Matrix Computations
and
Signal Processing

Charles Van Loan *

87-863

September 1987

Department of Computer Science
Cornell University
Ithaca, New York 14853-7501

*This work was partially supported by ONR contract N00014-83-K-640 and NSF contract DCR 86-2310, and the U.S. Army Research Office through the Mathematical Sciences Institute of Cornell University. It will appear in Signal Processing, Prentice-Hall, Simon Haykin (ed.)
Matrix Computations and Signal Processing

Charles Van Loan
Department of Computer Science
5126 Upson Hall
Cornell University
Ithaca, New York 14853

Abstract

The interaction between the signal processing and matrix computation areas is explored by examining some subspace dimension estimation problems that arise in a pair of direction-of-arrival algorithms: MUSIC and ESPRIT. We show that the intelligent handling of these numerical problems requires a successful intermingling of perturbation theory, sensible problem formulation, and reliance upon unitary matrix methods.

This work is partially supported by ONR contract NO0014-83-K-640 and NSF contract DCR 86-2310. It will appear in Signal Processing, Prentice-Hall, Simon Haykin (ed).
Introduction

Signal processing is an application area that has profited by recent research developments in matrix computations. In this paper we examine the synergism between these two fields as suggested by the following diagram:

![Diagram showing the relationship between Perturbation Theory, Algorithmic Development, and Problem Formulation.]

As a vehicle for studying these interactions between computing, mathematics, and engineering we have chosen to examine selected portions of the MUSIC and ESPRIT methods for Direction-of-Arrival (DOA) estimation. The basic problem in DOA estimation is to compute the location of \( d \) unknown signals given the output of \( n \) sensors. Various assumptions about the signals, the noise, and the array geometry must, of course, be made. We only focus on the interesting linear algebra associated with particular implementations of the two algorithms.

In both the MUSIC and ESPRIT procedures the number of signal sources \( d \) is estimated by computing the multiplicity of certain eigenvalues. Rank/dimension/multiplicity calculations in the presence of roundoff error and fuzzy data are notoriously tricky and require a good bit of advanced numerical linear algebra. This makes the MUSIC and ESPRIT algorithms ideal for illustrating the more "philosophical" connections between matrix computations and signal processing.

The order of presentation is as follows. First we make some general remarks about perturbation theory and how it can be used in the assessment of a numerical procedure. For the sake of simplicity, we use Gaussian elimination as an example and discuss the importance of the
"nearness to singularity" concept. Unitary matrix methods are then shown to be crucial to the intelligent handling of various rank determination problems. The singular value, CS, and Schur decompositions are stressed. After this "trip" into numerical linear algebra we focus on the MUSIC and ESPRIT computations and discuss aspects of their reliable implementation.

Our coverage of these DOA estimation methods is by no means conclusive. There are a few new results but many interesting MUSIC/ESPRIT research questions remain. And again, we are merely using these techniques to dramatize the value of "unitary matrix methodology" in signal processing.

### Perturbation Theory and Algorithm Assessment

Developing an effective algorithm for a problem and understanding the associated perturbation theory go hand-in-hand in scientific computation. We review various stability/perturbation concepts in the simple setting of linear equation solving. The quality of a linear equation solver cannot be assessed without an understanding of \( Ax = b \) sensitivity. How does \( x \) change if the elements of \( A \) and \( b \) are perturbed? An elementary perturbation theory (c.f Golub and Van Loan (1983,p.24ff)) tells us that if \((A + \Delta A)(x + \Delta x) = (b + \Delta b)\) and both \(\|\Delta A\|_2/\|A\|_2\) and \(\|\Delta b\|_2/\|b\|_2\) are \(O(\epsilon)\) with \(\epsilon << 1\), then

\[
\|\Delta x\|_2 \approx \epsilon \kappa_2(A).
\]

(H)

Here \(\kappa_2(A) = \|A\|_2 \|A^{-1}\|_2\) is the 2-norm condition of \(A\) with respect to inversion. It is easy to show that \(\kappa_2(A) \geq 1\) (always) and that as \(A\) gets "close" to being singular, \(A^{-1}\) and \(\kappa_2(A)\), blow up. The heuristic (H) says that \(O(\epsilon)\) changes in the data \(A\) and \(b\) usually induce \(O(\epsilon \cdot \kappa_2(A))\) changes in the solution \(x\).

This result can be used to analyze the floating point performance of Gaussian elimination with pivoting. In particular, if a system \(Ax = b\) is
solved by this method then the computed solution \( \hat{x} \) exactly solves a "nearby" problem in the sense that

\[
(A + E) \hat{x} = b \quad \| E \|_2 \approx u \| A \|_2
\]

where \( u \) is the machine precision. This shows that the method is stable, i.e., the algorithm does not compound the underlying mathematical sensitivity of the problem. This, does not imply that \( \hat{x} \) is accurate for if we interpret (E) using (H) then the best thing we can say about \( \hat{x} \)'s relative error is that \( \| \hat{x} - x \|_2 / \| x \|_2 \approx u \kappa_2(A) \). A stable linear equation solver such as Gaussian elimination with pivoting cannot be faulted for producing inaccurate results if the matrix \( A \) has a large condition number relative to the machine precision \( u \).

For many of the basic problems in linear algebra, the perturbation/condition number theory has been worked out and provides a lot of practical guidance in the assessment of algorithms and computed results. See Golub and Van Loan (1983), Stewart (1973), Stewart (1977), and Van Loan (1987).

**Rank Determination and the Singular Value Decomposition**

In the linear equation problem, a central issue concerns nearness to singularity. More generally we have the problem of estimating the dimension of rectangular matrix range, i.e., rank. From the perspective of pure mathematics, the notion of matrix rank is very crisp:

If \( A \in \mathbb{C}^{m \times n} \) and \( m \geq n \), then \( A \) has rank \( n \) if and only if it has \( n \) independent columns.

Full column rank is a yes-no, 0-1 proposition. Either a matrix has it or it does not. Unfortunately, fuzzy data and inexact arithmetic complicate the practical treatment of rank. Special tools are needed and one of the most useful in this regard is the singular value decomposition (SVD).
Theorem 1 (SVD)

If $A \in \mathbb{C}^{m \times n}$ ($m \geq n$) then there exist unitary $U \in \mathbb{C}^{m \times m}$ and $V \in \mathbb{C}^{n \times n}$ such that $U^H A V = \Sigma = \text{diag}(\sigma_1, ..., \sigma_n)$ where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$. The $\sigma_k$ are called the singular values. The columns of $U$ and $V$ are referred to as the corresponding left and right singular vectors.

Proof.

See Golub and Van Loan (1983, )

The SVD provides quantitative answers to a number of important questions that arise in practical signal processing work.

Q1. How close is a matrix to one of lower rank?

A1. If $A$ has singular values $\sigma_1 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0$ then $\text{rank}(A) = r$ and

$$\min \| A - B \|_2 = \sigma_r$$

$$\text{rank}(B) < r$$

In particular, if $A \in \mathbb{C}^{n \times n}$ is nonsingular then $\sigma_n$ is the distance to the set of singular matrices. Thus, $1/\kappa_2(A) = \sigma_n/\sigma_1 = \sigma_n / \| A \|_2$ is a relative measure of nearness to singularity.

Q2. What is the range and null space of a matrix?

A2. If $U = [u_1, ..., u_m]$ and $V = [v_1, ..., v_n]$ are column partitionings of the left and right singular vector matrices and $\text{rank}(A) = r$ then $\text{Null}(A) = \text{span} \{ v_{r+1}, ..., v_n \}$ and $\text{Range}(A) = \text{span} \{ u_1, ..., u_r \}$. 
Q3. How close are two $k$-dimensional subspaces of $\mathbb{C}^n$?

A3. Suppose the $n$-by-$k$ matrices $Y = [y_1, \ldots, y_k]$ and $Z = [z_1, \ldots, z_k]$ have orthonormal columns and that $S_1 = \text{Range}(Y)$ and $S_2 = \text{Range}(Z)$. If $\sigma_1 \geq \cdots \geq \sigma_k$ are the singular values of $Y^HZ$ then

$$\text{dist}(S_1, S_2) = \min_{y \in S_1, z \in S_2} \| y - z \|_2 = \sqrt{1 - \sigma_1^2}$$

Q4. How close are two matrices to having a common null vector?

A4. If $A$ and $B$ are $m \times n$ matrices and $\sigma_1 \geq \cdots \geq \sigma_n$ are the singular values of

$$C = \begin{bmatrix} A \\ B \end{bmatrix}$$

then there exist matrices $E_A$ and $E_B$ satisfying $\| E_A \|_2, \| E_B \|_2 \leq \sigma_n$ with the property that

$$\text{Null}(A + E_A) \cap \text{Null}(B + E_B) \neq \{0\}.$$ 

Proofs of these and other SVD properties can be found in Golub and Van Loan (1983).

Computing the SVD

There are a number of ways to compute the SVD. The most important for us is the Golub-Reinsch (1970) procedure which is a derivative of the symmetric QR algorithm. It stably exploits the connection between the
SVD of $A \in \mathbb{C}^{m \times n}$ and the $n$-by-$n$ Hermitian eigenvalue decomposition $U^H (A^H A) U = \Sigma^H \Sigma$. Implementations may be found in the software packages LINPACK and EISPACK. In either instance the computed $\hat{U}$ and $\hat{V}$ are "unitary" to machine precision $u$ meaning that

$$\| \hat{U}^H \hat{U} - I_n \|_2, \; \| \hat{V}^H \hat{V} - I_n \|_2 \approx u$$

One can also show that the computed singular values are the exact singular values of a matrix $A + \Delta A$ where $\| \Delta A \|_2 \approx u \| A \|_2$. From this result it is possible to show that the computed singular values $\hat{\sigma}_k$ satisfy

$$| \hat{\sigma}_k - \sigma_k | \approx u \cdot \sigma_1 \quad k = 1:n$$

This implies that the SVD is guaranteed to detect near rank-deficiency in practice. That is, if $A$ is close to rank deficient then $\sigma_n / \sigma_1$ and its computed analog $\hat{\sigma}_n / \hat{\sigma}_1$ would be small in an order of magnitude sense.

Other methods can be used for SVD computations. If $A$ is sparse then the Lanczos algorithm may be of interest. (See Golub, Luk, and Overton (1981).) In multiprocessor environments the block Jacobi method has exhibited some potential. (See Bischof (1986a,1986b) and Van Loan (1986).) If just the smallest singular value and associated singular vectors are required, then inverse iteration can be effective. See (Van Loan (1987)).

Throughout this paper we tacitly assume that all SVD computations are performed with the LINPACK implementation of the Golub-Reinsch algorithm.

**Unitary Matrix Methods**

The SVD algorithm is just one member of the unitary matrix method family. As we show in subsequent sections, the intelligent handling of numerical rank usually involves use of these methods in conjunction with the SVD. We summarize them for later reference.
(1) **QR Factorization**

If $A \in \mathbb{C}^{m \times n}$ then there exists an unitary $Q \in \mathbb{C}^{m \times m}$ such that $Q^H A = R$ is upper triangular. This is effectively Gram-Schmidt orthogonalization. If $A = [a_1, \ldots, a_n]$ has rank $n$ and $Q = [q_1, \ldots, q_m]$ then span $\{a_1, \ldots, a_k\} = \text{span} \{q_1, \ldots, q_k\}$, $k = 1:n$.

(2) **Schur Decomposition**

If $A \in \mathbb{C}^{n \times n}$ then there exists a unitary $Q \in \mathbb{C}^{n \times n}$ such that $Q^H A Q = T$ is upper triangular. The diagonal of $T$ is made up of $A$'s eigenvalues. If $Q = [q_1, \ldots, q_n]$ then span$\{q_1, \ldots, q_k\}$ is an invariant subspace associated with the eigenvalues $t_{11}, \ldots, t_{kk}$.

(3) **Hermitian Schur Decomposition**

If $A \in \mathbb{C}^{n \times n}$ is Hermitian then there exists a unitary $Q \in \mathbb{C}^{n \times n}$ such that $Q^H A Q = D = \text{diag}(\lambda_1, \ldots, \lambda_n)$. If $Q = [q_1, \ldots, q_n]$ is a column partitioning then $A q_k = \lambda_k q_k$, $k = 1:n$.

(4) **Generalized Schur Decomposition**

If $A, C \in \mathbb{C}^{m_1 \times n}$ ($m_1 \geq n$) and $B, D \in \mathbb{C}^{m_2 \times n}$ ($m_2 \geq n$) then there exist unitary $Q, U, V$, and $Z$ of appropriate dimension such that the matrices $T_A = U^H A Q$ and $T_B = V^H B Q$ are lower triangular and the matrices $T_C = U^H C Z$ and $T_D = V^H D Z$ are upper triangular. Since $Q^H (A^H C - \lambda B^H D) Z = T_A^H T_C - \lambda T_B^H T_D$, it follows that this matrix is singular whenever $\lambda = \alpha_i \gamma_i / \beta_i \delta_i$ where the $\alpha_i, \beta_i, \gamma_i$ and $\delta_i$ are the diagonals of $T_A$, $T_B$, $T_C$, and $T_D$, respectively.
(5) CS Decomposition

If

\[ Q = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \]

satisfies \( Q^H Q = I_n \) and \( m_1 \) and \( m_2 \) are each larger than \( n \), then there exist unitary \( U_1 \in \mathbb{C}^{m_1 \times m_1} \), \( U_2 \in \mathbb{C}^{m_2 \times m_2} \), and \( V \in \mathbb{C}^{n \times n} \) such that

\[
\begin{bmatrix}
U_1 & 0 \\
0 & U_2
\end{bmatrix}^H
\begin{bmatrix}
Q_1 \\
Q_2
\end{bmatrix} V =
\begin{bmatrix}
C \\
S
\end{bmatrix}
\]

where

\[
C = \text{diag}(\cos(\theta_1), \ldots, \cos(\theta_n))
\]

\[
S = \text{diag}(\sin(\theta_1), \ldots, \sin(\theta_n))
\]

The QR factorization is implemented in LINPACK. The Schur and Hermitian Schur decompositions are part of EISPACK as is the generalized Schur decomposition for the special case \( A = B = I_n \), i.e., the problem \( Cx = \lambda Dx \). An algorithm for the generalized Schur decomposition is discussed in Van Loan (1975). The CS decomposition is described in Davis and Kahan (1970), Stewart(1977), Paige and Saunders (1981), Stewart(1983), and Van Loan (1985). Algorithms are given in the last two references. Note that it amounts to a pair of SVDs.
Three Examples of Practical Rank/Subspace Dimension Determination

The CS and generalized Schur decompositions turn out to be quite important in our discussion of the MUSIC and ESPRIT methods. Before we pursue this we step through some simpler practical problems that illustrate the value of unitary matrix methods in subspace dimension estimation.

(a) Eigenvalue Multiplicity

It is often necessary to deduce the multiplicity of a computed eigenvalue. Suppose $A$ is 2-by-2 and that a unitary matrix $Q$ is found with the property that

$$Q^H(A+E)Q = T = \begin{bmatrix} \lambda_1 & 1 \\ 0 & \lambda_2 \end{bmatrix} \quad \| E \| \approx u \| A \|$$

where $u$ is the machine precision. This says that the computed eigenvalues $\lambda_1$ and $\lambda_2$ are the exact eigenvalues of a matrix near to $A$. (The eigenvalue routines in EISPACK permit one to make such statements.) Under what conditions may we assume that the original $A$ has a repeated eigenvalue? A possible criteria is

$$| \lambda_1 - \lambda_2 | < u \| A \|$$

However, it is not hard to show that if $| \lambda_1 - \lambda_2 | = \epsilon \ll 1$ then

$$\sigma_2(A - \lambda_1 I) = O(\epsilon^2).$$

It then follows from SVD theory that $\lambda_1$ is a multiple eigenvalue of a matrix $A + \Delta A$ with $\| \Delta A \|_2 = O(\epsilon^2)$. Thus, $A$ could be a lot closer to a
defective matrix than the mere inspection of the computed eigenvalues reveals. A more reasonable criteria for eigenvalue multiplicity in the \( n = 2 \) case would be

\[
\left| \lambda_1 - \lambda_2 \right|^2 < \| A \|
\]

The role of the SVD in eigenvalue multiplicity determination is discussed in Golub and Wilkinson (1976) and Kagstrom and Ruhe (1980).

(b) Rank Deficient Least Squares Solution

It can be shown that if \( U^H A V = \Sigma \) is the SVD of \( A \) with \( U = [u_1, \ldots, u_m] \)
and \( V = [v_1, \ldots, v_n] \), then

\[
x_{LS} = \sum_{k=1}^{r} \left( \frac{u_k^H b}{\sigma_k} \right) v_k, \quad r = \text{rank}(A)
\]

is the minimum 2-norm solution of the least squares problem

\[
\min \| A x - b \|_2 \quad \text{subject to} \quad A \in \mathbb{C}^{m \times n}, \ b \in \mathbb{C}^m, \ m \geq n
\]

The associated minimum residual is then given by

\[
\rho_{LS} = \| A x_{LS} - b \|_2 = \| \left[ u_{r+1} \ldots u_m \right]^H b \|_2
\]

In general, computed singular values are never exactly zero and so we need some procedure for computing an estimate \( \hat{r} \) of the rank \( r \). One possibility is to let \( \hat{r} \) be the largest integer so that

\[
\hat{\sigma}_r \geq \varepsilon \cdot \hat{\sigma}_1
\]
where $\epsilon$ is some small parameter that may depend upon the machine precision and/or the accuracy of the data. With this criteria

$$
\hat{x}_{LS} = \hat{r} \sum_{k=1}^{\hat{r}} \left( \hat{u}_k \hat{T}_b / \hat{\sigma}_k \right) \hat{v}_k
$$

can be regarded as a reasonable approximation to the true $x_{LS}$. Here, the "hat" notation is used to designate computed quantities. Whether or not this is an appropriate way to address a near rank deficient least squares problem depends upon the application. However, with the SVD of $A$ available, the ramifications of an individual $\hat{r}$ choice can be readily explored.

(c) System Controllability

Suppose we are given the system

\begin{equation}
\dot{x}(t) = Ax(t) + u(t) \cdot b \quad x(0) = x_0
\end{equation}

where $A \in \mathbb{R}^{nxn}$, $b \in \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$. Is it possible find a control function $u(t)$ such that $x(T) = x_T$ where $x_T \in \mathbb{R}^n$ is some desired "state" that we wish the system to be in at time $T > 0$? Clearly, this may not be possible. For example, if $b = 0$ then it is impossible to "control" the $x$-vector as required. More generally, if the above problem is to have a solution then $b$ must not be deficient in certain directions that can be defined in terms of $A$'s eigensystem. Here are two of the many ways that the "controllability" of (S) can be characterized:

The system (S) is controllable iff $W_1 = [b, Ab, A^2 b, ..., A^{n-1} b]$ is nonsingular.
The system (S) is controllable iff \( W_2 = \int_0^T e^{At} b b^T e^{A^T t} \, dt \) is nonsingular.

These are "0-1", yes-no characterizations. Our intuition tells us that if either \( W_1 \) or \( W_2 \) are nearly singular, then (S) must somehow be "hard" to control. The situation is thus ripe for singular value analysis. The smallest singular value of either \( W_1 \) or \( W_2 \) may be taken as a measure of how close the system (S) is to being uncontrollable. The use of SVD and related techniques in computational control theory has led to a healthier engineering perspective. Tools now exist for measuring things like nearness to uncontrollability that can assist in the design of robust systems. The spirit of numerical linear algebra's role in control theory is very well illustrated in Paige (1981).

The MUSIC Problem

The MUSIC procedure is due to Schmidt (1979, 1981, 1986). We assume that the vector \( x \in \mathbb{C}^n \) of received waveforms satisfies

\[
x = Af + n_x
\]

where \( A \in \mathbb{C}^{n \times d} \) is a function of the arrival angles and array element locations, \( f \in \mathbb{C}^d \) is the vector of incident signals, and \( n_x \in \mathbb{C}^n \) is the noise. If

\[
S_1 = E(xx^H) \quad \text{(signal+noise covariance)}
\]

\[
S_2 = E(n_x n_x^H) \quad \text{(noise covariance)}
\]

\[
P = E(ff^H) \quad \text{(incident signal covariance)}
\]
then it is easy to see that

\[ S_1 = APA^H + S_2 \]

The idea behind MUSIC is that under reasonable assumptions we can characterize the number of signals \( d \) in terms of the generalized eigenvalues of \( S_1 - \lambda S_2 \).

**Theorem 2.**

Let \( S_2 \in \mathbb{C}^{nxn} \) and \( P \in \mathbb{C}^{dxd} \) be Hermitian positive definite matrices and assume \( S_1 = APA^H + S_2 \) where \( A \in \mathbb{C}^{nxd} \) has rank \( d \). Then \( \lambda = 1 \) is the smallest zero of the polynomial \( p(\lambda) = \det(S_1 - \lambda S_2) \) and it has multiplicity \( n-d \).

**Proof.**

If \( S_1 z = \lambda S_2 z \) with \( 0 \neq z \in \mathbb{C}^n \) then

\[ z^H S_1 z = z^H (S_2 + APA^H) z = \lambda z^H S_2 z . \]

Thus, \( \lambda = 1 + z^H (APA^H) z / z^H S_2 z \). It follows that \( \lambda = 1 \) iff \( A^H z = 0 \). Since \( \text{dim}(\text{Null}(A^H)) = n-d \) it follows \( \lambda = 1 \) has multiplicity \( n-d \).

After the number of signals \( d \) is computed in MUSIC, the DOAs are found by solving a nonlinear minimization problem that requires orthonormal vectors \( z_{d+1}, \ldots, z_n \) such that

\[ (M) \quad \text{span}(z_1, \ldots, z_{n-d}) = \{ x \mid S_1 x = \lambda_{\text{min}} S_2 x \} \]

In practice we must estimate the covariance matrices \( S_1 \) and \( S_2 \). To this end we assume that a "signal-plus-noise" matrix \( A_S \) and a "noise alone" matrix \( A_N \) have been collected such that
\[ A_S^H A_S \approx E(x^H x) = APA^H + E(n_x n_x^H) \approx APA^H + A_N^H A_N \]

The generalized eigenvalue computations now become *generalized singular value* computations. In particular we must compute an orthonormal set \( \{ z_{d+1}, \ldots, z_n \} \) such that

\[(M') \quad \text{span}\{z_{d+1}, \ldots, z_n\} = \{ x \mid A_S^H A_S z \approx \lambda_{\text{min}} A_N^H A_N z \}\]

where \( \lambda_{\text{min}} \) is the smallest root of \( p(\lambda) = \det(A_S^H A_S - \lambda A_N^H A_N) \). Recognize \((M')\) as an approximation to \((M)\). The "\(\approx\)" is necessary because \( A_S^H A_S \approx S_1 \) and \( A_N^H A_N \approx S_2 \) and \( \lambda_{\text{min}} \) might not be repeated. Before we show how to cope successfully with these approximations it is instructive to examine what may be called an "eigenvector approach" to \((M')\).

**MUSIC and the Generalized Singular Value Problem**

Generalized singular value problems of the form

\[ A_S^H A_S x = \lambda A_N^H A_N x \]

have a number of nice properties that we state without proof.

**Theorem 3.**

If \( A_S \in \mathbb{C}^{m_1 \times n} \) \((m_1 \geq n)\) and \( A_N \in \mathbb{C}^{m_2 \times n} \) \((m_2 \geq n)\), then there exists a nonsingular \( x \in \mathbb{C}^{n \times n} \) such that

\[ x^H (A_S^H A_S) x = \text{diag}(\alpha_1, \ldots, \alpha_n) = D_S \quad \alpha_k \geq 0 \]

and

\[ x^H (A_N^H A_N) x = \text{diag}(\beta_1, \ldots, \beta_n) = D_N \quad \beta_k \geq 0 \]
Proof.

See Golub and Van Loan (1983,p.314) \( \square \)

The quantities \( \sqrt{\frac{\alpha_k}{\beta_k}} \) are called the generalized singular values of the pair \( \{ A_S,A_N \} \) and the columns of \( X \) the associated generalized singular vectors. Note that if \( X = [ x_1, \ldots, x_n ] \) is a column partitioning, then

\[
\beta_k (A_S^H A_S) x_k - \alpha_k (A_N^H A_N) x_k = 0 \quad k = 1:n
\]

Let's assume that all the \( \beta_k \) are nonzero. It follows that if \( \lambda_k = (\alpha_k / \beta_k)^2 \) then \( (A_S^H A_S - \lambda_k A_N^H A_N) x = 0 \). One approach to \( (M') \) would be to

(a) Find \( X \) and order the columns so that the \( \lambda_k = \alpha_k / \beta_k \) range from large to small.

(b) Determine \( d \) such that \( \lambda_d > \lambda_{d+1} \geq \epsilon \geq \lambda_{d+1} \geq \cdots \geq \lambda_n \geq 0 \) for some small tolerance \( \epsilon \).

(c) Compute the QR decomposition \( ZR = [ x_{d+1}, \ldots, x_n ] \). The columns of \( Z \) form the desired orthonormal basis.

The "standard" methods for computing \( X \) are all flawed and are instructive to look at. Again we assume that \( A_N \) has full rank which guarantees nonzero \( \beta_k \).

**Method 1.**

1. \( S_1 = A_S^H A_S ; S_2 \leftarrow A_N^H A_N \) /* Form cross-products:
2. \( S_2 = LL^H \) /* \( L = \) Cholesky lower triangle
3. Solve: \( LW = S_1 ; LC = W^H \) /* \( C = L^{-1} S_1 L^{-H} \)
4. \( Q^H C Q = D_S = \text{diag}(\alpha_k) \) /* Compute \( C \)'s eigensystem, \( Q^H Q = I \)
5. Solve: \( L^H X = Q \) /* \( D_N = I_n \)
This method is of dubious quality in that if $A_N$ is nearly rank deficient then $C$ will be highly contaminated with error. This makes it impossible to compute the small eigenvalues accurately. These are precisely the eigenvalues of interest in MUSIC.

**Method 2.**

1. $S_1 = A_S^H A_S ; S_2 \leftarrow A_N^H A_N$ /*Form cross-products:
2. $S_2 = U_2 D U_2^H$, $D = \text{diag}(\lambda_i)$ /* Hermitian Schur decomposition
3. $Y \leftarrow UD^{-1/2}$ ; $C \leftarrow Y^H S_1 Y$ /* $Y^H S_2 Y = I$
4. $Q^H C Q = \text{diag}(\alpha_1, ..., \alpha_n)$ /* Compute $C$'s eigensystem, $Q^H Q = I$
5. $X = Y Q$ ; /* $D_N = I_n$

This is similar to Method 1 except that ill-conditioning in $A_N$ is immediately identified because the eigensystem of $S_2$ is obtained. $C$ will again be very poorly determined if $S_2$ is nearly rank deficient.

**Method 3.**

1. $U_2^H A_N V_2 = \text{diag}(\sigma_k)$ /* SVD
2. $Y \leftarrow V_2 \text{ diag}(1/\sigma_1, ..., 1/\sigma_n)$
3. $A_S \leftarrow A_S Y$ /* $Y^H S_2 Y = I$
4. $U_1^H A_S V_1 = D$ /* SVD
5. $X \leftarrow Y V_1$ /* $D_S = D^H D ; D_N = I_n$

The only difference between this and Methods 1 and 2 is that the cross products $A_S^H A_S$ and $A_N^H A_N$ are avoided. It is well-known that a significant loss of information can occur in numerical cross products. That is why the method of normal equations is often unsatisfactory when solving least square problems. That is why the singular values of a matrix
A are not obtained by computing the eigenvalues of $A^H A$. See Golub and Van Loan (1983, p. 143, 289).

The point in presenting these three methods is to see how vulnerable the MUSIC computations are to rank deficiency. Unfortunately, near rank deficiency in $A_N$ is not unusual. It is sometimes possible to circumvent this by interchanging the roles of $A_S$ and $A_N$ in the above and hoping that the columns of $A_S$ are strongly independent. Instead of pursuing this line we present a method that doesn't care so much about near rank deficiency in the data.

A Completely Unitary Approach to MUSIC

The following approach to problem (M') is proposed in Speiser and Van Loan (1984).

Method 4.

1. Compute the QR factorization

$$
\begin{bmatrix}
A_S \\
A_N
\end{bmatrix} = \begin{bmatrix}
Q_1 \\
Q_2
\end{bmatrix} R
$$

where $Q_1$ and $Q_2$ have the same size as $A_S$ and $A_N$ respectively and $R \in C^{nxn}$ is upper triangular. Assume that $R$ is nonsingular, i.e., that $\text{Null}(A_S) \cap \text{Null}(A_N) = \{ 0 \}$.

2. Compute the CS decomposition

$$
\begin{bmatrix}
Q_1 \\
Q_2
\end{bmatrix} = \begin{bmatrix}
U_1 & 0 \\
0 & U_2
\end{bmatrix} \begin{bmatrix}
C \\
S
\end{bmatrix} V
$$

where $U_1$, $U_2$, and $V$ are unitary, $C = \text{diag}(\cos(\theta_k))$, and
\[ S = \text{diag}(\sin(\theta_k)) \text{ with } 0 \leq \theta_1 \leq \ldots \leq \theta_n \leq \pi/2. \] It follows that if \( X = R^{-1}V \) then \( X^H(\tilde{A}_S^H\tilde{A}_S - \mu^2 \tilde{A}_N^H\tilde{A}_N)X = \xi^HC - \lambda \tilde{S}^H\tilde{S} \) and so the generalized singular values are specified by \( \mu_k = \cot(\theta_k) \).

3. Define \( \hat{d} \) by \( c_{\hat{d}} > \epsilon + c_n \geq c_{\hat{d}+1} \geq \ldots \geq c_n \geq 0 \) where \( \epsilon > 0 \) is a small positive tolerance. Here, \( c_k = \cos(\theta_k) \).

4. Compute the QR factorization of the product \( ZT = R^HV \) where \( Z = [z_1 \ldots, z_n] \) is unitary and \( T \in \mathbb{C}^{n \times n} \) is upper triangular. Since \( X = R^{-1}V = (V^HR)^{-1} = ((R^HV)^H)^{-1} = ((ZT)^H)^{-1} = ZT^{-H} \) and \( T^{-H} \) is lower triangular, it follows that \( \text{span}\{z_{\hat{d}+1} \ldots, z_n\} = \text{span}\{x_{\hat{d}+1} \ldots, x_n\} \).

Note that both \( \hat{d} \) and the basis \( \{z_{\hat{d}+1} \ldots, z_n\} \) are found without any inversions or cross-products, the computations that undermine the reliability of Methods 1-3. Moreover, Method 4 is not prone to the sensitivity of the eigenvectors \( \{x_{\hat{d}+1} \ldots, x_n\} \). It is possible for an eigenspace to be well-conditioned even though the eigenvectors that define it are not. This is a common theme in many applications that require a basis for an eigenspace. Orthonormal bases are usually preferable to eigenvector bases from the numerical point of view. This is why the Schur decomposition is preferable to the Jordan decomposition when doing invariant subspace computation.

We conclude with a result about what Method 4 actually computes in light of the tolerance \( \epsilon \) that is used to define \( \hat{d} \).

**Theorem 4.**

The vectors \( \{z_{\hat{d}+1} \ldots, z_n\} \) produced by Method 4 exactly span the minimum singular value subspace for a problem \( \tilde{A}_S^H\tilde{A}_S - \lambda \tilde{A}_N^H\tilde{A}_N \) where

\[
\begin{bmatrix}
\tilde{A}_S \\
\tilde{A}_N
\end{bmatrix} -
\begin{bmatrix}
A_S \\
A_N
\end{bmatrix}
\leq \epsilon \| R \|_2
\]
Proof.

Define \( \tilde{C} = \text{diag}(\cos(\tilde{\theta}_k)) \) and \( S = \text{diag}(\sin(\tilde{\theta}_k)) \) where \( \tilde{\theta}_k = \theta_k \) if \( k \leq \hat{d} \) and \( \tilde{\theta}_k = \theta_n \) is \( k > \hat{d} \). Set \( \tilde{\mathbf{A}}_S = U_1 \tilde{C} V_H R \) and \( \tilde{\mathbf{A}}_N = U_2 \tilde{S} V_H R \). Since

\[
\tilde{\mathbf{A}}_S - \mathbf{A}_S = U_1 (\tilde{C} - C) V_H R
\]

and

\[
\tilde{\mathbf{A}}_N - \mathbf{A}_N = U_2 (\tilde{S} - S) V_H R
\]

it follows that

\[
\left\| \begin{bmatrix} \tilde{\mathbf{A}}_S \\ \tilde{\mathbf{A}}_N \end{bmatrix} - \begin{bmatrix} \mathbf{A}_S \\ \mathbf{A}_N \end{bmatrix} \right\|_2 \leq \epsilon \left\| R \right\|_2 \Box
\]

Thus, if MUSIC is implemented using Method 4 it solves a "nearby problem" exactly.

The ESPRIT Problem

ESPRIT avoids the nonlinear minimization in MUSIC. It does this by comparing output \( x \) and \( y \) from a pair of sensor arrays (\( X \) and \( Y \)), one a translate of the other. Details may be found in Paulraj, Roy, and Kailath (1986) and Roy, Paulraj, and Kailath (1986). A comparison of the MUSIC and ESPRIT procedures can be found in Roy, Paulraj, and Kailath (1987).

In ESPRIT the output of the two arrays is modelled as follows:

\[
x(t) = \mathbf{A} s(t) + n_x(t)
\]

\[
y(t) = \mathbf{A} \Phi s(t) + n_y(t)
\]
where (it can be shown), \( A \in \mathbb{C}^{n \times d} \), \( n_x \) and \( n_y \) are the noise vectors, \( s(t) \in \mathbb{C}^d \) is the vector of source signals, and \( \Phi \) is a diagonal unitary matrix whose diagonal entries are easy functions of the DOAs. Under certain assumptions we have

\[
S_{XX} = E(xx^H) = ASA^H + \sigma^2 I \\
S_{YY} = E(yy^H) = A\Phi S\Phi^H A^H + \sigma^2 I \\
S_{XY} = E(xy^H) = A\Phi^H A^H
\]

In ESPRIT, the computation of \( \Phi \) is based upon the following result.

**Theorem 5.**

Suppose \( A \in \mathbb{C}^{n \times d} \) has rank \( d \), \( S \in \mathbb{C}^{d \times d} \) is Hermitian positive definite, and \( \Phi \in \mathbb{C}^{d \times d} \). If \( S_{XX} = ASA^H + \sigma^2 I \) and \( S_{YY} = A\Phi S\Phi^H A^H + \sigma^2 I \) then \( \sigma^2 \) is their smallest eigenvalue and in either case it has multiplicity \( n-d \). Likewise if \( V = [v_1, ..., v_n] \) is unitary with

\[
S_{XX} v_k = \sigma^2 v_k \quad k = d+1:n
\]

then \( A^H V = [B^H \ 0] \) where \( B \in \mathbb{C}^{d \times d} \) and thus,

\[
V^H S_{XX} V = \begin{bmatrix}
BSB^H + \sigma^2 I_d & 0 \\
0 & \sigma^2 I_{n-d}
\end{bmatrix}
\]

and

\[
V^H S_{YY} V = \begin{bmatrix}
B\Phi S\Phi^H B^H + \sigma^2 I_d & 0 \\
0 & \sigma^2 I_{n-d}
\end{bmatrix}
\]
If \( S_{XY} = AS\phi^HA^H \) then

\[
\nu^H S_{XY} \nu = \begin{bmatrix}
BS\phi^HB^H & 0 \\
0 & 0
\end{bmatrix}
\]

There are \( d \) complex numbers \( \lambda_1, \ldots, \lambda_d \) for which

\[
\text{rank}( (S_{XX} - \sigma^2 I) - \lambda S_{XY} ) = d-1
\]

and these are precisely the \( \lambda \) that make

\[
BSB^H - \lambda BS\phi^HB^H = BS(1 - \lambda \phi^H)B^H
\]

singular.

**Proof.**

If \( S_{XX} \nu = \lambda \nu \) with \( \| \nu \|_2 = 1 \), then

\[
\lambda = \nu^H (ASA^H + \sigma^2 I) \nu = \sigma^2 + (A^H \nu)^H S(A^H \nu)
\]

It follows that \( \sigma^2 \) is the smallest eigenvalue of \( S_{XX} \) and that it has multiplicity \( n-d = \dim(\text{Null}(A^H)) \). If \( S_{XX} \nu = \sigma^2 \nu \) then \( \nu \in \text{Null}(A^H) \). Thus, if \( \nu = [v_1, \ldots, v_n] \in \mathbb{C}^{n \times n} \) is unitary with \( \text{span}(v_{d+1}, \ldots, v_n) = \text{Null}(A^H) \), then \( A^H \nu \) has the form specified in the hypothesis. The rest of the proof is straightforward. \( \square \)

In ESPRIT \( \phi \) is diagonal and so we may take \( \phi = \text{diag}(\lambda_1, \ldots, \lambda_d) \) where \( \lambda = \lambda_k \) forces

\[
\text{rank}( (S_{XX} - \sigma^2 I) - \lambda S_{XY} ) = d-1
\]

If the matrices \( S_{XX}, S_{YY}, \text{and } S_{XY} \) are known exactly then there are several ways to compute \( \phi \).
**Method E1.**

1. Compute the Hermitian Schur decomposition of $S_{XX} : V^H S_{XX} V = D$.

   Since $\lambda = \sigma^2$ is an eigenvalue of multiplicity $n-d$ we may assume

   $$V^H S_{XX} V = \begin{bmatrix} \text{diag}(\sigma_k^2) & 0 \\ 0 & \sigma^2 \end{bmatrix} \begin{bmatrix} d \\ n-d \end{bmatrix}$$

   If $V = [v_1, ..., v_n]$ it follows that $\text{span}(v_{d+1}, ..., v_n) = \text{Null}(A^H)$.

2. Compute $W = V^H S_{XY} V$ which must have the form

   $$V^H S_{XX} V = \begin{bmatrix} W_{11} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} d \\ n-d \end{bmatrix}$$

3. Compute $\lambda_1, ..., \lambda_d$ such that $\det(\text{diag}(\sigma_k^2 - \sigma^2) - \lambda_k W_{11}) = 0$ for $k=1:d$ and set $\Phi = \text{diag}(\lambda_1, ..., \lambda_d)$.

**Method E2.**

1. Compute the Schur decomposition of $S_{XY} : Q^H S_{XY} Q = T$. Since this matrix has a nullspace of dimension $n-d$ we can choose $Q$ such that

   $$Q^H S_{XY} Q = \begin{bmatrix} T_{11} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} d \\ n-d \end{bmatrix}$$
Note that if $Q = [ q_1, \ldots, q_n ]$ then span \{ $q_{d+1}$, \ldots, $q_n$ \} = Null ($A^H$).

2. Compute $Q^H S_{XX} Q = W$ which must have the form

$$Q^H S_{XX} Q = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma^2 \end{bmatrix}$$

3. Compute $\lambda_1, \ldots, \lambda_d$ such that $\det (W_{11} - \sigma^2 I) - \lambda_k T_{11} = 0$, $k = 1:d$.
Set $\phi = \text{diag}(\lambda_1, \ldots, \lambda_d)$.

**Method E3.**

Essentially the same as Method E1 but with $S_{YY}$ replacing $S_{XX}$.

Thus, the approach taken in all these methods is to factor out the common null space Null($A^H$) and solve the remaining $d$-by-$d$ generalized eigenproblem.

In practice, a great deal of care must be exercised because we will only have approximations to $S_{XX}$, $S_{XY}$ and $S_{YY}$. Indeed, if we collect $m$ snapshots \{ $x(t_k)$, $y(t_k)$ \} \[k = 1:m\], of $X$ array and $Y$ array output and form the matrices

$$A_X = (1/\sqrt{m}) [ x(t_1), \ldots, x(t_m) ]^T \in \mathbb{C}^{m \times n}$$

$$A_Y = (1/\sqrt{m}) [ y(t_1), \ldots, y(t_m) ]^T \in \mathbb{C}^{m \times n}$$

then $A_X^H A_X \approx S_{XX}$, $A_Y^H A_Y \approx S_{YY}$, and $A_X^H A_Y \approx S_{XY}$. The trouble now is that we must now factor an approximate null space out of the approximate problem.
\[(A_X^H A_X - \sigma^2 I) - \lambda A_X^H A_Y\]

Generalized eigenvalue problems of the form \(A - \lambda B\) where \(A\) and \(B\) have nearly intersecting nullspaces (the ESPRIT situation) are notoriously ill-conditioned and great care must be exercised.

**A Singular Value Approach**

This corresponds to Method E1. Note that if

\[U_X^H A_X V_X = \Sigma_X = \text{diag}(\sigma_{k,X})\]

is the SVD of \(A_X\) and

\[\sigma_{d,X} > \epsilon + \sigma_{n,X} \geq \sigma_{d+1,X} \geq \cdots \geq \sigma_{n,X} = \sigma_{\text{min},X}\]

where \(\epsilon\) is a small tolerance, then the corresponding columns of \(V_X\) are an approximate basis for \(\text{Null}(A^H)\):

\[\text{span}\{v_{d+1,X}, \ldots, v_{n,X}\} \approx \text{Null}(A^H)\]

This is because \(A_X^H A_X \approx S_{XX} = A S A^H + \sigma^2 I\). Likewise, if

\[U_Y^H A_Y V_Y = \Sigma_Y = \text{diag}(\sigma_{k,Y})\]

is the SVD of \(A_Y\) and

\[\sigma_{d,Y} > \epsilon + \sigma_{n,Y} \geq \sigma_{d+1,Y} \geq \cdots \geq \sigma_{n,Y}\]

then

\[\text{span}\{v_{d+1,Y}, \ldots, v_{n,Y}\} \approx \text{Null}(A^H)\].
Unfortunately, these two singular vector approximations to Null(A^H) may differ, even in their choice of \( \hat{d} \). A way around this difficulty that assigns "equal weight" to the X and Y data is to compute the intersections of the subspaces

\[
\text{span}\{ v_{d+1,X}, \ldots, v_{n,X} \} \cap \text{span}\{ v_{d+1,Y}, \ldots, v_{n,Y} \}
\]

for various values of \( \hat{d} \) and settle on that value that gives the largest possible dimension. Subspace intersections can be computed using the SVD, see Golub and Van Loan (1983, p.430.). Moreover, if we have

\[
v_X^H v_Y = V = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{bmatrix} \hat{d} \\ n-\hat{d} \end{bmatrix}
\]

and \( \| V_{12} \|_F = \delta \), then \( \text{span}\{ v_{d+1,X}, \ldots, v_{n,X} \} \) and \( \text{span}\{ v_{d+1,Y}, \ldots, v_{n,Y} \} \) are "within \( \delta \)" of being the same subspace. This follows from CS decomposition theory. Here, \( \| \cdot \|_F \) denotes the sum-of-squares norm. Note how easy it is to compute \( \| V_{12} \|_F \) as a function of \( d \). This could be used in an intelligent way to compute the critical basis for Null(A^H).

Once we have unitary \( V = [ v_1, \ldots, v_n ] \) with \( \text{span}\{ v_{d+1}, \ldots, v_n \} \approx \text{Null}(A^H) \) and an estimate \( \hat{\sigma}^2 \) of \( \sigma^2 \), say \( \hat{\sigma}^2 = \sigma_{n,X}^2 \), then our problem transforms as follows

\[
V^H(A_X^H U_X U_X^H A_X - \hat{\sigma}^2 I) - \lambda A_X^H U_X U_X^H A_Y \} V =
\begin{bmatrix} B^H - \sigma^2 I & E_{12} \\ E_{21} & E_{22} \end{bmatrix} - \lambda \begin{bmatrix} B^H C & F_{12} \\ F_{21} & F_{22} \end{bmatrix}
\]
Here, B and C are the upper \( \tilde{\alpha} \)-by-\( \tilde{\alpha} \) portions of \( U_X^H A_X V \) and \( U_Y^H A_Y V \) respectively and the \( E_{ij} \) and \( F_{ij} \) are small in norm. How small depends upon the quality of the approximations \( \text{span}\{v_{d+1}, \ldots, v_n\} \approx \text{Null}(A^H) \), \( A_X^H A_X \approx S_{XX} \), \( S_{YY} \approx A_Y^H A_Y \) and \( S_{XY} \approx A_X^H A_Y \). There remains the problem of solving the \( d \)-by-\( d \) eigenproblem

\[
\det[(B^H B - \sigma^2 I) - \lambda B^H C] = \det[(\tilde{B} + \sigma I)(\tilde{B} - \sigma I) - \lambda B^H C] = 0
\]

where \( B \) is a Hermitian matrix that satisfies \( B^2 = B^H B \). This can be found from the SVD \( B = U_B \Sigma_B V_B^H \). Just set \( B = V_B \Sigma_B V_B^H \).

To avoid cross products and inverses the above determinantal equation could be solved using the technique in Van Loan (1975) that computes the generalized Schur decomposition described earlier. In general, the computed \( \lambda \) will not be on the unit circle as they should be in theory. Thus, it is plausible to set \( \Phi = \text{diag}(\text{arg}(\lambda_k)) \).

Although this procedure relies on unitary matrices throughout, we have been unable to show that this implementation of ESPRIT computes the exact DOAs of a "nearby" problem.

**A Generalized Schur Approach**

We next outline a unitary matrix approach to the ESPRIT problem that corresponds to Method E2.

1. Compute unitary \( Q \) and \( V \) such that

\[
V^H A_X Q = T
\]
\[
Q^H A_Y V = S
\]

are upper triangular. Note that \( V^H A_X^H A_Y V = TS = R \in \mathbb{C}^{n \times n} \) is the Schur decomposition of \( A_X^H A_Y \) and so \( V \) can be chosen to order the eigenvalues from largest to smallest in absolute value.
It is possible to do this without forming $A_H^H A_Y$ using the algorithm in Van Loan (1975). Determine $\hat{d}$ so that

$$|r_{dd}| = |s_{d+1,d+1}^t \hat{d} \hat{d}^t| > |r_{nn}| + \epsilon = |s_{nn}^t \hat{d} \hat{d}^t| + \epsilon \geq |s_{d+1,d+1}^t \hat{d} \hat{d}^t|$$

where $\epsilon$ is a small tolerance. Note that if $V = [v_1, \ldots, v_n]$ then $\text{span}(v_{d+1}^\top, \ldots, v_n^\top) \approx \text{Null}(A^H)$.

2. If $U^H(A^H V)$ is upper triangular, then it has the form

$$U^H(A^H V) \approx \begin{bmatrix} W_{11} & 0 \\ 0 & \sigma^2 I \end{bmatrix}$$

3. Compute the generalized eigenvalues of the problem

$$(\tilde{W}_{11} + \sigma I)(\tilde{W}_{11} - \sigma I) - \lambda \cdot \text{S}(1:d,1:d) \cdot \text{T}(1:d,1:d)$$

using the algorithm in Van Loan (1975) and set $\Phi = \text{diag}(\arg(\lambda_k))$.

Here $W_{11}$ is a Hermitian matrix that satisfies $\tilde{W}_{11}^2 = W_{11}^H W_{11}$.

As with the singular value approach, we are not able to show that this implementation of ESPRIT solves a nearby problem. Thus, the stability properties of ESPRIT are unclear to us although good unitary methods exist for the computations.

**Conclusions**

We have shown how the difficult subspace dimension estimation problems in MUSIC and ESPRIT can be handled. In the case of MUSIC, we are able to show that the computed DOAs are exact for "nearby data". This shows that the method is stable.

ESPRIT is conceptually much simpler, but involves a trickier eigenvalue computations. More research is necessary to examine how the ESPRIT DOAs are effected by the choice of $d$ and the computed basis for
Null($A^H$). In the mean time, simulations suggest that ESPRIT is pretty reliable prompting us to conjecture that there is a favorable perturbation analysis of the method that awaits discovery.
References


