SPARSE VECTOR LINEAR PREDICTION MATRICES WITH MULTIDIAGONAL STRUCTURE

Davor Petrinović, Davorka Petrinović

Faculty of Electrical and Computer Engineering
Department of Electronic Systems and Information Processing
University of Zagreb, Croatia
davor.petrinovic@fer.hr, davorka.petrinovic@fer.hr

ABSTRACT

A modification of the classical vector linear prediction (VLP) problem is presented. The introduced technique called the sparse VLP (sVLP) is based on the assumption that each component of a single LSF vector is highly correlated only to a few neighboring vector components of consecutive vectors, while the correlation between distant vector components can be ignored. This leads to simplification of predictor matrices in a way that for a chosen number of neighboring components, predictor matrices obtain multidiagonal form. It is shown that prediction gain resulting from sVLP is only slightly lower than for the case of full matrix predictors but with significant reduction of computation, both for coding and predictor design.

1. INTRODUCTION

One of the popular techniques for decorrelation of the vector processes is the Vector Linear Prediction (VLP). It is a vector generalization of scalar linear prediction that is widely accepted for most speech applications. VLP is usually employed in the Vector Predictive Coding [1] that is again a vector generalization of the DPCM scalar coding. VLP has been applied to several signal processing areas, but this paper will focus mainly on its application to the interframe predictive coding of the LPC parameters. Due to the stationary behavior of the speech signal during sustained phonemes, successive LPC parameter vectors are highly correlated. Switched-adaptive interframe vector prediction (SIVP) [2] and covariance and autocorrelation frame-adaptive methods [3] are some of techniques based on VLP, that remove this correlation (or redundancy) prior to coding. Although the VLP maximally exploits intraframe as well as interframe correlation between successive speech parameter vectors, it is computationally very intensive and the design process may become ill-conditioned for higher prediction orders [3] thus requiring the utilization of sophisticated algorithms for matrix inversion.

For certain classes of vector processes the dominant correlation is the one between same vector components of consecutive vectors. Sometimes the correlation between one component and several neighboring components is also significant and should be exploited while the correlation between distant vector components can be ignored. This fact is utilized in the technique proposed in this paper that is named sparse VLP (sVLP). It is a VLP technique modified in a way that the prediction of a single vector component is not based on all other vector components from the previous vectors but only on a few components of each vector that are close neighbors to the one being predicted. In other words, the predictor matrices have nonzero elements only on the main diagonal and several auxiliary diagonals. Therefore it is named sparse vector prediction. It will be shown that sVLP results in only slightly lower prediction gain than the conventional VLP, but with several times lower computational complexity.

2. CLASSICAL FORMULATION OF THE VLP

Although a very comprehensive formulation of the VLP problem was presented in [3], the main equations will be repeated in this section since the proposed technique is based on the modification of the original expressions.

An M-th order vector linear predictor that predicts the current k-dimensional vector $\mathbf{x}(n)$ [#] based on its M preceding vectors, is defined with M predictor matrices $\mathbf{A}_j \in \mathbf{R}^{k\times k}$, $j = 1, 2, \ldots, M$ so the predicted vector $\hat{\mathbf{x}}(n)$ is obtained by:

$$\hat{\mathbf{x}}(n) = \sum_{j=1}^{M} \mathbf{A}_j \mathbf{x}(n-j). \tag{1}$$

The vector predictor defined in (1) can be reformulated as k scalar predictors, each resulting with a prediction of a single vector component. So, the predicted value of the i-th vector component $\hat{x}_i(n)$ can be expressed as:

$$\hat{x}_i(n) = \sum_{j=1}^{M} \mathbf{a}_{ji}^{\mathrm{T}} \mathbf{x}(n-j), \quad i = 1, 2, \ldots, k \tag{2}$$

where $\mathbf{a}_{ji}^{\mathrm{T}}$ represents the i-th row of each of the predictor matrices $\mathbf{A}_j$. The prediction residual vector, $\mathbf{e}(n)$, is defined as the difference between the actual vector $\mathbf{x}(n)$ and the predicted vector $\hat{\mathbf{x}}(n)$ so the i-th component of $\mathbf{e}(n)$ can be expressed as:

$$e_i(n) = x_i(n) - \sum_{j=1}^{M} \mathbf{a}_{ji}^{\mathrm{T}} \mathbf{x}(n-j). \tag{3}$$

If the equation (3) is evaluated on a given time interval $n \in [1,P]$, it can be expressed in the matrix form as:

$$\mathbf{f}_i = \mathbf{y}_i - \mathbf{Z}\mathbf{b}_i \tag{4}$$

where $\mathbf{Z}$, $\mathbf{f}_i$, $\mathbf{y}_i$ and $\mathbf{b}_i$ are defined as follows:

$$\mathbf{Z} = \begin{bmatrix} \mathbf{x}^{\mathrm{T}}(0) & \mathbf{x}^{\mathrm{T}}(-1) & \cdots & \mathbf{x}^{\mathrm{T}}(1-M) \\ \mathbf{x}^{\mathrm{T}}(1) & \mathbf{x}^{\mathrm{T}}(0) & \cdots & \mathbf{x}^{\mathrm{T}}(2-M) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{x}^{\mathrm{T}}(P-1) & \mathbf{x}^{\mathrm{T}}(P-2) & \cdots & \mathbf{x}^{\mathrm{T}}(P-M) \end{bmatrix} \in \mathbf{R}^{P\times Mk}, \tag{5}$$

[#] Throughout the paper, vectors without transposition are assumed to be column vectors.

$$\mathbf{f}_i = [\, e_i(1), e_i(2), \ldots, e_i(P) \,]^{\mathrm{T}} \in \mathbf{R}^P, \tag{6}$$

$$\mathbf{y}_i = [\, x_i(1), x_i(2), \ldots, x_i(P) \,]^{\mathrm{T}} \in \mathbf{R}^P, \tag{7}$$

$$\mathbf{b}_i = [\, \mathbf{a}_{1i}^{\mathrm{T}}, \mathbf{a}_{2i}^{\mathrm{T}}, \ldots, \mathbf{a}_{Mi}^{\mathrm{T}} \,]^{\mathrm{T}} \in \mathbf{R}^{Mk}. \tag{8}$$

Minimization of the squared prediction error of the i-th component of $\mathbf{e}(n)$ leads to optimal predictor coefficients in row vectors $\mathbf{a}_{ji}^{\mathrm{T}}$, found by solving the normal equation:

$$\mathbf{Z}^{\mathrm{T}}\mathbf{Z}\mathbf{b}_i = \mathbf{Z}^{\mathrm{T}}\mathbf{y}_i . \tag{9}$$

If the equation (9) is expressed for all vector components $i = 1, 2, \ldots, k$, vectors $\mathbf{b}_i$ and $\mathbf{y}_i$ are replaced by matrices $\mathbf{B}$ and $\mathbf{Y}$. After performing matrix multiplication $\mathbf{Z}^{\mathrm{T}}\mathbf{Z}$ and $\mathbf{Z}^{\mathrm{T}}\mathbf{Y}$, (9) can be compactly written as:

$$\sum_{j=1}^{M} \mathbf{C}_{mj}\mathbf{A}_j^{\mathrm{T}} = \mathbf{C}_{m0}, \quad m = 1, 2, \ldots, M \tag{10}$$

where

$$\mathbf{C}_{mj} = \sum_{n=1}^{P} \mathbf{x}(n-m)\,\mathbf{x}^{\mathrm{T}}(n-j) \quad \in \mathbf{R}^{k\times k}. \tag{11}$$

If P is chosen to be equal to the number of vectors in the analysis frame, $P = L$, then (10) and (11) represent covariance formulation of VLP.

The autocorrelation method of VLP can also be expressed from (9), taking into account that vectors $\mathbf{x}(n)$ outside the analysis frame $n \in [1, L]$ are intentionally set to zero so the prediction error is defined on the interval $[1, P]$ where $P = L + M$. The solution of the autocorrelation method is determined by:

$$\sum_{j=1}^{M} \mathbf{R}_{|m-j|}\mathbf{A}_j^{\mathrm{T}} = \mathbf{R}_m \quad m = 1, 2, \ldots, M \tag{12}$$

where

$$\mathbf{R}_m = \sum_{n=1}^{L-m} \mathbf{x}(n)\,\mathbf{x}^{\mathrm{T}}(n+m) \quad \in \mathbf{R}^{k\times k}. \tag{13}$$

3. sVLP TECHNIQUE

The proposed sVLP technique represents the modification of the classical formulation of the VLP problem in the sense that it exploits intraframe correlation only between few neighboring vector components. Since it works on subvectors of the original vector process, the symbol $\breve{\mathbf{x}}_i(n)$ will be used to denote a subvector of $\mathbf{x}(n)$ enclosing the i-th component and is obtained by:

$$\breve{\mathbf{x}}_i(n) = \mathbf{I}_1(i)\mathbf{x}(n) \tag{14}$$

where $\mathbf{I}_1(i) \in \mathrm{R}^{(u(i)-v(i)+1)\times k}$ is a linear operator that extracts rows $v(i)$ to $u(i)$ from vector $\mathbf{x}(n)$ and is given by:

$$\mathbf{I}_1(i) = \begin{array}{c} \begin{matrix} 1 & \ldots & v-1 & v & \ldots & i & \ldots & u & u+1 & \ldots & k \end{matrix} \\ \begin{bmatrix} 0 & \ldots & 0 & 1 & 0 & & & \ldots & \ldots & & 0 \\ & & & \ddots & \ddots & \ddots & & & & & \\ \vdots & & & & 0 & 1 & 0 & & & & \vdots \\ & & & & & \ddots & \ddots & \ddots & & & \\ 0 & & \ldots & \ldots & & & 0 & 1 & 0 & \ldots & 0 \end{bmatrix} \end{array}. \tag{15}$$

In another words, vector $\breve{\mathbf{x}}_i(n)$ is comprised of those components of $\mathbf{x}(n)$ that are positioned around the i-th component where indexes $v(i)$ and $u(i)$ determine the first and the last extracted component.

Let the integer d define the neighborhood of the i-th component in a way that (d-1) components above and (d-1) components below the i-th component are extracted from $\mathbf{x}(n)$. The indexes $v(i)$ and $u(i)$ can thus be expressed as:

$$v(i) = \max(1, i-(d-1)), \tag{16}$$

$$u(i) = \min(i+(d-1), k). \tag{17}$$

The expressions (16) and (17) also take in the account the special cases when the index i is either close to 1 or close to k therefore some symmetrical components enclosing the i-th component do not exist. Thus, the dimension $D(i)$ (18) of vector $\breve{\mathbf{x}}_i(n)$ varies, depending on index i, from d to (2d-1).

$$D(i) = u(i) - v(i) + 1 \tag{18}$$

Linear prediction of the i-th component of the vector $\mathbf{x}(n)$, denoted as $\hat{x}_i'(n)$, can be calculated from M preceding subvectors $\{\breve{\mathbf{x}}_i(n-j)\}, j = 1, 2, \ldots, M$, according to:

$$\hat{x}_i'(n) = \sum_{j=1}^{M} \breve{\mathbf{a}}_{ji}^{\mathrm{T}}\breve{\mathbf{x}}_i(n-j), \quad i = 1, 2, \ldots, k. \tag{19}$$

This equation can also be given for all k vector components simultaneously in a conventional matrix form as in (1), with prediction based on $\mathbf{x}(n$-1) to $\mathbf{x}(n$-$M)$ rather then on subvectors, as in:

$$\hat{\mathbf{x}}'(n) = \sum_{j=1}^{M} \mathbf{A}_j'\mathbf{x}(n-j) \tag{20}$$

but with predictor matrices $\mathbf{A}_j'$ of the multidiagonal form:

$$\begin{aligned} \mathbf{A}_j' &= \{a_{im}\}_j \qquad i = 1, 2, \ldots, k \\ &\qquad\qquad\quad m = 1, 2, \ldots, k \\ a_{im} &\neq 0, \quad |i-m| < d \\ a_{im} &= 0, \quad |i-m| \geq d \end{aligned} \tag{21}$$

The number of diagonals of $\mathbf{A}_j'$ with nonzero coefficients is (2d-1), while all other elements of matrices are equal to zero.

With $\mathbf{A}_j'$ defined as in (21), the i-th row of predictor matrices, $\breve{\mathbf{a}}_{ji}^{\mathrm{T}}$(19), can be expressed as in (22), where the elements of $\breve{\mathbf{a}}_{ji}^{\mathrm{T}}$ correspond to nonzero elements of the i-th row of the respective predictor matrix $\mathbf{A}_j'$:

$$\breve{\mathbf{a}}_{ji}^{\mathrm{T}} = \mathbf{I}_2(i)\,\mathbf{A}_j'\,\mathbf{I}_1^{\mathrm{T}}(i). \tag{22}$$

Operator $\mathbf{I}_2(i) \in \mathbf{R}^k$ is defined as:

$$\mathbf{I}_2(i) = \begin{matrix} 1 & & i-1 & i & i+1 & & k \end{matrix} \\ \begin{bmatrix} 0 & \ldots & 0 & 1 & 0 & \ldots & 0 \end{bmatrix}. \tag{23}$$

The expression for sVLP technique analogous to (4) is:

$$\mathbf{f}_i = \mathbf{y}_i - \breve{\mathbf{Z}}_i\breve{\mathbf{b}}_i \tag{24}$$

where $\breve{\mathbf{Z}}_i \in \mathbf{R}^{P\times MD(i)}$ and $\breve{\mathbf{b}}_i \in \mathbf{R}^{MD(i)}$ are equal to:

$$\breve{\mathbf{Z}}_i = \begin{bmatrix} \breve{\mathbf{x}}_i^{\mathrm{T}}(0) & \breve{\mathbf{x}}_i^{\mathrm{T}}(-1) & \cdots & \breve{\mathbf{x}}_i^{\mathrm{T}}(1-M) \\ \breve{\mathbf{x}}_i^{\mathrm{T}}(1) & \breve{\mathbf{x}}_i^{\mathrm{T}}(0) & \cdots & \breve{\mathbf{x}}_i^{\mathrm{T}}(2-M) \\ \vdots & \vdots & \vdots & \vdots \\ \breve{\mathbf{x}}_i^{\mathrm{T}}(P-1) & \breve{\mathbf{x}}_i^{\mathrm{T}}(P-2) & \cdots & \breve{\mathbf{x}}_i^{\mathrm{T}}(P-M) \end{bmatrix} \tag{25}$$

$$\breve{\mathbf{b}}_i = [\, \breve{\mathbf{a}}_{1i}^{\mathrm{T}}, \breve{\mathbf{a}}_{2i}^{\mathrm{T}}, \ldots, \breve{\mathbf{a}}_{Mi}^{\mathrm{T}} \,]^{\mathrm{T}} \tag{26}$$

so the normal equation (9) becomes:

$$\breve{\mathbf{Z}}_i^{\mathrm{T}}\breve{\mathbf{Z}}_i\breve{\mathbf{b}}_i = \breve{\mathbf{Z}}_i^{\mathrm{T}}\mathbf{y}_i \tag{27}$$

Expression (27) can be compactly written in a form similar to (10) as:

$$\begin{bmatrix} \breve{\mathbf{C}}_{11,i} & \breve{\mathbf{C}}_{12,i} & \cdots & \breve{\mathbf{C}}_{1M,i} \\ \breve{\mathbf{C}}_{21,i} & \breve{\mathbf{C}}_{22,i} & \cdots & \breve{\mathbf{C}}_{2M,i} \\ \vdots & \vdots & \vdots & \vdots \\ \breve{\mathbf{C}}_{M1,i} & \breve{\mathbf{C}}_{M2,i} & \cdots & \breve{\mathbf{C}}_{MM,i} \end{bmatrix} \begin{bmatrix} \breve{\mathbf{a}}_{1i} \\ \breve{\mathbf{a}}_{2i} \\ \vdots \\ \breve{\mathbf{a}}_{Mi} \end{bmatrix} = \begin{bmatrix} \breve{\mathbf{c}}_{10,i} \\ \breve{\mathbf{c}}_{20,i} \\ \vdots \\ \breve{\mathbf{c}}_{M0,i} \end{bmatrix} \tag{28}$$

where

$$\breve{\mathbf{C}}_{mj,i} = \sum_{n=1}^{P} \breve{\mathbf{x}}_i(n-m)\,\breve{\mathbf{x}}_i^{\mathrm{T}}(n-j) \;\in\; \mathbf{R}^{D(i)\times D(i)}, \tag{29}$$

$$\breve{\mathbf{c}}_{m0,i} = \sum_{n=1}^{P} \breve{\mathbf{x}}_i(n-m)\,\breve{x}_i(n) \;\in\; \mathbf{R}^{D(i)}. \tag{30}$$

By solving the matrix equation (28), the nonzero coefficients of the i-th row of all predictor matrices $\mathbf{A}'_j$, j = 1, 2, …, M, are obtained. This procedure is repeated for all rows (i = 1, 2, …, k).

The elements of matrices $\breve{\mathbf{C}}_{mj,i}$ in (29) can be obtained from classical VLP covariance matrices $\mathbf{C}_{mj}$ (11) by extracting submatrices along the main diagonal, that corresponds to:

$$\breve{\mathbf{C}}_{mj,i} = \mathbf{I}_1(i)\,\mathbf{C}_{mj}\,\mathbf{I}_1^{\mathrm{T}}(i). \tag{31}$$

Analogously, the vectors $\breve{\mathbf{c}}_{m0,i}$ can be obtained from $\mathbf{C}_{m0}$:

$$\breve{\mathbf{c}}_{m0,i} = \mathbf{I}_1(i)\,\mathbf{C}_{m0}\,\mathbf{I}_2^{\mathrm{T}}(i). \tag{32}$$

If the submatrices $\breve{\mathbf{C}}_{mj,i}$ and $\breve{\mathbf{C}}_{mj,i+1}$ of two successive vector components are examined, it can be observed that they differ from each other in at most (4d-3) elements. Therefore, from the aspect of computational efficiency, the equation (31) is preferred compared to (29). Furthermore, not all elements of $\mathbf{C}_{mj}$ need to be computed since only (4d-3) central diagonals of $\mathbf{C}_{mj}$ are required in (31).

Expressions (28), (29) and (30) with P equal to the number of vectors in the analysis frame, L, define the covariance formulation for sVLP technique.

If the conditions for the autocorrelation method are applied to the matrix $\breve{\mathbf{Z}}_i$, with $P = L + M$, the product $\breve{\mathbf{Z}}_i^{\mathrm{T}}\breve{\mathbf{Z}}_i$ results with a supermatrix comprised of matrices:

$$\breve{\mathbf{R}}_{m,i} = \sum_{n=1}^{L-m} \breve{\mathbf{x}}_i(n)\,\breve{\mathbf{x}}_i^{\mathrm{T}}(n+m) \;\in\; \mathbf{R}^{D(i)\times D(i)} \tag{33}$$

while the product $\breve{\mathbf{Z}}_i^{\mathrm{T}}\mathbf{y}_i$ consists of vectors:

$$\breve{\mathbf{r}}_{m,i} = \sum_{n=1}^{L-m} \breve{\mathbf{x}}_i(n)\,x_i(n+m) \;\in\; \mathbf{R}^{D(i)} \tag{34}$$

so the matrix equations for solving predictor coefficients of the i-th row of each of the predictor matrices $\mathbf{A}'_j$ are:

$$\begin{bmatrix} \breve{\mathbf{R}}_{0,i} & \breve{\mathbf{R}}_{1,i}^{\mathrm{T}} & \cdots & \breve{\mathbf{R}}_{M-1,i}^{\mathrm{T}} \\ \breve{\mathbf{R}}_{1,i} & \breve{\mathbf{R}}_{0,i} & \cdots & \vdots \\ \vdots & \vdots & \vdots & \breve{\mathbf{R}}_{1,i}^{\mathrm{T}} \\ \breve{\mathbf{R}}_{M-1,i} & \cdots & \breve{\mathbf{R}}_{1,i} & \breve{\mathbf{R}}_{0,i} \end{bmatrix} \begin{bmatrix} \breve{\mathbf{a}}_{1i} \\ \breve{\mathbf{a}}_{2i} \\ \vdots \\ \breve{\mathbf{a}}_{Mi} \end{bmatrix} = \begin{bmatrix} \breve{\mathbf{r}}_{1,i} \\ \breve{\mathbf{r}}_{2,i} \\ \vdots \\ \breve{\mathbf{r}}_{M,i} \end{bmatrix} \tag{35}$$

The submatrices $\breve{\mathbf{R}}_{m,i}$ and subvectors $\breve{\mathbf{r}}_{m,i}$ can also be expressed in terms of $\mathbf{R}_m$:

$$\breve{\mathbf{R}}_{m,i} = \mathbf{I}_1(i)\,\mathbf{R}_m\,\mathbf{I}_1^{\mathrm{T}}(i), \tag{36}$$

$$\breve{\mathbf{r}}_{m,i} = \mathbf{I}_1(i)\,\mathbf{R}_m\,\mathbf{I}_2^{\mathrm{T}}(i). \tag{37}$$

4. EXPERIMENTAL RESULTS

The VLP technique employing sparse predictor matrices was evaluated using SIVP [2] scheme with first order prediction and up to 32 switched matrices. The LPC speech model was represented with Line Spectrum Frequencies (LSF) since this parameter set shows very good properties for interframe coding [2].

The parameters of the database used for evaluation were: 4 speakers (2 male/2 female) 5 minutes each, autocorrelation LPC method with order k =10, with pre-emphasis, sampling rate 8 kHz, and 25ms Hamming window with frame rate of 50 frames/s.

The switched-predictor matrices were designed by an iterative technique similar to the one used in [4] but with no quantizer in the loop. The design procedure starts by initializing the predictor matrices with values obtained for example, by classification based on averages of the instantaneous correlation vectors [2]. The iterations are then performed using all input vectors as follows. For each input vector, the predictor is selected based on the highest prediction gain and the covariance matrices of the selected predictor are updated with outer vector products. At the end of each iteration, a new set of prediction matrices is calculated from accumulated covariance matrices, and is then used for next iteration. The iteration procedure terminates upon reaching a chosen minimal difference of prediction gain between two successive iterations.

If the sum of squared components of each predictor matrix obtained by classical VLP is calculated along the main diagonal and compared to the total matrix sum, it can be observed, that the main diagonal alone contains most of the predictor energy. As an illustration, for the case of 2 switched matrices, the 'strong correlation' matrix comprises almost 99% and the 'weak correlation' matrix nearly 81% of the total predictor energy. For the case of two additional auxiliary diagonals (5-diagonal case) , these numbers rise to 99,5% and 86% respectively. This fact motivates consideration of sparse matrix technique.

To investigate the performance of sVLP, the open-loop prediction gain was calculated for different predictor matrix forms and different number of switched matrices. The number of diagonals was varied from a single diagonal (d = 1) up to the full matrix (d = 10). For each number of diagonals, SIVP procedure was performed with 2, 4, 8, 16 and 32 switched matrices, all having the same structure. The resulting prediction gain curves are shown on Figure 1. with dotted lines. It can be observed that the maximum gain loss due to multidiagonal structure is for the case of 32 predictors. For example, for 5-diagonal matrices (d = 3), the loss is 0.12 dB and 0.37 dB for 2 and 32 predictors, respectively.

Since the sVLP technique disregards some of the correlation still present between components of the same vector, the performance can even be improved by removing this intraframe correlation prior to SIVP. This can be performed by multiplying the LSF vectors with a decorrelation matrix such as Discrete Cosine Transform (DCT) or Karhunen Loeve Transform (KLT) matrix. Both types of transformations were applied in sVLP evaluation. The simulations were repeated the same way as in the case with no intraframe decorrelation and the results are also shown in Figure 1., with solid lines for KLT and dashed lines for DCT. It is obvious that both transformations improve the resulting prediction gain,

with KLT being the best. The most significant improvements are obtained for cases with 1 and 3 diagonal matrices, which are, in fact, of the greatest interest. It can be observed that the prediction gain for 3-diagonal prediction with KLT is almost identical to the 5-diagonal case without intraframe decorrelation. Since the transformation is performed only once, the computational load of this transformation is negligible for higher number of switched matrices compared to the rest of the calculations. For example, if 8 or more predictors are used, than the 3-diagonal prediction with KLT is preferred to 5-diagonal case without intraframe decorrelation due to less computation.

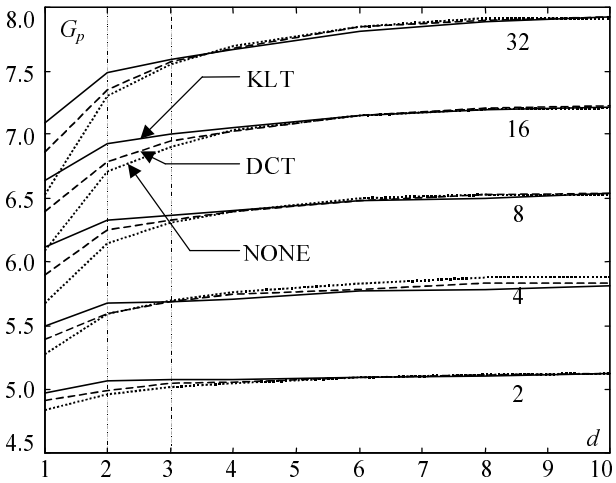

Figure 1. Prediction gain vs. variable d; for optimized multidiagonal predictors with no transformation (dotted) and combined with fixed intraframe decorrelation, KLT (solid) and DCT (dashed), for SIVP with 2 to 32 switched matrices.

The sVLP technique can be characterized as a method of suboptimal prediction where the predictor coefficients are optimized based on a given multidiagonal structure of the predictor. This can be further verified if the results for sVLP are compared to those obtained by the classical VLP method but with predictor matrices that are forced to be of the same multidiagonal form, that is performed by zeroing all the elements outside the central ($2d$-1) diagonals. The prediction gain for sVLP with no interframe decorrelation (solid line) and for classical VLP with the restrictions as stated above (dashed), are shown in Figure 2. As can be seen, the sVLP consistently performs better. For example, the classical VLP with 32 3-diagonal predictor matrices, results approximately with the same gain as 3-diagonal sVLP with 16 predictors.

The main motivation for investigating the technique of sparse prediction was the anticipated significant reduction of the computational complexity. As it was already discussed, sVLP with 3-diagonal and 5-diagonal matrices results in a slight decrease of the prediction gain compared to the full matrix predictors. On the other hand, the amount of computation is reduced by factor of 3.57 for 3-diagonal, 2.27 for 5-diagonal matrices and even 10 times for the special case of diagonal matrices. Furthermore, the numerical stability of the predictor design process, emphasized in [3] as an inherent problem, is improved since the matrices are smaller. For example, for the first order predictor with two switched predictor matrices, the condition numbers of covariance matrices obtained by classical VLP are 40.56 and 31.11, while for sVLP with 3-diagonal matrices, the mean value of condition numbers of all submatrices is 7.43 (max. 14.97) for the first matrix and 6.90 (max. 11.53) for the second.

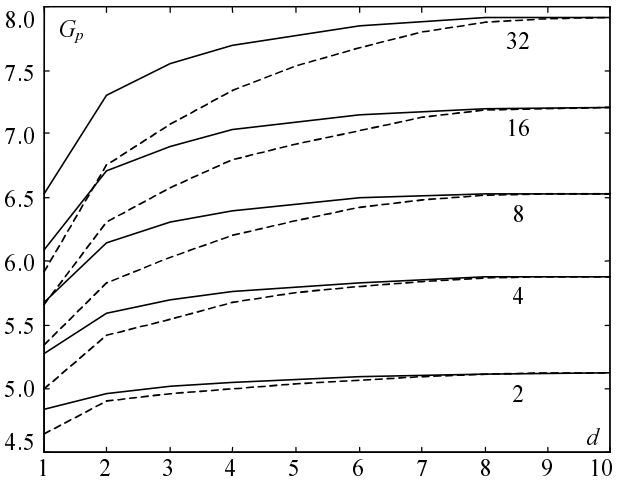

Figure 2. Prediction Gain vs. variable d; predictors designed with classical VLP (dashed) and optimized multidiagonal predictors designed with sVLP (solid), for SIVP with 2 to 32 switched matrices.

5. CONCLUSION

In this paper, a technique called sVLP, for vector linear prediction using sparse multidiagonal predictor matrices has been introduced. It has been shown that sVLP based on 3-diagonal or 5-diagonal predictor matrices results with prediction gain that is only slightly lower than for the case of the full matrix predictors. At the same time, the amount of computation is reduced 3.57 times for 3-diagonal and 2.27 times for 5-diagonal predictors compared to the classical VLP. The sVLP method can be used for coding systems with scalar and vector quantization (VQ) of the prediction residual. For scalar quantization, a fixed intraframe decorrelation (KLT or DCT) prior to SIVP improves the prediction gain. On the other hand, for VQ schemes it is expected that the remaining intraframe correlation of distant components would be covered for by the VQ itself, so the performance of sVLP and classical VLP would be almost identical.

6. REFERENCES

[1] Cupperman V., Gersho A., "Vector predictive coding of speech at 16 kbits/s", *IEEE Trans. Commun.*, vol. COM-33, no.7, 1985, pp. 685-696

[2] Yong M., Davidson G., Gersho A., "Encoding of LPC spectral parameters using switched-adaptive interframe vector prediction", *Proc. ICASSP*,1988,vol.1, pp. 402-405

[3] Chen J.H., Gersho A., "Covariance and autocorrelation methods for vector linear prediction, *Proc. ICASSP*, 1987, pp.1545-1548

[4] Wang S., Paksoy E., Gersho A., Product code vector quantization of LPC parameters, in *Speech and Audio Coding for Wireless and Network Applications*, editors: B. S. Atal, V. Cuperman, A. Gersho, Kluwer Academic Publishers, 1993, pp. 251-258