Journal of Magnetic Resonance **157,** 292–297 (2002)
doi:10.1006/jmre.2002.2593

COMMUNICATIONS

Improved Lanczos Algorithms for Blackbox MRS Data Quantitation

T. Laudadio,*,[1] N. Mastronardi,† L. Vanhamme,* P. Van Hecke,‡ and S. Van Huffel*

*Katholieke Universiteit Leuven, Department of Electrical Engineering, Division ESAT-SCD (SISTA), Kasteelpark Arenberg 10, 3001 Leuven-Heverlee, Belgium; †Istituto per le Applicazioni del Calcolo "M. Picone," Consiglio Nazionale delle Ricerche, Sez. Bari, Via Amendola 122/I, 70126 Bari, Italy; and ‡Katholieke Universiteit Leuven, Biomedical NMR Unit, O. & N., Herestraat 49,B, 3000 Leuven, Belgium

Received January 31, 2002; revised June 11, 2002

Magnetic resonance spectroscopy (MRS) has been shown to be a potentially important medical diagnostic tool. The success of MRS depends on the quantitative data analysis, i.e., the interpretation of the signal in terms of relevant physical parameters, such as frequencies, decay constants, and amplitudes. A variety of time–domain algorithms to extract parameters have been developed. On the one hand, there are so-called blackbox methods. Minimal user interaction and limited incorporation of prior knowledge are inherent to this type of method. On the other hand, interactive methods exist that are iterative, require user involvement, and allow inclusion of prior knowledge. We focus on blackbox methods. The computationally most intensive part of these blackbox methods is the computation of the *singular value decomposition* (SVD) of a Hankel matrix. Our goal is to reduce the needed computational time without affecting the accuracy of the parameters of interest. To this end, algorithms based on the Lanczos method are suitable because the main computation at each step, a matrix–vector product, can be efficiently performed by means of the fast Fourier transform exploiting the structure of the involved matrix. We compare the performance in terms of accuracy and efficiency of four algorithms: the classical SVD algorithm based on the QR decomposition, the Lanczos algorithm, the Lanczos algorithm with partial reorthogonalization, and the implicitly restarted Lanczos algorithm. Extensive simulation studies show that the latter two algorithms perform best.

Key Words: **magnetic resonance spectroscopy; biomedical signal processing; singular value decomposition; Lanczos methods.**

1. INTRODUCTION

Parameters of MRS signals provide direct information about the molecules of the organism under investigation: the frequency of the spectral components characterizes the identity of the molecules; the damping factor characterizes the mobility of the molecules, and the amplitude is directly proportional to the number of molecules. Accurate and efficient quantitation of MRS signals is the essential step before converting the estimated signal parameters into biochemical quantities (concentration, pH).

[1] E-mail: Laudadio@esat.kuleuven.ac.be.

A variety of advanced techniques based on a time–domain model function have been developed.

The function often used to model the N measured data points of a MRS signal is the sum of K exponentially damped complex sinusoids,

$$y_n = \check{y}_n + e_n = \sum_{k=1}^{K} a_k e^{j\phi_k} e^{(-d_k + j2\pi f_k)t_n} + e_n \qquad n = 0, 1, \ldots, N-1, \qquad [1.1]$$

where y_n is the nth measured data point, $\check{y}_n$ represents the model function rather than the actual measurements, $j = \sqrt{-1}$, a_k is the amplitude, ϕ_k the phase, d_k the damping factor and f_k the frequency of the kth sinusoid ($k = 1, 2, \ldots, K$), K is the number of the sinusoids, $t_n = n\Delta t + t_0$ with Δt the sampling interval, t_0 the time between the effective time origin and the first data point to be included in the analysis, and e_n is complex white noise with a circular Gaussian distribution.

In this paper we consider time–domain estimation methods which belong to the class of the so-called blackbox methods. A recent overview is given in (*11*). Their computationally most intensive part is the computation of the *singular value decomposition* (SVD) of a Hankel matrix. To reduce the needed computational time without affecting the accuracy of the parameters of interest, algorithms based on the Lanczos method are suitable. In fact, the main computation at each step, a matrix–vector product, can be efficiently performed by means of the fast Fourier transform exploiting the structure of the involved matrix.

The paper is organized as follows. In Section 2 the subspace-based parameter estimation method HSVD is presented.

In Section 3 four different algorithms for computing the SVD of a Hankel matrix are considered and, then, four HSVD-based methods are obtained:

- QR, the classical method based on the QR decomposition;
- HLSVD, the method based on the Lanczos algorithm;

1090-7807/02 $35.00

• PRO, the method based on the Lanczos algorithm with partial reorthogonalization;

• IRL, the method based on the implicitly restarted Lanczos algorithm.

In Sections 4 and 5 extensive simulation studies are described and the performances of the four methods are compared in terms of accuracy and efficiency. More precisely, in Section 4 we compare the computational efficiency and in Section 5 the statistical accuracy of these four methods. Finally, in Section 6 we formulate the main conclusions.

2. THE SUBSPACE-BASED PARAMETER ESTIMATION METHOD HSVD

HSVD is a subspace-based parameter estimation method in which the noisy signal is arranged in a Hankel matrix H (*1*). Its SVD allows computation of a "signal" subspace and a "noise" subspace. In fact, if H is constructed from a noiseless time–domain signal, the data matrix H has rank exactly equal to K, the number of exponentials that models the underlying signal. Due to the presence of the noise, H becomes a full-rank matrix. However, as long as the signal-to-noise ratio (SNR) of the signal is not too low, one can still define the "numerical" rank being approximately equal to K. Then, the "signal" subspace is found by truncating the SVD of the matrix H to rank K.

The method HSVD is described in the following steps:

• *Step 1.* We arrange the N data points defined in [1.1] in a Hankel matrix H of dimensions $L \times M$, with $N = L + M - 1$:

$$H = \begin{bmatrix} y_0 & y_1 & \cdots & y_{M-1} \\ y_1 & y_2 & \cdots & y_M \\ \vdots & \vdots & \vdots & \vdots \\ y_{L-1} & y_{L-2} & \cdots & y_{N-1} \end{bmatrix}. \qquad [2.2]$$

• *Step 2.* We compute the SVD of the Hankel matrix H,

$$H_{L\times M} = U_{L\times L}\Sigma_{L\times M}V^H_{M\times M},$$

where $\Sigma = \mathrm{diag}\,(\sigma_1, \sigma_2, \ldots, \sigma_q)$, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_q$, $q = \min(L, M)$, $U^H U = UU^H = I$, and $V^H V = VV^H = I$ contain respectively the left and right singular vectors, the superscript H denotes the Hermitian conjugate. We truncate H to a matrix H_K of rank K,

$$H_K = U_K \Sigma_K V_K^H \qquad [2.3]$$

where U_K and V_K are the first K columns of U and V, and Σ_K is the $K \times K$ upper-left submatrix of Σ.

• *Step 3.* We compute the least–squares (LS) solution E of the following overdetermined set of equations,

$$V_K^{(t)} E^H \approx V_K^{(b)}$$

where $V_K^{(b)}$ and $V_K^{(t)}$ are derived from V_K by deleting its first and last row, respectively.

• *Step 4.* The K eigenvalues of E yield the signal pole estimates

$$\hat{z}_k = e^{(-\hat{d}_k + j2\pi\hat{f}_k)\Delta t} \quad k = 1, \ldots, K,$$

from which estimates for $\hat{f}_k$ and $\hat{d}_k$ are found.

• *Step 5.* Filling in the estimated frequencies $\hat{f}_k$ and damping factors $\hat{d}_k$ into the model equation [1.1] yields the set of equations:

$$y_n \approx \sum_{k=1}^{K} c_k e^{(-\hat{d}_k + j2\pi\hat{f}_k)t_n} \quad n = 0, 1, \ldots, N-1.$$

From its least-squares solution $\hat{c}_k = \hat{a}_k e^{j\hat{\phi}_k}$, we find $\hat{a}_k$ and $\hat{\phi}_k$, the estimated amplitudes and phases.

The most time-consuming step of HSVD is the SVD of the matrix H, i.e., Step 2. In the next section we will briefly describe four different algorithms which compute the SVD.

3. THE SVD OF A HANKEL MATRIX

3.1. The Golub–Reinsch Algorithm

Various algorithms are available for computing the SVD of a matrix. The most reliable algorithm for dense matrices is due to Golub and Reinsch (*3*) and it is available in LAPACK (*4*). In this paper we refer to the HSVD-based method based on this algorithm as QR.

The Golub–Reinsch method computes the full SVD in a reliable way and takes approximately $2LM^2 + 4M^3$ complex multiplications for a $L \times M$ matrix. However, when only the computation of a few largest singular values and corresponding singular vectors is needed, the method is computationally too expensive. Moreover, it does not exploit the particular structure of the Hankel matrix H.

3.2. The Lanczos Algorithm

An efficient tool for computing the SVD of large and structured or sparse matrices is the Lanczos bidiagonalization (*5*). Given the rectangular $L \times M$ matrix H, the algorithm computes a sequence of vectors (Lanczos vectors) $u_j \in \mathbb{C}^L$ and $v_j \in \mathbb{C}^M$, where $\mathbb{C}$ is the set of complex numbers, and scalars α_j and β_j for $j = 0, 1, \ldots$. as follows: Choose a starting vector $p_0 \in \mathbb{C}^L$, $p_0 \neq 0$ and let $\beta_1 = \|p_0\|_2$, $u_1 = p_0/\beta_1$ and $v_0 \equiv 0$

for $j = 0, 1, \ldots$.
  $r_j = A^H u_j - \beta_j v_{j-1}$
  $\alpha_j = \|r_j\|_2$
  $v_j = r_j/\alpha_j$
  $p_j = Av_j - \alpha_j u_j$
  $\beta_{j+1} = \|p_j\|_2$
  $u_{j+1} = p_j/\beta_{j+1}$
end

After k steps, the lower bidiagonal matrix B_k is generated

$$B_k = \begin{bmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \beta_2 & \ddots & \\ & & \ddots & \alpha_k \\ & & & \beta_{k+1} \end{bmatrix}. \qquad [3.4]$$

In exact arithmetic the Lanczos vectors are orthonormal such that $U_{k+1} = [u_1, u_2, \ldots, u_{k+1}] \in \mathbb{C}^{L\times(k+1)}$, $V_k = [v_1, v_2, \ldots, v_k] \in \mathbb{C}^{M\times k}$, $U_{k+1}^H U_{k+1} = I_{k+1}$, and $V_k^H V_k = I_k$, where I_l is the $l \times l$ identity matrix. By construction, the columns of U_{k+1} and V_k satisfy the recurrences:

$$\alpha_j v_j = A^H u_j - \beta_j v_{j-1} \quad \text{and} \quad \beta_{j+1} u_{j+1} = A v_j - \alpha_j u_j.$$

We will refer to the columns of U_{k+1} as left Lanczos vectors and the columns of V_k as right Lanczos vectors.

At each step of the Lanczos algorithm, two matrix–vector products, $A^H u_i$ and $A v_i$, are performed. Taking into account the Hankel structure of the involved matrix H, this can be done in $O((L+M) \times \log_2(L+M))$ by means of the FFT, rather than in $O(LM)$. In exact arithmetic the singular values of B_k converge monotonically to those of H. Moreover, the largest and the smallest ones converge first (*2*). When the Lanczos bidiagonalization is carried out in finite precision arithmetic, error vectors accounting for the rounding errors at the jth step occur in the previous recurrence relations, and the orthogonality among the left and right Lanczos vectors is gradually lost. Moreover, multiple copies of the same singular values can arise.

A popular method called HLSVD, described in (*9*) and implemented in the freely available software package MRUI (http://www.mrui.uab.es/mrui/mruiHomePage.html), is based on a way to eliminate the extra copies of converged singular values.

More efficient algorithms exist and are mainly based on work by Paige (*8*), who carried out a thorough error analysis of the Lanczos algorithm and managed to find out when and where the loss of orthogonality takes place.

Among these algorithms there is PRO. Its central idea is that the level of orthogonality among the Lanczos vectors satisfies a recurrence relation which can be derived from the recurrence relations in finite precision arithmetic. These recurrences can be used as a practical tool for computing estimates of the level of orthogonality in an efficient way, and this information can be used to decide when to reorthogonalize and which Lanczos vectors need to be included in the reorthogonalization (*10*). A reliable implementation of this method is available in PROPACK (*6*).

There is another way to maintain orthogonality: to limit the size of the basis set and use restarting schemes.

Restarting means replacing the starting vector with an "improved" starting vector and computing a new Lanczos factorization with the new vector. The implicitly restarted Lanczos algorithm, IRL, is a technique which combines the implicitly shifted QR scheme (*5*) with a k-step Lanczos factorization, obtaining a truncated form of the implicitly shifted QR iteration. The numerical difficulties and storage problems normally associated with the Lanczos process are avoided. A reliable implementation of this method is included in ARPACK (*7*).

4. COMPARISON IN COMPUTATIONAL EFFICIENCY

We consider a simulated signal derived from an *in vivo* ^{31}P spectrum measured in the human brain and consisting of 256 complex data points and 11 exponentials, as defined in (*12*). Figure 1 shows the spectrum of the simulated signal. We consider Eq. [1.1] with a_k, d_k, f_k, and ϕ_k, $k = 1, \ldots, 11$, known and perturb the signal by adding complex white noise with a circular Gaussian distribution with standard deviation σ. The SNR for each peak is measured in decibels (dB) and defined as

$$\text{SNR peak } k \equiv 20 \log\left(\frac{a_k}{\sigma}\right). \qquad [4.5]$$

A low, intermediate, and high noise level are used ($\sigma = 5$, 15, 25, which corresponds to a SNR of 29.5, 20, and 15.6 dB for the middle peak of $\beta - ATP$). Our goal is to reconstruct the parameters a_k, d_k, f_k, and ϕ_k, $k = 1, \ldots, 11$, characterizing the signal y_n, and to compare the execution time in seconds when computing the first 11 singular values and vectors with the Lanczos methods PRO, the complex version of the corresponding routine available in PROPACK (*6*), HLSVD, and IRL. The computations are carried out on a PC with Pentium Intel 850 MHz in the Linux environment, in fortran 77 with machine precision $\omega \approx 2.22 \times 10^{-16}$, with the exception of HLSVD, which is

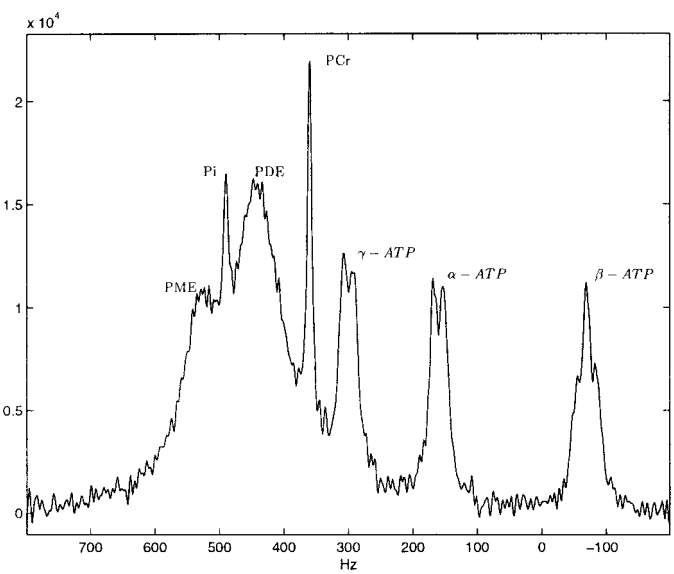

FIG. 1. Real part of the spectrum of the simulated ^{31}P MRS signal, obtained for SNR = 20.

TABLE 1
Mean Value of the Computational Times for 50 Runs

Length (signal) = 256			
SNR	PRO	HLSVD	IRL
29.5	0.0330	0.0470	0.0451
20	0.0492	0.0668	0.0578
15.6	0.0878	0.1496	0.1127
Length (signal) = 512			
SNR	PRO	HLSVD	IRL
29.5	0.0584	0.0886	0.0683
20	0.1220	0.1612	0.1498
15.6	0.2168	0.2990	0.2581

Note. Above: length (signal) = 256. Below: length (signal) = 512.

partly implemented in single precision. The latter code is very frequently used for water removal in the NMR spectra.

In Table 1 CPU times are reported for the simulated signal when considering 256 and 512 data points. We can see that PRO is clearly the fastest algorithm.

5. COMPARISON IN STATISTICAL ACCURACY

In this section we apply the four HSVD-based methods QR, HLSVD, PRO, and IRL to filter out the water signal in ^{1}H spectra. We illustrate their performances in terms of statistical accuracy via computer simulation studies. More precisely, we focus on the amplitude estimates of the metabolite signals after water removal, obtained using the four methods, and compare the quality of these estimates. The quality is measured as the relative root mean squared error (RRMSE) in percentage,

$$\text{RRMSE peak } k \equiv 100 \sqrt{\frac{1}{L} \sum_{l=1}^{L} \frac{\left(a_k - \tilde{a}_k^l\right)^2}{a_k^2}}, \qquad [5.6]$$

where L is the number of simulation runs and $\tilde{a}_k^l$ denotes the estimate of a_k obtained in simulation run l. The RRMSE is compared with the relative Cramer–Rao lower bound (CRB). The CRB indicates the best possible accuracy of an estimate among all unbiased estimators.

5.1. HSVD for Solvent Suppression

The ^{1}H spectrum contains the signal contribution of the water which can have a magnitude 10^3 to 10^4 larger than the magnitude of the metabolites of interest. A preprocessing step is necessary to remove the unwanted water contribution and it is obvious that it should influence the final parameter estimates of the metabolites of interest as little as possible and have a low computational complexity. HSVD provides a mathematical fit of the data by a sum of exponentially damped complex-valued sinusoids. Hence it can be used to approximate the complicated features of the water resonance, including its large tails. The fitted water region is subsequently subtracted from the original signal. We investigate the water suppression abilities of the four HSVD-based methods in terms of accuracy. We use the following scheme to process proton spectra:

1. The user specifies the model order K and a cutoff frequency f_r, which defines a so-called water region $[-f_r, f_r]$.
2. HSVD-based methods are used to model the original signal by a sum of K exponentially damped complex-valued sinusoids.

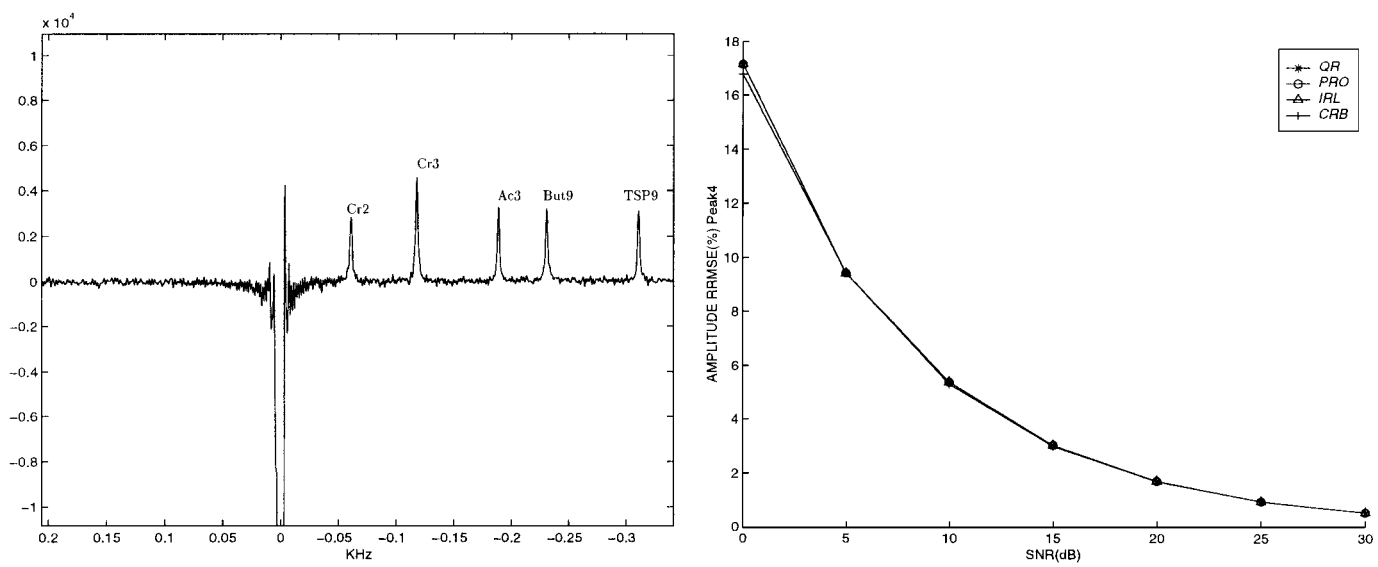

FIG. 2. Left: Real part of the spectrum of the noisy ^{1}H MRS signal containing the water peak obtained for SNR = 15. Right: CRB and RRMSE of amplitude estimates as a function of the SNR for Peak4 (But9) obtained for $K = 12$.

TABLE 2

RRMSE Results of AMARES after Preprocessing by PRO and HLSVD Methods for Peak4 (But9), Obtained for the Model Order Values $K = 10$, $K = 11$, and $K = 12$

SNR	PRO ($K = 10$)	HLSVD ($K = 10$)	PRO ($K = 11$)	HLSVD ($K = 11$)	PRO ($K = 12$)	HLSVD ($K = 12$)
0	17.132	17.140	17.154	17.168	17.151	17.164
5	9.4755	9.4761	9.4634	*****	9.4329	9.4139
10	5.4277	5.4533	5.3442	5.3444	5.3819	5.4936
15	3.1248	3.1246	3.0180	3.0186	3.0188	*****
20	1.8425	1.8420	1.6958	1.6959	1.6959	*****
25	1.1475	1.1467	0.95144	0.95147	0.95123	*****
30	0.79215	0.79112	0.53493	0.60617	0.53365	*****

3. The peaks with frequencies belonging to the user-defined water region are used to reconstruct the water peak, after which the reconstructed water signal is subtracted from the original signal.

4. The residual signal is quantified with AMARES, which minimizes the difference between the nonlinear model function and the data.

Since experimental signals contain errors introduced by factors such as unknown lineshape, data acquisition errors, and eddy currents, all inevitably present in *in vivo* experiments, we use simulated signals (see Fig. 2) to evaluate the performance of the proposed quantification scheme. The cutoff frequency is chosen equal to 35 Hz and the parameters of the seven peaks used to reconstruct the water resonance and of the five metabolite peaks can be found in (*13*). This implies that the correct order of the "signal" subspace is equal to 12. The added complex noise is white and circular Gaussian-distributed. The noise standard deviation σ is varied to simulate a number of SNRs.

In Fig. 2 the magnitude spectrum of the simulated signal is displayed (left) and the RRMSE results obtained from 400 simulation runs are compared with the CRB for the amplitude estimates of peak 4 (But9) (right) when considering the exact model order $K = 12$. We omit the estimation results for the other peaks because they are very similar. First of all, QR, PRO, and IRL perform in the same way and, above all, they perform in an accurate way. Moreover, we have to note that the RRMSE results obtained by HLSVD have not been plotted. The reason of such a choice is that when we consider SNR values greater or equal to 15, HLSVD sometimes fails in computing parameter estimates, thereby yielding too large values of the amplitude RRMSE. For SNR values lower than 15, HLSVD's behavior is comparable to the behavior of the other methods (see Table 2, $K = 12$).

TABLE 3

Mean Value of the Computational Times for 50 Runs Obtained for $K = 12$

SNR	PRO ($K = 12$)	HLSVD ($K = 12$)
0	0.3622	0.6096
5	0.3884	0.6744
10	0.4050	0.7522
15	0.3582	0.7730
20	0.3246	0.8082
25	0.2624	0.7100
30	0.2284	0.5428

The mean values on 50 runs of the execution times in seconds are also reported (see Table 3) when computing the first 12 singular values with PRO and HLSVD. PRO is faster and, moreover, more accurate than HLSVD.

In Table 2 RRMSE results for all methods have been displayed when considering different model order values: $K = 10$, $K = 11$, and $K = 12$. The results clearly show that HLSVD runs into problems if the model order is exactly estimated ($K = 12$); the failures rarely occur when the model order is approximately estimated ($K = 11$) and disappear if it is underestimated ($K = 10$). Then, we can conclude that HLSVD failure occurs when the model order is exact or overestimated. In case of underestimation, which is usually the case in practice when water removal is performed by using HLSVD, failures are unlikely to occur.

5.2. HSVD Accuracy Aspects

We examined several simulation runs in which HLSVD fails and noted that the problem arises in Step 2 of the HLSVD method. More precisely, checking the singular values computed by the algorithm, we can observe the presence of more copies of the same singular value and, consequently, the same number of copies of the corresponding right singular vectors in the matrix V_K. This implies that the LS solution E in Step 3 has determinant equal to 0, i.e., at least one of its eigenvalues is equal to 0. In Step 4 of HSVD, we extract frquency and damping factor estimates from the eigenvalues of E, the so-called signal poles. More precisely, if the kth complex signal pole is indicated as z_k, we have: $f_k = \text{imag}(\log(z_k))/2\pi\Delta t$ and $d_k = -\text{real}(\log(z_k))/\Delta t$, where the expression $\log(z_k)$ can be evaluated only for $z_k \neq 0$. In order to overcome HLSVD failure, we tried to convert the code to double precision. In fact, as already specified in Section 4, HLSVD code is partly implemented in single precision. Unfortunately, also after replacing the single precision, the code sometimes works and sometimes fails. In our investigation we noted that the code actually includes a procedure to eliminate extra copies of converged singular values: the Kats–van der Vorst procedure, based on some eigenvalues' properties and, in particular, on the "interlacing theorem" (*2*). This procedure avoids the reorthogonalization, but it is not accurate enough to circumvent the numerical problems due to the loss of orthogonality among the Lanczos vectors. We can conclude that HLSVD failure is due to its inability to detect only one copy of the same singular value. As noted in Section 5.1, this problem only occurs at large SNR values when the model order is exact or overestimated, which is rarely the case in NMR practice.

6. CONCLUSIONS

The HLSVD method is a blackbox method which computes the parameters of MRS signals and is very frequently used in NMR spectroscopy for water suppression. HLSVD requires the computation of the truncated singular value decomposition of a Hankel matrix H [2.3]. In this paper we have proposed two methods, PRO and IRL, to compute the truncated singular value decomposition of a Hankel matrix [2.3], based on the Lanczos method with partial reorthogonalization or complete reorthogonalization on a small subspace. Via extensive simulation studies, we have compared their performance in terms of accuracy and efficiency with the currently used HLSVD method. Our studies show that PRO and IRL outperform HLSVD in terms of computational efficiency and numerical reliability.[2]

ACKNOWLEDGMENTS

L. Vanhamme is a postdoctoral researcher with the F.W.O. (Fund for Scientific Research—Flanders). This paper presents research results of the EC, the Belgian Programme on Interuniversity Poles of Attraction (IUAP Phase V-10-29), initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture, of a Concerted Research Action (GOA) project of the Flemish Community, entitled "Mathematical Engineering for Information and Communication Systems Technology (MEFISTO)" and of the FWO projects G.0078.01, G.0269.02, and G.0270.02.

REFERENCES

1. H. Barkhuijsen, R. De Beer, and D. Van Ormondt, Improved algorithm for noniterative time–domain model fitting to exponentially damped magnetic resonance signals, *J. Magn. Reson.* **73,** 553–557 (1987).
2. J. K. Cullum and R. A. Willoughby, "Lanczos Algorithms for Large Symmetric Eigenvalue Computations," Vol. 1, "Theory," Birkhäuser, Boston, 1985.
3. G. H. Golub and C. Reinsch, Singular value decomposition and least squares solutions, *Numer. Math.* **14,** 403–420 (1970).
4. E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, "LAPACK Users' Guide," SIAM, Philadelphia, 1995.
5. G. H. Golub and C. F. Van Loan, "Matrix Computations," 3rd ed., Johns Hopkins Univ. Press, Baltimore, MD, 1996.
6. R. M. Larsen, Lanczos bidiagonalization with partial reorthogonalization, Ph.D. thesis, University of Aarhus, Denmark (1998).
7. R. B. Lehoucq, D. C. Sorensen, and C. Yang, "ARPACK Users's Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods," STAM, Philadelphia (1998).
8. C. C. Paige, The computation of eigenvalues and eigenvectors of very large sparse matrices, Ph.D. thesis, London University (1971).
9. W. W. F. Pijnappel, A. Van den Boogaart, R. de Beer, and D. van Ormondt, SVD-based quantification of magnetic resonance signals, *J. Magn. Reson.* **97,** 122–134 (1992).
10. H. D. Simon, The Lanczos algorithm with partial reorthogonalization, *Math. Comp.* **42,** 115–142 (1984).
11. L. Vanhamme, T. Sundin, P. Van Hecke, and S. Van Huffel, MR spectroscopy quantitation: A review of time-domain methods, *NMR Biomed.* **14**(4), 233–246 (2001).
12. L. Vanhamme, A. Van den Boogaart, and S. Van Huffel, Improved method for accurate and efficient quantification of MRS data with use of prior knowledge, *J. Magn. Reson.* **129,** 35–43 (1997).
13. L. Vanhamme, T. Sundin, P. Van Hecke, Ioannis Dologlou, and S. Van Huffel, Accurate quantification of ^{1}H spectra: From finite impulse response filter design for solvent suppression to parameter estimation, *J. Magn. Reson.* **139,** 189–204 (1999).

[2] The corresponding matlab and fortran codes can be obtained from the authors upon request.