Numerically robust transfer function modelling from noisy frequency domain data
A. Bultheel, M. Van Barel, Y. Rolain, and R. Pintelon

Abstract—Using vector orthogonal polynomials as basis functions for the representation of the rational form of a linear time-invariant system, in frequency domain identification problems, it is shown that the notorious numerical ill conditioning of these maximum likelihood problems can be overcome completely. For the identification of high order (100/100) systems operating over a wide frequency band, or even in the situation of over- or undermodelling, condition numbers less than 10 are reported for real measurements.

Index Terms—Discrete rational approximation, frequency domain identification, vector orthogonal polynomials, maximum likelihood

I. INTRODUCTION

Frequency domain identification of linear time-invariant systems has regained some interest during the last years ([1], [2]). The main driving force behind this revival is the use of periodic excitation signals for identification. Periodic excitation signals not only allow to calculate reliable and very cheap estimations of the non-parametric frequency response function from the division of the measured spectra, they do also yield a reliable measure of the stochastic uncertainty on these spectra at the cost of a low number (> 4) of repeated synchronized measurements [2], [3]. The knowledge of the frequency response function allows a visual interpretation of both the system and the estimated model behavior and the match between them. Selection of a simplified model based on application dependent criteria is possible through comparison of the simplified model with the measured non-parametric frequency response function. The knowledge of the variance of the measured spectral lines bounds the stochastic frequency response function variation. Model simplification is statistically sound whenever the simpler model remains in the frequency response function uncertainty bound with a certain probability. Over-simplification of the model can be detected by comparison of the model cost function with its known expected value. Combination of all these indicators leads to fully automatic model order selection [4], [5].

On the dark side, frequency domain methods are known to suffer from a poor numerical conditioning when the frequency span and/or the model order become large (more than 2 decades and an order more than 20). This numerical problem ruins both the modelling performance and the model order selection capability.

Several attempts have been made in the past to circumvent numerical degeneracy. These range from an appropriate scaling of the frequencies [2], to the decomposition of the numerator and denominator of the model equation in a separate basis of polynomials orthogonal with respect to the inner product defined by the normal equations of the estimator [2], [6]. Even if these approaches increase the limit on bandwidth and complexity, they do not remove numerical problems totally, neither do they extrapolate gracefully to multiple input multiple output (MIMO) systems. The approach proposed here solves the numerical conditioning issue perfectly (condition number = 1) for all the frequency domain methods whose cost function can be reduced to a linear weighted least squares problem or a weighted (generalized) total least squares problem. As such, it opens the way for automatic model estimation procedures in the frequency domain for both SISO and MIMO linear systems.

II. THE MODELLING PROBLEM

We consider a system whose exact input and output spectra are \( U_0(\Omega) \in \mathbb{C}^{n_u \times 1} \) and \( Y_0(\Omega) \in \mathbb{C}^{n_y \times 1} \), where \( \Omega \) stands for the time lag \( \omega^{-1} \) (discrete time) or the Laplace variable \( s \) (continuous time). We collect these in a vector \( S_0 = [Y_0^* - U_0^*]^T \in \mathbb{C}^{n_u \times 1} \), \( n = n_y + n_u \), (the star stands for complex conjugate transpose). The system model is \( G(\Omega, \theta) = A^{-1}(\Omega, \theta)B(\Omega, \theta) \), where \( P = [A \ B] \in \mathbb{P}^{n_y \times (n_y+n_u)} \) is a polynomial matrix containing the polynomials \( A \) and \( B \). Note that we use the left matrix fraction description which is suited for input-output measurements as well as for measurements of the transfer function. For a right matrix fraction description, the analysis is completely analogous. Our model is parameterized by the vector \( \theta \) which is to be determined. The spectra are however measured in discrete points so that the data available are not \( S_0(\Omega) \), but \( S(\Omega_k), k = 1, \ldots, F \) where \( S \) is corrupted by additive noise: \( S(\Omega) = S_0(\Omega) + N_S(\Omega) \). The covariance matrix of the noise is \( \Sigma(\Omega) = \text{cov}(N_S(\Omega)) \in \mathbb{C}^{n_y \times n_y} \) is supposed to be known. Note that the linearized error is given by \( E = PS = AY - BU \in \mathbb{C}^{n_y \times 1} \).

Our estimator will be obtained as the solution of a least squares problem which minimizes a cost function of the form

\[
V = \sum_{k=1}^{F} E_k^* W_k E_k = E^* W E, \tag{1}
\]

with \( E_k = E(\Omega_k, \theta) = P(\Omega_k, \theta)S(\Omega_k) \), \( P \neq 0 \). \( W_k \in \mathbb{C}^{n_y \times n_y} \) positive definite weights, while \( E = [E_1^* \cdots E_F^*]^T \in \mathbb{C}^{n_y \times F} \)
and $\mathbb{C}^{n_x \times 1}$ and $W = \text{diag}(W_1, \ldots, W_F)$. In the classical least squares problem, the weights $W_k$ are all $I_{n_y}$ but this results in a very noise sensitive solution. In the (total) least squares cost function they can be of a much more general form. They can depend on some initial estimate $\theta$ of the unknown parameters $\theta$. In general we form some $w_k \in \mathbb{C}^{n_x \times n_x}$ (possibly depending on $\theta$) and define

$$M(\theta) = \sum_{k=1}^{F} w_k C_{E_k}^2 w_k^*,$$

with $C_{E_k}^2 = \text{cov}(E_k)$, thus

$$C_{E_k}^2 = A_k C_{Y_k}^2 A_k^* + B_k C_{U_k}^2 B_k^* - 2\text{Herm}(A_k C_{Y_k}^2 U_k^* B_k^*),$$

where $A_k = A(\Omega_k, \theta)$, $B_k = B(\Omega_k, \theta)$, and $C_Z^2 = \text{cov}(X, Z)$ and $C_X^2 = C_X^2$. The weights for this generalized least squares problem are then $W_{k}^{\text{GLS}} = w_k^* M^{-1}(\theta) w_k$, which may depend on $\theta$. In the case of weighted generalized total least squares, the weights are $W_{k}^{\text{WTLS}} = w_k^* w_k / \text{trace}(M(\theta))$, which may also depend on $\theta$, see [7].

In general $W_k$ can explicitly depend on the solution $\theta$. For example in the maximum likelihood (ML) estimator for the nonlinear problem, the weight $W_k$ is given by

$$W_{k}^{\text{ML}} = W(\Omega_k, \theta) = [C_{E_k}^2]^{-1}.$$

In this case, it is a nonlinear problem that can be solved iteratively. However, the cost function is not convex and may have many local minima. Therefore no global minimum is guaranteed. A very good starting value for $\theta$ is essential for a successful identification of the model.

The following procedure can be followed. We start by a simple least squares problem (using weights $W_k = I_{n_y}$) or a generalized total least squares problem, using some estimate $\theta$ for the unknown parameters. This gives us a first estimate for $\theta$. This estimate is used to compute a better weight and with this weight, a new estimate is obtained etc., following a Sanathanan-Koerner iteration [8]. This iterative process will then eventually converge to a close approximation of the ML estimates [1]. Finally a true nonlinear iteration can be done to minimize the ML cost function.

### III. PARAMETERIZATION OF THE PROBLEM

Until now we have not been precise about what the parameters $\theta$ were. They should somehow be used in the representation of the polynomial matrix $P \in \mathbb{P}^{n_y \times n_y}$. To this end we shall stack all the $n_x n_y$ scalar polynomials in a long vector using the vec-operator. So, defining vec($M$) as the vector which stacks the columns of the matrix $M$ on top of each other. We then have

$$E = \text{vec}(E) = \text{vec}(PS) = \text{vec}(I_{n_y} PS)$$

$$= (S^T \otimes I_{n_y}) \text{vec}(P).$$

Here we use the Kronecker product. Given the matrices $F$ and $G$, then $F \otimes G$ is the block matrix whose $(i,j)$th entry is $F_{ij}G$. Setting $S_k = S_k^T \otimes I_{n_y}$ and $P_k = \text{vec}(P(\Omega_k))$, we can write the cost function (1) as

$$V(\theta) = \sum_{k} P_k^*(\theta) S_k^* W_k S_k P_k(\theta).$$

Note that the matrix $M_k = S_k^T W_k S_k$ is positive semi-definite.

From now on we have to parameterize the vector $P \in \mathbb{P}^{n_y n_x \times 1}$. Suppose that the maximal degree that appears in $P$, thus the maximal degree that appears in the entries of $A$ and $B$ is $n$ and suppose that $\{\phi_{d, j} \in \mathbb{P}^{n_y n_x} : d = 0, \ldots, n; j = 1, \ldots, n_y n_t\}$ forms a basis for the space of all possible $P$ where the first index refers to the degree and the second index to the $n_y n_t$ independent polynomials for each degree. Thus we can write $P(\theta) = \sum_{d=0}^{n} \sum_{j=1}^{n_y n_t} \theta_{d,j} \phi_{d,j}$. Let us group this in $n + 1$ blocks of size $n_y n_t$ as follows $\phi_d = [\phi_{d,1}, \ldots, \phi_{d, n_y n_t}] \in \mathbb{P}^{n_y n_t \times n_y n_t}$ and correspondingly $\theta_d = [\theta_{d,1}, \ldots, \theta_{d, n_y n_t}]^T \in \mathbb{P}^{n_y n_t \times 1}$. So we can rewrite $P = \sum_{d=0}^{n} \phi_d \theta_d = \varphi \theta$, with $\varphi = [\phi_0, \ldots, \phi_n] \in \mathbb{P}^{n_y n_t (n+1) \times n_y n_t}$. To illustrate this idea, suppose we consider the SISO case, where $n_y = n_x = 1$ and thus $n_t = 2$. Let $n_a = \partial(A)$, $n_b = \partial(B)$ hence $n = \max(n_a, n_b)$. The transfer function can then be written as

$$G(\Omega, \theta) = B(\Omega, \theta) A(\Omega, \theta) = \sum_{d=0}^{n} b_d \theta_d \Omega^d$$

Note that there are $n_a + n_b + 1$ free parameters. Now, setting

$$P = \begin{bmatrix} A \end{bmatrix} = \phi_0 \theta_0 + \cdots + \phi_n \theta_n = \sum_{d=0}^{n} \phi_d \theta_d.$$  

Denoting the first element of $\phi_d, j$ as $A_{d,j}$ and the second element of $\phi_d, j$ as $B_{d,j}$, then we have

$$G(\Omega, \theta) = B(\Omega, \theta) A(\Omega, \theta) = \sum_{d=0}^{n} \sum_{j=1}^{n_y n_t} B_{d,j}(\Omega) \theta_{d,j}$$

which seems to have $2n + 2$ parameters. However, suppose for example that $G$ is strictly proper, so that $n_y < n_x$. The $\theta_0, \ldots, \theta_n$ are all in $\mathbb{C}^{2 \times 1}$ and form $2n + 2$ parameters. Since the degree of $B$ is restricted to $n_y$, the second element in $\phi_d, j$ for $d = n_y + 1, \ldots, n_x$ has to be zero, which imposed $n_x - n_y$ linear conditions on the remaining parameters. Together with a normalization condition, this leaves us with $n_a - n_y - 1$ additional free parameters, giving a total of $2n + 2 + (n_x - n_y - 1) = n_x + n_t + 1$ degrees of freedom, just as in the previous representation.

Thus

$$V(\theta) = \theta^* \left[ \sum_{k} \varphi^* (\Omega_k) M_k \varphi(\Omega_k) \right] \theta = \theta^* \Phi^* M \Phi \theta$$

with $M = \text{diag}(M_1, \ldots, M_F)$ and $\Phi = [\varphi(\Omega_1)^*^*, \ldots, \varphi(\Omega_F)^*^*]$ has to be minimized under some degree constraint for $P$, which can be translated into certain $\theta_{d,j}$ being nonzero (e.g. chosen as 1). Without this constraint, the problem would have the trivial solution $\theta = 0$. Since the matrix in between brackets is positive semi-definite, we can write it as $L^* L$ so that we have to find the least squares solution of $L \theta = 0$ under the given constraints on $\theta$. 


These columns of \( L \) that correspond to \( \theta_{d,j} \) that are chosen to be 1, can be brought to the right-hand side, giving an inhomogeneous least squares problem of the form \( J\theta = r \), with a nontrivial solution.

Until now, we have assumed that the parameters \( \theta \) are complex. However, in most practical situations, \( \theta \) is real. In that case (2) should be replaced by

\[
V(\theta) = \theta^T \text{Re}(\Phi^* M \Phi) \theta
\]

and \( L \theta = 0 \) is to be replaced by \( L_{\text{re}} \theta = 0 \), while \( J \theta = r \) becomes

\[
J_{\text{re}} \theta = r_{\text{re}}
\]

where for any array \( M \), the \((\cdot)_{\text{re}} \) operator is defined as

\[
M_{\text{re}} = [\text{Re}(M)^T \ \text{Im}(M)^T]^T.
\]

Note that \( \text{Re}(L^* L) = L_{\text{re}}^T L_{\text{re}} \).

From now on we shall discuss the case of real \( \theta \). For a complex vector \( \theta \), just leave out the \((\cdot)_{\text{re}} \) and the \( \text{Re}(\cdot) \) from the notation.

IV. VECTOR ORTHOGONAL POLYNOMIALS: CONTROLLING THE CONDITION NUMBER

It is a well known fact that the Jacobian \( J_{\text{re}} \) of the least squares problem (3), (4) can be extremely ill conditioned for any but academic problems. For example, taking the standard basis with respect to the discrete inner product. One such basis was selected [9], [10]. These are polynomials orthogonal with respect to a discrete inner product. One such basis was imposed. Indeed, as a result of the orthogonalization, we have that all the computations will be real. The reordering of the columns is dictated by the degree structure that is imposed. Indeed, as a result of the orthogonalization, we have to minimize the cost function \( V(\theta) = \theta^T \theta \). Thus we should choose all \( \theta_{d,j} \) equal to zero except those that are taken to be nonzero by the degree conditions.

Suppose that the polynomial matrix \( A \) has a leading coefficient that is upper triangular and that \( A^{-1} B \) is proper. Then we assume a degree structure for the matrix polynomial \( P \) that is at most \( n - 1 \) for the \((i,j)\) entry with \( i > j \) and that is at most \( n \) for all the other ones. We place these in the matrix \( \partial P \).

Obviously, \( \text{vec}(\partial P) \) will then be the degree structure of the vector polynomial \( P \). For normalization we should take one of the entries of degree \( n \) to be monic. For example the one corresponding to the right bottom element of \( A \). That is element \( n_2^2 \) of the vector \( P \). In [14] it is described in detail how to compute a solution \( P \) with a general degree structure. By the above normalization, the algorithm will guarantee the upper triangular leading coefficient. Note that \( P \) and \( TP \) for any regular matrix \( T \) will give the same solution. Thus if the highest degree coefficient of \( A \) does not have a zero last column then, there is an equivalent solution that has a 1 in the right bottom corner. Note that even though the least squares solution depends on \( T \) when there are noise and or model errors, the maximum likelihood cost function does not depend on \( T \), so that also the ML solution does not depend on \( T \).

In an iterative refinement setting, the weight \( W_k \) will depend on the last value of \( \theta \) that was obtained. This results in a new \( \theta \)-value which is again plugged into the weights, etc. The application is straightforward, but it should be noted that in each step, the system that is solved has an optimal condition number so that there is almost no degradation of the numerical computation because of rounding errors.

To solve the true ML problem, we shall have to iterate to solve the nonlinear problem since the weight will depend on \( \theta \). If we have a certain value for \( \theta \) (the one corresponding to the current orthogonal basis), then we have to compute a correction \( \Delta \theta \). This correction is found by solving a linear least squares problem of the type considered. The Jacobian matrix of this system in each iteration step can again be optimally conditioned. However, this requires an orthogonal basis with respect to the weight (evaluated at the current \( \theta \)).
and the solution will be a correction \( \Delta \theta \), but this correction will be represented with respect to the new orthogonal basis for the current weights, while \( \theta \) was with respect to previous weights. Unfortunately there is no stable way to transform one basis into the next. Therefore, we fix the polynomial basis in the subsequent iteration steps (i.e., we fix the weights \( W_k \)). This implies that the condition number shall not be 1, but that it can grow during the iterations. However practical experience has shown that when the initial guess is good enough, the condition number will not grow above 10.

The classical approach is to start with a unity weight and then to solve a series of weighted linear estimation problems whose weighting depends on the parameter values of the previous iteration. This iterative process will then eventually converge to a close approximation of the ML estimates [1]. However, if the problem is too complex, the estimation obtained in the first step of the process (unity weight) may be inadequate to start the process successfully. Thereto, a parameter independent approximation of the ML weighting has been proposed for the SISO case [10]. All these linear estimation methods are combined with the proposed method to obtain the numerically stable methods used in the experimental part of the paper.

We summarize the advantages of the method developed in this paper.

1) It is recursive and fast and numerically stable.
2) In the SISO case, it gives not only a description of all the best least squares approximants of degree \( n \) but also all the solutions of lower degree.
3) One can impose an arbitrary degree structure for MIMO systems.
4) The algorithm can be efficiently implemented on a parallel computer.

V. SOME FURTHER COMPUTATIONAL ASPECTS

Obviously, the problem simplifies considerably if \( n_u = 1 \), i.e., in the MISO (hence a fortiori in the SISO) case because the Kronecker product is then avoided.

The determination of the degree of the approximant is part of the general problem. For the moment assume that it is given. Since we propose a recursive procedure, we shall compute not only the best approximant of degree \( n \) but also all the approximants of lower degree, so that a right decision on the degree of the approximant can be made.

Our normalization condition required a certain entry in the polynomial matrix \( P \) to be monic. It is not clear which entry to normalize.

It is perfectly possible that one has a large number of data points from which one can select a relatively small number of relevant points. For example, some point is really an “outlier” and one wants to get rid of that point after it has been introduced. In such a case the problem is a problem of downdating. A combination of a downdating and an updating step would then replace one point by another. This simulates a sliding window: new data are taken into account and old data are forgotten.

The principle for a downdating step is clear. In the polynomial case it is described for real data in [15], and for data on the unit circle in [16]. These algorithms are based on a direct algorithm. Indeed, a direct algorithm will start from a Hessenberg matrix and diagonalize it by orthogonal similarity transformations. So, some QR type algorithm is needed to diagonalize the Hessenberg matrix in one column, for example the last one. This isolates one point on the diagonal, which has to be \( z_F \). The unitary similarity transformation that achieved this has to be applied to the extended Hessenberg matrix and in the first block column, the corresponding weight will appear. Now the data \((w_F, z_F)\) can be deflated and the point has been removed.

VI. EXPERIMENTAL RESULTS

To show that the proposed method leads to practically applicable results, it has been decided to apply it on real measured data of a real world system. The systems are selected in such a way that most common types of practical non-ideal behavior are present in the examples. The first example is a complex mechanical vibrating structure with a very high number of vibration modes. It shows the ability of the method to cope with wideband, very high order systems.

The second example, the radial response of a CD head, illustrates the robustness of the method against lousy model order selection. This property can then be exploited to obtain an automatic model order selection procedure.

A. A very high order measured vibrating mechanical structure in the Laplace domain

A composed steel beam structure (fig.1) is excited by a mini-shaker. The force and the acceleration are measured by an impedance head mounted on the shaker using a stinger rod. Signals are acquired by an FFT-based dynamic signal analyzer (HP3562A). The excitation signal is a periodic chirp between 0 and 5000 Hz with a frequency resolution of 6.25 Hz. Fig.2 shows the magnitude of the frequency response function obtained by division of the output and input spectra. These spectra will then be used for the estimation.

The rational transfer function is modelled between 493 Hz and 5 kHz at \( F = 722 \) equidistant frequency lines. 32 consecutive synchronised measurements allow to obtain the mean value of the measured spectral lines and their variance as a function of vibration modes. It shows the ability of the method to cope with wideband, very high order systems.
Fig. 2. Magnitude of the measured (dashed line) and estimated (full line) transfer function of the vibrating structure.

Fig. 3. Phase of the measured (dashed line) and estimated (full line) transfer function of the vibrating structure.

Fig. 4. Magnitude of the measured (dashed line) and estimated (full line) transfer function and magnitude of the complex error between measurement and model (dotted line).

The estimated model can predict the linear dynamics of the system with a high accuracy. The model combines pole/zero pairs with high quality factors, that result in sharp resonances, with low quality factor roots to describe slower variations in the transfer function. The latter can not be determined at all if the frequency characteristic is sliced into smaller bands as proposed by generic modal analysis approaches.

The value of the ML cost function is 642 while the expected value [2] is $F = (n + d + 1)/2 = 601.5$. The estimated cost falls within the uncertainty bound of the theoretically predicted one (50% for 32 measurements). This means that the residue between model and measurement can be explained by the noise variation only.

The (linear) condition number of the Jacobian matrix degrades from 1 to 1.4 during the ML estimation. This is due to the fact that the orthogonal basis is not updated during the ML iteration. This is only a marginal degradation, and shows that the BTLS estimation, that was obtained in the last iteration of this estimator, matches the ML solution very closely. Calculation of the poles/zeros of the obtained model show a stable, minimum-in-phase behavior.

Based on the cost function and the excellent conditioning, one can conclude that the model captures the dynamics of the system adequately. The ML residual shown in fig.5, contains a residual correlation as a function of the frequency. Maximal residue magnitudes are mainly attained at the sharp resonances of the transfer function. This points to the presence of small non-linearities at these frequencies. Practical experience learns that this often happens for modal analysis measurements with sharp resonances.

B. Modeling a CD-radial positioner in discrete time

The device under test is a Philips CD320/00G CD drive modified to get access to the control loop of the radial servo system. All measurements were done in closed loop operation at the start of track 1. The maximal experiment time was 26.2s.

The closed loop system is excited by a HPE1445A VXI arbitrary waveform generator operating on the ZOH-mode without reconstruction filter. The sampling frequency was set to $f_s = 1 \times 10^7/2^{10} = 9765.6$Hz.
The system response is measured by HPE1430A VXI ADC cards with anti-aliasing protection on and sampling frequency equal to $f_s$. High impedance buffers with a gain of about 0.5 were used to isolate the plant from the 50Ω input impedance of the acquisition front end.

The excitation signal is a special odd multisine \cite{2} with $F$ frequencies at $kf_0$, $k = 1, 3, 9, 11, 17, 19, \ldots, 611$. The crest factor of the signal was compressed to about 1.55 starting from random initialized phases. The RMS value of the applied excitation signals was about 1 Volt.

The system is modelled in the Laplace domain. Several models are extracted. The degree of the transfer function model is varied from 2/2 to 6/6 in increments of 1 order and from 8/8 to 20/20 with increments of 2.

To obtain the final ML estimates, an initial GTLS estimation and a series of 10 BTLS estimations are performed as in the previous example. Results of the series of estimations are summarised in fig.6. The value of the expected cost function ranges from 219 for the simplest model to 200 for the most complex one. Clearly, the cost function obtained is too large, even for the most complex model. An additional test on the whiteness of residual shows that for model orders larger than 8/8, there are no linear unmodelled dynamics left, and that the increased value of the cost is caused by nonlinear distortions. Note that, even under severe overmodelling and undermodelling, the (linear) condition number of the ML estimator remains very close to the optimal value of 1 that is obtained as a result for the BTLS initial step. The numerical properties of the estimator are hereby shown to be robust to both overmodelling and undermodelling, and this opens the door to sensible and easy to interpret model order selection procedures. To show that the modelling as obtained for the different orders is indeed sensible, three estimated models are shown. The first model of order 3/3 is shown in fig.7. It results in a severe undermodelling of the measured characteristics, especially at low frequencies where the magnitude of the complex model error is much larger than the value of the measurement. A sensible model is obtained at the order 5/5 (fig.8). The low frequency dynamics are now nicely captured, the model error is about 20dB under the model value over the whole band. When the model order is further increased to 18/18, (fig.9), the overall level of the error decreases further, especially at the center of the band where the remaining bumps in the characteristic are now captured. However, the model contains 6 unstable poles, that are almost perfectly cancelled by very close zeros. The closed loop system resulting from this open loop model is still perfectly stable. Based on this
example, one can conclude that the proposed method is robust to over- and undermodelling problems. As such, the tool may be handy to the community of identification users, that have no extensive identification knowledge but require sensible models for use in their own expertise fields. Even if the initial model order is poorly selected, the method will come up with a sensible model that comes close to the behaviour of the system to be modelled.

C. A MIMO example

The measurement set up is as follows. An aluminium plate (185mm × 63mm × 1.5mm) hung by three nylon threads is excited by two mini-shakers via plexi-glass stinger rods (see Figure 10). The forces \((u_1, u_2)\) and accelerations \((y_1, y_2)\) at the excitations points are measured with impedance heads (B&amp;K 8001). These signals are amplified (charge amplifier B&amp;K 2635) and buffered \((Z_{in} > 5M\Omega, \ Z_{out} = 50\Omega)\) before being applied to the acquisition channels (HPE 1430A, \(Z_{in} = 50\Omega\)) of the VXI measurement device. The periodic excitation signals \((r_1, r_2)\) in Figure 11) are generated by two arbitrary function generators (HPE 1445A, \(Z_{out} = 50\Omega\)) at a sampling frequency \(f_s = 10\text{MHz}/2^{12} \approx 2.44\text{kHz}\). The output of the arbitrary function generators is lowpass filtered (7th order inverse Chebyshev filter with a cut off frequency of 1 kHz) before being applied to the mini-shaker. To reduce the effect of the inductive impedance of the shaker on the generator unit, an 18 Ω//5 W resistance is put in series with its input.

The measurement results: For each MIMO measurement two random phase multisines (= sum of harmonically related sinewaves with deterministic amplitude spectrum and random phase spectrum) with flat amplitude spectrum, interleaved frequency grids, and \(N = 4096\) points per period are used as excitation signals. Expressed in harmonic numbers w.r.t. the frequency resolution \(f_0 = f_s/N \approx 0.6\text{Hz}\) the interleaved frequency grids are for grid 1: 119:2:617 and for grid 2: 120:2:688, which corresponds to \(F = 500\) in a frequency band of about [71 Hz, 368 Hz].

To cope with the nonlinear distortions, 25 MIMO experiments with different realisations of the random phase multisines are performed (see [2]). Each MIMO experiment consists of \(n_u = 2\) MIMO measurements. Hence, four different random phase multisines \(r_{1j}\) (frequency grid 1), \(r_{21}\) (frequency grid 2), \(r_{12}\) (frequency grid 2), and \(r_{22}\) (frequency grid 1) are calculated for each MIMO experiment. In the \(j\)th measurement \((j = 1,2)\) of each experiment signals \(r_{1j}\) and \(r_{2j}\) are applied to shakers 1 and 2 respectively. Referring the observed input/output DFT spectra to the exactly known reference signals \(r_1, r_2\), (see Figure 11),

\[
U_R^{(n)}(k) = \tilde{U}^{(n)}(k)(R^{(n)}(k))^{-1}
\]

\[
y_R^{(n)}(k) = \tilde{Y}^{(n)}(k)(R^{(n)}(k))^{-1}
\]

Where \(\tilde{U}^{(n)}(k), \tilde{Y}^{(n)}(k)\) and \(R^{(n)}(k)\) are the 2 by 2 input, output en reference DFT spectra of the \(n\)th MIMO experiment, it is possible to calculate the sample means and sample covariance matrices of the measurements over the twenty five MIMO experiments (see [2], [3])

\[
X(k) = \begin{bmatrix} \tilde{X}_{11}(k) & \tilde{X}_{12}(k) \\ \tilde{X}_{21}(k) & \tilde{X}_{22}(k) \end{bmatrix}, \quad \tilde{X}_{ij}(k) = \frac{1}{N} \sum_{n=1}^{N} \tilde{X}_{ij}^{(n)}(k)
\]

\[
N\tilde{C}_{zz}(k) = \frac{1}{N-1} \sum_{n=1}^{N} \Delta z^{(n)}(k) \Delta z^{(n)*},
\]

\[
\Delta z^{(n)} = \tilde{z}^{(n)}(k) - \tilde{x}(k), \quad x = \text{vec}(Z)
\]
For ease of notation, the frequency arguments are dropped. The noisy input/output DFT spectra $U, Y$ resulting form $n_a$ independent MIMO experiments are related to the true $U_0, Y_0$ by

$$Y = Y_0 + N_Y, \quad U = U_0 + N_U.$$  

Using the first order Taylor series expansion

$$(U_0 + N_U)^{-1} \approx U_0^{-1} - U_0^{-1}N_UU_0^{-1}.$$  

we get

$$G = YU^{-1} = (Y_0 + N_Y)(U_0 + N_U)^{-1} = G_0 + N_YU_0^{-1} - G_0N_UU_0^{-1}.$$  

Using

$$\text{vec}(G_0N_0U_0^{-1}) = (U_0^{-1} \otimes G_0)\text{vec}(N_Y)$$  

and

$$\text{vec}(N_YU_0^{-1}) = \text{vec}(I_{n_a}N_YU_0^{-1}) = (U_0^{-1} \otimes I_{n_a})\text{vec}(N_Y)$$  

(see [17]), it can easily be verified that

$$G_{\text{vec}G} = (U_0^T \otimes I_{n_a})\text{vec}(Y_0U_0^{-1} \otimes I_{n_a}) + (U_0^T \otimes G_0)\text{vec}(Y_0U_0^{-1} \otimes G_0^*) - 2\text{Herm}((U_0^T \otimes I_{n_a})\text{vec}(Y_0U_0^{-1} \otimes G_0^*)).$$

Replacing the true values $U_0$ and $G_0$ by the measured values, and the exact noise covariance matrices by the sample noise covariance matrices finally gives (5).

**REFERENCES**


