CHAPTER 1

Exponential data fitting

Victor Pereyra,
Computational Sciences Research Institute
San Diego State University
San Diego, CA, USA
vpereyra@yahoo.com

Godela Scherer
Mathematics Department
University of Reading, UK.
A.Scherer@reading.ac.uk

Abstract. In this initial chapter we consider some of the basic methods used in the fitting of data by real and complex linear combinations of exponentials. We have selected the classes of methods that are most frequently used in many different fields: variable projections for solving this separable nonlinear least squares problem, derivatives and variants of Prony's method, which rely on evenly sampled data and take special advantage of the particular form of the approximation and finally the matrix-pencil method. We also have implemented some of these techniques and compared them in a few examples to support some comments on their advantages and disadvantages and exemplify their performance in terms of computing time and robustness, specially considering that this is a notoriously ill-conditioned problem in many cases.

Keywords Exponential data fitting; separable nonlinear least squares; Prony's method

1.1. Introduction

Fitting data with linear combinations of real or complex exponentials is pervasive within many disciplines in Sciences and Engineering. Since Gaspard Riche de Prony invented a method in 1795 [**14**] to solve this problem for evenly spaced samples there have been many developments and applications. We will survey some of the more successful ones and then let leading experts from different fields describe their applications and experiences.

One obvious reason why these types of approximation functions are important is that combinations of exponentials are solutions of homogeneous linear ordinary differential equations and as such they naturally model many different physical processes. Thus, if we have measurements of a quantity that can be modelled by the solution of such an equation, fitting this data to a linear combination of exponentials can give valuable information on decay rates or other material properties

of the physical system. Also, exponentials have good approximation properties on compact domains and, of course, complex exponentials lead to Fourier expansions.

If we know the exponents and are only interested in the coefficients of the linear combination and if we choose to minimize the l_2 norm of the residuals between observed and calculated values, then this fitting is a linear least squares problem. The interesting and more challenging case though, is when we want not only the weights but also the exponents, which leads to a nonlinear least squares problem. As indicated by Beylkin and Monzon [**5**], this is akin to Fourier series with adaptive exponents that can lead to more concise approximations as exemplified in a number of challenging examples.

It was observed (see [**17, 19, 21**]) that it is very useful to separate the treatment of the weights from that of the parameters appearing nonlinearly. Although the context of so-called separable problems is more general than fitting exponentials, the latter one turns out to have many successful applications.

Many algorithms have been developed for separable nonlinear least squares problems (**SNLS**). The *variable projections* algorithm designed and analyzed by G. Golub and V. Pereyra [**17**] uses nonlinear optimization techniques to compute the nonlinear parameters. A computer program (VARPRO was developed and put in the public domain. This program had great impact in many disciplines, as shown in the survey paper [**19**] and in several chapters of this book.

Another set of techniques, somewhat misnamed as "linear methods," are of importance because of their prevalence in the application fields. Among them are the Prony-type or polynomial methods, so called because the nonlinear parameters are obtained from the roots of a characteristic polynomial. The variants of this method differ in the way the coefficients of the polynomial are defined. An important variant is the *Modified Prony* method, originated by M. Osborne in [**37**]. It extends the "Prony method" for extracting sinusoidal or exponential signals from uniformly sampled time series data when there is no noise, to the case when the signal is imbedded in noise. The modified Prony method has been exhaustively analyzed [**39, 40**] for fitting with exponentials or other functions that satisfy a linear difference equation with constant coefficients, and a MATLAB program by G. K. Smyth, one of the coauthors, is available on the Internet [**56**]. In this approach, the coefficients of the polynomial are obtained by solving a generalized eigenproblem.

Another Prony-type method is the linear prediction algorithm presented in [**35, 51**]. It follows the general structure of the Prony method but uses a truncated singular value decomposition (TSVD) of a Hankel matrix defined by the noisy signal to solve for the polynomial coefficients. This allows one to determine the number of representative exponential terms of the fitting function that is appropriate for the noise level.

These algorithms use numerical linear algebra techniques, but generalized eigenvalues and zeros of polynomials are hardly linear problems, so that is the reason against cataloging them as "linear techniques".

A different approach that uses the separability of the approximating model is found in the subspace-based matrix-pencil methods. We outline below the HTLS/HSVD methods, extensively analysed and used by the Katholieke Universiteit Leuven group [**31, 51**]. (See also the chapter in this book by Sima, Poullet and Van Huffel.) They also start with a Hankel matrix of the data but define a multiple right-hand side linear least squares problem by comparing a Vandermonde

and an SV decomposition. The nonlinear parameters are obtained by solving this problem.

Last but not least, alternatively to separation of linear and nonlinear variables, nonlinear optimization techniques can also be applied directly to the full functional problem. A prominent position in this approach is held by the secant type code NL2SOL [**13**], used for example in AMARES, a software for biomedical applications. It has the advantage of ease in incorporating *a priori* knowledge about the parameters.

A detailed survey of exponential fitting with many references and interesting discussions can be found in [**25**]. Some pitfalls on the indiscriminate use of the results of exponential fitting are discussed in [**54**]. See also [**61**] for an interesting discussion on the potential ill-conditioning of these problems that was already pointed out by Lanczos in 1956 [**30**]. In the following sections we will review some of these methods and compare them on several examples.

Acknowledgement

We would like to thank P. C. Hansen and M. Saunders for reading carefully this chapter and making many suggestions that considerably improved its presentation.

1.2. Solving separable nonlinear least squares problems with variable projections

A separable nonlinear least squares problem was defined in [**17**], as one for which the model used to approximate the data is a linear combination of nonlinear functions that can depend on multiple parameters. The ith component of the residual vector is written as

$$r_i(\mathbf{a}, \boldsymbol{\alpha}) = y_i - \sum_{j=1}^{n} a_j \phi_j(\boldsymbol{\alpha}; t_i), \ i = 1, \ldots, N, \qquad N > n + k. \tag{1.1}$$

Here the t_i are independent variables associated with the observations y_i, while the $\mathbf{a} = \{a_j\}$ and the components of the k-dimensional vector $\boldsymbol{\alpha}$ are the parameters to be determined by minimizing the functional $||\mathbf{r}(\mathbf{a}, \boldsymbol{\alpha})||_2^2$, where $||\cdot||_2$ stands for the l_2 vector norm, i.e., the functional to be minimized is the sum of squares of the residuals. We can write this functional using matrix notation as

$$r(\mathbf{a}, \boldsymbol{\alpha}) = \|r(\mathbf{a}, \boldsymbol{\alpha})\|_2^2 = \|\mathbf{y} - \boldsymbol{\Phi}(\boldsymbol{\alpha})\mathbf{a}\|_2^2, \tag{1.2}$$

where the columns of the matrix $\boldsymbol{\Phi}(\boldsymbol{\alpha})$ correspond to the nonlinear functions $\phi_j(\boldsymbol{\alpha};t_i)$ of the k parameters $\boldsymbol{\alpha}$ evaluated at all the t_i values, and the vectors $\mathbf{a}$ and $\mathbf{y}$ represent the linear parameters and the observations respectively. The minimization problem is then,

$$\min_{\mathbf{a},\boldsymbol{\alpha}} \|r(\mathbf{a}, \boldsymbol{\alpha})\|_2^2. \tag{1.3}$$

Now it is easy to see that if we knew the nonlinear parameters $\boldsymbol{\alpha}$, then the linear parameters $\mathbf{a}$ could be obtained by solving the linear least squares problem:

$$\mathbf{a} = \boldsymbol{\Phi}(\boldsymbol{\alpha})^{+}\mathbf{y}, \tag{1.4}$$

which stands for the minimum-norm solution of the linear least squares problem (1.2) for fixed $\boldsymbol{\alpha}$, where $\boldsymbol{\Phi}(\boldsymbol{\alpha})^{+}$ is the Moore-Penrose generalized inverse of

$\mathbf{\Phi}(\alpha)$[1](which can be ill-conditioned or even rank-deficient). Substituting this $\mathbf{a}$ into the original functional gives the problem

$$\min_{\alpha} \|(\mathbf{I} - \mathbf{\Phi}(\boldsymbol{\alpha})\mathbf{\Phi}(\boldsymbol{\alpha})^{+})\mathbf{y}\|_2^2, \tag{1.5}$$

where the linear parameters have been eliminated. Some good references for non-numeric persons further reading on these basic concepts are **[7, 20]**.

We define,

$$\mathbf{r}_{VP}\ (\boldsymbol{\alpha}) = (\mathbf{I} - \mathbf{\Phi}(\boldsymbol{\alpha})\mathbf{\Phi}(\boldsymbol{\alpha})^{+})\mathbf{y}, \tag{1.6}$$

and call it the *Variable Projection (VP)* of $\mathbf{y}$. Its name stems from the fact that the matrix in parentheses is the projector onto the orthogonal complement of the column space of $\mathbf{\Phi}(\boldsymbol{\alpha})$, which we will denote in what follows by $\mathbf{P}^{\perp}_{\mathbf{\Phi}(\boldsymbol{\alpha})}$. We will also refer to $\|\mathbf{r}_{VP}\ (\boldsymbol{\alpha})\|_2^2$ as the *Variable Projection functional*.

This is a more powerful paradigm than the simple idea of alternating between minimization of the two sets of variables (such as the NIPALS algorithm of Wold and Lyttkens **[62]**), which can be proven theoretically and practically not to result, in general, in the same enhanced performance.

In summary, the Variable Projection algorithm consists of first minimizing (1.5) and then using the optimal value obtained for $\boldsymbol{\alpha}$ to solve for $\mathbf{a}$ in (1.4). One obvious advantage is that the iterative nonlinear algorithm used to solve the first minimization problem works in a space of smaller dimension and in consequence fewer initial guesses are necessary. However, the main payoff of this algorithm is the fact that, as the minima for the reduced functional are better defined than those for the full one, it always converges in fewer iterations than the minimization of the full functional, including convergence when the same minimization algorithm for the full functional diverges (see for instance **[27]**).

Therefore, a different reason to use the reduced functional is to observe from the above results that, since the linear parameters are determined by the nonlinear ones, then the full problem must be increasingly ill-conditioned as, and if, it converges to the optimal parameters. That is probably why the important problem of real or complex exponential fitting is so hard to solve directly. See for instance **[55]** for a theoretical discussion of this issue and an interesting application to the training of nonlinear neural networks **[49, 50, 43]**.

It was also proven in the original paper **[17]** that the set of stationary points of the original and reduced functionals are the same. This theorem has been reassuring to many practitioners and has been used to derive other theoretical results in similar situations. Further comments on the basic results can also be found in the textbooks of Seber and Wild **[53]** and Björck **[7]**.

1.3. Complex VARPRO

In this Section we consider the development of a Variable Projection type solver (VARPRO) for separable nonlinear least problems (SNLLSQ) **[17, 19]**, for the case in which the model is a linear combination of complex exponentials.

We will discuss the essential elements of a modern VARPRO type implementation, without attempting to reproduce all the aspects of the compact original 1973 one, which was constrained by the computer capabilities of that time frame. We

[1]The generalized inverse plays a similar role for rectangular matrices as the inverse does for square ones. For a definition see **[7]**.

can do a much simpler job now that memory is not an issue. Also the code can use reliable off-the-shelf open software as available.

Calculation of the VP functional and its derivative. There are two ways to calculate the necessary quantities and we explore them both: singular value Decomposition (SVD) or Linear least squares (LLSQ). The advantage of the first is that it gives good quantitative information about the condition of the problem and it facilitates its regularization if necessary.

The SVD of the complex matrix $\mathbf{\Phi} = \mathbf{UDV}^*$ always exist. $\boldsymbol{U}, \boldsymbol{V}$ are unitary and $\mathbf{D} = \text{diag}(\sigma_j)$ is diagonal with the same rectangular shape of $\mathbf{\Phi}$, and the * stands for transposed conjugate. The diagonal contains the singular values of $\mathbf{\Phi}$, which are real and non-negative. As in the real case, the SVD is rank revealing: small singular values (relative to the largest) are a sign of ill-conditioning, while in the extreme case, zero singular values indicate rank deficiency. In all cases, truncating the small singular values regularizes the problem and gives the best approximation of that rank to the original matrix in the Frobenius norm.

If we calculate the SVD of the $m \times n$ matrix $\mathbf{\Phi} = \mathbf{UDV}^*$, then the Variable Projection functional can be written as

$$r_{VP}(\boldsymbol{\alpha}) = \left\| \mathbf{U} \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{I}_{m-r} \end{bmatrix} \mathbf{U}^* \mathbf{y} \right\|_2^2 = \left\| \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{I}_{m-r} \end{bmatrix} \mathbf{U}^* \mathbf{y} \right\|_2^2,$$

where r is the numerical rank of $\mathbf{\Phi}$ and we can eliminate the first $\mathbf{U}$ because it is a unitary matrix. Thus, if we call $\widetilde{\mathbf{y}} = \mathbf{U}^* \mathbf{y}$, we have

$$r_{VP}(\boldsymbol{\alpha}) = \sum_{i=r+1}^{m} \widetilde{y_i}^2.$$

Unfortunately most SVD codes compute only the "thin" matrix $\mathbf{U}$ $(m \times n)$, which is not sufficient for this calculation. In that case we need to obtain $\mathbf{a}$ as indicated below in 1.9 and then calculate the residual directly as in 1.1.

Gradient. For the complex exponential case, since the gradient of the functional will be real, we need to be careful with how we compute it. First of all we observe that for any complex vector $\mathbf{f}(z)$,

$$D(\mathbf{f}^*\mathbf{f}) = D\mathbf{f}^*\mathbf{f} + \mathbf{f}^* D\mathbf{f} = D\mathbf{f}^*\mathbf{f} + (D\mathbf{f}^*\mathbf{f})^* = 2\Re(D\mathbf{f}^*\mathbf{f}).$$

Here D stands for the *Fréchet* derivative.

Therefore, for $\mathbf{f} = \mathbf{P}_{\mathbf{\Phi}}^{\perp}\mathbf{y}$, we will have

$$\frac{1}{2}\nabla r_{VP}(\boldsymbol{\alpha}) = \Re(\mathbf{y}^* D\mathbf{P}_{\mathbf{\Phi}}^{\perp *}\mathbf{P}_{\mathbf{\Phi}}^{\perp}\mathbf{y}) = -\Re(\mathbf{y}^*(\mathbf{P}_{\mathbf{\Phi}}^{\perp} D\mathbf{\Phi}\mathbf{\Phi}^+ + (\mathbf{P}_{\mathbf{\Phi}}^{\perp} D\mathbf{\Phi}\mathbf{\Phi}^+)^*)\mathbf{P}_{\mathbf{\Phi}}^{\perp}\mathbf{y}),$$

where the expanded expression comes from the derivative of the pseudoinverse as derived in **[17, 19]**. This is a 3-dimensional tensor consisting of the gradients with respect to the vector α of each component of $\mathbf{\Phi}$:

$$D\mathbf{\Phi} = \left\{ \frac{\partial \Phi_{i,j}}{\partial \alpha_l} \right\}.$$

The columns of $\mathbf{\Phi}$ are exponentials sampled at the data points and therefore $D\mathbf{\Phi}$ can be easily generated. But, because of the properties of the pseudoinverse,

$\mathbf{\Phi}^+\mathbf{P}_{\mathbf{\Phi}}^{\perp} = \mathbf{0}$, and therefore the first term in the formula above drops out, leaving

$$\frac{1}{2}\nabla r_{VP}(\boldsymbol{\alpha}) = -\Re(\mathbf{y}^*(\mathbf{P}_{\mathbf{\Phi}}^{\perp}D\mathbf{\Phi}\mathbf{\Phi}^+)^*\mathbf{P}_{\mathbf{\Phi}}^{\perp}\mathbf{y}) = -\Re(\mathbf{y}^*\mathbf{\Phi}^{+*}D\mathbf{\Phi}^*\mathbf{r}_{VP}). \tag{1.7}$$

Hessian. Levenberg-Marquardt's method for solving nonlinear least squares problems (as well as the Gauss-Newton method) uses a simplified Hessian that does not require second derivatives. This approximation is constructed with the Jacobian matrix of the vector residual:

$$\mathbf{H}(\boldsymbol{\alpha}) = \mathbf{J}^*\mathbf{J}, \tag{1.8}$$

where $\mathbf{J} = D\mathbf{P}_{\mathbf{\Phi}}^{\perp}\mathbf{y}$.

From [**17**] we know that

$$\mathbf{J} = -(\mathbf{P}_{\mathbf{\Phi}}^{\perp}D\mathbf{\Phi})\mathbf{\Phi}^+\mathbf{y} - ((\mathbf{P}_{\mathbf{\Phi}}^{\perp}D\mathbf{\Phi})\mathbf{\Phi}^+)^*\mathbf{y}.$$

Kaufman [**26**] has introduced a simplification that does not impair the efficiency of the iterative method and makes the cost of the iterations similar to that for the full functional. Kaufman's simplification consists of dropping the second term in the formula above. This is justified by observing that during calculation of the Hessian some of the terms cancel out. In summary, we can use in 1.8 the approximation:

$$\mathbf{J} = -(\mathbf{P}_{\mathbf{\Phi}}^{\perp}D\mathbf{\Phi})\mathbf{a}.$$

Ruano and his collaborators [**49, 50**] have introduced an interesting analysis that seems to indicate that from Kaufman's idea follows that there is actually a family of equally suitable Jacobians and they have introduced an even more simplified one that works well in their applications.

Alternative: no SVD's. Observing the formulas above we see that there are several multiplications by $\mathbf{\Phi}^+$. These multiplications can be interpreted as LLSQ solves, since

$$\mathbf{\Phi}^+\mathbf{y} = \mathbf{a} \qquad \Leftrightarrow \qquad \min_a \|\mathbf{\Phi}\mathbf{a} - \mathbf{y}\|_2^2.$$

$\mathbf{\Phi}^+\mathbf{y} = \mathbf{a}$ is an overdetermined system of linear equations and therefore a whole sub-space of $\mathbf{y}$'s gets mapped into the same $\mathbf{a}$. Thus, if we solve first these LLSQ problems (same matrix, multiple right-hand sides, a very economical proposition), then

$$\begin{aligned} r_{VP} &= \|\mathbf{y} - \mathbf{\Phi}\mathbf{a}\|_2^2, \\ \frac{1}{2}\nabla \mathbf{r}_{VP}(\boldsymbol{\alpha}) &= -\Re(\mathbf{a}^*D\mathbf{\Phi}^*\mathbf{r}_{\mathbf{VP}}), \\ \mathbf{J} &= -\mathbf{P}_{\mathbf{\Phi}}^{\perp}D\mathbf{\Phi}\mathbf{a}. \end{aligned}$$

Regularized VARPRO using PRAXIS. Sometimes is more expedient to use an optimization code that does not require derivatives to minimize $r_{VP}(\alpha)$. A good choice is the intelligent search method of R. Brent, implemented in the program PRAXIS [**8**]. Observe that by choosing the numerical rank of $\mathbf{\Phi}$ appropriately we will be regularizing the problem in case of severe ill-conditioning or actual rank deficiency (see [**22**] for detailed discussions on ill-conditioned problems). Finally we obtain $\mathbf{a}$ by using a regularized version of (1.4):

$$\mathbf{a} = \mathbf{\Phi}^+\mathbf{y} = \mathbf{V}\mathbf{D}_r^+\mathbf{U}^*\mathbf{y} = \mathbf{V}\mathbf{D}_r^+\widetilde{\mathbf{y}} = \sum_{j=1}^{r} v_{ij}\widetilde{y}_j/\sigma_j, \tag{1.9}$$

where we have assumed that the singular values are in descending order and that we have chosen to truncate the SVD after the first r components, so that $\sigma_{r+1}/\sigma_1 < \tau$, for a given threshold τ.

1.4. Prony-type or polynomial methods

In the next sections we consider the following simplified problem. Given N equally spaced samples of a signal (t_i, y_i), $\quad i = 1, \cdots, N$, $t_i = i\Delta t$, $\Delta t = \frac{1}{N}$, use the model function

$$\mu(t) = \sum_{j=1}^{n} a_j \phi_j(\alpha_j) = \sum_{j=1}^{n} a_j e^{\alpha_j t}, \tag{1.10}$$

with N and n to be defined for each method, in order to interpolate or best fit the data in the least squares sense.

Prony's classical method interpolates a sequence of $N = 2n$ observations by a linear combination of n exponentials, separating the computation of the nonlinear and linear parameters. The method takes advantage of the fact that the $z_j \equiv e^{\alpha_j \Delta t}$ satisfy a linear difference equation that can be written as a recurrence equation,

$$\sum_{k=1}^{n+1} \delta_k E^{k-1}\, \mu(t) = 0, \tag{1.11}$$

where E is the translation operator, $E\,\mu(t) = \mu(t + \triangle t)$, and $\boldsymbol{\delta} = (\delta_1, \delta_2, \cdots, \delta_{n+1})$ are called the *Prony recurrence parameters.*

A more convenient matrix form for the recurrence equation is

$$\mathbf{X}_\delta^\top \boldsymbol{\mu} = \mathbf{0} \tag{1.12}$$

where $\boldsymbol{\mu} = (\mu(t_1), \mu(t_2), \cdots, \mu(t_N))$, and the $(N, N-n)$ matrix $\mathbf{X}_\delta$ is the rectangular Toeplitz matrix

$$\mathbf{X}_\delta = \begin{pmatrix} \delta_1 & & & \\ \cdot & \cdot & & \\ \cdot & & \cdot & \delta_1 \\ \cdot & & & \cdot \\ \delta_{n+1} & & & \cdot \\ & & & \cdot \\ & & & \delta_{n+1} \end{pmatrix}.$$

Alternatively, if $\mathbf{y} = (y_1, y_2, \cdots, y_N)$ is the vector of data and $\mathbf{Y}(\mathbf{y})$ the $(N - n, n + 1)$ Hankel matrix defined using this vector,

$$\mathbf{Y}(\mathbf{y}) = \begin{pmatrix} y_1 & y_2 & \cdots & y_{n+1} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & & & \\ & & & \\ & & & \\ & & & \\ \cdot & \cdot & \cdot & \cdot \\ y_{N-n} & \cdot & \cdots & y_N \end{pmatrix},$$

then

$$\mathbf{X}_\delta^{\mathbf{T}} \mathbf{y} = \mathbf{Y}(\mathbf{y})\boldsymbol{\delta}.$$

Returning to equation (1.11), the z_j are the roots of the characteristic polynomial associated with $\boldsymbol{\delta}$

$$\delta_{n+1} z^n - \delta_n z^{n-1} - \cdots - \delta_1 = 0.$$

The unknown coefficients δ_k can be determined, assuming that there is no error in the observations, from the linear system of equations

$$\sum_{k=1}^{n+1} \delta_k E^{k-1}\, y_i = 0, \qquad i = 1, \cdots, n.$$

For the system to be determined, one unknown must be fixed and the Prony choice is $\delta_{n+1} = 1$.

There have been several attempts at adapting the Prony technique to the more general, *approximation* problem, i.e., a generalization to the overdetermined case, when $N \gg 2n$, leading to a least squares approximation to determine the δs. The structure of these Prony-type algorithms is: *Nonlinear stage:*

- Determine the Prony recurrence parameters from a least squares formulation.
- Determine the roots of the characteristic equation.
- Determine the nonlinear parameters α_j by taking $\ln(z_j)$.

Linear stage:

- Insert the α_j into the model and solve the resulting linear least squares problem in the a_j.

The algorithms differ in the techniques used to determine the Prony parameters. The current-day Prony's method and the Pisarenko or covariance method fail for large data sets (see **[40]**). We will describe in more detail the modified Prony method **[37, 38, 39, 40]** and the linear predictor method **[35, 51]**. The following table lists the best known algorithms.

Method	Technique
Classic Prony	Linear system $\mathbf{X}_\delta^\top \mathbf{y} = 0$
Prony	$\min_\delta \mathbf{y}^\top \mathbf{X}_\delta \mathbf{X}_\delta^\top \mathbf{y}, \quad \delta_{n+1} = 1$
Pisarenko	$\min_\delta \mathbf{y}^\top \mathbf{X}_\delta \mathbf{X}_\delta^\top \mathbf{y}, \quad \|\boldsymbol{\delta}\|_2 = 1$
Linear predictor	$\min_\delta \mathbf{y}^\top \mathbf{X}_\delta \mathbf{X}_\delta^\top \mathbf{y}, \quad \delta_K = 1,\ K > n$
Modified Prony	$\min_\delta \mathbf{y}^\top \mathbf{X}_\delta \mathbf{X}_\delta^+ \mathbf{y}, \quad \|\boldsymbol{\delta}\|_2 = 1$

The modified Prony method. The modified Prony method described in **[38]** estimates any function $\mu(t)$ that solves a linear homogeneous difference equation. This includes linear combinations of real and complex exponentials and damped/undamped sinusoids, without an *a priori* knowledge of how many terms fit best, but automatically adapting to the most appropriate number, and also, as in the other Prony-type methods, avoiding the evaluation of exponentials.

It will be assumed (see **[38]**) that the minimization problem (1.3) has a single isolated minimum for $\boldsymbol{\alpha}$ in an appropriate subset, and that $\Phi(\boldsymbol{\alpha})$ is continuously differentiable and has full rank there.

To set up a least squares formulation for the Prony parameters $\boldsymbol{\delta}$ we go back to the reduced minimization problem obtained in Section **??** for the nonlinear parameters α

$$r_{VP}(\boldsymbol{\alpha}) = \|\mathbf{r}_{VP}\ (\boldsymbol{\alpha})\|_{\mathbf{2}}^{\mathbf{2}} = ||\mathbf{y} - \mathbf{P}_{\boldsymbol{\Phi}}\mathbf{y}||_2^2 = ||\mathbf{P}_{\boldsymbol{\Phi}}^{\perp}\mathbf{y}||_2^2. \tag{1.13}$$

Here, $\mathbf{P}_{\mathbf{\Phi}} = \mathbf{\Phi}(\boldsymbol{\alpha})\mathbf{\Phi}(\boldsymbol{\alpha})^{+}$ is the projection onto the column space of $\mathbf{\Phi}$ and $\mathbf{P}_{\mathbf{\Phi}}^{\perp}$ is therefore the projection onto its orthogonal complement. But, if we set $\boldsymbol{\mu} = \mathbf{\Phi}(\boldsymbol{\alpha})\mathbf{a}$, and use (1.12), then $\mathbf{P}_{\Phi}^{\perp}$ is also the projection onto the column space of $\mathbf{X}_{\delta}$ and the reformulation as a minimization problem with respect to $\boldsymbol{\delta}$ uses the functional

$$r_{VP}(\boldsymbol{\delta}) = \mathbf{y}^{\mathrm{T}}\mathbf{P}_{X_{\delta}}\mathbf{y} = \mathbf{y}^{\mathrm{T}}\mathbf{X}_{\delta}(\mathbf{X}_{\delta}^{\mathrm{T}}\mathbf{X}_{\delta})^{-\mathbf{1}}\mathbf{X}_{\delta}^{\mathrm{T}}\mathbf{y}. \tag{1.14}$$

In this case $\mathbf{X}_{\delta}^{+} = (\mathbf{X}_{\delta}^{\mathrm{T}}\mathbf{X}_{\delta})^{-\mathbf{1}}\mathbf{X}_{\delta}^{\mathrm{T}}$, because assuming that $\mathbf{\Phi}(\boldsymbol{\alpha})$ is a full-rank matrix implies that $\mathbf{X}_{\delta}$ has full column rank as well. When using the Prony parameters $\boldsymbol{\delta}$ there is one more variable to be determined than when using the minimisation in $\boldsymbol{\alpha}$ (the characteristic polynomial is not monic); thus an additional condition must be added, for example that $\boldsymbol{\delta}$ is normalized: $\boldsymbol{\delta}^{T}\boldsymbol{\delta} = 1$.

Including a constant term, i.e., choosing $\alpha_1 = 0$ so that

$$\mu(t) = a_1 + \sum_{i=2}^{n} a_i e^{\alpha_i t},$$

is equivalent to imposing a constraint $\sum_{j=1}^{n+1} \delta_j = 0 = \mathbf{g}^{\top}\boldsymbol{\delta} = 0$ on the parameters, with $\mathbf{e}^{\top} = (1, 1, \cdots, 1)^{\top}$.

The objective function to be minimized is then

$$F(\boldsymbol{\delta}, \lambda, \nu) = r_{VP}(\boldsymbol{\delta}) + \lambda(1 - \boldsymbol{\delta}^{\top}\boldsymbol{\delta}) + 2\nu e^{\boldsymbol{T}}\boldsymbol{\delta}, \tag{1.15}$$

with λ, ν Lagrange multipliers.

Differentiating with respect to $\boldsymbol{\delta}$ and the Lagrange multipliers one obtains the necessary minimization conditions, which take the form of a generalized eigenproblem: Determine λ and $\mathbf{v}$ so that,

$$(\mathbf{A} - \lambda\mathbf{P})\mathbf{v} = 0, \tag{1.16}$$

$$\mathbf{v}^{T}\mathbf{P}\mathbf{v} = 1 \tag{1.17}$$

where

$$\mathbf{A} = \begin{pmatrix} \mathbf{B}_{\delta} & \mathbf{g} \\ \mathbf{g}^{\top} & \mathbf{0} \end{pmatrix}, \quad \mathbf{v} = \begin{pmatrix} \boldsymbol{\delta} \\ \nu \end{pmatrix}, \quad \mathbf{P} = \begin{pmatrix} \mathbf{I}_{n+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}.$$

The symmetric $(n+1) \times (n+1)$ matrix $\mathbf{B}_{\delta}(\boldsymbol{\delta})$ has elements:

$$\mathbf{B}_{\delta_{ij}} = \mathbf{y}^{\mathrm{T}}\mathbf{X}_{\delta_i}(\mathbf{X}_{\delta}^{\mathrm{T}}\mathbf{X}_{\delta})^{-\mathbf{1}}\mathbf{X}_{\delta_j}^{\mathrm{T}}\mathbf{y} - \mathbf{y}^{\mathrm{T}}\mathbf{X}_{\delta}(\mathbf{X}_{\delta}^{\mathrm{T}}\mathbf{X}_{\delta})^{-\mathbf{1}}\mathbf{X}_{\delta_i}^{\mathrm{T}}\mathbf{X}_{\delta_{\mathbf{j}}}(\mathbf{X}_{\delta}^{\mathrm{T}}\mathbf{X}_{\delta})^{-\mathbf{1}}\mathbf{X}_{\delta}^{\mathrm{T}}\mathbf{y},$$

where $\mathbf{X}_{\delta_j} = \partial\mathbf{X}_{\delta}/\partial\delta_j$.

In the case of a model without constant terms, the matrices involved are reduced to $\mathbf{A} = \mathbf{B}_{\delta}, \quad \mathbf{v} = \boldsymbol{\delta}, \quad \mathbf{P} = \mathbf{I}$.

It can be shown **[38, 39]** that the Lagrange multiplier λ must be zero at a solution of the generalized eigenproblem. The similarity to an eigenproblem suggests the use of an iterative algorithm for linear eigenproblems, where at each step the eigenvalue nearest to zero is chosen as the new $\lambda^{(k+1)}$ and the corresponding vector as $\mathbf{v}^{(k+1)}$. Convergence is assumed when $\lambda^{(k+1)}$ is small compared to $\|\mathbf{B}_{\delta}\|_2$.

The detailed minimization algorithm is described in **[39]**, and the simplifications for exponential fitting are sketched in **[40]**. See also **[38]** for some practical considerations, among them that the algorithm seems to be relatively insensitive to the starting values. The algorithm has been analyzed **[38]** as a nonlinear vector iteration in $\mathbf{v}$ and although it was not possible to obtain an estimate for the convergence rate it was asserted that the iteration will be successful if the functional at

the minimum $r_{VP}(\boldsymbol{\delta}^*)$ is small; in data fitting problems this requires small experimental error and that the model be appropriate, i.e., the number of exponential terms must be the correct one.

After the Prony parameters $\boldsymbol{\delta}$ are estimated, the rate constants $z_j = e^{\alpha_j \Delta t}$ are recovered as the roots of the characteristic polynomial:

$$p(z) = \delta_{n+1} z^n + \delta_n z^{n-1} + ... + \delta_1 = 0.$$

Unfortunately, for large values of N, i.e., small Δt (remember $\Delta t = \frac{1}{N}$), this is an ill-conditioned problem because the roots will cluster around 1. Osborne [**41**] analyses this case showing favorable asymptotic results for the convergence of VARPRO type methods that use the Gauss-Newton minimization approach.

There remains the important question of the relation between the critical point sets for the problems $r_{VP}(\boldsymbol{\delta})$ and $\|\mathbf{r}(\mathbf{a}, \boldsymbol{\alpha})\|_2^2$. In fact, the two sets may be different:

$$\min_{\boldsymbol{\delta}} r_{VP}(\boldsymbol{\delta}) \leq \min_{\boldsymbol{\alpha}} r(\boldsymbol{a}, \boldsymbol{\alpha}).$$

The Prony parametrization is more general and may yield a larger set of solutions, including for example repeated roots of the characteristic polynomial. There is, however, a close relation between the two sets as the theorem in [**40**] proves:

"The Prony parametrization does in fact solve the exponential fitting problem in the sense that if $\boldsymbol{\alpha}$ is a minimizer of problem (1.13), then the corresponding elementary symmetric functions give Prony parameters that satisfy the necessary condition (1.16)."

Fast linear prediction method. As mentioned in the Introduction, the fitting technique suggested in [**35**] and [**51**] follows the general structure of the Prony method, but in the process also determines the appropriate number of exponential terms that best represent the data.

In general, even though one knows that the data $\mathbf{y} = (y_1, y_2, \cdots, y_N)$ can be modeled by $y_i \approx \mu(t_i)$, with $\mu(t) = \sum_{i=1}^{n} a_i e^{\alpha_i t}$, where α_1 may be zero, the correct number of terms is generally not known. The model satisfies a difference equation, stated in the digital signal processing literature as a forward linear predictor with coefficients $\mathbf{f} = (f_1, f_2, \cdots, f_K)$, for any $K \geq n$,

$$\mu(t_i) = \sum_{k=1}^{K} f_k \mu(t_{i-k}),$$

or recast in the recurrence equation format,

$$\sum_{k=1}^{K+1} \delta_k \mu(t_{i-k+1}) = 0, \quad i = K+1, ..., N. \tag{1.18}$$

Here $\delta_1 = 1$ and $\delta_{K+1} = f_K,\ k = 1, \cdots, K$.

We define now a $(N-K, K+1)$ Hankel matrix $\bar{\mathbf{Y}}(\boldsymbol{\mu})$ using the $\mu(t_i)$. If the model has n exponential terms, the Hankel matrix has rank n and that is independent of the choice of K. The rank can be computed from the SVD of the "exact" matrix $\bar{\mathbf{Y}}(\boldsymbol{\mu})$, where the singular values $\bar{\sigma}_i$ will be zero from $n+1$ onwards: $\bar{\sigma}_1 \geq \bar{\sigma}_2 \geq \cdots \geqq \bar{\sigma}_n > \bar{\sigma}_{n+1} = \cdots = \bar{\sigma}_{K+1} = 0$.

Unfortunately one does not have the exact $\boldsymbol{\mu}$ but the noisy data $\mathbf{y}$. In this case, if one computes the SVD of the $(N-K, K+1)$ Hankel matrix $\mathbf{Y}(\mathbf{y}) = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$, then the singular values σ_i will all generally be different from zero. One can define n as the *numerical rank* with respect to a given tolerance τ [**20**, p. 261] if $\frac{\sigma_{n+1}}{\sigma_1} < \tau$.

This tolerance should be consistent with the data precision; for example, if the data have p correct decimal digits a good choice is $\tau = 10^{-p}$.

The matrix $\mathbf{Y}(\mathbf{y})$ can then be approximated by the truncated SVD (TSVD) expansion [**20**]:

$$\mathbf{Y}(\mathbf{y}) \approx \mathbf{U}_n \mathbf{\Sigma}_n \mathbf{V}_n^\top, \tag{1.19}$$

where $\mathbf{\Sigma}_n$ $(n \times n)$, contains the non-zero singular values of $\mathbf{Y}(\mathbf{y})$ and $\mathbf{U}_n$ $((N-K)\times n)$, $\mathbf{V}_n^\top$ $(n \times (K+1))$ are the corresponding sub-blocks of the unitary matrices involved in the SVD.

One could compute the Prony parameters $\boldsymbol{\delta}$ using the recurrence equation and the matrix $\bar{\mathbf{Y}}(\boldsymbol{\mu})$ if the signals were noiseless: $y_i = \mu(t_i)$,[2]

$$\bar{\mathbf{Y}}(\boldsymbol{\mu})_{:,1:K}\, \mathbf{f} = \bar{\mathbf{Y}}(\boldsymbol{\mu})_{:,K+1}. \tag{1.20}$$

However, this relation is only approximate for the $\mathbf{Y}(\mathbf{y})$ and

$$\mathbf{Y}(\mathbf{y})_{:,1:K}\, \mathbf{f} \approx \mathbf{Y}(\mathbf{y})_{:,K+1},$$

needs to be solved by linear least squares. Inserting the TSVD into (1.20) gives an under-determined system of equations for $\mathbf{f}$:

$$\min \|\mathbf{f}\|_2^2 \text{ such that } \mathbf{V}_{n\,1:K,1:n}^\top\, \mathbf{f} = \mathbf{V}_{n\,K+1,1:n}{}^\top.$$

After this system is solved for the $K+1$ Prony parameters $\boldsymbol{\delta}$, the relevant roots $z_j = e^{\alpha_j \Delta t}$ of the characteristic equation corresponding to the recurrence (1.18) are computed. Note that there are n roots that should be separated from the other $K-n$ extraneous roots [**51**].

There remain several practical issues: one is the choice of K. In order to obtain a reliable value of n, K should be large - for some applications between $N/3$ and $N/2$. This implies costly computations of both the SVD of $\mathbf{Y}(\mathbf{y})$ and the roots of the characteristic polynomial. In [**35**] there are some pre-processing steps that might reduce these costs.

1.5. Subspace or matrix-pencil method HTLS/HSVD

A subspace-based method starts with the model $\mu(t_i)$ for $t_i = i\triangle t$, $i = 0, \cdots, N-1$, rewritten with the change of variable $e^{\alpha_j t_i} = e^{\alpha_j i \Delta t} = z_j^i$:

$$\mu(t_i) = \sum_{j=1}^{n} a_j z_j^i. \tag{1.21}$$

To describe the algorithm we will assume that the data are noiseless $y_i = \mu(t_i)$. The first step is to arrange the model values in an $L \times M$ Hankel matrix, with L and M greater than n and $L + M = N - 1$, the number of data samples:

$$\bar{\mathbf{Y}}(\boldsymbol{\mu}) = \begin{pmatrix} \mu(t_0) & \mu(t_1) & \dots & \dots & \mu(t_M) \\ \mu(t_1) & \mu(t_2) & \dots & \dots & \mu(t_{M+1}) \\ . & . & . & . & . \\ . & . & . & . & . \\ \mu(t_L) & \mu(t_{L+1}) & . & . & \mu(t_{N-1}) \end{pmatrix}.$$

[2]For convenience we use the *colon notation* for matrices (see [**20**], pp.7): if $\mathbf{A}_{m\times n}$, then $\mathbf{A}_{k,:}$ designates the whole kth-row, whereas $A_{k,i:j}$ denotes the positions between the ith and the jth columns. Similarly for columns.

The optimal values for L and M will be discussed later. It is easy to see that $\bar{\mathbf{Y}}$ can be expressed in terms of matrices where the a_j and z_j appear explicitly - the so-called Vandermode decomposition,

$$\bar{\mathbf{Y}}(\boldsymbol{\mu}) = \bar{\mathbf{S}}\bar{\mathbf{C}}\bar{\mathbf{T}}^{\top}.$$

Here $\bar{\mathbf{S}}_{(L+1)\times n}$ and $\bar{\mathbf{T}}_{(M+1)\times n}$ are Vandermonde matrices defined by the vector $\mathbf{z} = (z_1,\, z_2, \cdots, z_n)$, and $\bar{\mathbf{C}} = \mathrm{diag}(a_1,\, a_2, \cdots, a_n)$,

$$\bar{\mathbf{Y}}(\boldsymbol{\mu}) = \begin{pmatrix} 1 & 1 & ... & 1 \\ z_1 & z_2 & ... & z_n \\ ... & ... & ... & ... \\ z_1^L & z_2^L & ... & z_n^L \end{pmatrix} \begin{pmatrix} a_1 & & & \\ & a_2 & & \\ & & & a_n \end{pmatrix} \begin{pmatrix} 1 & 1 & ... & 1 \\ z_1 & z_2 & ... & z_n \\ ... & ... & ... & ... \\ z_1^M & z_2^M & ... & z_n^M \end{pmatrix}^{\top}.$$

An interesting property of the matrix $\bar{\mathbf{S}}$ is *shift-invariance*. If $\mathbf{Z} = \mathrm{diag}(\mathbf{z})$, it can easily be proved that

$$\bar{\mathbf{S}}_{2:L+1,:} = \bar{\mathbf{S}}_{1:L,:}\mathbf{Z},$$

where $\bar{\mathbf{S}}_{2:L+1,:}$, $\bar{\mathbf{S}}_{1:L,:}$ are derived by removing respectively the first or the last row from $\bar{\mathbf{S}}$.

On the other hand, the rank of $\bar{\mathbf{Y}}(\boldsymbol{\mu})$ is n. Therefore, in terms of the thin SVD [**20**, p. 72], $\bar{\mathbf{Y}}(\boldsymbol{\mu})$ can be written as $\bar{\mathbf{Y}} = \bar{\mathbf{U}}_n\bar{\mathbf{\Sigma}}_n\bar{\mathbf{V}}_n^{\top}$, where $\bar{\mathbf{\Sigma}}_n$ contains the non-zero singular values of $\bar{\mathbf{Y}}(\boldsymbol{\mu})$, and $\bar{\mathbf{U}}_n$ $((L+1)\times n)$, $\bar{\mathbf{V}}_{\mathbf{n}}^{\top}$ $(n\times(M+1))$ are the corresponding sub-blocks of the unitary matrices involved in the normal SVD.

Comparing this expression with the Vandermonde decomposition, one can see that the columns of $\bar{\mathbf{S}}$ and $\bar{\mathbf{U}}_n$ generate the same subspace and can therefore be obtained one from the other by a multiplication with a non-singular matrix $\bar{\mathbf{Q}}$:

$$\bar{\mathbf{U}}_n = \bar{\mathbf{S}}\bar{\mathbf{Q}}.$$

But then $\bar{\mathbf{U}}_n$ inherits the shift-invariance property of $\bar{\mathbf{S}}$:

$$\bar{\mathbf{U}}_{n\,2:L+1,:} = \bar{\mathbf{U}}_{n\,1:L,:}\bar{\mathbf{Q}}^{-1}\mathbf{Z}\bar{\mathbf{Q}}. \tag{1.22}$$

The matrix $\bar{\mathbf{Q}}^{-1}\mathbf{Z}\bar{\mathbf{Q}}$ that can be computed from this equation is similar to $\mathbf{Z}$. This implies that by calculating the eigenvalues of $\bar{\mathbf{Q}}^{-1}\mathbf{Z}\bar{\mathbf{Q}}$ one has the elements of the vector $\mathbf{z}$.

In the real case, with noisy data, if the noise-to-signal ratio is small enough, these calculations can be repeated "approximately" using the Hankel matrix $\mathbf{Y}(\mathbf{y})$ instead. Now, as in the previous section, one determines the numerical rank of $\mathbf{Y}(\mathbf{y})$ with respect to a given tolerance. Assuming that it is n, the matrix $\mathbf{Y}(\mathbf{y})$ can be approximated by the rank-n matrix obtained by the TSVD,

$$\mathbf{Y}(\mathbf{y}) \approx \mathbf{Y}_n = \mathbf{U}_n\mathbf{\Sigma}_n\mathbf{V}_n^{\top},$$

where $\mathbf{\Sigma}_n = \mathrm{diag}(\sigma_1, \cdots, \sigma_n)$. The matrix $\mathbf{Y}(\mathbf{y})$ has no Vandermonde decomposition, so the shift-invariance equation (1.22) is only approximately valid and the next problem must be solved by least squares:

$$\mathbf{U}_{n\,2:L+1,:} \approx \mathbf{U}_{n\,1:L,:}\mathbf{Q}^{-1}\mathbf{Z}\mathbf{Q}. \tag{1.23}$$

One could use ordinary least squares, or the more adequate total least squares (in Magnetic Resonance Spectroscopy (MRS) applications these are known as the HSVD or the HTLS method, respectively), which has been found to be better for problems with noise-contaminated data. This is so, because the assumption used in TLS is that there are errors in both the matrix and the right-hand side and the idea is to minimize both. The application of TLS to the present problem is carefully

TABLE 1. Properties of various methods. FLP is the fast linear prediction method and M-P the HSVD matrix-pencil method.

Property	Mod. Prony	FLP	M-P	VARPRO
Non-uniform spacing	no	no	no	yes
Ill-conditioned $\mathbf{\Phi}(\boldsymbol{\alpha})$	no	yes	yes	yes
Equality constraints	no	no	no	yes
Complex models	yes	yes	yes	yes
Estimates "best" # of terms	no	yes	yes	no
Needs parameter initial guess	yes	no	no	yes

described in [**24**] for multimensional TLS problems. Both methods involve an SVD computation, either of $\mathbf{U}_{n\,1:L,:}$ for ordinary least squares, or of the augmented matrix $[\mathbf{U}_{n\,1:L};\, \mathbf{U}_{n\,2:L+1,:}]$ for HTLS.

The matrix $\mathbf{Y}(\mathbf{y})$ should be chosen as square as possible [**31**, p. 25]. The size of $\mathbf{Y}(\mathbf{y})$ will be decisive in whether an iterative Lanczos method with reorthogonalization (for larger sets) or the golub-Kahan QR based algorithm is more efficient for the computation of the SVD (see [**24**, Chapter 5] or [**20**, Chapter 9]).

Once the nonlinear parameters are known, in a second stage the linear ones are obtained, as in the Prony-type methods, by solving an appropriate linear least squares problem.

1.6. Numerical results

There are several codes based on variable projections in the public domain. A basic version can be found in Netlib [**64**] under the name of VARPRO. An extension for problems with multiple right-hand sides is VARP2, also in [**64**]. In the Port library, at the same site, there are careful implementations by Gay and Kaufman of versions for the case of constrained and unconstrained separable nonlinear problems. All these apply to linear combinations of real exponentials and many other basis functions.

We include in the Appendix an executable for a GUI based version of VARPRO using the Gay and Kaufman code as the computational engine. This program includes a pre-packaged catalogue of the most commonly used functions, such as sigmoids, Gaussians, etc., in addition to exponentials (see Documentation in the Appendix) and it is fairly straightforward to use. This code allows multiple input variables $\mathbf{t}$, which is quite useful for training Neural Networks [**43**] (Sigmoids or Radial functions) and other applications.

As mentioned in the introduction, the only available version of the modified Prony algorithm is the Matlab program by G. K. Smyth [**56**]. Although it is only implemented for models with no constant term, it can be easily modified to include this option and that is what we used in the numerical results below.

The matrix-pencil methods are straightforward to implement and there are a number of references in the specialized literature of programs tailored to specific problems in signal processing and high-resolution imaging [**31, 51**]. Again, we implemented a basic Matlab version, using ordinary least squares (HSVD) to solve problem (1.23). Table 1 summarizes some of the advantages and disadvantages of the different algorithms as described in the literature.

The variable projections implementation that we used permits initial guesses for the nonlinear parameters or provides a number of initial values at random and chooses the computations that give the best results. The modified Prony algorithm also has options, either to input suitable initial values or to compute them. For these two methods both approaches were tried.

Neither the fast linear prediction algorithm nor the matrix-pencil method require initial values. A drawback is then that they do not allow for a restriction on the possible parameter values, i.e., for the incorporation of some *a priori* information. On the other hand, when approximating data, these two subspace-based methods have as advantage that they automatically choose the most appropriate number of exponential terms. Of course, any of the methods can be run repeatedly with different number of terms and the best results (based on RMS, say) can be chosen, although there may be pitfalls associated with this approach..

All methods except VARPRO compute polynomial roots or eigenvalues of the form: $e^{\alpha_j \Delta t}$, which are therefore sensitive to the size of $\triangle t$. An additional difference, polynomial rootfinding, used in modified Prony and forward linear prediction, is an ill-conditioned problem, even more so for multiple, or clusters of roots, as is the case in some applications. On the other hand, the eigenvalue problem for a symmetric matrix, as is $\mathbf{Q^{-1}ZQ}$ for the matrix-pencil methods HTLS/HSVD, is well-conditioned.

The conditioning or sensitivity of a nonlinear least squares problem to changes in the data, i.e., an estimate of how well the parameters can be determined, is given, to a first approximation, by the condition number[3] of the Jacobian $\mathbf{J}$ of $\mathbf{r}$: $\mathbf{J}_{ij} = \frac{\partial r_i}{\partial x_j}$, at the minimum $\mathbf{x}^* \equiv (\mathbf{a}, \boldsymbol{\alpha})$.

A necessary condition for a critical point to be a minimum is that the matrix $\mathbf{H}$ be positive definite, with $\mathbf{H} = \mathbf{J}^\top \mathbf{J} + \sum_{i=1}^{N} r_i \mathbf{G}_i$, where $\mathbf{G}_i$ is the Hessian of a component r_i: $\mathbf{G}_{i_{jk}} = \frac{\partial^2 r_i}{\partial x_j \partial x_k}$. In [**7**, Chapter 9] a more geometrical interpretation using the normal curvature matrix is given.

The data sets were chosen to test two data fitting applications, parameter estimation and data representation. We include timings as a reference, although the Matlab implementations are not optimal and cannot be directly compared with the VARPRO Fortran one. For the VARPRO runs with random initial guesses, the listed time is an average of the times for 40 trials. In the tables below, we list under (# l) and (# nl) the minimum number of correct decimals of the linear and nonlinear parameters computed by the programs. The tests were run under Windows with an Intel T9300, 2.5GHz, chip.

Simulated data problems. In the following two tests we try to recover the parameters of a linear combination of exponentials to which noise has been added. A measure of the sensitivity of the parameters to data perturbations can be derived if one assumes that $r(\mathbf{a}, \boldsymbol{\alpha})$ 1.2 is well approximated by a quadratic function in a neighbourhood of the point $\mathbf{x}^* \equiv (\mathbf{a}, \boldsymbol{\alpha})$,

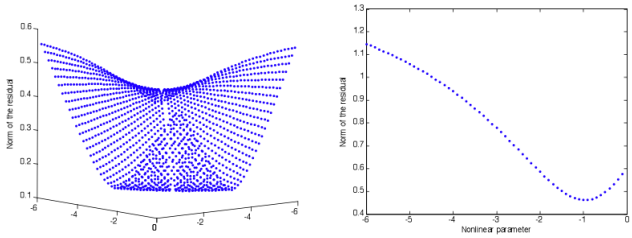
$$\mathbf{H} \triangle \mathbf{x} = -\mathbf{J}^\top \mathbf{r}. \tag{1.24}$$

Here, $\triangle \mathbf{x}$ is the parameter's perturbation.

For more details see [**61**] and also the chapter "Two exponential models for optically stimulated luminiscence" in this volume.

[3]The Euclidean condition number of a rectangular matrix $\mathbf{A}$ is $\kappa_2(\mathbf{A}) = \|\mathbf{A}\|_2 \left\|\mathbf{A}^+\right\|_2$.

FIGURE 1.6.1. Norm of residual surface as function of the nonlinear parameters. The diagonal, where the parameters coincide and there is a discontinuity is plotted separately.



Test #1. This data set is taken from **[36]**. It has 46 data points (y_i, t_i), $i = 0, \cdots, 45$, with $t_i = 0.02i$ and $y_i \approx \mu(t_i) = 4e^{-4t_i} - 4e^{-5t_i}$, with noise added to the model so that the y_i have between one and two correct decimals (all the values of y_i are in the interval $[0, 0.35]$). The methods, VARPRO, modified Prony, and matrix-pencil (HSVD) were tested.

The relative bounds for the parameters established using equation (1.24) give a large uncertainty region that is consistent with the large errors in the data:

$$\frac{|\delta a_1|}{|a_1|} \leq 12.19, \quad \frac{|\delta a_2|}{|a_2|} \leq 12.18, \quad \frac{|\delta \alpha_1|}{|\alpha_1|} \leq 1.32, \quad \frac{|\delta \alpha_2|}{|\alpha_2|} \leq 1.44.$$

A plot of the surface $r(\mathbf{a}, \boldsymbol{\alpha})$ as a function of the nonlinear parameters $\boldsymbol{\alpha} \in [-6, 0] \times [-6, 0]$, assuming that the optimal linear parameters $\mathbf{a}$ are computed via linear least squares for each $\boldsymbol{\alpha}$, shows a mostly flat surface, convex near the nonlinear parameters $(-4, -5)$, and with a "saddle" close to the $(0, 0)$ corner. For confluent parameter values $\alpha = \alpha_1 = \alpha_2$, there is a discontinuity in the surface, (the model has only one term). The plot of the curve $r(a, \alpha)$ (where $a = a_1 = a_2$) has a well defined minimum at $\alpha = 1.27$. VARPRO converges to this $\boldsymbol{\alpha}$ when starting from randomly chosen initial values (see Table). However, a check of the eigenvalues of $\mathbf{H}$ proves that this cannot be a local minimum as the matrix is not positive definite.

For the HSVD, two options were tried: allow the algorithm to estimate the appropriate number of exponential terms, given the level of noise, or force it to use the (in this case known) 2-term approximation. In this last case, even though the RMS is small, the method failed to return an approximation of the parameter values. The reason is the assumption on which the method is based, namely that the shift-invariance property (1.22) for the model matrix $\tilde{\mathbf{Y}}(\boldsymbol{\mu})$ case is approximately valid for the data matrix (1.23). For the present data set the noise is too large and this hypothesis does not hold. In fact, if instead of using the above data one decreases the noise level to 10^{-3} HSVD returns with 1-2 correct decimals, both, for the linear and the nonlinear parameters. RMS stands for Residual Mean Squares, i.e., the square root of the average sum of squares of residuals.

Test #2. This set **[35]** is obtained from a model with a constant term, $y_i \approx \mu(t_i) = 10^{-2} + 2e^{-0.5t_i} + 4e^{-t_i} + 8e^{-2t_i}$, where $t_i = 0.01i$ for $i = 0, \cdots, 999$. The y_i are derived from the model values by rounding to 6 decimal digits and adding noise of the order $O(10^{-3})$. Here, under method FLP, we list results from **[35]**.

TABLE 2. Results for Test #1

Method	Initial guess	# of terms	Max. rel. error	RMS	# digits	Time (sec.)
VARPRO	random	-	6.93e-1	5.33e-2	none	0.0018
"	$\alpha = (-1; -2)$	-	2.02e-1	1.49e-2	4l, 4nl	0.01
Mod. Prony	computed	-	2.38e-1	1.49e-2	2l, 3nl	0.46
"	$\alpha = (-1; -2)$	-	2.38e-1	1.49e-2	3l, 3nl	0.31
HSVD	-	2	2.35e-1	1.49e-2	none	0.06
"	-	estimated: 3	2.25e-1	1.48e-2	none	0.08

TABLE 3. Results for Test #2

Method	Initial guess	# of terms	Max. rel error	RMS	# digits	Time (sec.)
VARPRO	random	-	7.31e-4	3.14e-5	4l,4nl	0.05
Mod. Prony	computed	-	2.77e-1	1.89e-2	none	0.14
FLP	-	4	-	$\mathcal{O}(10^{-4})$	3l,3nl	-
HSVD	-	4	6.3e-4	3.14e-5	3l,3nl	1.92

TABLE 4. Parameter's uncertainty

parameter	bound
a_1	3×10^{-2}
a_2	3.16×10^{-2}
a_3	4.95×10^{-3}
a_4	7.47×10^{-3}
α_2	8.8×10^{-3}
α_3	1.31×10^{-2}
α_4	2.05×10^{-3}

The condition number of the Jacobian at the model parameters is $\kappa_2(\mathbf{J}) = 3.86 \times 10^3$, but the relative bounds for the parameters give a considerably smaller uncertainty region.

Here, the poor results of the modified Prony method can be explained because the roots of the characteristic polynomial $z_j = e^{\alpha_j \triangle t}$ are 0.9802, 0.9905 and 0.99501, i.e., they are close together, so that even a small perturbation in the coefficients $\boldsymbol{\delta}$ affects them.

Approximation of difficult functions with high accuracy. Next we present some results of exponential fitting for a couple of difficult functions mentioned in recent work by Beylkin, Monzon and Mohlenkamp [**4, 3**]. The challenge is that high precision is required and the problems are very ill-conditioned.

Test #3. We use the algorithms to fit $1/x$ sampled uniformly over the interval $[0.01, 1]$ (100 samples) with a linear combination of exponentials. To have an estimate for the appropriate number of terms, the data were arranged in a 55×45 Hankel matrix and the numerical rank was computed, suggesting the use of 16 terms for the approximations. However, the numerical tests with the different algorithms show that it is not possible, or of any advantage to use this many terms, as one can

TABLE 5. Results for Test #3

Method	# terms	Max. rel error	RMS	Time (sec.)
VARPRO	10 + constant	3.51e-7	3.39e-7	0.28
"	12 + constant	3.65e-7	3.17e-7	0.22
Mod. Prony	10 + constant	9.93e-4	5.37e-4	0.08
"	12 + constant	too ill-cond	-	-

TABLE 6. Results for Test #4

Method	# terms	Max. rel error	RMS	Time (sec.)
VARPRO	21	1.68e-4	2.81e-7	1.4
"	28	1.66e-4	2.81e-7	1.64
Mod. Prony	10 with constant term	0.99	0.18	0.09
"	10	too ill-cond	-	-
HSVD	13	5.63e-5	9.04e-8	1.7
"	21	5.11e-11	6.04e-14	1.65
"	99	2.33e-11	2.9e-14	1.85

see from the table below. HSVD cannot be used for this example (at $t_0 = 0$ the data is not defined).

Test #4. Finally, we consider the approximation of the Bessel function J_0, a damped oscillating function in the range $[0,\ 20\pi]$, using 1000 equally spaced sample points. In VARPRO we take as basis functions the real part of a complex exponential with a complex weight. Since $\Re(\phi(x)) = \Re[(a+ib)e^{(c+id)x}] = a\ e^{cx}(\cos(dx) - b/a\ \sin(dx))$, we can consider the real basis functions $\psi(x) = e^{cx}(\cos(dx) - \lambda\ \sin(dx))$ with real weights as our approximants. As explained above the number of terms in HSVD are chosen automatically, depending on the level of noise in the data. We considered the data correct up to 6, 12 and 16 decimals, for the 13, 21 and 99 terms approximation. The non-VARPRO methods use complex exponential approximations. In 1.6.2 we show the fit (true and approximated are indistinguishable) and the absolute error for VARPRO using 20 terms.

These results are quite competitive with those obtained by Beylkin *et al* using quite different techniques. What is very interesting in their approach is that the approximation of $1/x$, which might seem an elementary example, is transformed into a powerful tool to obtain approximations to the Green function when x is interpreted as the Laplacian. Coming from a totally different direction, related results are obtained in the article by Srivastava, Suaya, Pereyra, Suaya and Banerjee in Chapter 9 of this book.

Numerical results for the complex case. We have implemented the simplest method of section 1.3. We used subroutine CGESVD from LAPACK to calculate the SVD of the complex matrix $\mathbf{\Phi}$. We have run a number of tests using randomly generated target values for the nonlinear parameters to create artificial data sets in order to validate the algorithm. The main observation is that, as we

FIGURE 1.6.2. Results for J_o

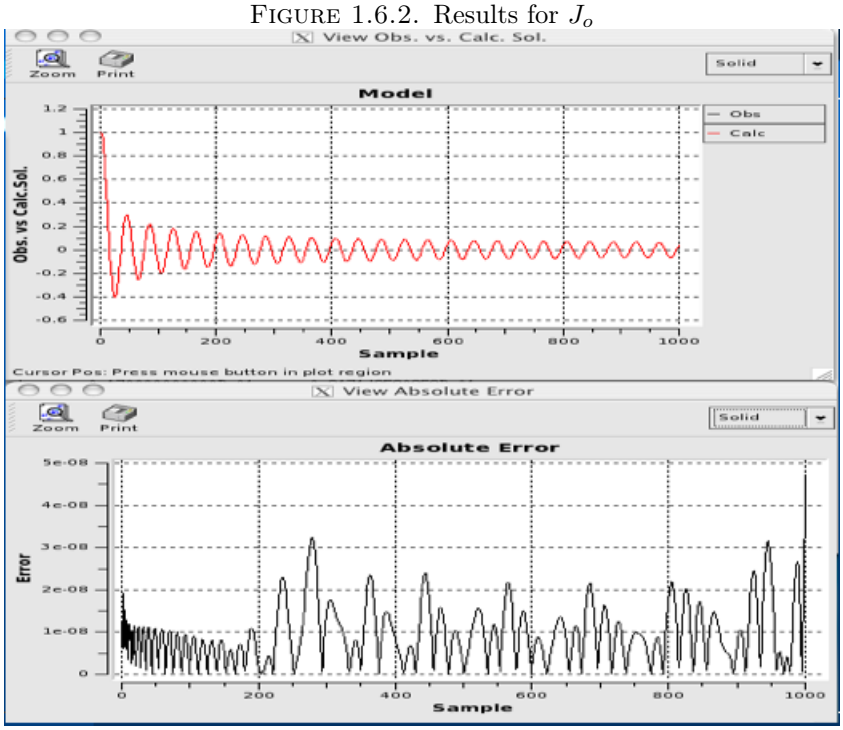


TABLE 7. Absolute errors on retrieved nonlinear parameters

$R(\alpha^* - \alpha)$	$I(\alpha^* - \alpha)$
-0.0072	-0.0045
0.12	0.19
-0.046	-0.073
0.0037	-0.015

have indicated before, it is wise to make several runs using different randomly chosen initial values and then select the best fit from them. For a particular case with 4 exponentials, we chose to make 10 runs and the 7th gave the smallest residual norm, namely 2.9 10^{-11}.

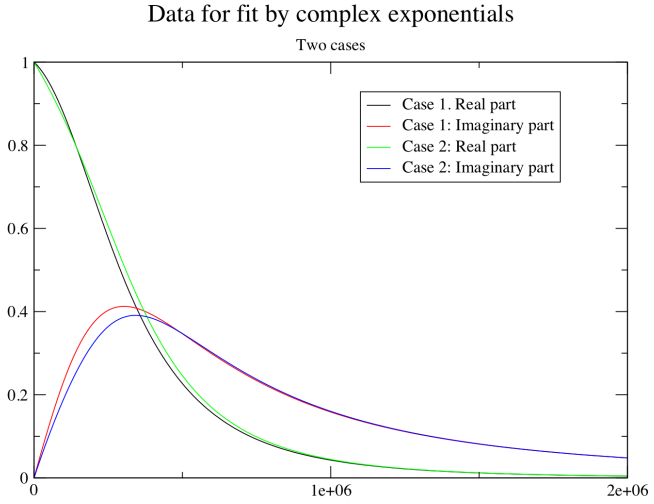
We consider now some typical real data for the problem discussed in Chapter 9 of this book by Srivastava, Suaya, Pereyra and Banerjee. The model for this problem is:

$$\mathbf{\Phi}(\lambda; \mathbf{a}, \boldsymbol{\alpha}) = \sum_{j=1}^{n} a_j e^{\alpha_j \lambda / z},$$

where $z = 2\pi 10^5(1+i)$, $i = \sqrt{(-1)}$ and λ is the independent variable. Thus, it can be writen as:

$$\mathbf{\Phi}(\lambda; \mathbf{a}, \boldsymbol{\alpha}) = \sum_{j=1}^{n} a_j e^{(1-i)\alpha_j \lambda / 4\pi 10^5},$$

FIGURE 1.6.3.



with α_j real and a_j complex. There are also some additional restrictions:

$$0 < \Re(a_j) < 1,\ \alpha_j \leq 0,\ \sum a_j = (1,\ 0).$$

The constraints on the nonlinear parameters α are introduced as a large penalty on the goal functional. However, for the VP algorithm there is no possibility of introducing constraints on the linear parameters, because they are eliminated from the problem and are recovered at the end as functions of the nonlinear ones. What happens is that through the definition of $\mathbf{a}$ (1.9), the constraints get translated into constraints on the nonlinear parameters.

In Table 8 we show some results of fitting the first data set with an increasing number of exponentials. We observe a consistent behavior, in the sense that an increasing number of exponentials improves the RMS. Two other good things seem to happen:

- The problems are not too ill-conditioned.
- The constraints on the linear parameters are satisfied automatically (to high precision!) - that is shown in column 3; recall that the a_i's are complex. This is altogether not very surprising if the model is a good approximation, because for small values of λ the data is essentially $(0, 1)$ and due to the small factor, the exponentials are essentially equal to 1 and therefore the model reduces to the sum of the linear parameters.

1.7. Some applications

A number of important applications of the above and other special methods are presented in the next chapters of this book. We survey here some additional applications related to classical and modern telecommunication and other problems, which can be cast as separable Nonlinear least squares **(SNLS)** problems of fitting

TABLE 8. Results for complex data. For cases 2, 4, 6, 8 we run only one trial; for case 10 we run 3 different randomly chosen initial values.

# exp.	RMS	Σa_i
2	0.025	(1.01,0.003)
4	0.0019	(0.9996,-2.7(-5))
6	0.00034	(0.99999, 5.2(-5))
8	0.000082	(0.9999998, -3.5(-6))
10	0.000031	(0.999989,-9.3(-6))

a linear combination of complex exponentials, where the linear coefficients represent amplitude while the nonlinear ones are the phases of the signals.

Roy and Kailath [**48**] describe in detail applications to practical signal processing problems. The objective there is to estimate from measurements a set of constant (time-independent) parameters upon which the received signal depends. Among these, high-resolution direction of arrival (DOA) estimation is important in many sensor systems such as radar, sonar, electronic surveillance, and seismic exploration. High-resolution frequency estimation is important in numerous applications, such as the design and control of robots and large flexible space structures. In such problems, the functional form of the underlying signals can often be assumed (e.g., narrow-band plane waves, cisoids). The quantities to be estimated are parameters in these functional descriptions, such as frequencies and directions of arrival for plane waves, or cisoid frequencies.

Several approaches have been developed through the years for solving these problems, including Capon's [**11**] maximum likelihood and Burg's [**9**] maximum entropy methods. These methods have significant limitations and Pisarenko was one of the first to consider the structure of the data model to estimate the parameters of cisoids in additive noise using a covariance approach. Schmidt [**52**] and Bienvenu [**6**] were the first to exploit correctly the measurement model in the case of a sensor array of arbitrary form. Schmidt's algorithm, MUSIC (MUltiple SIgnal Classification), which according to that author was inspired by the separation of variables technique, has been widely studied and was considered in an MIT study of that time as the most promising high-resolution algorithm. However, MUSIC's success came at a high computational cost that involved a search in parameter space and the storage of array calibration data.

Roy and Kailath developed a new algorithm, called ESPRIT, that dramatically reduced the computational cost and storage for sensor arrays that show what they call displacement invariance. These are arrays where the sensors come in matched pairs with identical displacement vectors.

Unfortunately, many of the earlier simplified algorithms are ineffective when some of the sources are coherent. This can stem from multipath effects or it can be introduced artificially to impede detection. Kumaresan and Shaw [**28**] and Cadzow [**10**] have studied in detail the application of separation of variables to this classical problem. More recently, a number of new algorithms have been developed to consider the more challenging problem of multiple broad-band source location. A variety of least squares modeling methods provide viable means for overcoming the difficulties of coherent sources.

Cadzow [**10**] presents a method that models the signal eigenvectors. These are linear combinations of steering vectors instead of the sensor signals, which introduces a smoothing effect and decreases the computational cost, while the use of the Variable Projection **(VP)** approach produces significant additional computational savings. As Roy and Kailath [**48**] indicate, **VP**-type algorithms were considered too expensive until fairly recently, thus justifying the use of the simplified **SVD** based ones. However, the increasing power of modern computers has rendered some of those arguments and simplified methods obsolete, especially in low signal-to-noise situations, where they do not work well.

Friedlander [**15**] has analyzed the sensitivity of the Maximum Likelihood method for the problem above. This is a separable problem and the sensitivity study involves the differentiation formulas of [**17**]. This analysis is valuable because the fast algorithms require a knowledge of the antenna array that is hard to come by in real situations, and thus have not been used as often as they deserve.

Talwar *et al.* [**58, 57**] have considered the problem of estimating co-channel digital signals using an antenna array when the spatial response of the array is unknown. Traditional techniques, such as MUSIC or ESPRIT, are dependent on the reliability of the array manifold. In the application the authors envision (mobile communications), the array manifold is poorly determined because of a highly variable propagation environment. They consider instead a block **SNLS** approach, which is both fast and reliable.

Rao and Arun [**46**] discuss the problem of estimating closely spaced frequencies of multiple, superimposed sinusoids from noisy measurements as a **SNLS** problem. This variant of the problem discussed earlier has wide applications in radio-astronomy, interference spectroscopy, seismic dataprocessing, and MR spectroscopy. Because of the cost of the computation, as compared to the simplified methods, **SNLS** is only advisable at low signal-to-noise ratios.

Zhou, Yip and Leung [**63**] consider the DOA problem for multiple moving targets by a passive array of sensors, a problem of great interest in communications, air traffic control, and tactical and strategic defense operations. In satellite and personal communication systems it is also advantageous to deploy sensor arrays to reject undesired signals. The classical techniques mentioned above deteriorate rapidly in the presence of moving targets, since they provide poor resolution because of the spread array spatial spectrum caused by the target motion. This deterioration increases with the number of sensors. Zhou *et al.* propose a maximum likelihood algorithm, where the target motion is assumed to be locally linear, which helps eliminate the spread spectrum effects and provides accurate target dynamical state estimates. Since they use the array signal model for an array of omnidirectional sensors, their approach leads to a separable problem that is solved by a **VP** method.

Lilleberg *et al.* from Nokia Mobile Phones [**32**] consider a near-far resistant iterative algorithm for multiuser signature sequence delay estimation. **VP** is used to separate the delay and data to be estimated, obtaining a so-called blind maximum likelihood estimator that does not require any knowledge of user amplitudes and data.

Heredia and Arce [**23**] have considered the splitting of a signal into a set of multilevel components as an **SNLS** problem. They use as a comparative example a system identification problem for wave propagation through a nonlinear multilayer channel, where they test the new concepts against Linear, Volterra, and Neural

Network alternatives. They show that the realization of piecewise linear filters with unknown thresholds leads to a **SNLS** problem. In the test problem they verify that the new approach can cope with the difficulties of the problem that trip the Volterra and Neural Network approaches.

Baum *et al.* [**1**] review the singularity expansion method (SEM) for quantifying the transient electromagnetic scattering from targets illuminated by pulsed EM radiation. The SEM theory suggests that the late-time scattered field of a target, interrogated by pulsed EM radiation, can be represented as a sum of natural-resonance modes. Since the excitation-independent natural frequencies depend upon the detailed size and shape of the target, the full complement of those frequencies is unique to a specified target and provides a potential basis for its identification. The first efforts to extract such natural frequencies from measured target pulse responses were based on Prony's method. However, in the practical low signal-to-noise environment in which this inverse problem occurs, only one or a few modes could be extracted reliably using that inherently unstable algorithm. Although several efforts have improved the reliability of Prony-based methods, realistic problems require a nonlinear approach, and since the problem is separable, **VP** has found another good application in the radar cross-section identification business.

In [**2**], Beece *et al.* use a **VP** algorithm in an exponential fitting problem associated with the effect of viscosity on the kinetics of the photochemical cycle of bacteriorhodopsin. Marque and Eisenstein [**34**] extend this work to consider pressure effects on the photocycle of purple membrane. By considering several kinetic data sets taken at the same temperature and pressure but with different monitoring wavelengths and an exponential model, they are able to use **VARP2** to separate the variables and efficiently solve a problem with multiple right-hand sides. The first to use this method in these type of problems was Richard Lozier [**33**], who motivated the development of the **VARP2** extension and became a champion in this field for many years (we thank Randy LeVeque for this insight).

A recent flurry of activity in using **VP** has occurred in the problem of superresolution, i.e., the combination of multiple resolution signals that requires registration (alignment) [**12, 47, 60**].

1.8. Appendix

In this Appendix we collect additional material that might be useful to some readers. First of all, we offer an executable and corresponding documentation for the Gay-Kaufman VARPRO code (nsf) **[16]** with a GUI that, we hope, will facilitate considerably its use: VARPRO Documentation, VARPRO code.
We also include a number of data sets related to the problems used to compare codes in this chapter. For tests #1 to #4 in the Numerical results section, there are corresponding files with extension .dat:
Test_1_t_y.dat,
Test_2_t_y.dat,
Test_3_t_y.dat,
Test_4_t_y.dat.

The files contain N records, each with a data pair (y_i, t_i), as described in section 1.6.

Bibliography

[1] C. E. Baum, E. J. Rothwell, K-M. Chen and D. P. Nyquist, *The singularity expansion method and its application to target identification.* Proc. IEEE **79:**1481-1492 (1991).

[2] D.Beece, S. F. Bowne, J. Czege, L. Eisenstein, H. Frauenfelder, D. Good, M. C. Marden, J. Marque, P. Ormos, L. Reinisch and K. T. Yue, *The effect of viscosity on the photocycle of bacteriorhodopsin.* Photochem. Photobiol. **33**:517-522 (1981).

[3] G. Beylkin and M. Mohlenkamp, *Numerical operator calculus in higher dimensions* Proc. Nat. Acad. Sci. U.S.A. **99**:10246-10251 (2002).

[4] G. Beylkin and L. Monzon, *On generalized Gaussian quadratures for exponentials and their applications.* App. Comput. Harm. Anal., **12**:332-373 (2002).

[5] G. Beylkin and L. Monzon, *On approximation of functions by exponential sums.* Appl. Comput. Harmon. Anal. **19**:17-48 (2005).

[6] G. Bienvenu and L. Kopp, *Adaptivity to background noise spatial coherence for high resolution passive methods.* Proc. IEEE on Acoustics, Speech, and Signal Processing, **5**:307-310 (1980).

[7] Å. Björck, *Numerical Methods for Least Squares Problems.* SIAM Pub., Philadelphia, PA (1996).

[8] R. P. Brent, *Algorithms for Minimization Without Derivatives.* Prentice Hall, Englewood Cliffs, NJ (1973).

[9] J. P. Burg, *Maximum entropy spectral analysis.* Soc. Exp. Geophys. 37th Annual Meeting Extended Abstracts (1967).

[10] J. A. Cadzow, *Multiple source location–The signal subspace approach.* IEEE Trans. on Acoustics, Speech, and Signal Processing **38:**1110-1125 (1990).

[11] J. Capon, *High-resolution frequency-wavenumber spectrum analysis.* Proc. IEEE **57:**1408-1418 (1969).

[12] J. Chung, E. Haber and J. Nagy, *Numerical method for coupled super-resolution.* Inverse Prob. **22**:1261-1272 (2006).

[13] J. E. Dennis, D. M. Gay and R. W. Welsch, *NL2SOL-An adaptive nonlinear least squares algorithm.* ACM TOMS **7**:369-383 (1981).

[14] Baron Gaspard Riche de Prony, *Essai experimental et analytique: sur les lois de la dilatabilite de fluides elastique et sur celles de la force expansive de la vapeur de l'alkool, a differentes temperatures.* J. Ecole Polyt. **1**:24-76 (1795).

[15] B. Friedlander, *Sensitivity analysis of the maximum likelihood direction-finding algorithm.* IEEE Trans. Aerosp. Electron. Syst. **26:**953-968 (1990).

[16] D. Gay and L. Kaufman, *NSF, port library*, http://www.netlib.org/port/dnsf.f (last accessed: 9/2/2009).

[17] G. H. Golub and V. Pereyra, *The differentiation of pseudoinverses and nonlinear least squares problems whose variables separate.* SIAM J. Numer. Anal. **10**:413-432 (1973).

[18] G. H. Golub and R. LeVeque, *Extensions and uses of the variable projection algorithm for solving nonlinear least squares problems.* Proc. American Numerical Analysis and Computer Conference (1979).

[19] G. H. Golub and V. Pereyra, *Separable nonlinear least squares: the Variable Projection method and its applications.* Inverse Prob. **19**:R1-R26 (2003).

[20] G. H. Golub and C. F. Van Loan; *Matrix Computations,* 3rd. ed., John Hopkins Univ. Press, Baltimore (1996).

[21] I. Guttman, V. Pereyra and H. D. Scolnik, *Least squares estimation for a class of non-linear models.* Technometrics, **1**5:209-218 (1973).

[22] P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems.* SIAM Pub., Philadelphia, PA (1998).

[23] E. A. Heredia and G. R. Arce, *Piecewise linear systems modeling based on a continuous threshold decomposition.* IEEE Trans. Signal Process. **44** (1996).

[24] S. Van Huffel and J. Vandewalle, *The Total Least Squares Problem*, SIAM Pub., Philadelphia, PA (1991).

[25] A. A. Istratov and O.F. Vyvenko, *Exponential analysis in physical phenomena.* Rev. Sc. Inst. **70**:1233-1257 (1999).

[26] L. Kaufman, *A variable projection method for solving separable non-linear least squares problems,* BIT **15**:49-57 (1975).

[27] F. T. Krogh, *Efficient implementation of a variable projection algorithm for nonlinear least squares problems.* Comm. ACM **17:**167-169 (1974).

[28] R. Kumaresan and A. K. Shaw, *Superresolution by structured matrix approximation.* IEEE Trans. Antennas Propag. **36:**34-44 (1988).

[29] S. Y. Kung, K. S. Arun and D. V. Bhaskar Rao, *State-space and singular value decomposition-based approximation methods for the harmonic retrieval problem*, J. Opt. Soc. Am. **73**: 1799-1811 (1983).

[30] C. Lanczos, *Applied Analysis.* Prentice-Hall, Englewood Cliffs, NJ (1956).

[31] T. Laudadio, *Subspace-Based Quantification of Magnetic Resonance Spectroscopy Data Using Biochemical Prior Knowledge*, Ph. D. Thesis, Faculty of Engineering, K. U. Leuven, Leuven, Belgium (2005).

[32] J. Lilleberg, E. Nieminen and M. Latva-aho, *Blind iterative multiuser delay estimator for CDMA.* Proc. IEEE Int. Symp. Personal Indoor and Mobile Radio Communications (PIMRC), pp. 565-568. Taipei, Taiwan (1996).

[33] R. H. Lozier, R. A. Bogomolni and W. Stoeckenius. *Bacteriorhodopsin: a light-driven proton pump in Halobacterium halobium.* Biophys. J. **15:**955-962 (1975).

[34] J. Marque and L. Eisenstein, *Pressure effects on the photocycle of purple membrane.* Biochem. **23**:5556-5563 (1984).

[35] H. B. Nielsen, *Multi-exponential fitting of low-field H NMR data*, Tecnical Report IMM-REP-2000-03, Dept. of Mathematical Modelling, Technical University of Denmark (2000).

[36] H. B. Nielsen, *UCTP test problems for unconstrained optimization*, Tecnical Report IMM-REP-2000-17, Dept. of Mathematical Modelling, Technical University of Denmark (2000).

[37] M. R. Osborne *A class of nonlinear regression problems*, in Data Representation, R. S. Anderssen and M. R. Osborne, eds., University of Queensland Press, St. Lucia, pp. 94-101 (1970).

[38] M. R. Osborne *Some special nonlinear least squares problems.* SIAM J. Numer. Anal. **12**:571-592 (1975).

[39] M. R. Osborne and G. K. Smyth, *A modified Prony algorithm for fitting functions defined by difference equations.* SIAM J. Sci. Comp. **1**2:362-382 (1991).

[40] M. R. Osborne and G. K. Smyth, *A modified Prony algorithm for exponential function fitting.* SIAM J. Sci. Comp. **16**:119-138 (1995).

[41] M. R. Osborne, *Separable least squares, variable projections, and the Gauss-Newton algorithm.* ETNA 28:1-15 (2007).

[42] J. M. Papy, L. De Lathauwer and S. Van Huffel, *Exponential data fitting using multilinear algebra: The single-channel and multi-channel case.* Numer. Lin. Alg. Appl. **12**:809-826 (2005).

[43] V. Pereyra, G. Scherer and F. Wong, *Variable projections neural network training.* Mathematics and Computers in Simulation **73**:231-243 (2006).

[44] V. Pereyra, *Fast computation of equispaced Pareto manifolds and Pareto fronts for multiobjective optimization problems.* Math. Comput. Simul. **79**:1935-1947 (2009).

[45] V. Pereyra, M. Saunders and J. Castillo, *Equispaced Pareto front construction for constrained multiobjective optimization.* CSRC, Tecn. Rep., San Diego State Univ. (2009).

[46] B. D. Rao, and K. S. Arun, *Model based processing of signals: a state space approach.* Proc. IEEE **80:**283-309 (1992).

[47] D. Robinson, F. Farsiu and P. Milanfar, *Optimal registration of aliased images using variable projection with applications to super-resolution.* Comp. J. (2007).

[48] R. Roy and T. Kailath, *ESPRIT-estimation of signal parameters via rotational invariance techniques.* IEEE Trans. Acoust. Speech Signal Process. **37:**984-995 (1989).

[49] Antonio E. Ruano, Pedro M. Ferreira, C. Cabrita and S. Matos, *Training neural networks and neuro-fuzzy systems: an unified view.* Proc. IFAC 15th Triennial World Congress (2002).

[50] Antonio E. Ruano, P. J. Fleming and D. I. Jones, *Connectionist approach to PID autotuning*. Cont. Th. Appl. IEEE Proc. D **139**:279-285 (2002).

[51] T. K. Sarkar and O. Pereira, *Using the matrix pencil method to estimate the parameters of a sum of complex exponentials*, IEEE Antennas Propag. **37**: 48-55 (1995).

[52] R. O. Schmidt, *Multiple emitter location and signal parameter estimation.* Proc. RADC Spectrum Estimation Workshop (1979).

[53] G. A. F. Seber and C. J. Wild, *Nonlinear Regression.* Wiley Interscience, New York (2003).

[54] R. I. Shrager and R. W. Hendler, *Some pitfalls in curve-fitting and how to avoid them: A case in point*. J. Biochem. Biophys. 36:157-173 (1998).

[55] J. Sjöberg, and M. Viberg, *Separable non-linear least squares minimization – possible improvements for neural net fitting*. IEEE Workshop in Neural Networks for Signal Processing. Amelia Island Plantation, FL (1997).

[56] G. Smyth, http://www.statsci.org/other/prony.html, 30 June 2009.

[57] S. Talwar, *Blind Space-Time Algorithms for Wireless Communication Systems*. Ph. D. Thesis, SCCM, Stanford University (1996).

[58] S. Talwar, M. Viberg and A. Paulraj, *Blind estimation of multiple co-channel digital signals arriving at an antenna array.* IEEE SP Letters **1:**29-31 (1994).

[59] M. L. Van Blaricum and R. Mittra, *Problems and solutions associated with Prony's method for processing transient data*, IEEE Trans. Antennas Propag. **AP-26**:174-182 (1978).

[60] P. Vandenwalle, *Super-Resolution From Unregistered Aliased Images*. Master Thesis in E. E., Katholieke Univ. Leuven, Belgium (2006).

[61] J. M. Varah, *On fitting exponentials by nonlinear least squares*. SIAM J. Sci. Stat. Comput. **6**:30-44 (1985).

[62] H. Wold and E. Lyttkens, *Nonlinear iterative partial least squares (NIPALS) estimation procedures*. Bull. ISI **43:**29-51 (1969).

[63] Y. Zhou, P. C. Yip and H. Leung, textitTracking the direction-of-arrival of multiple moving targets by passive arrays: algorithm. IEEE Trans. Signal Proc. **47:**2655-2666 (1999).

[64] http://www.netlib.org, 30 June 2009.