JOURNAL OF MAGNETIC RESONANCE **132,** 197–203 (1998)
ARTICLE NO. MN981425

Fast Removal of Residual Water in Proton Spectra

Leentje Vanhamme,* Ricardo D. Fierro,† Sabine Van Huffel,* and Ron de Beer‡

*Department of Electrical Engineering (ESAT), Katholieke Universiteit Leuven, Kard. Mercierlaan 94, 3001 Leuven, Belgium; †Department of Mathematics, California State University San Marcos, San Marcos, California; and ‡Department of Applied Physics, University of Technology of Delft, The Netherlands

Received November 3, 1997; revised February 19, 1998

Quantification of metabolites in ^{1}H spectra is difficult because of the presence of an unwanted water signal. Preprocessing, or removing the water contribution of a ^{1}H spectrum, in the time domain is usually done using the state-space approach HSVD. HSVD removes the residual water and its side lobes, thereby reducing the baseline for the metabolites of interest and allowing subsequent data analysis using more sophisticated nonlinear least squares algorithms. However, the HSVD algorithm is computationally expensive because it estimates the signal subspace using the singular value decomposition (SVD). We show here that replacing the SVD by a low-rank revealing decomposition speeds up the computations without affecting the accuracy of the wanted parameter estimates. © 1998 Academic Press

Key Words: **rank-revealing orthogonal decomposition; singular value decomposition; water removal; exponential data modeling; MRS data quantification.**

INTRODUCTION

Accurate and efficient quantification of MRS (magnetic resonance spectroscopy) signals is extremely important in medical diagnosis or biochemical analysis. In the time domain, nonlinear least squares (NLLS) algorithms such as VARPRO (*1*) and AMARES (*2*) have proven to be very reliable in recent years. These methods commonly use a sum of exponentially damped sinusoids (Lorentzian lines after Fourier transformation) to fit the N measured data points y_l. In particular, the model function is given by

$$y_l = \hat{y} + e_l = \sum_{k=1}^{K} a_k e^{j\phi_k} e^{(-d_k + j2\pi f_k)t_l} + e_l$$

$$l = 0, 1, \ldots, N-1, \qquad [1]$$

where K is the model order, $j = \sqrt{-1}$, a_k is the amplitude, ϕ_k is the phase angle, d_k is the damping factor, and f_k is the frequency of the kth sinusoid (for $k = 1, \ldots, K$). Here, $t_l = l\Delta t + t_0$, where Δt is the sampling interval (nonuniform sampling vectors are also valid), t_0 is the time between the effective time origin and the first data point to be included in the analysis, and e_l is complex white Gaussian noise. The caret on y indicates that this quantity represents the model function rather than the actual measurements. Other types of model line forms can be used. This class of methods provides maximum likelihood estimates in the case the model assumption is correct and the noise is white and Gaussian. In these algorithms biochemical prior knowledge can easily be incorporated to improve parameter accuracy.

In this paper we focus our attention on the quantification of ^{1}H signals. In the absence of water suppression techniques the ^{1}H signals are characterized by a dominating water peak that can be 10^3 to 10^4 larger than the metabolites of interest (which lie on the broad ''tails'' of the water resonance). Although instrumental methods can be used to suppress the water in the spectrum, it is impossible to eliminate the water completely without affecting the metabolites of interest over a relatively wide frequency range. Therefore, a water resonance always remains present in the signal. The intense water peak cannot be described by an analytic function, mainly because of magnetic field inhomogeneity and measurement suppression techniques. Thus, the water resonance makes it impossible for NLLS time domain methods to quantify the peaks of interest reliably. As a consequence, a preprocessing step is necessary to remove the unwanted water contribution.

Some of the preprocessing algorithms (*3–10*) to remove the residual water resonance use crude approximations to remove the wings of the water resonance, introduce changes in the peak area and phases of the resonances lying on the tails of the solvent resonance, or constitute too much of an extra computational burden to the spectral analysis. HSVD (*11*), however, a so-called black box method, has been found particularly useful (*12*). Although limited prior knowledge can be used in black box methods, they often provide a very good mathematical fit of the original data. HSVD can therefore be used to get a good fit of the water resonance, including its large tails. The fitted water region is subsequently subtracted from the original signal. Reliable and accurate NLLS algorithms are then used to analyze the residual signal to quantify the metabolites of interest.

Efficiency is of primary importance in MR spectroscopic imaging. A single metabolite image typically requires $32 \times 32 = 1024$ times the removal of the water resonance. The main drawback with HSVD, however, is the large computational load associated with the SVD of the data matrix. In this paper

1090-7807/98 $25.00

we show that the SVD can be replaced by a more efficient low-rank algorithm. It is shown here that the new, fast algorithm yields significant time savings in this application.

The paper is organized as follows. In the next section we will briefly outline HSVD. We will define low-rank revealing decompositions and show how they can be used to replace the SVD. In the simulation section we will show that this new method is as accurate as HSVD and we will also illustrate the obtained gain in efficiency. Finally, this gain in efficiency is illustrated by analyzing part of a MRS image.

To discuss and illustrate the methods, we use simulated and *in vivo* low-resolution spectra usually encountered in medical spectroscopy. The new method is applicable to high-resolution spectra obtained in other fields as well, since similar SVD-based algorithms have already been applied to such spectra (*13*). Further, the new method can be extended to the analysis of two-dimensional spectra (*14*) much like HSVD.

We mention that HLSVD, a fast version of HSVD, has already been developed in Ref. (*15*). HLSVD is based on the Lanczos procedure (*16*) and gives computational savings in most cases. However, HLSVD has the disadvantage that it can slow down in case of repeated or close singular values (*15*). The Lanczos method also suffers from the loss of orthogonality of the Lanczos vectors in finite precision and lacks rigorous supporting theory. The newly developed algorithm offers nearly the same computational savings as the Lanczos procedure but does not suffer from these drawbacks (*17*).

METHODS

HSVD

The underlying principles of HSVD are discussed in (*15*) and will not be repeated here. Next we give a short outline of the HSVD algorithm.

HSVD.

Step 1. Arrange the data points y_l, $l = 0, \ldots, N-1$, in a Toeplitz matrix T as follows:

$$T = \begin{bmatrix} y_{m-1} & y_m & \cdots & y_{N-1} \\ y_{m-2} & y_{m-1} & \cdots & y_{N-2} \\ \vdots & \vdots & \vdots & \vdots \\ y_0 & y_1 & \cdots & y_{n-1} \end{bmatrix},$$

$$m \geq K,\ N = m + n - 1. \qquad [2]$$

Step 2. Compute the SVD of the Toeplitz matrix T, that is,

$$T_{m\times n} = U_{m\times m}\Sigma_{m\times n}V^H_{n\times n},$$

$$\text{where } \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_p),\ \sigma_1 \geq \cdots \geq \sigma_p,$$

$p = \min(m, n)$, and the superscript H denotes the Hermitian conjugate. In order to obtain the best parameter accuracy it is recommended to choose T as square as possible (*18*), that is, $m = n(+1) = N/2$.

Step 3. Truncate T to a matrix T_K of rank K:

$$T_K = U_K\Sigma_K V_K^H.$$

U_K and V_K are, respectively, the first K columns of U and V; Σ_K is the $K \times K$ upper-left submatrix of Σ. The model order K is chosen equal to the number of sinusoids that comprise the measured signal. In case a water peak is present we have to take into account the non-Lorentzian lineshape of this peak. In practical situations only a few Lorentzians are needed to describe the water region (*12*).

Step 4. Compute the least squares solution $\hat{E}$ of the following (incompatible) system:

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

where $V_K^{(t)}$ and $V_K^{(b)}$ are derived from V_K by omitting its first and last row, respectively. Once E is estimated, its K eigenvalues give the signal pole estimates:

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

From these signal poles it is easy to obtain estimates of the dampings d_k and frequencies f_k.

Step 5. Finally, fill in the estimates $\hat{z}_k$, $k = 1, \ldots, K$, in the N model equations and compute the least squares solution $\hat{c}_k = \hat{a}_k e^{j\hat{\phi}_k}$, $k = 1, \ldots, K$, of

$$y_l \approx \sum_{k=1}^{K} c_k \hat{z}_k^l,\ l = 0, \ldots, N-1.$$

In this way estimates for the amplitudes a_k and phases ϕ_k are obtained.

Note that here the data are arranged in a Toeplitz matrix instead of a Hankel matrix and that the left signal subspace V_K is used instead of U_K as is done in (*15*). Both modifications have been made to make the similarity between HSVD and the new method as great as possible, but they do not change the properties of the algorithm.

The computationally most intensive part of the algorithm is the computation of the SVD of the $m \times n$ matrix T, which requires $\mathbb{O}(mn^2 + n^3)$ floating-point operations ($\mathbb{O}(\cdot)$ denotes the order of magnitude). The least squares solution $\hat{E}$ can be computed efficiently by making use of the Sherman–Morrison matrix-inversion formula (*16*). As can be seen from the preceding algorithm, the full SVD is not required. Instead, only the first K columns of V are required, which estimates the signal subspace. Since K is usually much smaller than n, much of the computational effort in computing a full SVD is wasted. Therefore, we introduce a new matrix decomposition recently introduced in numerical linear algebra

(*17*). These so-called low-rank revealing decompositions only compute approximations to the desired signal subspace, resulting in considerable computational savings.

HLR

A rank-revealing ULV (RR ULV) decomposition of T is of the following form:

$$T = \bar{U}\bar{L}\bar{V}^H,\ \bar{L} = \begin{pmatrix} \bar{L}_K & 0 \\ \bar{H} & \bar{E} \end{pmatrix}, \qquad [3]$$

where $\bar{L}_K$ is a lower triangular $K \times K$ matrix whose singular values approximate the first K singular values of T. The 2-norm of $(\bar{H}, \bar{E})$ is of the same order of magnitude as the $(K + 1)$th singular value of T. $\bar{U}$ and $\bar{V}$ are unitary matrices. $\bar{V}_K$, consisting of the first K columns of $\bar{V}$, approximates the signal space. Special *low-rank* revealing (LRR) decompositions have been developed (*17*) to handle the case in which the dimension of the signal subspace is small, that is, $K \ll n$. We can use a LRR algorithm to compute $\bar{V}_K$ and use it as an estimate for the signal subspace instead of computing a full SVD and truncating V to rank K. Several versions of LRR algorithms have been developed (*17, 19*). In case $K \ll n$ these algorithms are able to compute $\bar{V}_K$ much more efficiently than their SVD counterparts. The algorithm presented here, originally developed by R. Fierro, has the same properties as the algorithms published in (*17*), but makes optimal use of the Toeplitz structure of the original data matrix and therefore results in the largest computational savings (*19*). It is called the Product Form LULV algorithm (PFLULV). Next, we give a brief description of the algorithm.

First, an estimate $u_{est}^{(1)}$ of the left singular vector belonging to the largest singular value of T is estimated. Therefore the Lanczos method (*16*) is used. The stopping criterion for the Lanczos iterations is based on the current singular value estimate δ_j computed during the jth iteration and the method proceeds until $|\delta_j - \delta_{j-1}| < \eta\delta_j$, where $\eta \geq 0$ is a threshold or a maximum number of iterations is reached. For MRS applications the threshold can be set to $1e-03$ and the maximum number of iterations to 15, as done in this paper. Based on $u_{est}^{(1)}$ an estimate of the largest singular value $\sigma_{est}^{(1)}$ of T and corresponding right singular vector $v_{est}^{(1)}$ are computed, that is, $\sigma_{est}^{(1)} = \|T^H u_{est}^{(1)}\|$ and $v_{est}^{(1)} = (T^H u_{est}^{(1)})/\sigma_{est}^{(1)}$. The vectors $u_{est}^{(1)}$ and $v_{est}^{(1)}$ are then reduced to a complex unit vector by Householder transformations (*16*) represented by Householder matrices $P^{(1)}$ and $Q^{(1)}$:

$$P^{(1)}u_{est}^{(1)} = (e^{j\psi}, 0, \ldots, 0)^T \text{ and } Q^{(1)}v_{est}^{(1)} = (e^{j\varphi}, 0, \ldots, 0)^T. \qquad [4]$$

$\|\cdot\|$ denotes the Euclidean vector norm. In general the Householder matrix P is of the form $P = I - 2zz^H/z^Hz$, where I is the identity matrix, $z = x + \text{sign}(x(1))\|x\|(1, 0, \ldots, 0)^T$, $\text{sign}(x(1))$ is equal to the sign of the first element of x, and x is the vector to be reduced to a complex unit vector. Note that these Householder matrices are entirely defined by one vector z.

Since $P^{(1)}$ and $Q^{(1)}$ are Hermitian and unitary, we can easily deduce from Eq. [4] that the matrices have the following partition:

$$P^{(1)} = [u_{est}^{(1)}e^{-j\psi},\ P_2^{(1)}] \text{ and } Q^{(1)} = [v_{est}^{(1)}e^{-j\varphi},\ Q_2^{(1)}].$$

It is then easy to see that elements 2 through n of the first row of $L = P^{(1)}TQ^{(1)}$ are equal to zero:

$$u_{est}^{(1)^H}TQ_2^{(1)} = \sigma_{est}^{(1)}(v_{est}^{(1)^H}Q_2^{(1)}) = \sigma_{est}^{(1)}(0) = 0.$$

Throughout the algorithm the relation $T = \bar{U}^{(i)}L\bar{V}^{(i)^H}$ has to hold. Therefore, $\bar{U}^{(1)} = P^{(1)^H}$ and $\bar{V}^{(1)} = Q^{(1)}$. The matrices $\bar{U}^{(1)}$ and $\bar{V}^{(1)}$, however, need not be computed explicitly. Indeed, as shown earlier, the associated Householder vectors are therefore used and stored instead as shown in the following outline of PFLULV. The same procedure is then repeated on the submatrix $L(2{:}m, 2{:}n) = P_2^{(1)}TQ_2^{(1)}$, which, however, is not explicitly formed. The notation $L(i{:}m, j{:}n)$ is used here to denote a submatrix of L consisting of the rows i to m and columns j to n. In this way the original Toeplitz structure of T is preserved, allowing fast matrix-vector multiplications (*20*) using the fast Fourier transform (FFT) to be performed each time $u_{est}^{(i)}$ is estimated using the Lanczos procedure. Another big advantage of HLR is the fact that the $(m \times n)$ Toeplitz matrix need not be stored in computer memory since matrix-vector multiplications with T or T^H which need to be performed during the course of the algorithm are entirely defined by the original data vector $[y_0, \ldots, y_{N-1}]$ (*20*).

PFLULV.

Step 1. Initialize $\bar{U}^{(0)} \leftarrow [\ ]$ and $\bar{V}^{(0)} \leftarrow [\ ]$, where $[\ ]$ denotes the empty matrix.

Step 2. For $i = 1{:}\ K$

Step 2.1. Compute estimate $u_{est}^{(i)}$ of $L(i{:}m, i{:}n) = (\Sigma_{j=1}^{i-1} P_2^{(j)})^H \times T \times \Sigma_{j=1}^{i-1} Q_2^{(j)}$. Compute $\sigma_{est}^{(i)} = \|L(i{:}m, i{:}n)^H u_{est}^{(i)}\|$ and $v_{est}^{(i)} = (L(i{:}m, i{:}n)^H u_{est}^{(i)})/\sigma_{est}^{(i)}$.

Step 2.2. Determine the Householder vectors z_u and z_v which determine Householder matrices $P^{(i)}$ and $Q^{(i)}$ such that $P^{(i)}u_{est}^{(i)} = (e^{j\psi}, 0, \ldots, 0)^T$ and $Q^{(i)}v_{est}^{(i)} = (e^{j\varphi}, 0, \ldots, 0)^T$. Omit the first column of $P^{(i)}$ and $Q^{(i)}$ to define the related submatrices $P_2^{(i)}$ and $Q_2^{(i)}$.

Step 2.3. Append the Householder vectors z_u and z_v:

$$\bar{U}^{(i)} = \left[\bar{U}^{(i-1)}, \begin{bmatrix} 0 \\ z_u \end{bmatrix}\right] \text{ and } \bar{V}^{(i)} = \left[\bar{V}^{(i-1)}, \begin{bmatrix} 0 \\ z_v \end{bmatrix}\right],$$

0 represents the zero vector/matrix of appropriate dimensions.

End (For)

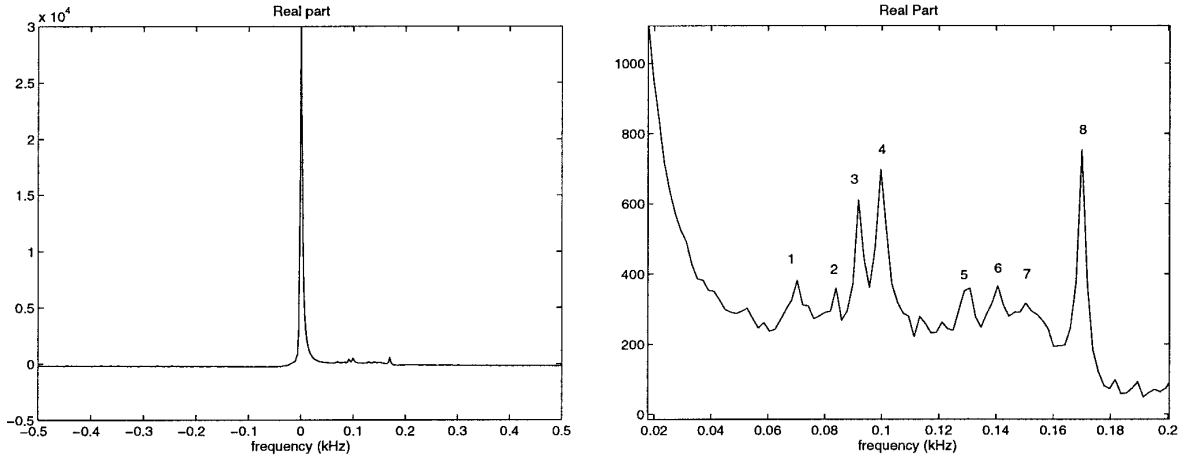

FIG. 1. Frequency domain representation of noisy simulated ^{1}H spectrum ($\sigma = 0.8$). In the left figure the entire signal is represented; the figure to the right zooms in on the region of interest.

Step 3. Use (*16*) Algorithm 5.1.2 to accumulate the K columns of $\bar{V}^{(K)}$ to obtain the estimate $\bar{V}_K$ for the signal subspace as follows:

$$\begin{bmatrix} I_{K-1} & 0 \\ 0 & Q^{(K)} \end{bmatrix} \cdots \begin{bmatrix} 1 & 0 \\ 0 & Q^{(2)} \end{bmatrix} Q^{(1)} \begin{bmatrix} I_K \\ 0 \end{bmatrix}$$

I_K represents the identity matrix of size $K \times K$.

The algorithm HLR is obtained by replacing the SVD by the PFLULV algorithm. The outline of HLR is then as follows:

HLR.

Step 1. Compute $\bar{V}_K$ using PFLULV.

Step 2–3. See *Step 4–5* of HSVD with V_K replaced by $\bar{V}_K$.

SIMULATIONS

In this section we address the accuracy and the efficiency of HLR compared to HSVD when used to preprocess ^{1}H spectra. To this end we perform a Monte Carlo study. The simulation signal we use is derived from an *in vivo* ^{1}H NMR echo signal and was previously used in (*21*). From the noiseless signal 400 noisy realizations were generated with noise standard deviation σ (both on the real and imaginary parts). One of the used noisy simulation signals is displayed in Fig. 1.

The following signal processing protocol is applied to all simulation signals:

1. The entire signal is fitted using HLR or HSVD, and the model order $K = 10$ is used. To this end the 512 data points are arranged in a 257×256 Toeplitz matrix.

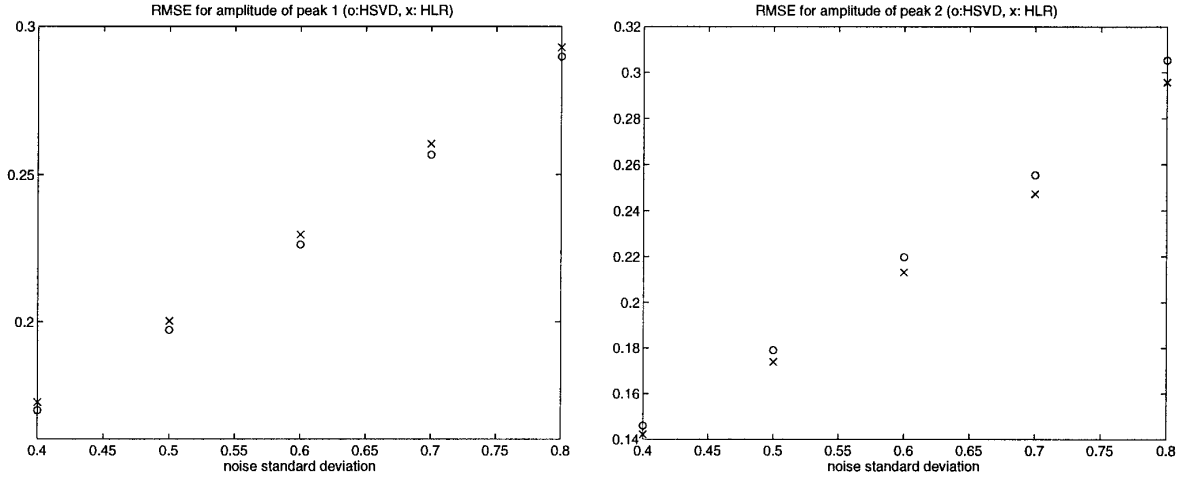

FIG. 2. RMSE for amplitudes of peak 1 and 2 of the simulation signal for different values of the noise standard deviation after removal of the water peak and quantification of the residual metabolites using AMARES. The crosses and the circles denote that the water peak was removed using HLR and HSVD, respectively. In the water removal preprocessing step the model order used was 10 and the data were arranged in a 257×256 data matrix.

TABLE 1
Comparison between SVD and LRR, HSVD and HLR in Terms of Flops

	$\sigma = 0.4$	$\sigma = 0.5$	$\sigma = 0.6$	$\sigma = 0.7$	$\sigma = 0.8$
^{fl}SVD/^{fl}LRR	57.4	57.2	57.0	56.8	56.2
^{fl}HSVD/^{fl}HLR	47.0	46.9	46.9	46.8	46.6

Note. In the first row the ratio of the number of flops (in MATLAB) needed by SVD to that required by the LRR algorithm PFLULV are displayed as a function of the noise level. In the second row the ratio of the number of flops required by HSVD to that required by HLR are displayed as a function of the noise level. The imposed model order K is 10 and the size of the data matrix used is 257×256.

2. The peaks with frequencies belonging to the water region (defined as the region below 20 Hz) are used to reconstruct the water peak.
3. The reconstructed signal is subtracted from the original signal.
4. The residual signal is quantified using AMARES (*2*) and estimates of the metabolites of interest are obtained.

We compared HLR and HSVD using the root mean-square error (RMSE) of the final parameter estimates. In Fig. 2 the RMSE of the amplitudes of peak 1 and 2 obtained by removing the water via HSVD and HLR and subsequent quantification of the residual signal with AMARES are compared. For peak 1 the RMSE obtained using HSVD preprocessing is slightly lower than that obtained using HLR preprocessing. For peak 2 the situation is reversed. For the other peaks the situation is similar—the obtained RMSEs of all parameters using HSVD preprocessing and HLR preprocessing are almost the same and neither of the two methods leads to an overall lowest RMSE.

It can be concluded that both methods, when used to subtract the water peak, lead to a comparable parameter accuracy of the metabolites of interest obtained after the final parameter estimation with a NLLS algorithm.

As a measure of efficiency we compare the number of flops (obtained by MATLAB) required by PFLULV to the number of flops required by the SVD for a 257×256 Toeplitz data matrix. Here a flop denotes a floating-point operation, either an addition or a multiplication. This measure gives a computer-independent comparison of the computational complexity of the two algorithms. We also compared the overall number of flops required by HSVD and HLR. The results for different noise levels are displayed in Table 1.

TABLE 2
Comparison between SVD and LRR, HSVD and HLR in Terms of CPU Times

	$\sigma = 0.4$	$\sigma = 0.5$	$\sigma = 0.6$	$\sigma = 0.7$	$\sigma = 0.8$
^{t}SVD/^{t}LRR	42.3	45.5	41.8	39.9	39.8
^{t}HSVD/^{t}HLR	37.4	40.2	36.9	35.2	35.2

Note. In the first row the ratio of the CPU time needed by SVD to that required by the LRR algorithm PFLULV are displayed as a function of the noise level. In the second row the ratio of the CPU time required by HSVD to that required by HLR are displayed as a function of the noise level. The imposed model order K is 10 and the size of the data matrix used is 257×256.

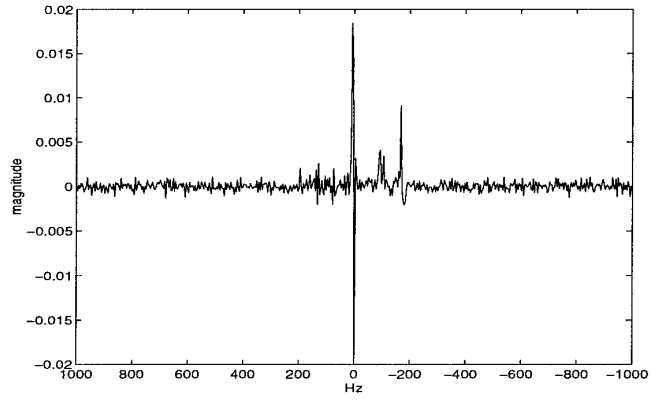

FIG. 3. Frequency domain representation of an *in vivo* ^{1}H signal of the MRSI data set. The water region is defined as all peaks with a frequency higher than −78 Hz.

To get an idea of the difference in actual execution time between HSVD and HLR we implemented both algorithms in FORTRAN 77 making use of the BLAS and LINPACK libraries (available from netlib (*22*)). The timing experiments were performed on a SUN ULTRA 2 (200 MHz). The results are displayed in Table 2. Although the timing results are compiler and computer dependent, Table 2 indicates the gain in efficiency in terms of actual CPU times is a factor of 35 to 40 in this particular case ($K = 10$, a 257×256 data matrix). The CPU time needed by HLR to quantify one of these simulation signals is of the order of 0.3 s as opposed to 11 s for HSVD. The subsequent analysis using AMARES takes about 2 s per signal. This shows that preprocessing using HSVD takes more time than the actual quantification using AMARES, which is of course unacceptable, especially when a large amount of data has to be analyzed.

APPLICATION TO *IN VIVO* MRS IMAGE

This fast algorithm is particularly useful when a lot of data needs to be processed as is the case in MRSI. To illustrate the gain in efficiency in the analysis of *in vivo* MRS signals we compare the performance of HLR and HSVD to analyze part of an MRSI data set. The data set under investigation was measured at the University of Alabama at Birmingham and provided by Dr. J. A. den Hollander, Center for NMR Research and Development, University of Alabama at Birmingham (see Acknowledgments). The measurements were performed on a 1.5-T ACS/S15 Philips Gyroscan, using a protocol developed by Philips Medical Systems (*23*).

Since we wanted to process only signals containing metab-

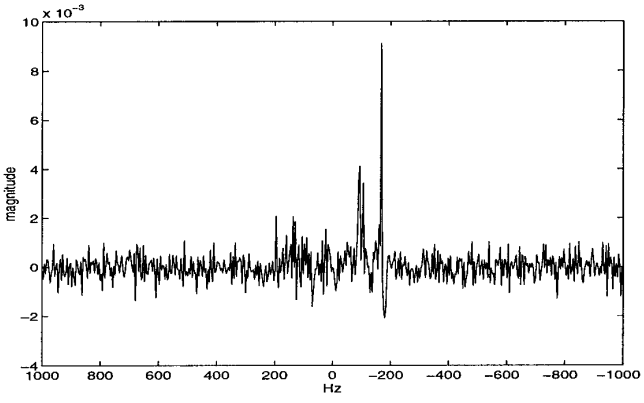

FIG. 4. Frequency domain representation of the *in vivo* ^{1}H signal of Fig. 3 after removal of the water contributions using HLR and a model order of 5.

olites of interest, 20 signals from the middle region of the image were selected and preprocessed. In these signals contributions from water have to be removed. If the entire image is to be processed the protocol explained in (*23*) can be used.

An exponentially damped sinusoid is assumed to contribute to the water peak if it has a frequency higher than −78 Hz. See Fig. 3 for a representative signal.

The number of data points used in the preprocessing stage was 512. Five exponentials were enough to remove the contributions from water. Figure 4 displays the signal of Fig. 3 after preprocessing with HLR (the imposed model order was 5).

To show, however, the influence of the chosen model order on the efficiency of the algorithm, model orders of 10 and 20 were also used. The total CPU time needed to preprocess 20 signals using HLR and a model order of 5 is only 3.15 s compared to 259.6 s using HSVD. The use of HLR in this case represents a reduction of a factor of 82.4 in CPU time. As illustrated in Table 3, the gain in efficiency decreases as more exponentials are estimated.

To show the influence on the number of data points used, the same analysis for different model orders is done using only 256 data points. The results are displayed in Table 4. As expected, the gain in efficiency decreases when the number of data points decreases. Although the use of only 256 data points is unacceptable for this type of application, it nonetheless shows the influence of the number of data points on the actual CPU times.

TABLE 3

Total CPU Time Needed by HSVD and HLR to Preprocess 20 Signals of an MRS Image Consisting of 512 Data Points Each for Imposed Model Orders of 5, 10, and 20

	$K = 5$	$K = 10$	$K = 20$
^{t}HSVD	259.6 s	254.3 s	263.7 s
^{t}HLR	3.15 s	7.0 s	19.8 s
^{t}HSVD/^{t}HLR	82.4	36.3	13.3

Note. The ratios between the needed CPU times are also depicted.

TABLE 4

Total CPU Time Needed by HSVD and HLR to Preprocess 20 Signals of an MRS Image Consisting of 256 Data Points Each for Imposed Model Orders of 5, 10, and 20

	$K = 5$	$K = 10$	$K = 20$
^{t}HSVD	20.9 s	20.1 s	20.4 s
^{t}HLR	2.16 s	4.07 s	10.77 s
^{t}HSVD/^{t}HLR	9.7	5.0	1.9

Note. The ratios between the needed CPU times are also depicted.

CONCLUSIONS

In this paper HLR is presented as an alternative to HSVD for the removal of the water peak in ^{1}H spectra. HLR uses a low-rank revealing decomposition to extract the signal subspace instead of a full SVD as done in HSVD. This results in a considerable improvement in efficiency without affecting the accuracy of the parameters of interest. The latter can be estimated after the preprocessing stage by means of an iterative nonlinear least algorithm such as AMARES.

The gain in efficiency depends on both the number of data points of the signal and the imposed model order, and is more pronounced when the number of data points becomes larger and the model order smaller. For a typical signal consisting of 512 data points and a model order of 10, the CPU time required by HLR is less than that required for HSVD by a factor of about 35 to 40.

ACKNOWLEDGMENTS

The authors thank Dr. J. A. den Hollander, Center for NMR Research and Development of the University of Alabama at Birmingham, for providing the MRSI data set.

The first author is a Ph.D. student funded by the IWT (Flemish Institute for Support of Scientific–Technological Research in Industry). The third author is a Research Associate with the F. W. O. (Fund for Scientific Research—Flanders). This work is supported by the Belgian Programme on Interuniversity Poles of Attraction (IUAP-4/2 & 24), initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture, by the EU Programme ''Training and Mobility of Researchers,'' project ERB 4061 PL 97-0945, and by a Concerted Research Action (GOA) project of the Flemish Community, entitled ''Model-based Information Processing Systems.''

REFERENCES

1. J. W. C. van der Veen, R. De Beer, P. R. Luyten, and D. van Ormondt, Accurate quantification of in vivo ^{31}P NMR signals using the variable projection method and prior knowledge, *Magn. Res. in Med.* **6,** 92–98 (1988).
2. 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).
3. A. Bielecki and M. H. Levitt, Frequency-selective double-quantum-filtered COSY in water, *J. Magn. Reson.* **82,** 562–570 (1989).
4. D. Marion, M. Ikura, and A. Bax, Improved solvent suppression in

one- and two-dimensional NMR spectra by convolution of time-domain data, *J. Magn. Reson.* **84,** 425–430 (1989).

5. Y. Kuroda, A. Wada, T. Yamazaki, and K. Nagayama, Postacquisition data processing method for suppression of the solvent signal, *J. Magn. Reson.* **84,** 604–610 (1989).
6. M. Derich and X. Hu, Elimination of water signal by postprocessing, *J. Magn. Reson. A* **101,** 229–232 (1993).
7. K. J. Cross, Improved digital filtering technique for solvent suppression. *J. Magn. Reson.* **101,** 220–224 (1993).
8. J. H. J. Leclerc, Distortation-free suppression of the residual water peak in proton spectra by postprocessing, *J. Magn. Reson. B* **103,** 64–67 (1994).
9. C. J. Craven and J. P. Waltho, The action of time-domain convolution filters for solvent suppression, *J. Magn. Reson. B* **106,** 40–46 (1995).
10. G. Zhu, D. Smith, and Y. Hua, Post-acquisition solvent suppression by singular-value decomposition, *J. Magn. Reson.* **124,** 286–289 (1997).
11. H. Barkhuysen, R. de Beer, and D. van Ormondt, Improved algorithm for noniterative time-domain model function to exponentially damped magnetic resonance signals, *J. Magn. Reson.* **73,** 553–557 (1987).
12. A. van den Boogaart, D. van Ormondt, W. W. F. Pijnappel, R. de Beer, and M. Ala-Korpela, Removal of the water resonance from ^{1}H magnetic resonance spectra, *in* ''Mathematics in Signal Processing III'' (J. G. McWhirter, Ed.), pp. 175–195. Clarendon, Oxford (1994).
13. R. de Beer, A. van den Boogaart, E. Cady, D. Graveron-Demilly, A. Knijn, K. W. Langenberger, J. C. Lindon, A. Ohlhoff, H. Serrai, and M. Wylezinska-Arridge, Multicentre quantitative data-analysis trial: The overlapping background problem, *in* ''Eurospin Annual 1995–1996. Istituto Superiore Di Sanita'' (F. Podo, W. M. M. J. Bovee, J. D. de Certaines, O. Henriksen, M. O. Leach and D. Leibfritz, Eds.), pp. 341–365, Rome (1996).
14. R. de Beer, D. van Ormondt, and W. W. F. Pijnappel, Maximum likelihood estimation of poles, amplitudes and phases from 2-D NMR time domain signals, *in* ''Proceedings, ICASSP 89,'' pp. 1504–1507, Glasgow, Scotland (1989).
15. 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).
16. G. H. Golub and C. F. Van Loan, ''Matrix Computations,'' Johns Hopkins Press, Baltimore (1993).
17. R. D. Fierro and P. C. Hansen, Low-rank revealing UTV decompositions, *Numerical Algorithms* **15,** 37–55 (1997).
18. S. Van Huffel, H. Chen, C. Decanniere, and P. Van Hecke, Algorithm for time-domain NMR data fitting based on total least squares, *J. Magn. Reson. A* **110,** 228–237 (1994).
19. R. D. Fierro, L. Vanhamme, and S. Van Huffel, Total least squares algorithms based on rank-revealing complete orthogonal decompositions, *in* ''Recent Advances in Total Least Squares techniques and Errors-in-Variables Modeling'' (S. Van Huffel, Ed.), pp. 99–116. SIAM, Philadelphia (1997).
20. C. F. Van Loan, ''Computational Frameworks for the Fast Fourier Transform,'' SIAM, Philadelphia (1992).
21. ''Eurospin Annual 1994'' (F. Podo, W. M. M. J. Bovée, J. D. de Certaines, O. Henriksen, M. O. Leach, and D. Leibfritz, Eds.), Instituto superiore di sanità (1994).
22. ftp://ftp.netlib.org.
23. R. de Beer, F. Michiels, D. van Ormondt, B. P. O. van Tongeren, P. R. Luyten, and H. van Vroonhoven, Reduced lipid contamination in in vivo ^{1}H MRSI using time-domain fitting and neural network classification, *Magn. Reson. Im.* **11,** 1019–1026 (1993).