NTIA Report 02-395

Estimation of System Gain and Bias Using Noisy Observations with Known Noise Power Ratio

Stephen D. Voran

U.S. DEPARTMENT OF COMMERCE
Donald L. Evans, Secretary

Nancy J. Victory, Assistant Secretary
for Communications and Information

September 2002

ACKNOWLEDGEMENT

The author gratefully acknowledges the contributions of Stephen Wolf to this work. He applied several of the algorithms presented in this report to objective and subjective video quality data and provided valuable feedback on the algorithms.

CONTENTS

FIGURES

TABLES

ACRONYMS

CJM	constrained joint minimization
DE	direct estimation
LSMS	least squares manual splitting
SNR	signal-to-noise ratio
SVD	singular value decomposition
TLS	total least squares
WJM	weighted joint minimization
XLS	extended least squares

Estimation of System Gain and Bias Using Noisy Observations with Known Noise Power Ratio

Stephen Voran*

The identification of linear systems from input and output observations is an important and well-studied topic. When both the input and output observations are noisy, the resulting problem is sometimes called the "errors in variables" problem. Existing work on this problem deals with the identification of multivariate systems and thus results in algorithms that are necessarily somewhat complex and often involve iteration. In this report we treat an important special case of the problem: estimation of a system bias and a system gain from noisy observations of system input and output. In addition, we invoke an input-output noise power ratio constraint. This constraint can also be interpreted as a parameter that moves the problem in a continuous fashion between two limiting cases, each of which is a conventional least-squares problem. We do not model the input signal, and we place minimal restrictions on the input and output observation noises. We develop five different low-complexity closed-form solutions to the problem. The final two are the most satisfying and we explore these further through simulations. Our original motivation for working on this problem came from the need to calibrate objective and subjective estimates of perceived video or speech quality. We expect that our solutions may also find applications in remote sensing, active noise reduction, echo cancellation, channel estimation, and channel equalization.

Key words: gain estimation, bias estimation, linear system identification, input noise, errors in variables, total least squares, extended least squares, audio quality estimation, speech quality estimation, video quality estimation

1. INTRODUCTION

ITS conducts research on the objective estimation of perceived video and speech quality. As part of this work it is often necessary to scale and shift subjective viewing or listening test results in order to compensate for the use of different laboratories, different populations of subjects, or different languages. The appropriate scaling (gain) and shifting (bias) factors are jointly estimated from subjective test results and objective estimates of perceived quality. Both of these can be considered to be noisy observations of a true underlying perceived quality value. This estimation problem is closely related to the problem of identifying linear systems from noisy input and output observations.

* The author is with the Institute for Telecommunication Sciences, National Telecommunications and Information Administration, U.S. Department of Commerce, 325 Broadway, Boulder, Colorado 80305.

The identification of linear systems from input and output observations is an important and well-studied topic. Much of this work has been based on noisy observations of system output and noise-free observations of system input. This is also described as the "errors in measurements" situation. While this approach is clearly useful in numerous situations, there are also plenty of situations where system identification must proceed from noisy output observations and noisy input observations. This could be described as the "errors in variables and measurements" situation, but it typically is identified by the shorter "errors in variables" moniker. A history of efforts at system identification using noisy output and input observations is provided in [1] which in turn points to [2] for coverage of earlier work.

The classical solution to this problem is given by total least-squares (TLS) [3],[4] and numerous other innovative approaches have been offered as well. A minimum description length approach is based on the argument that the best approximation to the underlying linear system is the one that allows the observations to be encoded with the fewest bits [5]. This requires a joint probabilistic model for the observations and the noises. Examples of cumulant-based solutions can be found in [6] and [1]. In general, cumulant-based solutions require multiple steps, or may even be iterative. A frequency-domain approach utilizing bispectra or trispectra (subject to certain restrictions on the observation noises) is given in [7].

The solution in [8] stems from minimization of a residual norm (or output observation noise) subject to an absolute constraint on a coefficient perturbation matrix (or input observation noise). This solution requires a singular value decomposition (SVD) of the data matrix. In [9] this work is extended to include a family of absolute constraints where each applies a different portion of the coefficient matrix perturbation. An extended least squares (XLS) formulation of the problem is described in [10]. This formulation seeks to minimize a weighted combination of the model mismatch (or input observation noise) and measurement inaccuracies (or output observation noise) and two iterative algorithms result from this formulation. In certain specific cases, the XLS approach is equivalent to optimization of a joint maximum *a posteriori* – maximum likelihood criterion [11]. The problem formulation of XLS is connected to our weighted joint minimization problem formulation as described later in this report.

In general, the existing work allows for identification of multivariate systems and thus results in algorithms that are necessarily somewhat complex, and often involve iteration. In this report, we treat a relatively simple but important special case of the problem that yields low complexity closed-form solutions. This version of the problem is also unique in that a single piece of side information is used: the input-output noise power ratio r^2. We simply wish to estimate a scalar gain (or equivalently a scalar attenuation) and bias using noisy observations of system input and output. We make no assumptions about the signal, we do not build a model for the signal, and we do not perform any spectral analysis.

The situation is described in Figure 1 and by

$$\boldsymbol{y} + \xi_y = a_T\left(\boldsymbol{x} + \xi_x\right) + b_T\mathbf{1}, \tag{1}$$

where $\boldsymbol{x} = \left[x_1,\ x_2,\ x_3,\ \ldots,\ x_n\right]^{\mathrm{T}}$ and $\boldsymbol{y} = \left[y_1,\ y_2,\ y_3,\ \ldots,\ y_n\right]^{\mathrm{T}}$ are the noisy observation vectors (corresponding to the input $\boldsymbol{w_T}$ and the output $\boldsymbol{z_T}$ respectively), a_T and b_T are the true scalar system gain and bias to be estimated, $\boldsymbol{\xi_x}$ and $\boldsymbol{\xi_y}$ are length n column vectors of observation noise, and **1** is the length n column vector of ones. The additive noises in Figure 1 are defined to be $-\boldsymbol{\xi_x}$ and $-\boldsymbol{\xi_y}$ rather than the more conventional $\boldsymbol{\xi_x}$ and $\boldsymbol{\xi_y}$. The only practical consequence of this definition choice is more readable mathematical derivations throughout this report.

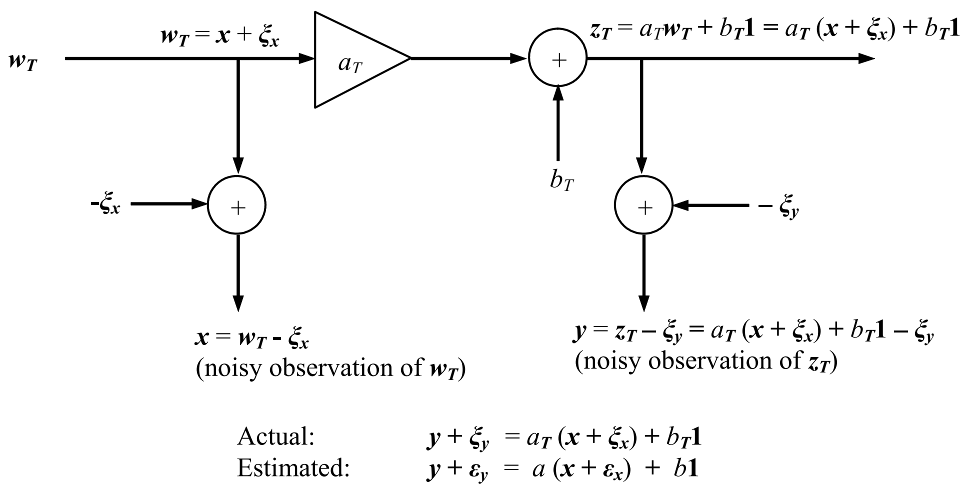

Figure 1. Block diagram for noisy observations of a system with gain and bias.

The absolute noise power levels are unknown, but they do conform to a cost-weighted noise power ratio constraint:

$$0 < \frac{\mathrm{E}\left|\boldsymbol{C}\xi_x\right|^2}{\mathrm{E}\left|\boldsymbol{C}\xi_y\right|^2} = r^2, \tag{2}$$

where $\boldsymbol{C}$ is an $n \times n$ diagonal cost matrix populated with $\left\{c_i\right\}_{i=1}^{n}$ and

$$0 < c_i,\ \mathrm{i} = 1 \text{ to } n \text{ and } \sum_{i=1}^{n} c_i^2 = 1. \tag{3}$$

$\boldsymbol{C}$ can be used to appropriately weight the noisy observations if such weighting is indicated by the details of a particular problem under consideration. Otherwise $\boldsymbol{C}$ can be set to $\frac{1}{\sqrt{n}}\boldsymbol{I}_{n\times n}$ where $\boldsymbol{I}_{n\times n}$ is the $n \times n$ identity matrix. Given a pair of noisy observation vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, a cost matrix $\boldsymbol{C}$, and the noise power ratio r^2, we seek to find $a \neq 0$, b, $\boldsymbol{\varepsilon_x}$, and $\boldsymbol{\varepsilon_y}$ such that

$$\boldsymbol{y} + \boldsymbol{\varepsilon}_y = a\left(\boldsymbol{x} + \boldsymbol{\varepsilon}_x\right) + b\mathbf{1} \tag{4}$$

and

$$\frac{\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2}{\left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2} = r^2. \tag{5}$$

We will use a, b, $\boldsymbol{w}$, $\boldsymbol{\varepsilon_x}$, and $\boldsymbol{\varepsilon_y}$ as estimates of a_T, b_T, $\boldsymbol{w_T}$, $\boldsymbol{\xi_x}$, and $\boldsymbol{\xi_y}$ respectively. While a and b are the estimates sought, in some cases the estimation algorithms also generate $\boldsymbol{w}$, $\boldsymbol{\varepsilon_x}$, and $\boldsymbol{\varepsilon_y}$ as well.

In Figure 1, the system input is $\boldsymbol{w_T}$ and we have access only to noisy observations $\boldsymbol{x}$ and $\boldsymbol{y}$. If we take $\boldsymbol{z_T}$ to be the system output, then we have modeled a noise-free system that only scales and shifts the input $\boldsymbol{w_T}$. If we take $\boldsymbol{y}$ to be the system output, then we have modeled a noisy system that scales, shifts, and adds noise to the input $\boldsymbol{w_T}$. Figure 2 shows a mathematically equivalent interpretation of Figure 1. Here we consider $\boldsymbol{x}$ to be an observed noise-free input that is contaminated by the noise $\boldsymbol{\xi_x}$ before it enters the system that scales and shifts it.

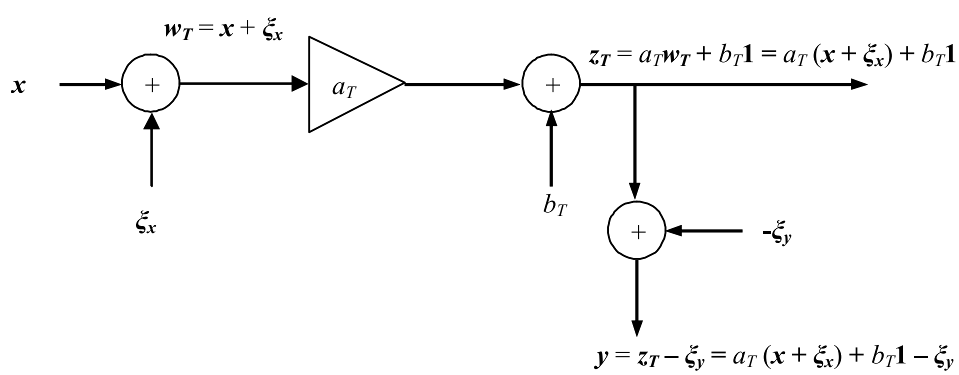

Figure 2. Alternative interpretation of Figure 1.

Note that the input-output noise power ratio r^2 can be viewed as the parameter that lets us move in a continuous fashion between two conventional least squares problems. As r goes to zero, the input noise vanishes (we have the errors in measurements case), and the problem that remains looks like a conventional least-squares problem. As r gets large, in relative terms the output noise vanishes (we have the errors in variables case) and the problem that remains looks like a closely related least-squares problem. In the case of the constrained joint minimization algorithm described later in this report (see Section 5), we find that the solutions are similarly parameterized by r, and they reduce to least-squares solutions in the limiting cases $r = 0$ and $r \rightarrow \infty$.

Note also that at this point we have not made any assumptions about the observation noises ξ_x and ξ_y. They could even be correlated to the signal (as in echoes or distortion products) but we will see later that the best estimates result from the case of uncorrelated observation noises.

The problem we have described is a realistic one when environmental conditions or the physics of a situation yield prior knowledge of the ratio of the two observation noises, but not the absolute level of the observation noises. This might apply to multiple observations of signals by imperfect instruments or transducers (to perform interferometry or to form synthetic apertures), or multiple replications of an experiment at different times or places (as in meta-analysis [12]). Thus we expect that solutions to this problem may find applications in remote sensing, active noise reduction, echo cancellation, channel estimation, and channel equalization.

In these applications, system physics often dictate that the bias is zero. Our original motivation for working on this problem comes from the objective estimation of perceived video or speech quality (see for example [13]-[16]) and in these applications the bias term is generally not zero. We maintain that subjective viewing or listening tests provide noisy observations ($\boldsymbol{x}$ and $\boldsymbol{y}$) of an underlying true value of perceived quality ($\boldsymbol{w}$). Similarly, objective quality estimates also give noisy observations of that true value. Subjective tests are not absolute; results may require scaling (gain) and shifting (bias) to compensate for the use of different laboratories, different populations of subjects, or different languages. But objective estimates are absolute. To advance the state of the art in objective estimation of perceived video and speech quality, it is necessary to relate objective estimator results to a number of disparate subjective test results, using a different gain and bias factor for each subjective test. These gain and bias factors must be estimated from noisy subjective ($\boldsymbol{x}$) and objective ($\boldsymbol{y}$) observations of the underlying true perceived video or speech quality. The absolute noise levels are not known. However, given our experience with objective estimators, we do have some confidence that the noise in the subjective test results is smaller than the noise in the objective estimates. In other words, $r \leq 1$. This report provides tools for estimating the gain and bias factors that relate the results of a single subjective test with the corresponding objective estimates of perceived quality.

In Sections 2-6, we develop five algorithms for estimating gain and bias in the situation described in Figure 1. We have adopted the following names for these algorithms: “least squares plus manual splitting” (LSMS), “total least squares” (TLS), “weighted joint minimization” (WJM), “constrained joint minimization” (CJM), and “direct estimation” (DE). The first four algorithms require no assumptions about the noises beyond their known noise power ratio given in (2). These algorithms are derived by minimizing four different noise quantities. The DE algorithm places additional restrictions on the noises, and does not involve the minimization of a noise. Rather, it follows directly from the algebraic manipulations of the noisy observations.

We have generated simulation results for numerous situations including tone, chirp, noise, and speech signals and using both correlated and uncorrelated observation noises. These results and some discussion are presented in Section 7. The appendix summarizes the five algorithms through the use of several tables.

2. LEAST SQUARES PLUS MANUAL SPLITTING (LSMS)

The derivation of the five algorithms is facilitated by the following definitions:

$$m_x = \sum_{i=1}^{n} c_i^2 x_i \quad , \quad m_y = \sum_{i=1}^{n} c_i^2 y_i \quad , \tag{6}$$

$$\hat{\boldsymbol{x}} = \boldsymbol{C}\left(\boldsymbol{x} - m_x \mathbf{1}\right) \quad , \quad \hat{\boldsymbol{y}} = \boldsymbol{C}\left(\boldsymbol{y} - m_y \mathbf{1}\right) \; , \tag{7}$$

$$\rho = \frac{\hat{\boldsymbol{x}}^{\mathrm{T}} \hat{\boldsymbol{y}}}{|\hat{\boldsymbol{x}}||\hat{\boldsymbol{y}}|} \quad , \quad \operatorname{sign}(\rho) = \begin{cases} 1 & \text{when} \quad 0 < \rho, \\ -1 & \text{otherwise,} \end{cases} \tag{8}$$

and

$$\boldsymbol{w} = \left[w_1,\ w_2,\ w_3,\ \ldots,\ w_n\right]^{\mathrm{T}} . \tag{9}$$

Note that m_x and m_y are cost-weighted means, $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ are cost-weighted, shifted versions of $\boldsymbol{x}$ and $\boldsymbol{y}$ such that $\boldsymbol{C}\hat{\boldsymbol{x}}$ and $\boldsymbol{C}\hat{\boldsymbol{y}}$ are zero mean. Note also that ρ is the normalized cross-correlation of $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$. We assume that $0 < |\hat{\boldsymbol{x}}|$ and $0 < |\hat{\boldsymbol{y}}|$. If multiple pairs of observations are available, then quantities like $|\hat{\boldsymbol{x}}|, |\hat{\boldsymbol{y}}|, \hat{\boldsymbol{x}}^T \hat{\boldsymbol{y}}, m_x$, and m_y are estimated by averaging over all available observations.

The first algorithm, LSMS, comes from a conventional least-squares step followed by the manual splitting of the least-squares residual into two vectors. We can rewrite (4) as

$$a\boldsymbol{x} + b\mathbf{1} = \boldsymbol{y} + \left(\boldsymbol{\varepsilon}_y - a\boldsymbol{\varepsilon}_x\right) = \boldsymbol{y} + \boldsymbol{\varepsilon}_{LSMS} . \tag{10}$$

Conventional weighted least squares allows us to minimize $\varepsilon_{LSMS}^2 = \left|\boldsymbol{C}\boldsymbol{\varepsilon}_{LSMS}\right|^2$ with respect to a and b. The results are:

$$a_{LSMS} = \frac{|\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}}|}\rho \quad , \quad b_{LSMS} = m_y - a_{LSMS} m_x \quad , \tag{11}$$

$$\boldsymbol{C}\boldsymbol{\varepsilon}_{LSMS} = a_{LSMS}\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}} = \frac{|\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}}|}\rho\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}}, \quad \left|\boldsymbol{C}\boldsymbol{\varepsilon}_{LSMS}\right|^2 = |\hat{\boldsymbol{y}}|^2\left(1 - \rho^2\right). \tag{12}$$

Next we manually decompose $\boldsymbol{\varepsilon}_{LSMS}$ into a pair of perfectly correlated noise estimates $\boldsymbol{\varepsilon}_x$ and $\boldsymbol{\varepsilon}_y$ that attain the desired noise power ratio r^2 given in (5). Let

$$g = \frac{|a_{LSMS}|\, r}{|a_{LSMS}|\, r + 1} \tag{13}$$

and note that $0 < g < 1$. We use g to decompose $\boldsymbol{\varepsilon}_{LSMS}$ into two portions:

$$\boldsymbol{\varepsilon}_{LSMS} = (1-g)\,\boldsymbol{\varepsilon}_{LSMS} + g\boldsymbol{\varepsilon}_{LSMS}. \tag{14}$$

From (10) we see that

$$\boldsymbol{\varepsilon}_{LSMS} = \boldsymbol{\varepsilon}_y - a_{LSMS}\boldsymbol{\varepsilon}_x. \tag{15}$$

Equating the right hand sides of (14) and (15) leads us to define

$$\boldsymbol{\varepsilon}_y = (1-g)\,\boldsymbol{\varepsilon}_{LSMS} \quad \text{and} \quad \boldsymbol{\varepsilon}_x = -\frac{g}{a_{LSMS}}\boldsymbol{\varepsilon}_{LSMS}. \tag{16}$$

It follows that

$$\boldsymbol{C\varepsilon}_x = \frac{-r\,\mathrm{sign}(\rho)}{1+r|a_{LSMS}|}\left(a_{LSMS}\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}}\right),\ \ \boldsymbol{C\varepsilon}_y = \frac{1}{\left(1+r|a_{LSMS}|\right)}\ \left(a_{LSMS}\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}}\right), \text{ and } \ \frac{|\boldsymbol{C\varepsilon}_x|^2}{|\boldsymbol{C\varepsilon}_y|^2} = r^2. \tag{17}$$

The total cost-weighted noise power is

$$|\boldsymbol{C\varepsilon}_x|^2 + |\boldsymbol{C\varepsilon}_y|^2 = \frac{r^2+1}{\left(1+|a_{LSMS}|\,r\right)^2}|\hat{\boldsymbol{y}}|^2\left(1-\rho^2\right). \tag{18}$$

When the noise power ratio is unity ($r=1$), and both cost-weighted shifted observations have unit power ($|\hat{\boldsymbol{x}}|^2 = |\hat{\boldsymbol{y}}|^2 = 1$), (18) reduces to

$$|\boldsymbol{C\varepsilon}_x|^2 + |\boldsymbol{C\varepsilon}_y|^2 = 2\,\frac{1-|\rho|}{1+|\rho|}. \tag{19}$$

We can show that $\hat{\boldsymbol{x}}^T\left[\boldsymbol{C}\boldsymbol{\varepsilon}_{LSMS}\right]=0,$ so the weighed error vector is orthogonal to the modified noisy observation vector $\hat{\boldsymbol{x}}$. The two weighted noise estimates $\boldsymbol{C}\boldsymbol{\varepsilon}_x$ and $\boldsymbol{C}\boldsymbol{\varepsilon}_y$ are perfectly correlated since they are formed by scaling a single noise vector. Figure 3 shows the geometry of the LSMS result in a simple two-dimensional case where $m_x = m_y = 0$, $r \approx 2$, and $a \approx 0.75$. The figure shows the various additive relationships among the pertinent quantities in the LSMS algorithm, as well as scaling and orthogonality. The LSMS solution for this simple case can be described geometrically as follows. First $\boldsymbol{C}\boldsymbol{\varepsilon}_{LSMS}$ and a are found by projecting $\hat{\boldsymbol{y}}$ onto $\hat{\boldsymbol{x}}$, then $a\boldsymbol{C}\boldsymbol{w}$ is found by dividing $\boldsymbol{C}\boldsymbol{\varepsilon}_{LSMS}$ into $-a\boldsymbol{C}\boldsymbol{\varepsilon}_x$ and $\boldsymbol{C}\boldsymbol{\varepsilon}_y$ according to r.

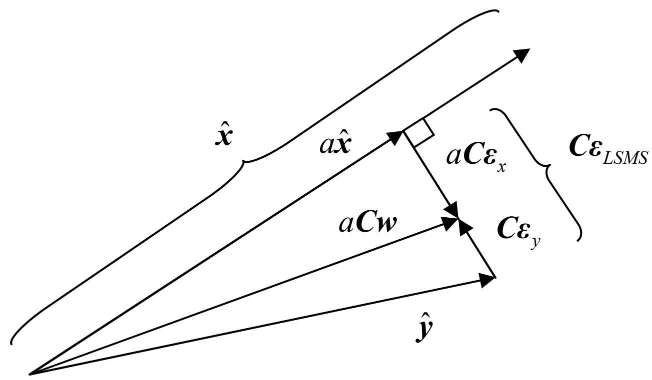

Figure 3. Example geometry of LSMS result.

The LSMS algorithm satisfies the noise power ratio constraint given in (5). However, the estimates of a and b generated by the LSMS algorithm depend only on $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$. While $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ do implicitly provide the LSMS algorithm with information about r, the LSMS algorithm does not make explicit use of the r value that is available to the algorithm. This is inconsistent with our desire to use that side-information advantageously. An additional disadvantage is that the quantity minimized by the LSMS algorithm (15) is somewhat unnatural. Finally, the LSMS algorithm generates perfectly correlated noise estimates, but in real situations input and output noise sources are unlikely to be perfectly correlated.

3. TOTAL LEAST SQUARES (TLS)

The shortcomings of the LSMS algorithm motivate us to minimize a more meaningful quantity — the total cost-weighted noise power:

$$\varepsilon_{TLS}^2 = \left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 + \left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2. \tag{20}$$

The problem is now to minimize ε_{TLS}^2 with respect to $\boldsymbol{w}$, a, and b and this results in a TLS algorithm.

A necessary condition for minimizing ε_{TLS}^2 is

$$\frac{\partial}{\partial w_i}\varepsilon_{TLS}^2 = 0,\ \ i = 1 \text{ to } n \tag{21}$$

and this results in

$$\boldsymbol{w} = \frac{1}{1+a^2}\left[\boldsymbol{x} + a\left(\boldsymbol{y} - b\mathbf{1}\right)\right]. \tag{22}$$

An additional necessary condition for minimizing ε_{TLS}^2 is

$$\frac{\partial}{\partial b}\varepsilon_{TLS}^2 = 0 \tag{23}$$

and this results in

$$b = m_y - a\sum_{i=1}^{n} c_i^2 w_i. \tag{24}$$

Using (22) in (24) and solving for *b* yields

$$b_{TLS} = m_y - a_{TLS} m_x. \tag{25}$$

We can now use results (22) and (25) to rewrite the estimated noise terms and the total cost-weighted noise power:

$$\boldsymbol{\varepsilon}_x = \frac{-a}{1+a^2}\left[a\left(\boldsymbol{x} - m_x\mathbf{1}\right) - \left(\boldsymbol{y} - m_y\mathbf{1}\right)\right], \tag{26}$$

$$\boldsymbol{\varepsilon}_y = \frac{1}{1+a^2}\left[a\left(\boldsymbol{x} - m_x\mathbf{1}\right) - \left(\boldsymbol{y} - m_y\mathbf{1}\right)\right], \tag{27}$$

$$\varepsilon_{TLS}^2 = \left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 + \left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2 = \frac{1}{1+a^2}\left|a\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}}\right|^2. \tag{28}$$

Now (28) allows us to minimize ε_{TLS}^2 with respect to the single variable *a*. The necessary condition

$$\frac{\partial}{\partial a}\varepsilon_{TLS}^2 = 0 \tag{29}$$

yields

$$a_{TLS} = \begin{cases} \dfrac{\left(\dfrac{|\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}}|} - \dfrac{|\hat{\boldsymbol{x}}|}{|\hat{\boldsymbol{y}}|}\right) \pm \sqrt{\left(\dfrac{|\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}}|} - \dfrac{|\hat{\boldsymbol{x}}|}{|\hat{\boldsymbol{y}}|}\right)^2 + 4\rho^2}}{2\rho}, & \rho \neq 0, \\ 0, & \rho = 0. \end{cases} \tag{30}$$

To show that (30) is also a sufficient condition we must show that

$$0 < \frac{\partial^2}{\partial a^2} \varepsilon_{TLS}^2 \bigg|_{a_{TLS}}. \tag{31}$$

Condition (31) is satisfied if we pick the correct sign in (30). Thus we arrive at the solution

$$a_{TLS} = \begin{cases} \dfrac{\left(\dfrac{|\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}}|} - \dfrac{|\hat{\boldsymbol{x}}|}{|\hat{\boldsymbol{y}}|}\right) + \sqrt{\left(\dfrac{|\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}}|} - \dfrac{|\hat{\boldsymbol{x}}|}{|\hat{\boldsymbol{y}}|}\right)^2 + 4\rho^2}}{2\rho}, & \rho \neq 0, \\ 0, & \rho = 0. \end{cases} \tag{32}$$

Thus ε_{TLS}^2 is minimized by (24) and (32), resulting in a total cost-weighted noise of

$$\varepsilon_{TLS}^2 = \frac{1}{1 + a_{TLS}^2} \left| a_{TLS} \hat{\boldsymbol{x}} - \hat{\boldsymbol{y}} \right|^2. \tag{33}$$

In the simple case where both cost-weighted shifted observations have unit power ($|\hat{\boldsymbol{x}}|^2 = |\hat{\boldsymbol{y}}|^2 = 1$), (32) reduces to a_{TLS}=sign(ρ) when $\rho \neq 0$ and (33) reduces to

$$\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 + \left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2 = 1 - |\rho|. \tag{34}$$

From (26) and (27) it is clear that

$$\frac{\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2}{\left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2} = a^2. \tag{35}$$

Note that (20) can also be minimized via the SVD-based TLS algorithm as described in [3] and [4]. By invoking the SVD, this approach can deal with the more general multivariate case. When applied to the present problem, the SVD approach would require that we extract the primary left singular vector $\boldsymbol{u}$ from the n x 2 observation matrix $[\hat{\boldsymbol{x}}\ \hat{\boldsymbol{y}}]$. We would then form the projection operator $\boldsymbol{P}_u = \boldsymbol{u}\left(\boldsymbol{u}^T\boldsymbol{u}\right)^{-1}\boldsymbol{u}^T = \boldsymbol{u}\boldsymbol{u}^T$. This operator projects the noisy observations onto the best fitting rank one subspace as defined by the span of $\boldsymbol{u}$. Thus the weighted noise estimates can be found by the orthogonal projection $-\boldsymbol{C}\boldsymbol{\varepsilon}_x = \left(\boldsymbol{I}-\boldsymbol{P}_u\right)\hat{\boldsymbol{x}}$ and $-\boldsymbol{C}\boldsymbol{\varepsilon}_y = \left(\boldsymbol{I}-\boldsymbol{P}_u\right)\hat{\boldsymbol{y}}$. This approach ultimately yields the solution given above, but with significant extra complexity.

We can show that $[\boldsymbol{C}\boldsymbol{w}]^T[\boldsymbol{C}\boldsymbol{\varepsilon}_x] = [\boldsymbol{C}\boldsymbol{w}]^T[\boldsymbol{C}\boldsymbol{\varepsilon}_y] = 0$, so both weighted noise estimates are orthogonal to the weighted input vector $\boldsymbol{w}$. From (26) and (27) it is clear that the TLS algorithm generates weighted noise estimates $\boldsymbol{C}\boldsymbol{\varepsilon}_x$ and $\boldsymbol{C}\boldsymbol{\varepsilon}_y$ that are perfectly correlated. The geometry of the TLS result in a simple two-dimensional case with $m_x = m_y = 0$, $a \approx 0.75$, is shown in Figure 4. A geometric description of the TLS solution for this simple case follows. Given that $a\boldsymbol{C}\boldsymbol{\varepsilon}_x$ and $\boldsymbol{C}\boldsymbol{\varepsilon}_y$ must be orthogonal to $a\boldsymbol{C}\boldsymbol{w}$ as shown, we can swing $a\boldsymbol{C}\boldsymbol{w}$ and slide a to find the unique solution that minimizes ε^2_{TLS}.

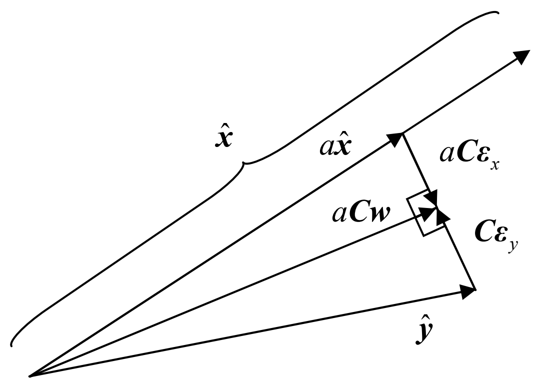

Figure 4. Example geometry of TLS result.

We have not included the noise power ratio constraint r^2 in the derivation of the TLS algorithm, and consequently that noise power ratio will not be satisfied in general. Further, the TLS algorithm generates perfectly correlated noise estimates, but in real situations input and output noise sources are unlikely to be perfectly correlated. However, the result in (35) provides motivation for the WJM algorithm described in the next section. On the positive side, the TLS algorithm does minimize a natural and meaningful quantity.

4. WEIGHTED JOINT MINIMIZATION (WJM)

The TLS results suggest that minimizing a weighted version of (20) could lead to a weighted minimum noise solution that satisfies the noise power ratio constraint r^2 given in (5). Equation (35) suggests the value of this weight. Thus for the WJM algorithm we define

$$\varepsilon_{WJM}^2 = \left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 + \frac{r}{|a|}\left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2 \tag{36}$$

and we seek to minimize ε_{WJM}^2 with respect to $\boldsymbol{w}$, a, and b. Note that (36) reproduces a special case of the XLS problem formulation in [10]. The notation of [10] defines a model matrix $\boldsymbol{A}$, a parameter vector $\boldsymbol{\theta}$, a model weight matrix $\boldsymbol{W_g}$, and a data weight matrix $\boldsymbol{W_x}$. If we define these four XLS variables to be

$$\boldsymbol{A} = \left[\boldsymbol{\text{-}y}\ \boldsymbol{w}\ \boldsymbol{1}\right], \qquad \boldsymbol{\theta} = \begin{bmatrix} a \\ b \end{bmatrix}, \qquad \boldsymbol{W_g} = \frac{r}{|a|}\boldsymbol{C}^2, \qquad \text{and} \qquad \boldsymbol{W_x} = \boldsymbol{C}^2, \tag{37}$$

and note that the noise-free input $\tilde{\boldsymbol{x}}$ in [10] corresponds exactly to our input $\boldsymbol{w}$, then the cost function $C\left(\tilde{\boldsymbol{x}},\boldsymbol{\theta}\right)$ given in equation (11) of [10] becomes our equation (36) exactly:

$$\begin{aligned} C\left(\tilde{\boldsymbol{x}},\boldsymbol{\theta}\right) &= \left[1\ \boldsymbol{\theta}^{\boldsymbol{T}}\right]\boldsymbol{A}^{\boldsymbol{T}}\boldsymbol{W_g}\boldsymbol{A}\begin{bmatrix} 1 \\ \boldsymbol{\theta} \end{bmatrix} + \left(\boldsymbol{x}-\tilde{\boldsymbol{x}}\right)^T\boldsymbol{W_x}\left(\boldsymbol{x}-\tilde{\boldsymbol{x}}\right) \\ &= \frac{r}{|a|}\left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2 + \left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 = \varepsilon_{WJM}^2 . \end{aligned} \tag{38}$$

Because the WJM problem is constrained significantly relative to the general XLS problem, our procedure for minimizing our equation (36) is very different from the procedure given for minimizing (11) in [10].

Our solution steps parallel those in the TLS algorithm, but with some additional complexity introduced by the weighting factor. We first find the necessary conditions

$$\boldsymbol{w} = \frac{1}{1+r|a|}\left[\boldsymbol{x} + r\,\text{sign}(a)\left(\boldsymbol{y} - b\boldsymbol{1}\right)\right] \tag{39}$$

and

$$b = m_y - a\sum_{i=1}^{n} c_i^2 w_i . \tag{40}$$

Using (39) in (40) and solving for b yields

$$b_{WJM} = m_y - a_{WJM} m_x. \tag{41}$$

We can now use results (39) and (41) to rewrite the noise estimates and ε_{WJM}^2 :

$$\boldsymbol{\varepsilon}_x = \frac{-r\,\text{sign}(a)}{1+r|a|}\left[a\left(\boldsymbol{x}-m_x\mathbf{1}\right)-\left(\boldsymbol{y}-m_y\mathbf{1}\right)\right], \tag{42}$$

$$\boldsymbol{\varepsilon}_y = \frac{1}{1+r|a|}\left[a\left(\boldsymbol{x}-m_x\mathbf{1}\right)-\left(\boldsymbol{y}-m_y\mathbf{1}\right)\right], \tag{43}$$

$$\varepsilon_{WJM}^2 = \frac{r}{|a|\left(1+r|a|\right)}\left|a\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right|^2. \tag{44}$$

Equation (44) allows us to minimize ε_{WJM}^2 with respect to the single variable a. The resulting necessary and sufficient condition is

$$a_{WJM} = \frac{\text{sign}(\rho)\ \ |\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}} + \text{sign}(\rho) r\hat{\boldsymbol{y}}| - |r\hat{\boldsymbol{y}}|}. \tag{45}$$

From (42) and (43) we see that the total cost-weighted noise is

$$\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 + \left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2 = \frac{r^2+1}{\left(1+r|a_{WJM}|\right)^2}\left|a_{WJM}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right|^2 \tag{46}$$

and the noise power ratio constraint is satisfied:

$$\frac{\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2}{\left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2} = r^2. \tag{47}$$

In the simple case where the noise power ratio is unity ($r = 1$), and both cost-weighted shifted observations have unit power ($|\hat{\boldsymbol{x}}|^2 = |\hat{\boldsymbol{y}}|^2 = 1$), (45) reduces to

$$a_{WJM} = \frac{\text{sign}(\rho)}{\sqrt{2\left(1+|\rho|\right)}-1} \tag{48}$$

and (46) reduces to

$$|\boldsymbol{C\varepsilon}_x|^2 + |\boldsymbol{C\varepsilon}_y|^2 = 2\left[2 - \sqrt{2(1+|\rho|)}\right]. \tag{49}$$

Equations (42) and (43) make it clear that the WJM algorithm generates weighted noise estimates $\boldsymbol{C\varepsilon}_x$ and $\boldsymbol{C\varepsilon}_y$ that are perfectly correlated. On the other hand, the WJM algorithm does not result in any general orthogonality relations. The geometry of the WJM result in a simple two-dimensional case with $m_x = m_y = 0$, $r \approx 2$, $a \approx 0.75$, is shown in Figure 5.

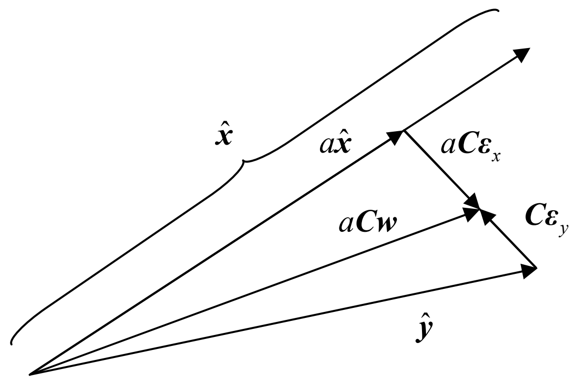

Figure 5. Example geometry of WJM and CJM results.

The WJM algorithm minimizes a weighted version of the total noise power that is less contrived than the quantity minimized in the LSMS algorithm, but not as natural as the quantity minimized in the TLS algorithm. The WJM algorithm does satisfy the noise power ratio constraint. On the other hand, the WJM algorithm generates perfectly correlated noise estimates, but in real situations input and output noise sources are unlikely to be perfectly correlated.

5. CONSTRAINED JOINT MINIMIZATION (CJM)

In order to minimize a more natural noise quantity (as in TLS) and yet enforce the noise power ratio constraint in (5), we invoke a Lagrange multiplier λ and perform a constrained minimization on the total cost-weighted noise power

$$\varepsilon^2_{CJM} = |\boldsymbol{C\varepsilon}_x|^2 + |\boldsymbol{C\varepsilon}_y|^2 + \lambda\left[|\boldsymbol{C\varepsilon}_x|^2 - r^2|\boldsymbol{C\varepsilon}_y|^2\right],\ 0 < \lambda. \tag{50}$$

In the following we seek to minimize ε^2_{CJM} with respect to $\boldsymbol{w}$, a, and b and this leads to the CJM algorithm. The solution steps partially parallel those in the TLS algorithm, but some significant differences are introduced by the Lagrange multiplier.

The necessary condition

$$\frac{\partial}{\partial w_i}\varepsilon_{CJM}^2 = 0,\ \ i = 1\,\text{to}\,n \tag{51}$$

results in

$$(1+\lambda)^2 c_i^2 (w_i - x_i)^2 = (1-\lambda r^2)^2 a^2 c_i^2 \left(a w_i - (y_i - b)\right)^2, \qquad i = 1\,\text{to}\,n, \tag{52}$$

and (52) can be solved for $\boldsymbol{w}$ to get

$$\boldsymbol{w} = \frac{1}{(1+\lambda)+(1-\lambda r^2)a^2}\left[(1+\lambda)\boldsymbol{x} + (1-\lambda r^2)a(\boldsymbol{y} - b\mathbf{1})\right]. \tag{53}$$

We can sum (52) over i and invoke the weighted noise power ratio constraint in (5) to get the relationship

$$(1+\lambda) = \pm\frac{(1-\lambda r^2)a}{r}. \tag{54}$$

Using (54) in (53) allows us to eliminate λ, giving

$$\boldsymbol{w} = \frac{1}{a \pm r^{-1}}\left[(\boldsymbol{y} - b\mathbf{1}) \pm r^{-1}\boldsymbol{x}\right], \text{when}\ a \neq \pm r^{-1}. \tag{55}$$

The two sign choices in (55) come from the single sign choice in (54) and thus they must match. In other words, (55) describes two, not four possible solutions for $\boldsymbol{w}$.

Another necessary condition for minimizing ε_{CJM}^2 is

$$\frac{\partial}{\partial b}\varepsilon_{CJM}^2 = 0 \tag{56}$$

and this results in

$$b = m_y - a\sum_{i=1}^{n} c_i^2 w_i. \tag{57}$$

Using (55) in (57) gives

$$b_{CJM} = m_y - a_{CJM} m_x. \tag{58}$$

We can now use results (55) and (58) to rewrite the noise estimates

$$\boldsymbol{\varepsilon}_x = \frac{-1}{a \pm r^{-1}}\left[a\left(\boldsymbol{x} - m_x\mathbf{1}\right) - \left(\boldsymbol{y} - m_y\mathbf{1}\right)\right], \tag{59}$$

$$\boldsymbol{\varepsilon}_y = \frac{\pm 1}{r\left(a \pm r^{-1}\right)}\left[a\left(\boldsymbol{x} - m_x\mathbf{1}\right) - \left(\boldsymbol{y} - m_y\mathbf{1}\right)\right]. \tag{60}$$

Note that all three of the sign choices in (59) and (60) must match. From (59) and (60) it is clear that the cost-weighted noise power ratio constraint in (5) is satisfied by the necessary conditions derived so far. Further, the total cost-weighted noise power is

$$\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 + \left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2 = \left(1 + r^{-2}\right)\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 = \frac{\left(1 + r^{-2}\right)}{\left(a \pm r^{-1}\right)^2}\left|a\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}}\right|^2. \tag{61}$$

We can minimize (61) by minimizing $\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2$ with respect to the single variable a. The necessary condition

$$\frac{\partial}{\partial a}\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2 = 0 \tag{62}$$

yields

$$a_{CJM} = \frac{r\dfrac{\left|\hat{\boldsymbol{y}}\right|}{\left|\hat{\boldsymbol{x}}\right|} \pm \rho}{r\rho \pm \dfrac{\left|\hat{\boldsymbol{x}}\right|}{\left|\hat{\boldsymbol{y}}\right|}} \tag{63}$$

and the two sign choices must match. From (63) it follows that

$$a_{CJM} \pm r^{-1} = \frac{1}{r} \cdot \frac{\left|\hat{\boldsymbol{x}} \pm r\hat{\boldsymbol{y}}\right|^2}{r\hat{\boldsymbol{x}}^T\hat{\boldsymbol{y}} \pm \left|\hat{\boldsymbol{x}}\right|^2} \tag{64}$$

and

$$\left|a_{CJM}\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}}\right|^2 = \frac{\left|\hat{\boldsymbol{x}}\right|^2\left|\hat{\boldsymbol{y}}\right|^2\left(1 - \rho^2\right)\left|\hat{\boldsymbol{x}} \pm r\hat{\boldsymbol{y}}\right|^2}{\left(r\hat{\boldsymbol{x}}^T\hat{\boldsymbol{y}} \pm \left|\hat{\boldsymbol{x}}\right|^2\right)^2} \tag{65}$$

with all sign choices matching. Thus we can rewrite (61) as

$$\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2+\left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2=\left(1+r^{-2}\right)\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2=\frac{\left|\hat{\boldsymbol{x}}\right|^2\left|\hat{\boldsymbol{y}}\right|^2\left(1-\rho^2\right)\left(r^2+1\right)}{\left|\hat{\boldsymbol{x}}\right|^2\pm 2r\hat{\boldsymbol{x}}^T\hat{\boldsymbol{y}}+\left|\hat{\boldsymbol{y}}\right|^2}. \tag{66}$$

From (66) it is clear that the total cost-weighted noise power will be minimized by picking the positive sign whenever $0<\hat{\boldsymbol{x}}^T\hat{\boldsymbol{y}}$ and picking the negative sign otherwise. This allows us to eliminate the sign choices from previous results (55), (59), (60), (63), (65), and (66) respectively:

$$\boldsymbol{w}=\frac{1}{a+\operatorname{sign}(\rho)r^{-1}}\left[\left(\boldsymbol{y}-b\mathbf{1}\right)+\operatorname{sign}(\rho)r^{-1}\boldsymbol{x}\right],\qquad \text{when } a+\operatorname{sign}(\rho)r^{-1}\neq 0, \tag{67}$$

$$\boldsymbol{\varepsilon}_x=\frac{-1}{a+\operatorname{sign}(\rho)r^{-1}}\left[a\left(\boldsymbol{x}-m_x\mathbf{1}\right)-\left(\boldsymbol{y}-m_y\mathbf{1}\right)\right], \tag{68}$$

$$\boldsymbol{\varepsilon}_y=\frac{\operatorname{sign}(\rho)}{r\left(a+\operatorname{sign}(\rho)r^{-1}\right)}\left[a\left(\boldsymbol{x}-m_x\mathbf{1}\right)-\left(\boldsymbol{y}-m_y\mathbf{1}\right)\right], \tag{69}$$

$$a_{CJM}=\operatorname{sign}(\rho)\frac{r\frac{\left|\hat{\boldsymbol{y}}\right|}{\left|\hat{\boldsymbol{x}}\right|}+\left|\rho\right|}{r\left|\rho\right|+\frac{\left|\hat{\boldsymbol{x}}\right|}{\left|\hat{\boldsymbol{y}}\right|}}, \tag{70}$$

$$\left|a_{CJM}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right|^2=\frac{\left|\hat{\boldsymbol{x}}\right|^2\left|\hat{\boldsymbol{y}}\right|^2\left(1-\rho^2\right)\left|\hat{\boldsymbol{x}}+\operatorname{sign}\left(\rho\right)r\hat{\boldsymbol{y}}\right|^2}{\left(r\left|\hat{\boldsymbol{x}}^T\hat{\boldsymbol{y}}\right|+\operatorname{sign}\left(\rho\right)\left|\hat{\boldsymbol{x}}\right|^2\right)^2}, \tag{71}$$

$$\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2+\left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2=\left(1+r^{-2}\right)\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2=\frac{\left|\hat{\boldsymbol{x}}\right|^2\left|\hat{\boldsymbol{y}}\right|^2\left(1-\rho^2\right)\left(r^2+1\right)}{\left|\hat{\boldsymbol{x}}+\operatorname{sign}\left(\rho\right)r\hat{\boldsymbol{y}}\right|^2}. \tag{72}$$

Now we can use (70) to show that

$$a_{CJM}+\operatorname{sign}(\rho)r^{-1}=\frac{1}{r}\operatorname{sign}(\rho)\frac{\left|\hat{\boldsymbol{x}}+\operatorname{sign}\left(\rho\right)r\hat{\boldsymbol{y}}\right|^2}{\left|\hat{\boldsymbol{x}}\right|^2+r\left|\hat{\boldsymbol{x}}^T\hat{\boldsymbol{y}}\right|}\neq 0 \tag{73}$$

so the condition in (67) is satisfied.

To show that (70) is a sufficient condition for the minimization of $\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2$ we must show that

$$0 < \frac{\partial^2}{\partial a^2}\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2\Bigg|_{a_{CJM}} . \tag{74}$$

Using (73) and (71) we can show that

$$\begin{aligned}\frac{\partial^2}{\partial a^2}\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2\Bigg|_{a_{CJM}} &= \frac{2\left(a_{CJM}+\operatorname{sign}(\rho)r^{-1}\right)^2\left|\hat{\boldsymbol{x}}\right|^2-2\left|a_{CJM}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right|^2}{\left(a_{CJM}+\operatorname{sign}(\rho)r^{-1}\right)^4} \\ &= \frac{2\left|\hat{\boldsymbol{x}}+\operatorname{sign}(\rho)r\hat{\boldsymbol{y}}\right|^2\left|\hat{\boldsymbol{x}}\right|^2\left(\left|\hat{\boldsymbol{x}}\right|^2+2r\left|\hat{\boldsymbol{x}}^T\hat{\boldsymbol{y}}\right|+\rho^2\left|r\hat{\boldsymbol{y}}\right|^2\right)}{\left(a_{CJM}+\operatorname{sign}(\rho)r^{-1}\right)^4 r^2\left(\left|\hat{\boldsymbol{x}}\right|^2+r\left|\hat{\boldsymbol{x}}^T\hat{\boldsymbol{y}}\right|\right)^2} > 0\end{aligned} \tag{75}$$

and we conclude that (70) is both a necessary and sufficient condition for the minimization of $\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2$ and hence for the minimization of $\left|\boldsymbol{C}\boldsymbol{\varepsilon}_x\right|^2+\left|\boldsymbol{C}\boldsymbol{\varepsilon}_y\right|^2$, resulting in the total cost-weighted noise level given in (72).

Note that the CJM algorithm generalizes the two conventional least-squares approaches to fitting $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ and it reproduces the conventional least squares solutions in the limiting cases $r = 0$ and $r \rightarrow \infty$. When $r = 0$, (70) gives

$$a_{CJM} = \rho\frac{\left|\hat{\boldsymbol{y}}\right|}{\left|\hat{\boldsymbol{x}}\right|} \tag{76}$$

which is the standard result for the least-squares problem

$$a\hat{\boldsymbol{x}} = \hat{\boldsymbol{y}} + \boldsymbol{\varepsilon}_y . \tag{77}$$

When $r \rightarrow \infty$, (70) gives

$$a_{CJM} = \frac{1}{\rho}\frac{\left|\hat{\boldsymbol{y}}\right|}{\left|\hat{\boldsymbol{x}}\right|} \tag{78}$$

which is the standard result for the least-squares problem

$$\frac{1}{a}\hat{\boldsymbol{y}} = \hat{\boldsymbol{x}} + \boldsymbol{\varepsilon}_x . \tag{79}$$

Note however that (70) is not just a linear combination of (76) and (78).

In the simple case where both cost-weighted shifted observations have unit power ($|\hat{\boldsymbol{x}}|^2 = |\hat{\boldsymbol{y}}|^2 = 1$) and $r = 1$, (70) reduces to

$$a_{CJM} = \text{sign}(\rho) \tag{80}$$

and (72) reduces to

$$|\boldsymbol{C}\boldsymbol{\varepsilon}_x|^2 + |\boldsymbol{C}\boldsymbol{\varepsilon}_y|^2 = 1 - |\rho| \tag{81}$$

which are the same as the TLS results for this simple case.

Equations (68) and (69) make it clear that the CJM algorithm generates weighted noise estimates $\boldsymbol{C\varepsilon}_x$ and $\boldsymbol{C\varepsilon}_y$ that are perfectly correlated. On the other hand, the CJM algorithm does not result in any general orthogonality relations. The geometry of the CJM result in a simple two-dimensional case with $m_x = m_y = 0$, $r \approx 2$, $a \approx 0.75$, is shown in Figure 5. The CJM algorithm satisfies the noise power ratio constraint (5) and it minimizes a very natural quantity: the total cost-weighted noise power (72). On the negative side the CJM algorithm generates perfectly correlated noise estimates, but in real situations input and output noise sources are unlikely to be perfectly correlated. We consider it to be the most pleasing of the four algorithms presented so far.

6. DIRECT ESTIMATION (DE)

A more direct estimation algorithm is possible when certain restrictions on the noises ξ_x and ξ_y are satisfied. This DE algorithm results from algebraic manipulations of the noisy observations – it does not involve any minimization. It also differs from the other algorithms presented here in that the gain estimation is accomplished without explicit decomposition of the noisy observations $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ into signal and noise estimates.

In analogy to all of the previous algorithms, we will eventually estimate the bias as in (58). This motivates us to start with the equivalent bias-free problem where m_x and m_y have been removed from the noisy observations and hence are no longer a part of the problem. This is described in Figure 6 and by

$$\hat{\boldsymbol{x}} = \boldsymbol{C}\left(\boldsymbol{w_T} - \xi_x\right), \qquad \hat{\boldsymbol{y}} = \boldsymbol{C}\left(a_T \boldsymbol{w_T} - \xi_y\right), \qquad \hat{\boldsymbol{y}} + \boldsymbol{C}\xi_y = a_T\left(\hat{\boldsymbol{x}} + \boldsymbol{C}\xi_x\right). \tag{82}$$

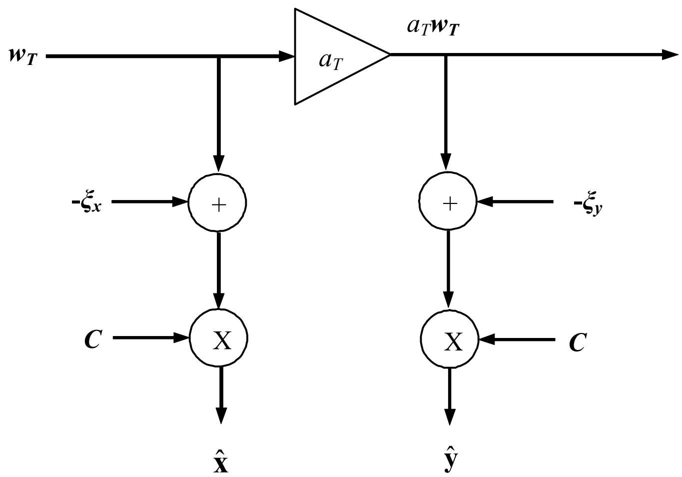

Figure 6. Block diagram for bias-free version of the estimation problem with m_x and m_y removed.

We require that $\boldsymbol{w}$ and ξ_x be independent, that $\boldsymbol{w}$ and ξ_y be independent, and that ξ_x and ξ_y also be independent. We also require that $\boldsymbol{C}^2\xi_x$ and $\boldsymbol{C}^2\xi_y$ have zero mean. Under these conditions, we can calculate expectations over ξ_x and ξ_y:

$$\mathrm{E}\left|\hat{\boldsymbol{x}}\right|^2 = \left|\boldsymbol{C}\boldsymbol{w}_{\boldsymbol{T}}\right|^2 + \mathrm{E}\left|\boldsymbol{C}\xi_{\boldsymbol{x}}\right|^2 = \left|\boldsymbol{C}\boldsymbol{w}_{\boldsymbol{T}}\right|^2 + r^2\mathrm{E}\left|\boldsymbol{C}\xi_y\right|^2, \tag{83}$$

$$\mathrm{E}\left|\hat{\boldsymbol{y}}\right|^2 = a_T^2\left|\boldsymbol{C}\boldsymbol{w}_{\boldsymbol{T}}\right|^2 + \mathrm{E}\left|\boldsymbol{C}\xi_y\right|^2, \text{ and} \tag{84}$$

$$\mathrm{E}(\hat{\boldsymbol{x}}^{\mathrm{T}}\hat{\boldsymbol{y}}) = a_T\left|\boldsymbol{C}\boldsymbol{w}_{\boldsymbol{T}}\right|^2. \tag{85}$$

Note also that (83) enforces the noise power ratio constraint in (2).

We then multiply (84) by r^2, subtract the result from (83), divide by (85), and solve for a to find

$$a_T = \frac{-k \pm \sqrt{k^2 + 4r^2}}{2r^2}, \qquad \text{where } k = \frac{\mathrm{E}\left|\hat{\boldsymbol{x}}\right|^2 - r^2\mathrm{E}\left|\hat{\boldsymbol{y}}\right|^2}{\mathrm{E}(\hat{\boldsymbol{x}}^{\mathrm{T}}\hat{\boldsymbol{y}})}. \tag{86}$$

For a meaningful solution, we require sign(a_T)=sign(ρ), and this requirement can be satisfied by

$$a_T = \frac{-k + \operatorname{sign}(\rho)\sqrt{k^2 + 4r^2}}{2r^2}. \tag{87}$$

Thus we have used the noisy observations $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ to calculate a value for a_T. This calculated value of a_T then naturally becomes our estimate of the system gain a_{DE}. Thus the expansion of (87) yields

$$a_{DE} = \begin{cases} \dfrac{\left(\dfrac{|\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}}|} r^2 - \dfrac{|\hat{\boldsymbol{x}}|}{|\hat{\boldsymbol{y}}|}\right) + \sqrt{\left(\dfrac{|\hat{\boldsymbol{y}}|}{|\hat{\boldsymbol{x}}|} r^2 - \dfrac{|\hat{\boldsymbol{x}}|}{|\hat{\boldsymbol{y}}|}\right)^2 + 4r^2\rho^2}}{2r^2\rho}, & \rho \neq 0, \\ 0, & \rho = 0. \end{cases} \tag{88}$$

For notational clarity, (88) uses $|\hat{\boldsymbol{x}}|, |\hat{\boldsymbol{y}}|$, and ρ in place of $\sqrt{\mathrm{E}|\hat{\boldsymbol{x}}|^2}, \sqrt{\mathrm{E}|\hat{\boldsymbol{y}}|^2}$, and $\dfrac{\mathrm{E}(\hat{\boldsymbol{x}}^{\mathrm{T}}\hat{\boldsymbol{y}})}{\sqrt{\mathrm{E}|\hat{\boldsymbol{x}}|^2}\sqrt{\mathrm{E}|\hat{\boldsymbol{y}}|^2}}$ respectively. In practice, each of these quantities will be estimated by conventional means using all available observations.

We estimate the bias b as before:

$$b_{DE} = m_y - a_{DE} m_x. \tag{89}$$

Note that when $r = 1$, the result for a_{DE} in (88) reduces to the TLS result for a_{TLS} in (30). Thus for the special case of estimating a single system gain and a bias we might think of the DE algorithm as a generalization of the TLS algorithm. This is somewhat unexpected since the TLS algorithm is based on the minimization of an error expression but the DE algorithm is not.

We have required $\boldsymbol{w_T}$ and ξ_x, $\boldsymbol{w_T}$ and ξ_y, and ξ_x and ξ_y, to be pairwise independent. From a geometric viewpoint these independence constraints correspond to orthogonality constraints, so no 2-dimensional example can be drawn. Figure 4 does show two of the three required orthogonality constraints, so we might consider this to be a 2-dimensional projection of a simple 3-dimensional DE example. In this figure we can view the DE algorithm operation geometrically as swinging $a\boldsymbol{Cw}$ and sliding a to find the unique solution that solves the noise power ratio constraint given in (2).

The DE algorithm generates estimates of a and b without any minimization. The noise power ratio constraint is directly incorporated into the DE algorithm derivation and thus it is always satisfied. We consider the DE and CJM algorithms to be the most pleasing algorithms for estimating gain and bias. We do note that the DE algorithm does not provide decompositions of noisy observations into signal and noise estimates.

7. EXAMPLE SIMULATION RESULTS AND DISCUSSION

This section contains the results of computer simulations of the five gain and bias estimation algorithms. The bias estimation result is identical in the five algorithms. Further, the bias issue is effectively removed from the problem before gain estimation. Thus we are able to simplify this section by treating only the case b=0, with no loss of generality in terms of the estimation of a.

Figure 7 shows gain estimates for the five algorithms for the case $a_{dB} = 20\log_{10}(a) = 0$, $-40 \le r_{dB} = 20\log_{10}(r) \le 40$. Note that r_{dB} controls the noise generators in our simulations and thus this is the true value of r_{dB}. All algorithms use this true value of r_{dB} as an input as well. The input $\boldsymbol{w}$ is a chirp that sweeps from a normalized frequency of $\frac{\pi}{40}$ to $\frac{\pi}{4}$ radians/sample. The noise vectors ξ_x and ξ_y contain white Gaussian noise and are uncorrelated to $\boldsymbol{w}$ and to each other. For this problem we define an intuitive total signal-to-total noise ratio (SNR) as

$$SNR = 10\log_{10}\left(\frac{|\boldsymbol{w}|^2 + |a\boldsymbol{w}|^2}{|\boldsymbol{\xi}_x|^2 + |\boldsymbol{\xi}_y|^2}\right) \tag{90}$$

and set it to 10 dB. All vectors have length n=4096 and we use $\frac{1}{\sqrt{n}}\boldsymbol{I}_{n\times n}$ for the cost function $\boldsymbol{C}$.

For the example shown in Figure 7, all five gain estimates show similar variances but they show very different means. As r_{dB} gets large and negative, the LSMS mean estimation error vanishes. This is expected since this case corresponds to noise-free input observations and that is the situation that least-squares is meant to handle. The TLS algorithm has small mean estimation error only around r_{dB} = 0. This is also expected since TLS tries to minimize both noise components equally. The WJM algorithm has small mean estimation error only when r_{dB} is large and positive. The CJM algorithm has small mean estimation error near r_{dB} = 0 and when $|r_{dB}|$ gets large, where it tends toward one of two different least-squares solutions. The DE algorithm has small mean estimation error across the full range shown. As expected, the CJM and DE algorithms give the best estimates across a wide range of r values. This result stands over a good range of SNR values, a values, N values, and input signal types including other chirps, tones, speech, and noise. Thus through the remainder of this section, we present simulation results for only the CJM and DE algorithms.

Figures 8-16 show example errors in gain estimates for true gain values between -20 and +20 dB. The input and noises of Figure 7 are used again. The results are almost identical for the other signal types mentioned above. As before, n=4096. Figures 8-10 use SNR=20 dB, Figures 11-13 use SNR=10 dB, and Figures 14-16 use SNR=0 dB. Within each triple of figures we use three values of r_{dB} :-10, 0, and +10 dB.

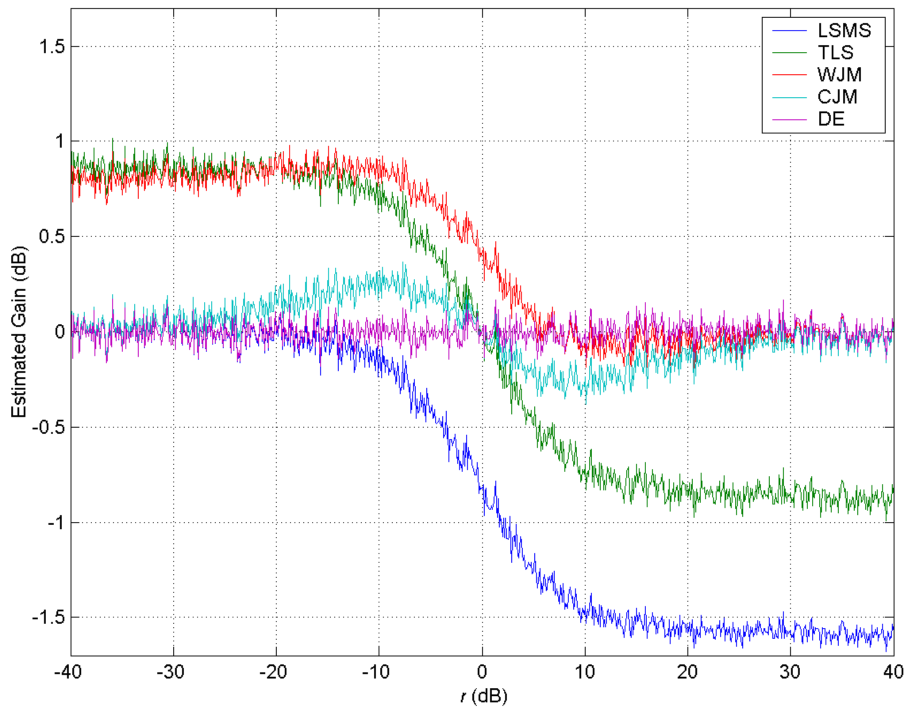

Figure 7. Example results for five estimation algorithms; true gain is 0 dB.

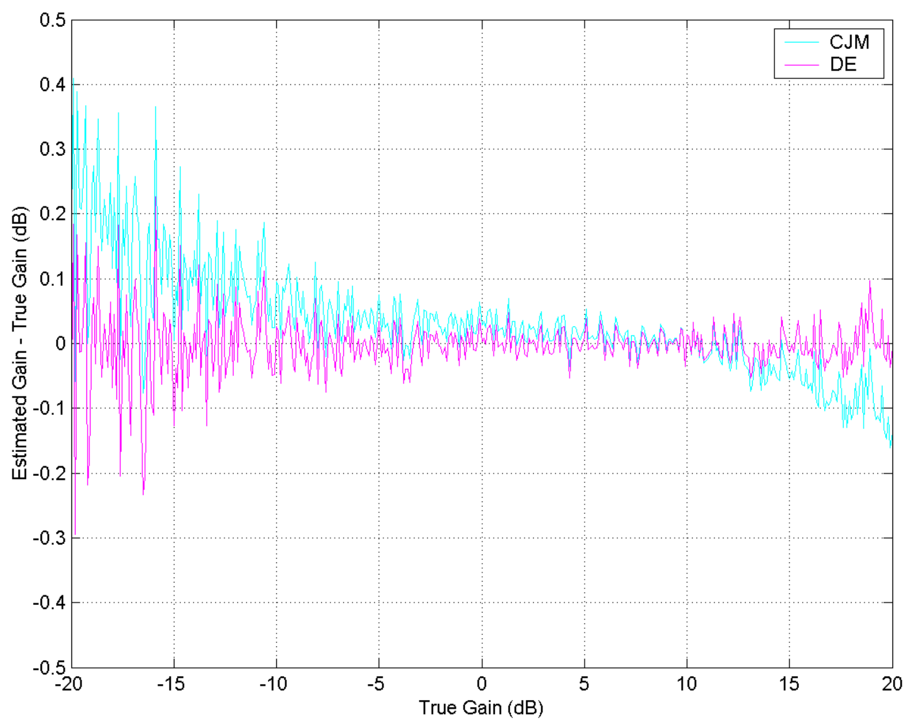

Figure 8. Example estimation errors for the case *SNR*=20 dB, *r* =-10 dB.

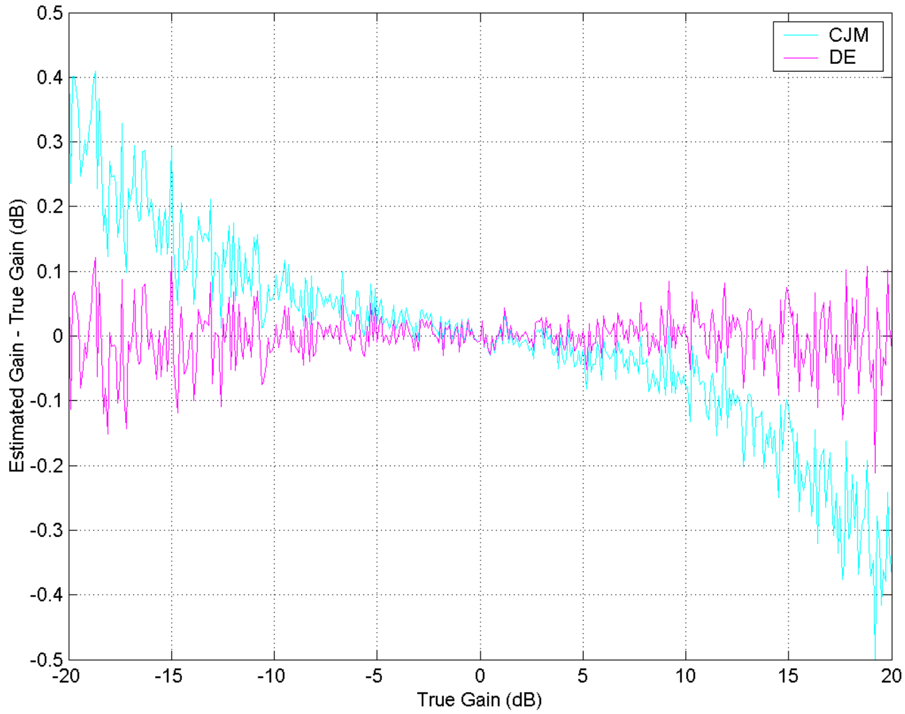

Figure 9. Example estimation errors for the case *SNR*=20 dB, *r* =0 dB.

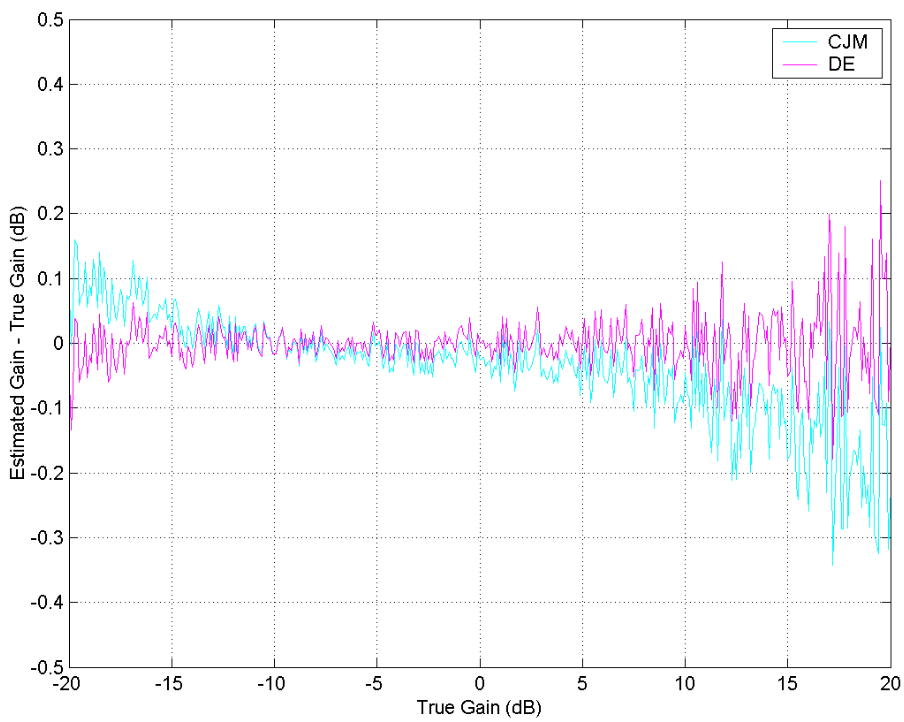

Figure 10. Example estimation errors for the case *SNR*=20 dB, *r* =10 dB.

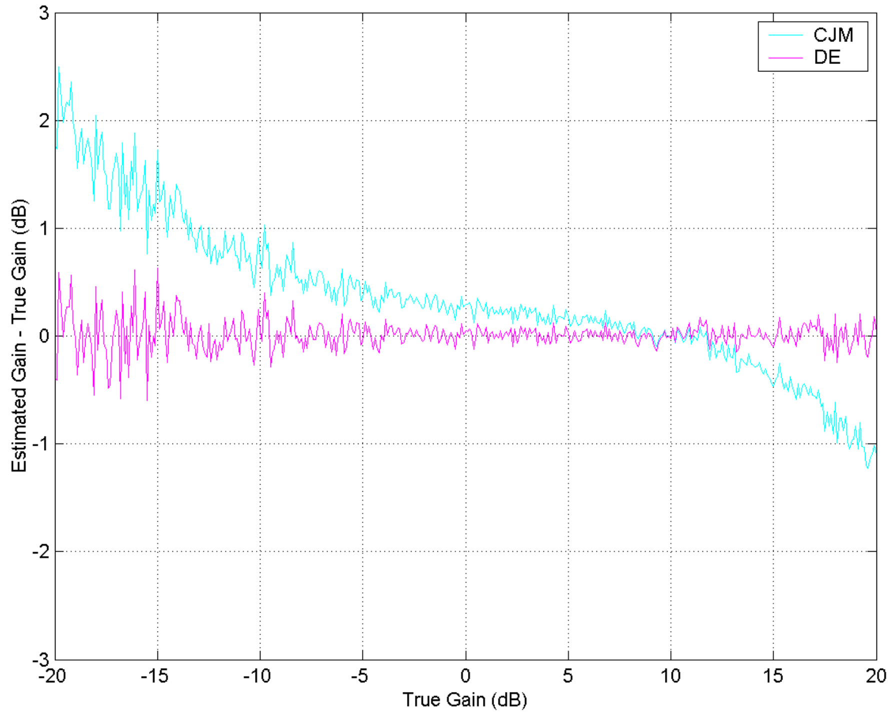

Figure 11. Example estimation errors for the case *SNR*=10 dB, *r* =-10 dB.

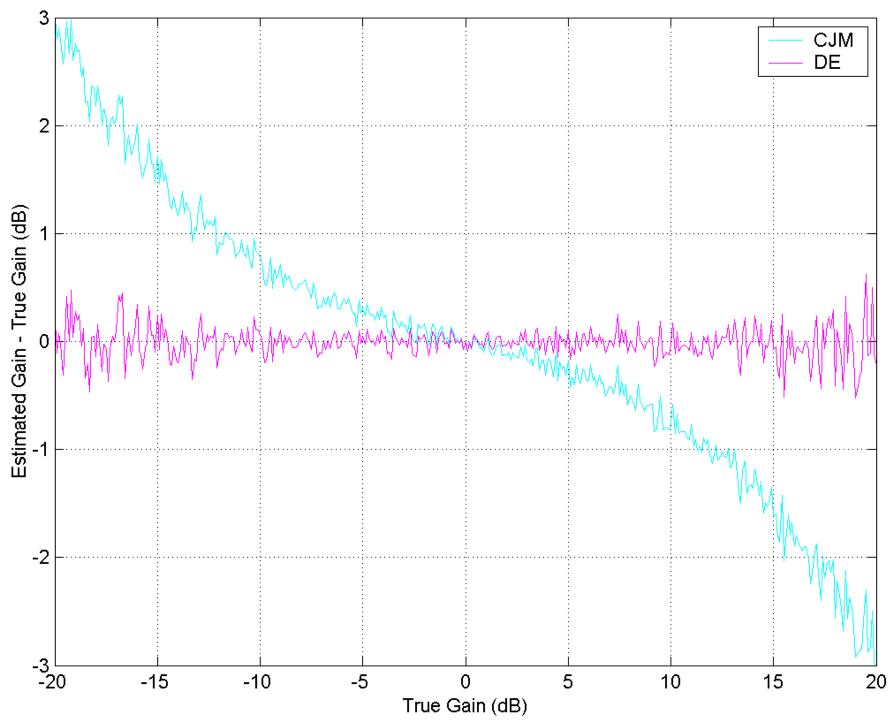

Figure 12. Example estimation errors for the case *SNR*=10 dB, *r* =0 dB.

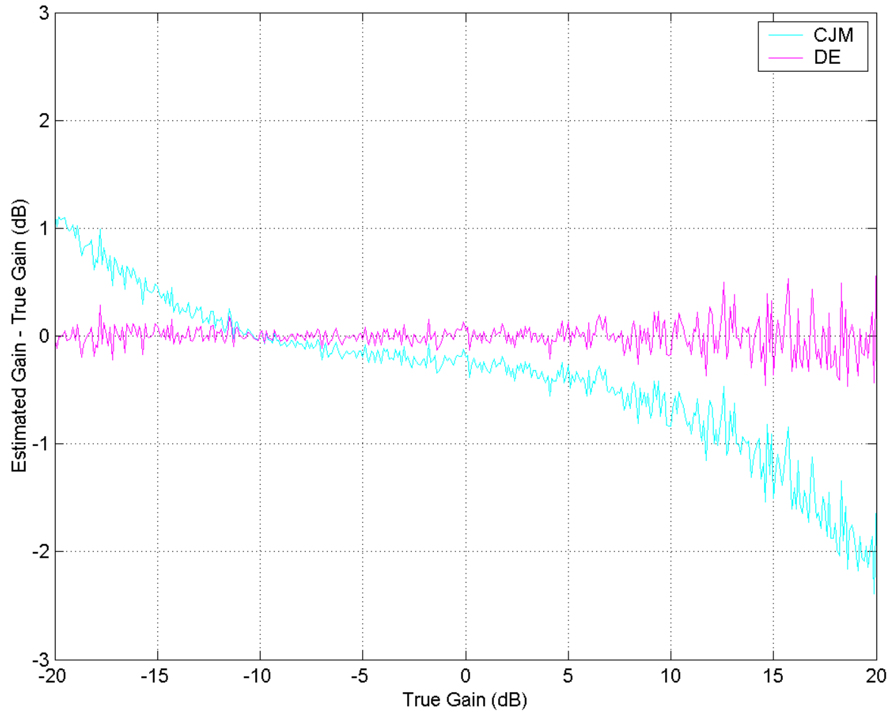

Figure 13. Example estimation errors for the case *SNR*=10 dB, *r* =10 dB.

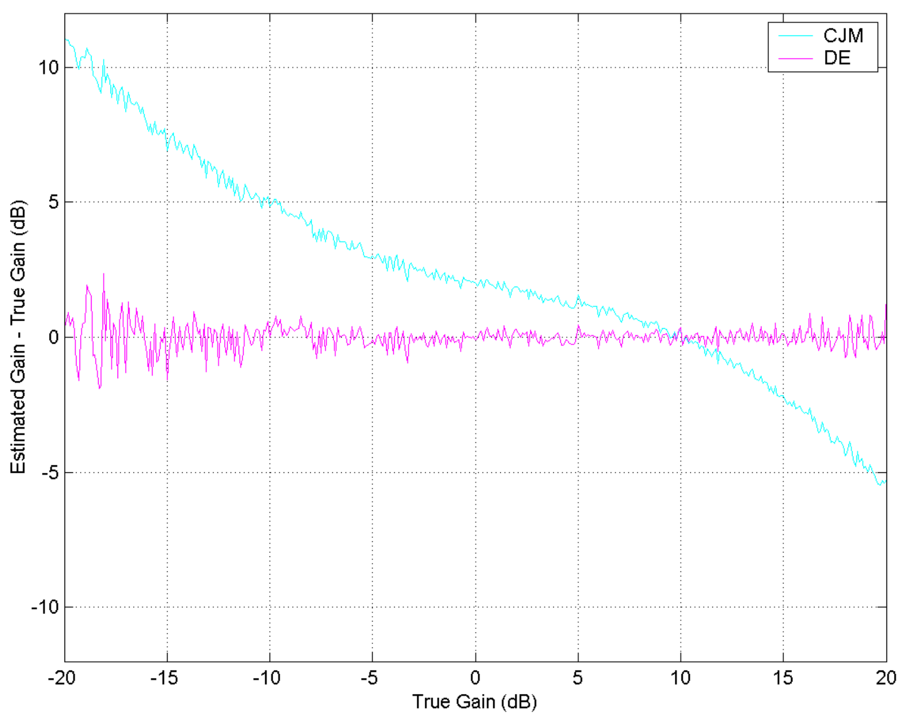

Figure 14. Example estimation errors for the case *SNR*=0 dB, *r* =-10 dB.

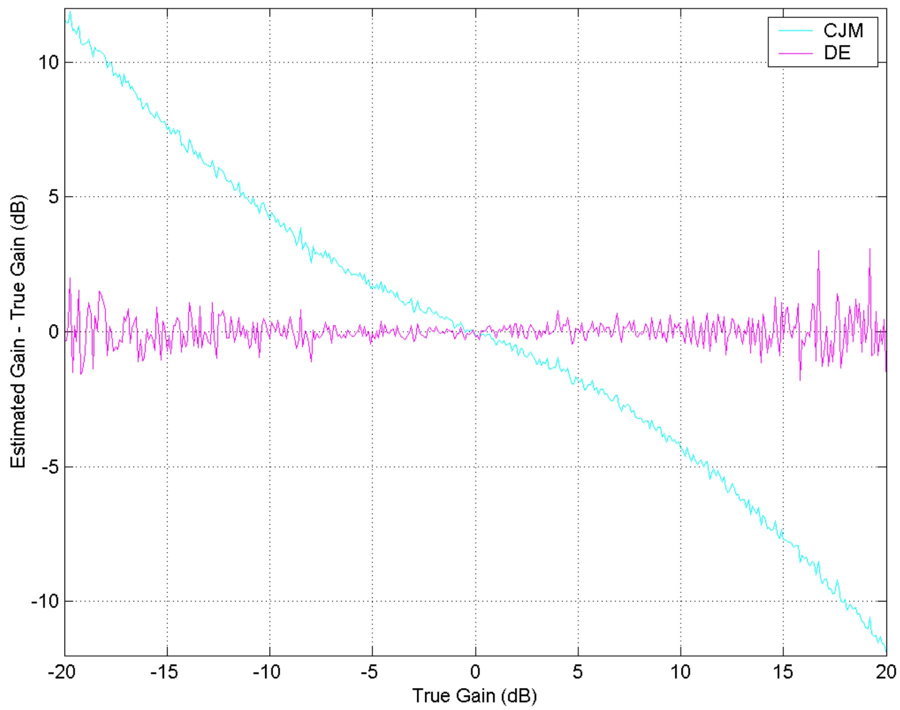

Figure 15. Example estimation errors for the case *SNR*=0 dB, *r* =0 dB.

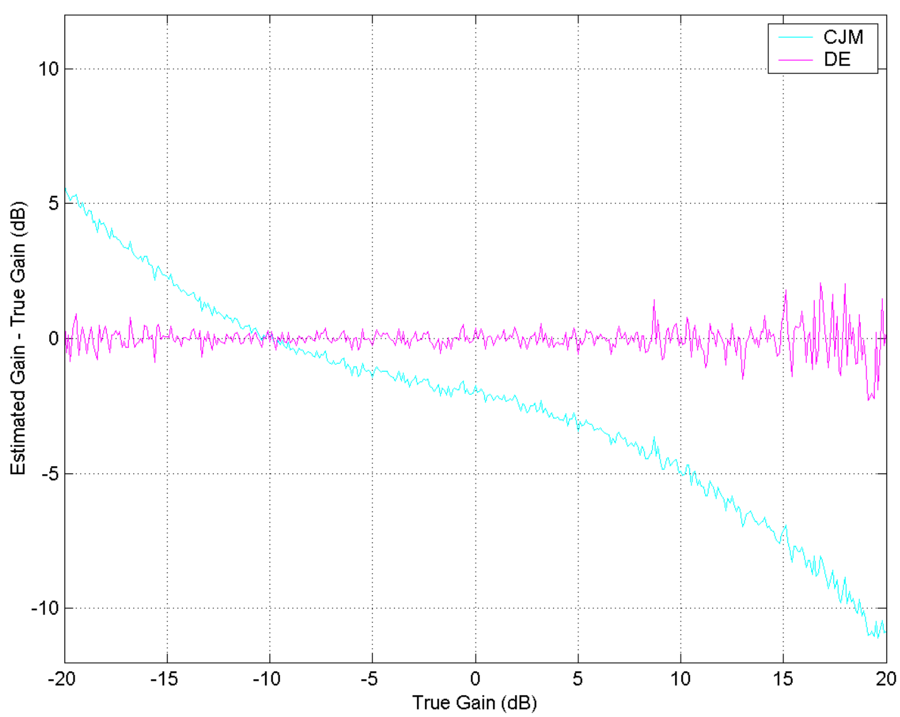

Figure 16. Example estimation errors for the case *SNR*=0 dB, *r* =10 dB.

As expected, Figures 8-16 show that the estimation errors are smaller at higher *SNR* values. These figures also indicate that DE gain estimation errors have nearly zero mean. The CJM gain estimation errors have zero mean only when $a_{dB} \approx -r_{dB}$. The CJM algorithm overestimates gain when $a_{dB} < -r_{dB}$ and it underestimates gain when $a_{dB} > -r_{dB}$. Both algorithms show a smaller estimation error variance near $a_{dB} = 0$, and that variance increases as $|a_{dB}|$ increases. It also appears that the DE estimation error variance is smaller when $a_{dB} \approx -r_{dB}$ than when $a_{dB} \approx r_{dB}$. Finally, we note that the CJM and DE estimation error variances are generally similar. The exception comes at lower *SNR* values (e.g. *SNR*=0 dB) where the CJM error variance is markedly smaller than the DE error variance when $|a_{dB}| \gg 0$.

Figure 17 provides an example correlated noise case. Here the noise vectors ξ_x and ξ_y contain a mixture of harmonic distortion (correlated to ***w***) and white Gaussian noise (uncorrelated to ***w***). For this example *SNR* = 20 dB and *r* = 0 dB. The chirp of Figure 7 is used for the signal ***w***. Across most of the range of true gain values, the CJM algorithm provides a smaller mean estimation error than the DE algorithm. We note here that the DE algorithm is explicitly predicated on an uncorrelated noise environment while the CJM algorithm is not explicitly predicated on an uncorrelated noise environment. In fact, while the CJM algorithm makes no assumptions about noise correlation, it does generate noise estimates that are perfectly correlated. Thus one might expect the CJM algorithm to be better suited to correlated noise environments. However, this expectation is not confirmed by our simulations. For some correlated noise environments, the DE algorithm performs better than the CJM algorithm. Results for correlated noise environments show modest dependence on signal type but strong dependence on noise type.

Figure 18 shows example results for a case where the CJM and DE algorithms are given inaccurate values of *r*. This situation would occur when the actual *r* value in the noisy observations is different from an estimated, measured, or modeled *r* value. The signal and noises of Figure 7 are used, true r_{dB}=0 dB, *SNR*=10 dB, and *n*=4096. These conditions would reproduce Figure 12, except that the estimation algorithms were given inaccurate values of *r*. The traces labeled "*r* 10 dB low" result from *r* =-10 dB and the traces labeled "*r* 10 dB high" result from *r* =10 dB. For these two situations, the CJM algorithm provides better estimates across most of the range of true gain values used in the simulation and this result is largely invariant to signal type. But this is not a general result. For some other *r* inaccuracies and true gain values, the DE algorithm will provide better estimates than the CJM algorithm.

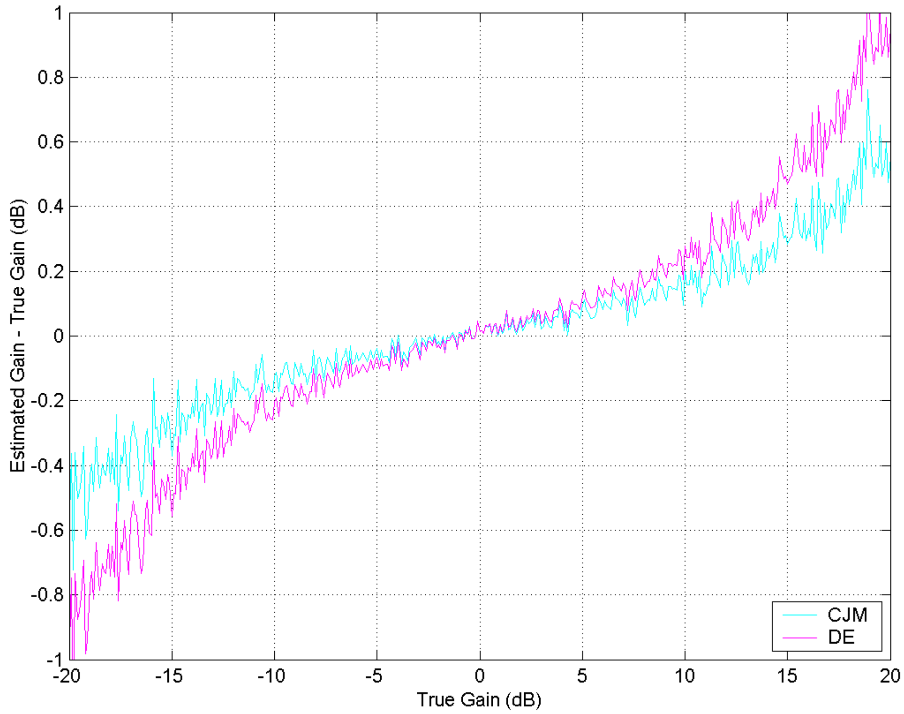

Figure 17. Example estimation errors for correlated noise environment.

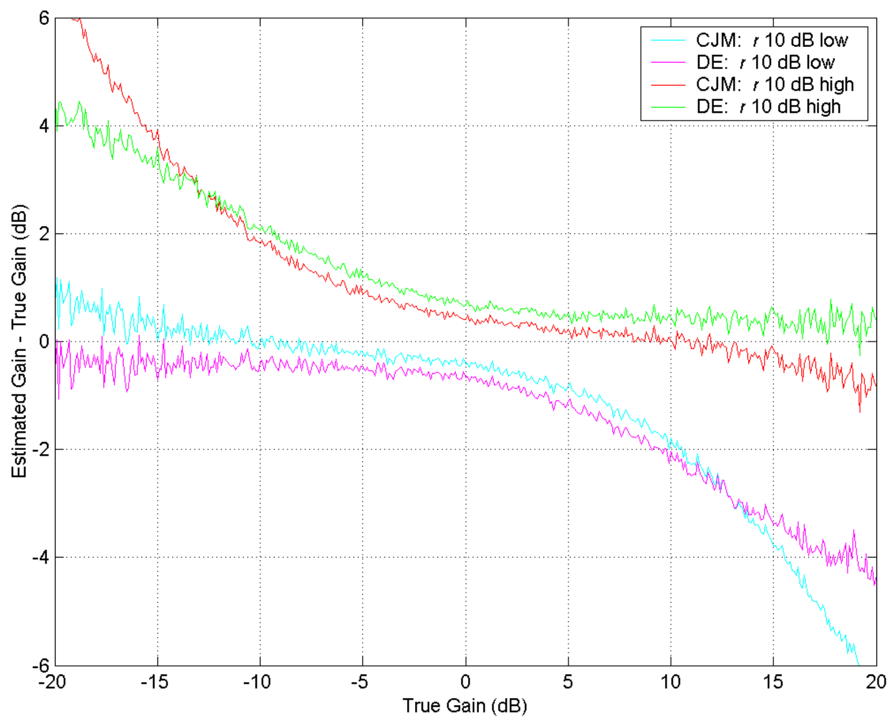

Figure 18. Example estimation errors for the case of inaccurate *r* values.

From these results we conclude that the DE algorithm would generally be the best choice when noise is uncorrelated and r is well-known. When noise is correlated, or r is not well known, one should do simulations of the specific situation to determine if the CJM or DE algorithm is more likely to provide correct estimates. Note also that there are bias vs. variance trade-offs at work here. For example Figure 15 shows that when $SNR = 0$ dB, and $r = 0$ dB, if the true gain is 15 dB, then the CJM algorithm gives a high bias (high mean error) low variance estimate of that true gain, while the DE algorithm gives a low bias, high variance estimate of that true gain.

Finally, we mention here three additional algorithm properties. First, we note that we have been able to show that $\text{sign}(a)=\text{sign}(\rho)$ for all five algorithms and this phase consistency is certainly a desirable property. Second we note a scaling property displayed by the LSMS, WJM, CJM and DE algorithms. For these four algorithms, we have been able to show that if the inputs $\boldsymbol{x}$, $\boldsymbol{y}$, and r yield the gain estimate a, then the inputs $\boldsymbol{x}$, $\alpha\boldsymbol{y}$, and r/α yield the gain estimate αa. Third we note that the LSMS, TLS, WJM, and CJM algorithms decompose noisy observations into signal and noise estimates and thus have potential use in denoising applications.

8. SUMMARY

We have provided background context and references for the general problem of identifying linear systems from noisy input and output observations and have noted that the existing work in this area allows for identification of multivariate systems and thus is fairly complex. We introduced the important special case of the estimation of just a system gain and bias from noisy input and output observations and added a single piece of side-information to the problem: the noise power ratio, r. We described the potential utility of solutions to this special case of the problem. We then derived five different solutions.

The LSMS algorithm comes from a conventional least-squares step followed by the manual splitting of the least-squares residual into two vectors. This algorithm satisfies the noise power ratio constraint but the gain and bias estimates do not directly depend on r, and this is inconsistent with our desire to use that side-information advantageously. An additional disadvantage is that the quantity minimized by the LSMS algorithm is somewhat unnatural.

The TLS algorithm minimizes a total noise power which is a more natural quantity, but it does not make use of r, and thus the noise power ratio constraint is not satisfied in general. We derived our TLS algorithm for this specific problem from first principles. Our algorithm has only a single step and does not require a SVD. The TLS algorithm does provide motivation for the weighting factor used in the WJM algorithm.

The WJM algorithm minimizes a weighted version of the total noise power that is more natural than the quantity minimized in the LSMS algorithm, but not as natural as the quantity minimized in the TLS algorithm. The WJM algorithm does satisfy the noise power ratio constraint.

The CJM algorithm derivation invokes a Lagrange multiplier which allows for the simultaneous minimization of a natural quantity while enforcing the noise power ratio constraint. This algorithm generalizes two conventional least-squares solutions and it reproduces conventional least squares solutions in the limiting cases $r = 0$ and $r \to \infty$.

The DE algorithm differs from the other four algorithms in that its derivation does not include the minimization of any noise quantity. Rather the DE algorithm results from algebraic manipulations of the noisy observations. The DE algorithm requires additional assumptions on the observation noises. Unlike the other algorithms, the DE algorithm does not provide decompositions of noisy observations into signal and noise components. The DE algorithm can be interpreted as a generalization of the TLS algorithm, even though they were derived in very different ways.

From a mathematical perspective, we observed that the CJM and DE algorithms seemed most satisfying. A fundamental distinction between these two algorithms lies in the area of noise correlation. While the DE algorithm is explicitly predicated on an uncorrelated noise environment, the CJM algorithm actually produces noise estimates that are perfectly correlated. We conducted simulations that confirmed that these two algorithms would be most likely to give correct estimates of system gain. In our simulations we found that the CJM and DE estimation error variances are generally similar, but the estimation error means can be quite different. We concluded that when noise is uncorrelated and r is well-known, the DE algorithm would generally be the best choice. When noise is correlated, or r is not well known, further simulations of the specific situation are required to determine if the CJM or DE algorithm is more likely to provide correct estimates.

We have also provided simplified 2-dimensional geometric interpretations for the LSMS, TLS, WJM, and CJM algorithms, noting the orthogonal and parallel vector relationships that would extend to higher dimensional examples. For convenience, the algorithm details are summarized in tabular form in the appendix.

9. REFERENCES

[1] J.K. Tugnait, "Stochastic system identification with noisy input using cumulant statistics," *IEEE Transactions on Automatic Control*, vol. 37, no. 4, pp. 476-485, Apr. 1992.

[2] T. Söderström, "Identification of stochastic linear systems in presence of input noise," *Automatica*, vol. 17, no. 5, pp. 713-725, Sep. 1981.

[3] G.H. Golub and C.F. Van Loan, *Matrix Computations*, Baltimore, MD: Johns Hopkins University Press, 1989.

[4] L.L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*, Reading, MA: Addison-Wesley, 1991.

[5] J. Rissanen, "Estimation of errors-in-variables models," *Proceedings of the 27th IEEE Conference on Decision and Control*, Austin, Texas, Dec. 1988, vol. 3, pp. 1828-1830.

[6] J.M.M. Anderson and G.B. Giannakis, "Noisy input/output system identification using cumulants and the Steiglitz-McBride algorithm," *IEEE Transactions on Signal Processing*, vol. 44, no. 2, pp. 1021-1024, Feb. 1996.
[7] J. K. Tugnait, "Identification of multivariable stochastic linear systems using integrated polyspectrum given noisy input-output data," *Proceedings of the IEEE Signal Processing Workshop on Higher-Order Statistics*, Banff, Alta, Canada, Jul. 1997, pp. 239-244.
[8] S. Chandrasekaran, G.H. Golub, M. Gu, and A.H. Sayed, "Parameter estimation in the presence of bounded modeling errors," *IEEE Signal Processing Letters*, vol. 4, no. 7, pp. 195-197, Jul. 1997.
[9] A.H. Sayed and S.Chandrasekaran, "Parameter estimation with multiple sources and levels of uncertainties," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 680-692, Mar. 2000.
[10] A. Yeredor, "The extended least squares criterion: Minimization algorithms and applications," *IEEE Transactions on Signal Processing*, vol. 49, no. 1, pp. 74-86, Jan. 2001.
[11] A. Yeredor, "The joint MAP-ML criterion and its relation to ML and to extended least squares," *IEEE Transactions on Signal Processing*, vol. 48, no. 12, pp. 3484-3492, Dec. 2000.
[12] L.V. Hedges and I. Olkin, *Statistical Methods for Meta-Analysis*, Orlando, Florida: Academic Press, 1985.
[13] S. Wolf, "Measuring the end-to-end performance of digital video systems," *IEEE Transactions on Broadcasting*, vol. 43, no. 3, pp. 320-328, Sep., 1997.
[14] C.J. van den Branden Lambrecht, D.M. Costantini, G.L. Sicuranza, and M. Kunt, "Quality assessment of motion rendition in video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 5, pp. 766-782, Aug. 1999.
[15] S.D. Voran, "Objective estimation of perceived speech quality, part II: Evaluation of the measuring normalizing block technique," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 4, pp. 383-390, Jul. 1999.
[16] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra, "Perceptual evaluation of speech quality (PESQ) – A new method for speech quality assessment of telephone networks and codecs," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 2001*, Salt Lake City, May 2001, vol. 2, pp. 749-752.

APPENDIX: SUMMARY OF ESTIMATION ALGORITHMS

In this appendix we summarize the five gain and bias estimation algorithms in tabular form. Table A-1 summarizes the mathematical constructions that lead to each of the five algorithms. Table A-2 gives the preparation steps that are common to all five algorithms. Table A-3 shows how each of the five algorithms calculates estimates of *a* and *b*. This table also gives the resulting noise power ratio. Tables A-4 and A-5 provide additional algorithm results that may be of interest.

Table A-1. Mathematical Constructions Leading to the Five Algorithms

Algorithm	Assumptions beyond Figure 1	Quantity Minimized	Additional Constraints
Least squares plus manual splitting (LSMS)	None	$\varepsilon_{LSMS}^2 = \left\lvert \boldsymbol{C}\left(\boldsymbol{\varepsilon}_y - a\boldsymbol{\varepsilon}_x\right)\right\rvert^2$	$\dfrac{\lvert \boldsymbol{C}\boldsymbol{\varepsilon}_x\rvert^2}{\lvert \boldsymbol{C}\boldsymbol{\varepsilon}_y\rvert^2} = r^2$
Total least squares (TLS)	None	$\varepsilon_{TLS}^2 = \lvert \boldsymbol{C}\boldsymbol{\varepsilon}_x\rvert^2 + \lvert \boldsymbol{C}\boldsymbol{\varepsilon}_y\rvert^2$	None
Weighted joint minimization (WJM)	None	$\varepsilon_{WJM}^2 = \lvert \boldsymbol{C}\boldsymbol{\varepsilon}_x\rvert^2 + \dfrac{r}{\lvert a\rvert}\lvert \boldsymbol{C}\boldsymbol{\varepsilon}_y\rvert^2$	None
Constrained joint minimization (CJM)	None	$\varepsilon_{CJM}^2 = \lvert \boldsymbol{C}\boldsymbol{\varepsilon}_x\rvert^2 + \lvert \boldsymbol{C}\boldsymbol{\varepsilon}_y\rvert^2$	$\dfrac{\lvert \boldsymbol{C}\boldsymbol{\varepsilon}_x\rvert^2}{\lvert \boldsymbol{C}\boldsymbol{\varepsilon}_y\rvert^2} = r^2$
Direct estimation (DE)	$\boldsymbol{w}$ and ξ_x are independent $\boldsymbol{w}$ and ξ_y are independent ξ_x and ξ_y are independent $\boldsymbol{C}^2\xi_x$ and $\boldsymbol{C}^2\xi_y$ are zero mean	No Minimization	$\dfrac{\mathrm{E}\lvert \boldsymbol{C}\xi_x\rvert^2}{\mathrm{E}\lvert \boldsymbol{C}\xi_y\rvert^2} = r^2$

Table A-2. Preparation Steps Common to the Algorithms

Step	Equations
1. Calculate cost-weighted means.	$m_x = \sum_{i=1}^{n} c_i^2 x_i \quad , \quad m_y = \sum_{i=1}^{n} c_i^2 y_i$
2. Calculate cost-weighted shifted versions of ***x*** and ***y***.	$\hat{\boldsymbol{x}} = C\left(\boldsymbol{x} - m_x \mathbf{1}\right) \quad , \quad \hat{\boldsymbol{y}} = C\left(\boldsymbol{y} - m_y \mathbf{1}\right)$
3. Calculate normalized cross correlation.	$\rho = \dfrac{\hat{\boldsymbol{x}}^{\mathrm{T}} \hat{\boldsymbol{y}}}{\lvert\hat{\boldsymbol{x}}\rvert \lvert\hat{\boldsymbol{y}}\rvert}$
4. Calculate correlation sign.	$\mathrm{sign}(\rho) = \begin{cases} 1 & \text{when} \quad 0 < \rho \\ -1 & \text{otherwise} \end{cases}$

Table A-3. Basic Algorithm Results

Algorithm	Estimated Bias	Estimated Gain	Noise Power Ratio
LSMS	$b_{LSMS} = m_y - a_{LSMS} m_x$	$a_{LSMS} = \frac{\lvert \hat{\boldsymbol{y}} \rvert}{\lvert \hat{\boldsymbol{x}} \rvert} \rho$	$\frac{\lvert \boldsymbol{C\varepsilon}_x \rvert^2}{\lvert \boldsymbol{C\varepsilon}_y \rvert^2} = r^2$
TLS	$b_{TLS} = m_y - a_{TLS} m_x$	$a_{TLS} = \begin{cases} \frac{\left(\frac{\lvert \hat{\boldsymbol{y}} \rvert}{\lvert \hat{\boldsymbol{x}} \rvert} - \frac{\lvert \hat{\boldsymbol{x}} \rvert}{\lvert \hat{\boldsymbol{y}} \rvert} \right) + \sqrt{\left(\frac{\lvert \hat{\boldsymbol{y}} \rvert}{\lvert \hat{\boldsymbol{x}} \rvert} - \frac{\lvert \hat{\boldsymbol{x}} \rvert}{\lvert \hat{\boldsymbol{y}} \rvert} \right)^2 + 4\rho^2}}{2\rho}, & \rho \neq 0 \\ 0, & \rho = 0 \end{cases}$	$\frac{\lvert \boldsymbol{C\varepsilon}_x \rvert^2}{\lvert \boldsymbol{C\varepsilon}_y \rvert^2} = a^2$
WJM	$b_{WJM} = m_y - a_{WJM} m_x$	$a_{WJM} = \frac{\operatorname{sign}(\rho) \ \lvert \hat{\boldsymbol{y}} \rvert}{\lvert \hat{\boldsymbol{x}} + \operatorname{sign}(\rho) r \hat{\boldsymbol{y}} \rvert - \lvert r \hat{\boldsymbol{y}} \rvert}$	$\frac{\lvert \boldsymbol{C\varepsilon}_x \rvert^2}{\lvert \boldsymbol{C\varepsilon}_y \rvert^2} = r^2$
CJM	$b_{CJM} = m_y - a_{CJM} m_x$	$a_{CJM} = \operatorname{sign}(\rho) \frac{r \frac{\lvert \hat{\boldsymbol{y}} \rvert}{\lvert \hat{\boldsymbol{x}} \rvert} + \lvert \rho \rvert}{r \lvert \rho \rvert + \frac{\lvert \hat{\boldsymbol{x}} \rvert}{\lvert \hat{\boldsymbol{y}} \rvert}}$	$\frac{\lvert \boldsymbol{C\varepsilon}_x \rvert^2}{\lvert \boldsymbol{C\varepsilon}_y \rvert^2} = r^2$
DE	$b_{DE} = m_y - a_{DE} m_x$	$a_{DE} = \begin{cases} \frac{\left(\frac{\lvert \hat{\boldsymbol{y}} \rvert}{\lvert \hat{\boldsymbol{x}} \rvert} r^2 - \frac{\lvert \hat{\boldsymbol{x}} \rvert}{\lvert \hat{\boldsymbol{y}} \rvert} \right) + \sqrt{\left(\frac{\lvert \hat{\boldsymbol{y}} \rvert}{\lvert \hat{\boldsymbol{x}} \rvert} r^2 - \frac{\lvert \hat{\boldsymbol{x}} \rvert}{\lvert \hat{\boldsymbol{y}} \rvert} \right)^2 + 4r^2\rho^2}}{2r^2\rho}, & \rho \neq 0, \\ 0, & \rho = 0 \end{cases}$	$\frac{\mathrm{E} \lvert \boldsymbol{C} \xi_x \rvert^2}{\mathrm{E} \lvert \boldsymbol{C} \xi_y \rvert^2} = r^2$

Table A-4. Additional Algorithm Results

Algorithm	Weighted Input Noise Estimate $\boldsymbol{C\varepsilon}_x$	Weighted Output Noise Estimate $\boldsymbol{C\varepsilon}_y$	Total Weighted Noise Power $\left\lvert\boldsymbol{C\varepsilon}_x\right\rvert^2+\left\lvert\boldsymbol{C\varepsilon}_y\right\rvert^2$
LSMS	$\frac{-r\,\mathrm{sign}(\rho)}{1+r\lvert a_{LSMS}\rvert}\left[a_{LSMS}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right]$	$\frac{1}{\left(1+r\lvert a_{LSMS}\rvert\right)}\left[a_{LSMS}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right]$	$\frac{r^2+1}{\left(1+\lvert a_{LSMS}\rvert r\right)^2}\lvert\hat{\boldsymbol{y}}\rvert^2\left(1-\rho^2\right)$
TLS	$\frac{-a_{TLS}}{1+a_{TLS}^2}\left[a_{TLS}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right]$	$\frac{1}{1+a_{TLS}^2}\left[a_{TLS}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right]$	$\frac{1}{1+a_{TLS}^2}\left\lvert a_{TLS}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right\rvert^2$
WJM	$\frac{-r\,\mathrm{sign}(\rho)}{1+r\lvert a_{WJM}\rvert}\left[a_{WJM}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right]$	$\frac{1}{1+r\lvert a_{WJM}\rvert}\left[a_{WJM}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right]$	$\frac{r^2+1}{\left(1+r\lvert a_{WJM}\rvert\right)^2}\left\lvert a_{WJM}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right\rvert^2$
CJM	$\frac{-r\,\mathrm{sign}(\rho)}{1+r\lvert a_{CJM}\rvert}\left[a_{CJM}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right]$	$\frac{1}{1+r\lvert a_{CJM}\rvert}\left[a_{CJM}\hat{\boldsymbol{x}}-\hat{\boldsymbol{y}}\right]$	$\frac{\lvert\hat{\boldsymbol{x}}\rvert^2\lvert\hat{\boldsymbol{y}}\rvert^2\left(1-\rho^2\right)\left(r^2+1\right)}{\left\lvert\hat{\boldsymbol{x}}+\mathrm{sign}(\rho)\,r\hat{\boldsymbol{y}}\right\rvert^2}$
DE	Not Calculated	Not Calculated	Not Calculated

Table A-5. Additional Algorithm Results

Algorithm	Estimated Input $\boldsymbol{w}$	Special Case $\lvert\hat{\boldsymbol{x}}\rvert^2 = \lvert\hat{\boldsymbol{y}}\rvert^2 = 1,\ r = 1$	
		Estimated Gain	Total Weighted Noise Power $\lvert \boldsymbol{C}\boldsymbol{\varepsilon}_x \rvert^2 + \lvert \boldsymbol{C}\boldsymbol{\varepsilon}_y \rvert^2$
LSMS	$\frac{1}{1+r\lvert a_{LSMS}\rvert}\left(\boldsymbol{x}+r\,\mathrm{sign}(\rho)\,(\boldsymbol{y}-b\mathbf{1})\right)$	ρ	$2\frac{1-\lvert\rho\rvert}{1+\lvert\rho\rvert}$
TLS	$\frac{1}{1+a_{TLS}^2}\left[\boldsymbol{x}+a_{TLS}\,(\boldsymbol{y}-b\mathbf{1})\right]$	$\mathrm{sign}(\rho)$	$1-\lvert\rho\rvert$
WJM	$\frac{1}{1+r\lvert a_{WJM}\rvert}\left[\boldsymbol{x}+r\,\mathrm{sign}(\rho)(\boldsymbol{y}-b\mathbf{1})\right]$	$\frac{\mathrm{sign}(\rho)}{\sqrt{2(1+\lvert\rho\rvert)}-1}$	$2\left[2-\sqrt{2(1+\lvert\rho\rvert)}\right]$
CJM	$\frac{1}{1+r\lvert a_{CJM}\rvert}\left[\boldsymbol{x}+r\,\mathrm{sign}(\rho)(\boldsymbol{y}-b\mathbf{1})\right]$	$\mathrm{sign}(\rho)$	$1-\lvert\rho\rvert$
DE	No Estimate	$\mathrm{sign}(\rho)$	$1-\lvert\rho\rvert$