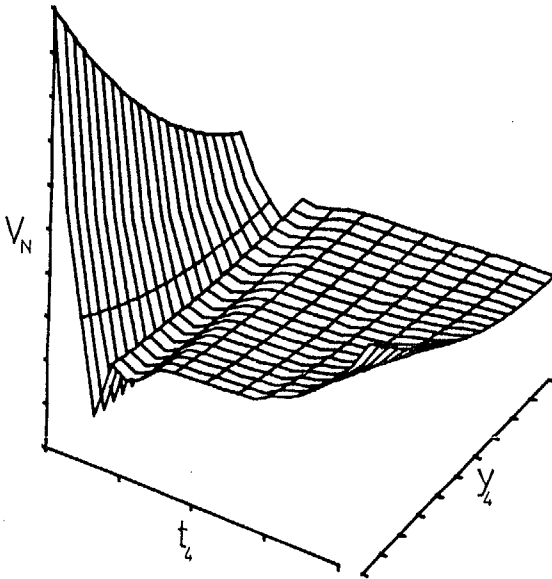


Figure 5.2a 3-D plot of loss function V_N .

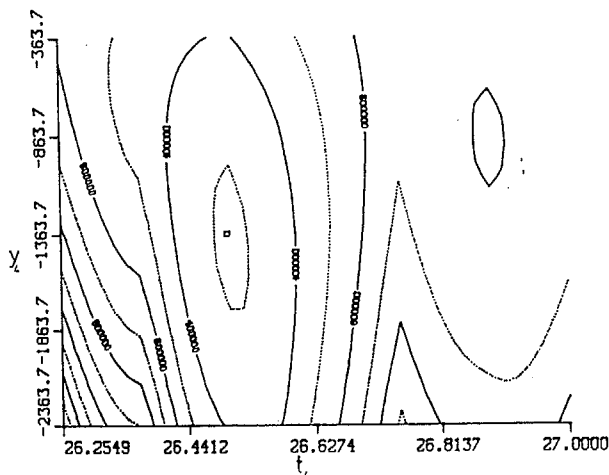


Figure 5.2b Level curves of V_N .

As we can see there seems to be a local minimum in the upper right hand corner of Figure 5.2b. To test that it is not a saddle point, the optimization was started in points in the vicinity of this suspected local minimum. We then actually found starting points which converged to this point, confirming our suspicions.

Also notice in Figure 5.2 that the loss function V_N is far more sensitive to changes in the time parameter t_4 , than to changes in the amplitude y_4 . For more details of test results and implementation aspects see Isaksson and Millnert (1984).

6. CONCLUSIONS

We have in this paper shown how the input signal and system parameters can be estimated simultaneously. The major advantage of the algorithm is that it handles the involved computations in an efficient way. The application in this paper was a problem from speech coding to estimate the parameters of a glottal pulse together with the vocal tract parameters. The algorithm as such is however not limited to this class of problems. The approach might for instance be suitable for the Geoscience problem described in Mendel (1978).

As mentioned earlier, some problems were connected with the speech coding application. We mentioned the problem of how to handle frames with no pronounced glottis signal and the question of local minima. It should be stressed that those problems are not related to the estimation algorithm as such. For this type of questions one should instead focus on the parametrization of the glottal pulse.

REFERENCES

- Ahlén, A. (1984). An input estimation approach to differentiation of noise data. Report UPTec 8450R, Institute of Technology, Uppsala University, Uppsala, Sweden.
- Dahlquist, G. and Å. Björk (1974). Numerical methods, Prentice-Hall.
- Dongarra, J. J. et al (1979). LINPACK User's Guide, SIAM.
- Fant, G. (1980). Voice source dynamics, STL-QPSR, Vol 2-3.
- Golub, G. H. and V. Pereyra (1973). The differentiation of Pseudo-inverses and nonlinear least squares problems whose variables separate. SIAM J. Numer. Anal. Vol 10, No 2 p 413.
- Hedelin, P. (1984). A Glottal LPC-vocoder, Proceedings ICASSP,84, San Diego.
- Isaksson, A. and M. Millnert (1984). Input estimation with application to speech coding, Report LiTH-ISY-I-0696, Department of Electrical Engineering, Linköping University, Linköping, Sweden.
- Ljung, L. and T. Söderström (1983). Theory and Practice of Recursive Identification, MIT Press.
- Markel, J. D. and A. H. Gray (1976). Linear Prediction of Speech, Springer-Verlag.
- Mendel, J., N. E. Nahi and M. Chan (1979). Synthetic seismograms using the state-space approach, Geophysics, Vol 44, No 5 p 880-895.
- Strang, G. (1980). Linear Algebra and its Applications, Academic Press.