Recursive Least Squares with Linear Constraints — **Source link** ↗

X. Rong Li, Yunmin Zhu

Topics: Recursive least squares filter, Non-linear least squares, Total least squares, Iteratively reweighted least squares and Explained sum of squares

Related papers:

- Solving least squares problems

- Numerical Methods for Least Squares Problems

- Brief paper: A new approach to linear regression with multivariate splines

- Differential constraints for bounded recursive identification with multivariate splines

- A general form for recursive adaptive algorithms leading to an exact recursive CMA

RECURSIVE LEAST SQUARES WITH LINEAR CONSTRAINTS

YUNMIN ZHU* AND X. RONG LI†

Abstract. Recursive Least Squares (RLS) algorithms have wide-spread applications in many areas, such as real-time signal processing, control and communications. This paper shows that the unique solutions to linear-equality constrained and the unconstrained LS problems, respectively, always have exactly the same recursive form. Their only difference lies in the initial values. Based on this, a recursive algorithm for the linear-inequality constrained LS problem is developed. It is shown that these RLS solutions converge to the true parameter that satisfies the constraints as the data size increases. A simple and easily implementable initialization of the RLS algorithm is proposed. Its convergence to the exact LS solution and the true parameter is shown. The RLS algorithm, in a theoretically equivalent form by a simple modification, is shown to be robust in that the constraints are always guaranteed to be satisfied no matter how large the numerical errors are. Numerical examples are provided to demonstrate the validity of the above results.

1. Introduction. The least squares (LS) approach has wide-spread applications in many fields, such as statistics, numerical analysis, and engineering. Its greatest progress in the 20th century was the development of the recursive least squares (RLS) algorithm, which has made the LS method one of the few most important and widely used approaches for real-time applications in such areas as signal and data processing, communications and control systems. Considerable efforts and significant achievements have been made in developing even more efficient RLS algorithms.

In many practical problems to which the LS method is applicable, the solutions have to satisfy certain constraints. For this reason, the study of constrained LS problems has received considerable attention. For example, four chapters of the book [25] were dedicated to the LS problems with linear constraints. Various algorithms have been developed for these constrained LS problems. To our knowledge, however, recursive (and thus computationally efficient) LS algorithms for constrained problems were not available prior to the work presented here.

In this paper, we first show that the solution to the linear LS problems with linear-equality constraints and the unconstrained linear RLS solution have an identical

recursive form. The two solutions differ only in that they correspond to different initial conditions (initial solutions). Based on the above RLS solution for linear-equality constraints, a recursive algorithm for the linear LS problems with linear-inequality constraints is presented. Then, an easily implementable RLS algorithm for the linear LS problems with linear constraints is proposed, along with its convergence to the exact RLS solution. Both the exact and inexact RLS algorithms are shown to converge to the true parameter under some assumptions. A numerically robust RLS algorithm is presented which ensures that the constraints are satisfied even in the presence of numerical errors. These results enable a solution to the linear LS problems with linear constraints that is highly efficient and robust for real-time applications.

2. Problem Statement and Motivation. In many scientific and engineering problems one has to solve the following minimization problem — the unconstrained Least Squares (LS) problem:

$$(1) \qquad \min_{\theta} \quad S_n(\theta) = \sum_{i=1}^{n} |y_i - \theta^* \mathbf{x}_i|^2 = (Y_n - \theta^* X_n)(Y_n - \theta^* X_n)^*,$$

where $y_i \in C^1$ is a scalar; $\mathbf{x}_i \in C^r$ and the estimated parameter $\theta \in C^r$ are r-dimensional vectors; C^1 and C^r denote complex number and complex vector space of r dimension, respectively; the superscript " $*$ " stands for complex conjugate and transpose; and Y_n and X_n are defined as

$$(2) \qquad Y_n = [y_1, y_2, \ldots, y_n] \in C^{1 \times n}, \qquad X_n = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in C^{r \times n}.$$

This optimization problem can be found in various fields. Consider, for example, a typical adaptive filtering problem described as follows [23]: Given a sequence of input signals $\{x_i\}$, and a desired response sequence $\{y_i\}$, the objective is to identify a linear FIR (finite impulse response) filter with coefficient vector θ of length r, such that the filter's response to the input sequence $\{x_i\}$ is in some sense close to the desired response $\{y_i\}$. Often the criterion for closeness is given by the sum of squares of the differences between the actual and desired responses at different time instants. In this case the problem formulation reduces to the above minimization problem, where y_i is the desired response at time i, $\mathbf{x}_i = [x_i, x_{i-1}, ..., x_{i+1-r}]^*$ and θ is the vector of tap weights of the FIR filter.

In practice, it is often necessary to impose constraints on a LS solution. For example, in curve fitting, inequality constraints may arise from such requirements as monotonicity, nonnegativity, and convexity. Equality constraints may arise from the need to guarantee continuity (and possibly smoothness) of the curves. Such constrained optimization problems can be found in many application areas. For example, in some adaptive filtering applications, such as minimum-variance distortionless response filtering, it is desirable to design a linear filter that minimizes its average output power

while maintaining its response to some specific frequencies of interest to be constant. A spatial counterpart of the above temporal filtering problem is the adaptive beamforming, which ensures that the beamformer θ provides a constant gain $A_1\theta = B_1$ along the direction A_1^* of the desired signal and simultaneously nullify the direction A_2^* of the interference (i.e., $A_2\theta = 0$), where $(A_1 \; A_2)' = A$ and $(B_1 \; B_2)' = B$ (for details see e.g., [34, 15, 23]). One popular class of such constraints is linear-equality constraints; that is, the solution θ to (1) has to satisfy the following system of linear algebraic equations

$$(3) \qquad\qquad\qquad\qquad A\theta = B,$$

where $A \in C^{d \times r}$ and $B \in C^d$ are given matrices. Following [25], we will refer to this LS problem with linear-equality constraints as the **LSE problem**.

When all of the above matrix spaces are restricted on the corresponding real spaces, another popular constraint is linear-inequality constraint

$$(4) \qquad\qquad\qquad\qquad A\theta \geq B,$$

where the inequality "$\geq$" between two matrices is componentwise — the inequality holds for each element. We will refer to this LS problem with linear-inequality constraints as the **LSI problem**. For example, in many estimation applications, it is known that θ is nonnegative from its physical meaning. In the above beamforming example, when a new interference comes in, such as in mobile communications, one may have to change the original weight vector to reject it as much as possible and at the same time do not make the original beam pattern changed too much. In this case, one may optimize the weight vector θ so that the gain can change in a suitable range. The preceding problem formulation is a linear-inequality constrained LS problem.

In fact, when $A = 0$, $B = 0$, constraints (3) and (4) vanish. In other words, the unconstrained LS problem is a special case of the constrained LS problem with the constraints [(3) or (4)] relaxed.

3. Previous Results. The solution of the above unconstrained LS problem for θ using the data y_i and $\mathbf{x}_i$ through time n will be denoted by θ_n. It is well-known that the unique minimum-norm LS solution to the unconstrained problem (1) is

$$(5) \qquad\qquad\qquad\qquad \theta_n = (X_n^*)^+ Y_n^*$$

where superscript "+" stands for the Moore-Penrose inverse. The unconstrained problem (1) has a unique solution (i.e., the minimum-norm solution (5) becomes the unique solution), given by

$$(6) \qquad\qquad\qquad\qquad \theta_n = (X_n X_n^*)^{-1} X_n Y_n^*$$

if and only if $X_n X_n^*$ is nonsingular. This is a batch solution and is thus not suitable for real-time applications because its computational complexity increases with n.

Important progress was made half a century ago when Plackett [29] and others (such as Woodbury [36]) demonstrated that if $X_n X_n^*$ is nonsingular, the unconstrained problem (1) has the following recursive solution

$$(7) \qquad \theta_{n+1} \; = \; \theta_n + K_{n+1}(y_{n+1} - \theta_n^* \mathbf{x}_{n+1})^*,$$

$$(8) \qquad K_{n+1} \; = \; P_n \mathbf{x}_{n+1}/(1 + \mathbf{x}_{n+1}^* P_n \mathbf{x}_{n+1}),$$

$$(9) \qquad P_{n+1} \stackrel{\text{def}}{=} (X_{n+1} X_{n+1}^*)^{-1} = (P_n^{-1} + \mathbf{x}_{n+1} \mathbf{x}_{n+1}^*)^{-1} = (I - K_{n+1} \mathbf{x}_{n+1}^*) P_n,$$

where the last equality in (9) follows from the Matrix Inversion Lemma [3, 4]:

$$(10) \qquad (A + B D^{-1} B^*)^{-1} = A^{-1} - A^{-1} B (D + B^* A^{-1} B)^{-1} B^* A^{-1},$$

where A and D are both Hermitian positive definite matrices. The recursive equation (9) plays a crucial role in the recursion (7)–(9), and generally, when $X_n X_n^*$ is singular, no recursion similar to (7)–(9) is available.

This recursive LS (RLS) solution greatly promotes the application of LS method in many fields where real-time processing plays a key role, including signal processing, communications and control (see, e.g., [17, 7, 6, 27, 23, 26]). Compared with the batch solution (6), the recursive solution (7)–(9) offers important advantages: It is in a recursive form, free of matrix inversion operation, and thus has a much reduced computational complexity. As such, it is much more suitable for real-time applications because the number of algebraic operations and required memory locations is reduced from $O(r^3)$ (order of r^3) to $O(r^2)$ per cycle (i.e., per new data point). Moreover, even more efficient algorithms (known as "fast algorithms") have been developed based on recursion (7)–(9), such as [28, 10], which further reduce the computational complexity to $O(r)$ by taking advantage of some properties (e.g., Toeplitz) of the matrices involved. A major drawback of these fast algorithms is that they have poor numerical properties. This drawback has been eliminated in the numerically robust fast algorithms developed more recently, such as [30, 9, 8], based on QR decomposition with Givens rotations or the Householder transformation. Mapping of RLS algorithms onto (e.g., systolic) array processors to further reduce the latency in the processing has received considerable attention in recent years (see, e.g., [23]).

The practical importance of developing efficient algorithms for solving LS problems has been and will continue to be the thrust for the past and future research efforts. The progress in the study of constrained LS problems has been less satisfactory. The most popular method in engineering for the LSE problem [i.e., with linear-equality constraint (3)] is based on the use of Lagrange multipliers. It converts

the constrained problem to an unconstrained problem of a higher dimension. Without any additional assumption on matrix A, however, it can provide in general only necessary conditions for the solution, even when $X_n X_n^*$ is nonsingular.

By using an orthogonal basis of the null space of A, the LSE problem has been shown [21, 25] to have a unique solution

$$(11) \qquad \theta_n = A^+ B + (X_n^* P)^+ (Y_n^* - X_n^* A^+ B),$$

if and only if $\binom{A}{X_n^*}$ has full rank r and (3) is consistent, otherwise (11) is the unique solution of minimum norm.

The LSE problem can also be solved by direct elimination [25, 5]: Express some elements of θ_n in terms of the remaining elements using the linear-equality constraint (3) if it is consistent and underdetermined. Then the constrained problem is reduced to an unconstrained LS problem for the remaining elements. In the weighting method [25, 5, 16, 1], the original LS cost function is augmented by an additional one with a large weight that corresponds to the LS formulation of the linear constraints. This formulation is equivalent to the original one only when the weight approaches infinity, which leads to numerical problems. Another method is based on unconstrained RLS algorithms combined with projections onto the constrained sets, such as in [14, 23], which has a heavy computational burden. Other methods include the use of generalized singular value decomposition, updating [4, 5], QR decomposition and other numerical method based algorithms (see, e.g., [12, 2, 13, 24, 33, 19, 11, 18, 5, 23]), which are nonrecursive in nature (albeit iterative for some of them) and thus are less attractive computationally.

A variety of methods have been developed for solving the LS problem with linear-inequality constraints numerically (see, e.g., [32, 20, 22, 5]). Most of them are based on constrained quadratic programming, including the dual and primal methods, in particular, the active set method [5].

None of the above algorithms for constrained LS problems is recursive in nature and thus they are not suitable for real-time applications. To our knowledge, the feasible directions based numerical algorithm of [31] was the only one that utilizes the old solution to compute the new solution when new data are available, where the old feasible tableau is used to obtained the new tableau by a procedure (but the procedure to find the LS solution from the tableau is still not recursive). It is thus important and natural to look for constrained LS solutions that can be calculated *recursively*. It would be ideal if the recursion were identical to (7)–(9) since were this the case, it would facilitate further development greatly — for example, some of the fast RLS algorithms developed for the unconstrained LS problems might be applicable to constrained LS problems with little modification and thus the computational complexity

could be reduced to $O(r)$ per cycle immediately.

In this paper, we show in Section 4 that the solution to the LSE problem (i.e., with linear-equality constraints) indeed can be calculated in exactly the same recursion as (7)–(9) for the unconstrained problem. The only difference between the constrained and unconstrained RLS solutions lies in their initial values. Section 5 presents a robust RLS solution that is guaranteed to satisfy the constraints no matter how large the computational errors of the recursive quantities are. Since the LS solution subject to a linear-inequality constraint is either the corresponding unconstrained LS solution (when it is inside the constraint set) or the corresponding linear-equality LS solution (if on the boundary of the constraint set), the above recursive solutions are extended in Section 6 to linear-inequality constraint case. This extension takes advantage of the fact that the LSE and unconstrained solutions have the same recursion. We propose a simple initialization for the RLS algorithm for linear constraints in Section 7, and show in Section 8 that the difference between the RLS algorithm initialized this way and the exact RLS solution converges to zero as more and more data are used. We also show that the exact and inexact RLS algorithms converge, under some assumptions, to the true parameter if it satisfies the constraints. Numerical examples that support the theoretical results are given in Section 9. Various extensions of the results presented are discussed in Section 10. Conclusions are provided in Section 11 and proofs of all theorems are given in the Appendix.

4. Linear-Equality Constraints. Throughout this section we will not consider the two trivial cases where (3) has either a unique solution or no solution, since in these two cases, the LSE problem has the same unique solution or no solution, respectively. In other words, we assume that (3) is consistent and underdetermined. Hence, by the MP inverse theory [3], we have

$$(12) \qquad AA^+B = B, \quad A^+A \neq I,$$

and the general solution to (3) is given by

$$(13) \qquad \theta = A^+B + (I - A^+A)\xi,$$

where ξ is an arbitrary vector in C^r. Let

$$(14) \qquad P = I - A^+A.$$

Clearly, P is an orthogonal projector (since $P^2 = P = P^*$) and not of full rank provided $A \neq 0$.

By definition, the solution to the LSE problem is the vector θ satisfying the constraints (3) that minimizes the cost function $S_n(\theta)$ of (1); that is, it is one of the solutions given by (13) that minimizes (1). This idea leads to the following theorem

in which a general solution to the LSE problem is given without any assumption on X_n or Y_n.

THEOREM 1. *The general solution to problem (1) subject to (3) is given by*

$$(15) \qquad \theta_n = A^+B + (PX_nX_n^*P)^+X_n(Y_n^* - X_n^*A^+B) + P\xi,$$

where $\xi \in C^r$ is an arbitrary vector satisfying

$$(16) \qquad\qquad\qquad X_n^*P\xi = 0.$$

The proofs of this theorem and all those that follow can be found in the Appendix.

In many applications of the LS method, such as LS estimation, a unique solution to the LS problem is essential because for instance, there is no reason to believe that estimation error associated with a nonunique LS solution is small. A corollary regarding unique LS solutions to the LSE problem follows from Theorem 1.

COROLLARY 1. *The LS problem (1) subject to (3) has a unique solution, given by*

$$(17) \qquad \theta_n = A^+B + (PX_nX_n^*P)^+X_n(Y_n^* - X_n^*A^+B),$$

if and only if $\left(\begin{smallmatrix}A\\X_n^*\end{smallmatrix}\right)$ has full column rank r.

Note that if $X_nX_n^*$ is nonsingular then $\left(\begin{smallmatrix}A\\X_n^*\end{smallmatrix}\right)$ has full rank r but the converse is not true in general. In addition, if $A = 0$ and $B = 0$, then $P = I$, and (17) reduces to the unconstrained solution (6). If $\left(\begin{smallmatrix}A\\X_n^*\end{smallmatrix}\right)$ does not have full rank, then (17) is the minimum-norm solution. This is clear from its equivalence to (11), shown below.

$$(PX_nX_n^*P)^+P = [(PX_nX_n^*P)^*(PX_nX_n^*P)]^+(PX_nX_n^*P)^*P = (PX_nX_n^*P)^+$$
$$(X_n^*P)^+ = [(X_n^*P)^*(X_n^*P)]^+(X_n^*P)^* = (PX_nX_n^*P)^+X_n$$

Thus (17) and (11) are equivalent. However, it will become clear later that (17) is in a form more suitable for deriving its equivalent recursive form.

More usefully, the following theorem states that the unique LSE solution can always be written recursively in a form identical to the recursion (7)–(9) for the unconstrained RLS solution.

THEOREM 2. *The unique solution θ_n in (17) can always be written recursively as*

$$(18) \qquad\qquad \theta_{n+1} = \theta_n + K_{n+1}(y_{n+1} - \theta_n^*\mathbf{x}_{n+1})^*$$

$$(19) \qquad\qquad K_{n+1} = P_n\mathbf{x}_{n+1}/(1 + \mathbf{x}_{n+1}^*P_n\mathbf{x}_{n+1}),$$

$$(20) \qquad\qquad P_{n+1} \overset{\text{def}}{=} (PX_{n+1}X_{n+1}^*P)^+ = (I - K_{n+1}\mathbf{x}_{n+1}^*)P_n.$$

The only difference between unconstrained and linear-equality constrained RLS solutions lies in their initial values. If the initial values are $\theta_{n_0} = (X_{n_0}X_{n_0}^*)^{-1}X_{n_0}Y_{n_0}^*$

and $P_{n_0} = (X_{n_0} X_{n_0}^*)^{-1}$, θ_n in (7)–(9) is an unconstrained LS solution. When the initial values are $\theta_{n_0} = A^+ B + (PX_{n_0} X_{n_0}^* P)^+ X_{n_0} (Y_{n_0}^* - X_{n_0}^* A^+ B)$ and $P_{n_0} = (PX_{n_0} X_{n_0}^* P)^+$, it is a linear-equality constrained LS solution.

Theorem 2 is the most important result of this paper. Note that the two matrices $(X_{n+1} X_{n+1}^*)^{-1}$ and $(PX_{n+1} X_{n+1}^* P)^+$ in the two recursions or their initial values can never be equal provided $A \neq 0$ since P is not of full rank. As a result, the two RLS solutions are in general different even if they start from the same initial θ_{n_0}.

It is well known that the unconstrained LS problem has a unique solution if and only if $X_n X_n^*$ is nonsingular. Not surprisingly, the state x_0 of the linear system

$$x_{k+1} = F_k x_k + G_k u_k$$
$$y_k = H_k x_k + I_k u_k$$

can be uniquely determined from finitely many output y_k's and input u_k's (i.e., is *observable*) if and only if $X_n X_n^*$ is nonsingular, where $X_n = [(H_0 \Phi_{0,0})^*, (H_1 \Phi_{1,0})^*, \ldots, (H_n \Phi_{n,0})^*]$ and $\Phi_{k,0} = F_{k-1} F_{k-2} \cdots F_0$ for $k > 0$ and $\Phi_{k,0} = I$ for $k \leq 0$. As such, the nonsingularity of $X_n X_n^*$ is also the condition for the observability of the above system. The above results imply that if it is known that the state x_0 satisfies linear-equality constraints $Ax_0 = B$, then the observability of the system can be relaxed to $\binom{A}{X_n^*}$ having full rank. This concept has important applications in e.g., power system state estimation (see e.g., [37]), where virtual measurements are introduced when the system is unobservable without constraints. These virtual measurements are obtained from various circuit laws and are actually equality constraints. The above results explains why the introduction of virtual measurements may make the system observable.

To use the recursion (18)–(20), we need to make sure $\binom{A}{X_n^*}$ has full rank r. Note that the sum of n dyads of $(r \times r)$-dimension has a rank no greater than $\min(r, n)$ and

$$[A^* \ X_n] \begin{bmatrix} A \\ X_n^* \end{bmatrix} = A^* A + X_n X_n^* = A^* A + \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^*.$$

Let m be the rank of A. As such, $n = r - m > 0$ is the smallest n for which $\binom{A}{X_n^*}$ may have full rank r. On the other hand, the columns of X_n are usually linearly independent in practice. Consequently, the first time we need to check whether $\binom{A}{X_n^*}$ has full rank or not is when $r - m$ data points are collected, and the exact recursion (18)–(20) can usually be started at this time.

5. Robust RLS Solution to LSE Problem. A recursive solution to the LSE problem was presented in the previous section. It is well-known that recursive algorithms are often vulnerable to numerical errors due to truncation and/or rounding off.

For the LSE problem, it is important to guarantee that the recursive solution always satisfies the linear constraints even in the presence of numerical errors. We derive below a robust RLS solution for the LSE problem that provides such a guarantee.

If θ_n is in the equality-constraint set, given by (13), it can always be written as

$$\theta_n = A^+ B + P \xi_n.$$

By (19), (20) and (A.11), we have $PK_{n+1} = K_{n+1}$. Hence, θ_{n+1} of (18) can be rewritten as

$$(21) \qquad \begin{aligned} \theta_{n+1} &= \theta_n + PK_{n+1}(y_{n+1} - \theta_n^* \mathbf{x}_{n+1})^* \\ &= A^+ B + P[\xi_n + K_{n+1}(y_{n+1} - \theta_n^* \mathbf{x}_{n+1})^*]. \end{aligned}$$

Note that numerical errors may render $PK_{n+1} \neq K_{n+1}$ and thus there is no guarantee for θ_{n+1} of (18) to remain in the constraint set given by (13). However, (22) indicates that if (18) of the recursion (18)–(20) is replaced by

$$(22) \qquad \theta_{n+1} = \theta_n + P[K_{n+1}(y_{n+1} - \theta_n^* \mathbf{x}_{n+1})^*]$$

then θ_{n+1} will always satisfy the linear constraints $A\theta_{n+1} = B$ no matter how large the numerical errors are, provided $A^+ B$ and P are calculated accurately, which is done off-line without any recursion.

6. Linear-Inequality Constraints. In this section, we apply and extend the results of Section 4 to give a recursive solution to the linear LS problem (1) subject to linear-inequality constraints (4) (i.e., the LSI problem).

Throughout this section we assume that all of the complex vector and matrix spaces in Section 4 are replaced by the corresponding real spaces, and the superscript "*" denotes transpose only. Furthermore, without loss of generality, by the definition of "$\geq$" we assume $\mathbf{x}_i \in R^{r \times 1}$, $y_i \in R$, $\theta \in R^{r \times 1}$, $A \in R^{d \times r}$, $B \in R^{d \times 1}$.

Applying quadratic programming theory to the LSI problem (see, e.g., ch. 10 of [35]), we have the following two propositions.

PROPOSITION 1. *If $X_n X_n^*$ is nonsingular, the LSI problem possesses a unique solution.*

PROPOSITION 2. *If $X_n X_n^*$ is nonsingular, the solution to the LSI problem is either the unconstrained LS solution or the solution to some linear-equality constrained LS problem related to (4) (for details, see ch. 10 (p. 245) of [35] and (23), (24) below).*

Note that under the nonsingularity assumption for $X_n X_n^*$, (a) both the unconstrained LS and LSE problems always have unique recursive solutions; (b) these recursive solutions possess the same form, given by (7)–(9) or (18)–(20); (c) the only difference between the solutions of the two LS problems lies in their initial values. In view of these, we describe below a recursive algorithm for solving the LSI problem, which was inspired by the active set method for constrained optimization.

A Recursive Solution to LSI Problem. Let A_j and B_j denote the jth row vectors of A and B, respectively, $j = 1, ..., d$. Rewrite constraint (4) as

$$(23) \qquad\qquad A_j\theta \geq B_j, j = 1, ..., d.$$

The corresponding equality constraints are

$$(24) \qquad\qquad A_j\theta = B_j, j = 1, ..., d.$$

From the above d equalities we have 2^d distinct combinations, including the case where no equality is selected, which correspond to all the combinations of the possible equality constraints selected from the d equality constraints plus the unconstrained case. Suppose that $X_n X_n^*$, $n \geq n_0$, is nonsingular. Then, the linear-inequality constrained RLS solution can be described as follows.

Step I (Initialization): Using all given data through time n_0, (6) and (9), (17) and (20), calculate all of the above 2^d different constrained and unconstrained LS solutions $\theta_{n_0}^i, P_{n_0}^i, i = 1, ..., 2^d$, by (17) and (6), respectively, as a group of initial values. Note that many of the $2^d - 1$ distinct equality-constraint sets may be empty, have a single element or be totally outside the original inequality-constraint set (23), which can be determined prior to receiving data. There is no need to calculate the associated LS solutions for the empty sets and for sets not satisfying (23). Assume there are s single-element sets, e empty sets, and l sets not satisfying (23).

Step II (Recursive Computation): Calculate $\theta_{n+1}^i, K_{n+1}^i, P_{n+1}^i, i = 1, ..., 2^d - e - l - s$, simultaneously by recursion (7)–(9) or (18)–(20) for each $n > n_0$.

Step III (Output): Select a unique θ_m that satisfies (23) and yields the smallest $S_m(\theta_m)$ in (1) of all $\theta_m^i, i \in \{1, ..., 2^d - e - l\}$, as the LS solution of (1) subject to (23). Here m denotes the time instants at which an (optimal) solution is needed for the LSI problem in practice.

Obviously, when the number of inequality constraints d is small, the above RLS solution to (1) subject to (23) offers significant advantages. In addition, it is particularly suitable for parallel processing.

7. Initialization of Constrained RLS Algorithms. In the above, we have derived the exact recursive solutions to the linear LS problem subject to (3) and (4), respectively. Only at time n_0 when $\binom{A}{X_n^*}$ has become of full rank, can they be initialized by the initial value $P_{n_0} = (PX_{n_0}X_{n_0}^*P)^+$ and θ_{n_0} of (17). In practice this is undesirable or even unacceptable.

Instead, we propose below a simple and easily implementable initialization scheme. Toward this end, let

$$\tilde{X}_n \stackrel{\text{def}}{=} [X_0, \quad X_n], \quad \tilde{Y}_n \stackrel{\text{def}}{=} [Y_0, \quad Y_n]$$

and consider the following modified LS problem

$$(25) \qquad \min_{\tilde{\theta}} \quad \tilde{S}_n = (\tilde{Y}_n - \tilde{\theta}^* \tilde{X}_n)(\tilde{Y}_n - \tilde{\theta}^* \tilde{X}_n)^*$$
$$= (Y_0 - \tilde{\theta}^* X_0)(Y_0 - \tilde{\theta}^* X_0)^* + (Y_n - \tilde{\theta}^* X_n)(Y_n - \tilde{\theta}^* X_n)^*,$$

subject to the linear constraints (3), where X_0 and Y_0 are to be determined. Denote by $\tilde{\theta}_n$ the solution to this modified problem using data through time n. Clearly, $\tilde{\theta}_n$ is in general not the exact recursive solution θ_n to the original LS problem (1). Nevertheless, it can be shown that under some mild conditions on X_n, $\tilde{S}_n$ converges to S_n and $\tilde{\theta}_n$ converges to θ_n as n increases. This is understandable because the first term in (26) is fixed while the second term is actually a sum of n dyads.

The trick with the initialization to be proposed is the following. Note that X_0 and Y_0 are up for us to choose. We could choose X_0 such that $X_0 X_0^*$ is nonsingular and thus the recursion (20) can be correctly initialized by $P_0 = (P X_0 X_0^* P)^+$. We could also choose Y_0 such that the initial $\tilde{\theta}_0$ is in the form of (17), given by

$$(26) \qquad \tilde{\theta}_0 = A^+ B + (P X_0 X_0^* P)^+ X_0 (Y_0^* - X_0^* A^+ B) = A^+ B + P\tilde{\xi},$$

where use has been made of (A.11) and $\tilde{\xi}$ depends on Y_0. As such, *$\tilde{\theta}_n$ can be obtained by the recursion (18)–(20) from the first time on.* Specifically, inspired by the following commonly used simple initialization for the unconstrained RLS algorithms (see, e.g., [23])

$$\theta_0 = 0, \quad P_0 = \alpha^{-1} I, \quad \alpha > 0$$

we propose the following simple initialization of the RLS algorithms for the LSE problems

$$(27) \quad P_0 = (P X_0 X_0^* P)^+ = (PRP)^+, \quad \text{for any Hermitian positive definite matrix } R,$$
$$(28) \quad \tilde{\theta}_0 = A^+ B + P\tilde{\xi}, \quad \text{for any vector } \tilde{\xi} \text{ of suitable dimension,}$$

For simplicity, we may choose e.g.,

$$(29) \qquad\qquad\qquad X_0 X_0^* = R = \alpha I, \quad \alpha > 0,$$

Clearly, $X_0 X_0^*$ is nonsingular. By (27), (29) and $P^2 = P$ (since P is a projector), we have $P_0^+ = \alpha P, P_0 = \alpha^{-1} P^+$.

For the LSI problem, we choose at most 2^d initial values of the above forms (as mentioned in Sec. 6, Step I), where $A^{(i)}$ and $B^{(i)}$ ($i = 1, ..., 2^d$) consist of one of the 2^d different combinations selected from the d equalities in (24). We calculate P_n^i and $\tilde{\theta}_n^i$, $i = 1, ..., 2^d$, by (18)–(20) using the above 2^d initial values for the equality constraints (24). Then at each time n (or only at some required time m), the solution to the LSI

problem is obtained by the one of $\tilde{\theta}_n^i$ that satisfies (23) and yields the smallest $\tilde{S}_n(\tilde{\theta}_n^i)$ in (26) of all $\tilde{\theta}_n^i$. To do so recursively, a recursion for $\tilde{S}_n(\tilde{\theta}_n^i)$ is required. This can be derived as follows. Let

$$(30) \qquad R_n = \tilde{X}_n \tilde{X}_n^* \in R^{r \times r}, \quad l_n = \tilde{X}_n \tilde{Y}_n^* \in R^{r \times 1}.$$

Obviously,

$$(31) \qquad R_n = R_{n-1} + \mathbf{x}_n \mathbf{x}_n^*, \quad l_n = l_{n-1} + y_n \mathbf{x}_n.$$

Thus, using (1) and (30), we have

$$\tilde{S}_n(\tilde{\theta}_n^i) = (\tilde{Y}_n - \tilde{\theta}_n^{i*} \tilde{X}_n)(\tilde{Y}_n^* - \tilde{X}_n^* \tilde{\theta}_n^i) = \tilde{Y}_n \tilde{Y}_n^* - 2\tilde{\theta}_n^{i*} \tilde{X}_n \tilde{Y}_n^* + \tilde{\theta}_n^{i*} \tilde{X}_n \tilde{X}_n^* \tilde{\theta}_n^i$$

$$(32) \qquad = \tilde{Y}_n \tilde{Y}_n^* - 2\tilde{\theta}_n^{i*}(l_{n-1} + y_n \mathbf{x}_n) + \tilde{\theta}_n^{i*}(R_{n-1} + \mathbf{x}_n \mathbf{x}_n^*) \tilde{\theta}_n^i.$$

In fact, since $\tilde{Y}_n \tilde{Y}_n^*$ does not depend on $\tilde{\theta}_n^i$, to yield a smallest $\tilde{S}_n(\tilde{\theta}_n^i)$ at each recursive step, only the follow needs to be evaluated

$$\tilde{\theta}_n^{i*}(R_{n-1} + \mathbf{x}_n \mathbf{x}_n^*) \tilde{\theta}_n^i - 2\tilde{\theta}_n^{i*}(l_{n-1} + y_n \mathbf{x}_n).$$

The nice thing about this initialization for the linear-constrained problems is that it is simple for implementation, it starts from the first time, and it converges to the exact solution and the true parameter, as shown below.

In the sequel, $\tilde{\theta}_n$ will be referred to as either the inexact RLS solution or simply the RLS algorithm and θ_n as the (exact) RLS solution.

8. Convergence of Constrained RLS Algorithms. As the data size n increases, there are three types of convergence for the LSE problem: (a) the inexact LS solution $\tilde{\theta}_n$ converges to the exact solution θ_n; (b) θ_n converges to the true parameter; (c) $\tilde{\theta}_n$ converges to the true parameter. Likewise for the LSI problem. We show below, for both the LSE and LSI problems, the convergence of $\tilde{\theta}_n$ to θ_n without any assumption on the relation between Y_n and X_n, and the convergence of $\tilde{\theta}_n$ and θ_n to the true parameter for an extremely popular linear model for Y_n and X_n.

Consider the convergence of $\tilde{\theta}_n$ to θ_n first. Note that $\tilde{X}_n \tilde{X}_n^* = X_0 X_0^* + X_n X_n^*$. By Theorem 2, the solution $\tilde{\theta}_n$ to (26) with the initialization (26) and (29) is given by

$$(33)$$
$$
\begin{aligned}
\tilde{\theta}_n &= A^+ B + [P\tilde{X}_n \tilde{X}_n^* P]^+ [\tilde{X}_n(\tilde{Y}_n^* - \tilde{X}_n^* A^+ B)] \\
&= A^+ B + [PX_0 X_0^* P + PX_n X_n^* P]^+ [X_0(Y_0^* - X_0^* A^+ B) + X_n(Y_n^* - X_n^* A^+ B)] \\
&= A^+ B + [P_0^+ + PX_n X_n^* P]^+ [PX_0(Y_0^* - X_0^* A^+ B) + X_n(Y_n^* - X_n^* A^+ B)].
\end{aligned}
$$

By (14), (A.1) and (A.11), we have $PA^+ B = 0, (PX_0 X_0^* P)(PX_0 X_0^* P)^+ X_0 = PX_0$. This and (27) lead to $P_0^+ \tilde{\theta}_0 = PRP\tilde{\theta}_0 = PX_0(Y_0^* - X_0^* A^+ B)$. Thus, (33) can be

written as

$$(34) \qquad \tilde{\theta}_n = A^+B + \left[\frac{1}{n}P_0^+ + \frac{1}{n}PX_nX_n^*P\right]^+ \left[\frac{1}{n}P_0^+\theta_0 + \frac{1}{n}X_n(Y_n^* - X_n^*A^+B)\right].$$

Note the following

$$
(35)
\begin{aligned}
&(\tfrac{1}{n}P_0^+ + \tfrac{1}{n}PX_nX_n^*P)^+(\tfrac{1}{n}P_0\theta_0 + \tfrac{1}{n}X_n(Y_n^* - X_n^*A^+B)) \\
&- (\tfrac{1}{n}PX_nX_n^*P)^+(\tfrac{1}{n}X_n(Y_n^* - X_n^*A^+B)) \\
&= (\tfrac{1}{n}P_0^+ + \tfrac{1}{n}PX_nX_n^*P)^+(\tfrac{1}{n}P_0\theta_0 + \tfrac{1}{n}X_n(Y_n^* - X_n^*A^+B)) \\
&- (\tfrac{1}{n}P_0^+ + \tfrac{1}{n}PX_nX_n^*P)^+(\tfrac{1}{n}X_n(Y_n^* - X_n^*A^+B)) \\
&+ (\tfrac{1}{n}P_0^+ + \tfrac{1}{n}PX_nX_n^*P)^+(\tfrac{1}{n}X_n(Y_n^* - X_n^*A^+B) \\
&- (\tfrac{1}{n}PX_nX_n^*P)^+(\tfrac{1}{n}X_n(Y_n^* - X_n^*A^+B)),
\end{aligned}
$$

The first difference on the right hand side of (35) converges to zero as $n \to \infty$ while the first inequality in Assumption A.1 of Theorem 4 below holds. From the first identity after (17) or (A.11) in the appendix it follows that $P_0^+ = (PRP)^+ = P(PRP)^+P$. By (A.17), we have $W^*P_0^+W = (W^*RW)^{-1}$. These two identities lead to

$$
(36)
\begin{aligned}
&(\tfrac{1}{n}P_0^+ + \tfrac{1}{n}PX_nX_n^*P)^+ = (P\tfrac{1}{n}P_0^+P + \tfrac{1}{n}PX_nX_n^*P)^+ \\
&= [P(P_0^+ + \tfrac{1}{n}X_nX_n^*)P]^+ = W[W^*(\tfrac{1}{n}P_0^+ + \tfrac{1}{n}X_nX_n^*)W]^{-1}W^* \\
&= W[\tfrac{1}{n}(W^*RW)^{-1} + \tfrac{1}{n}W^*X_nX_n^*W]^{-1}W^*.
\end{aligned}
$$

It follows from noting the second inequality in Assumption A.1 of Theorem 4 below and using matrix identity (10) to expand the inverse of the above square brackets that the second difference on the right hand side of (35) also tends to zero as $n \to \infty$. Hence, the difference between (17) and (33) is insignificant if n is large or P_0^+ is small (e.g., α in (29) is small). A by-product of the above analysis is that we should choose a small α value for the initialization.

Since the above RLS algorithm for the LSI problem is based on the RLS algorithm for the LSE problem, the difference between the above algorithm and the exact LS solution to the LSI problem should also tend to zero as n goes to infinity.

Note that the above convergence analysis is valid for all Y_0 (or $\tilde{\xi}$). However, the difference $\theta_n - \tilde{\theta}_n$ for finite n would usually not be the same if different Y_0's are used. It is better to choose some average value of Y_n as Y_0 if a priori information is available.

To show the convergence of the RLS algorithms when true parameter θ satisfies the linear constraints, we have the following two theorems.

THEOREM 3. *Suppose $\mathbf{x}_n$ and y_n are related by the following linear system*

$$(37) \qquad y_n = \theta^*x_n + e_n, \quad n = 1, 2, \dots ,$$

where θ satisfies the linear constraints (3) and e is some sequence. For the LSE problem, the unique exact RLS solution θ_n converges to the true parameter θ in (37)

satisfying the equality constraints (3) as n goes to infinity if and only if

$$(38) \qquad (\frac{1}{n}PX_nX_n^*P)^+ \frac{1}{n}X_nE_n^* \xrightarrow{n\to\infty} 0,$$

where $E_n = (e_1\ e_2\ \ldots\ e_n)$.

Clearly, the inexact solution $\tilde{\theta}_n$ (i.e., the RLS recursion (18)–(20) with the simple initialization) also converges to the true parameter θ in (37) satisfying the equality constraints (3) as n increases under the same condition for $\tilde{\theta}_n$ to converge to θ_n.

It is easy to see that the above condition for convergence is an extension of that for the unconstrained RLS algorithm (see, e.g., p. 177 of [27]).

THEOREM 4. *Suppose the following assumptions hold.*
A1. *There exists two constants c, $M > 0$ such that $\frac{1}{n}X_nX_n^* \geq cI, \forall n \geq k_0$ for some k_0 and $\frac{1}{n}X_n(Y_n^* - X_n^*A^+B) \leq MI\forall n$.*
A2. *$\frac{1}{n}X_nE_n^* \xrightarrow{n\to\infty} 0$, where E_n was defined in Theorem 3.*
A3. *The set of (or bounded by) the equality constraints is compact.*
Then for the LSI problem, both the inexact solution $\tilde{\theta}_n$ and the unique exact RLS solution θ_n converge to the true parameter θ in (37) satisfying the inequality constraints (4) as n goes to infinity.

In other words, both $\tilde{\theta}_n$ and θ_n as estimators of the true parameter θ are consistent for the linearly-constrained problem under the stated assumptions.

9. Numerical Examples. Before presenting examples, we emphasize first that the convergence we obtained in Section 8 is valid for every point under the stated assumptions. In the case of random data, this can be translated into the almost everywhere convergence if the convergence in the assumptions is also changed to almost everywhere convergence. As such, results of convergence in some average sense based on Monte Carlo simulations are not quite appropriate here since they are too weak. Instead, we should have the convergence for (almost) *every run.*

Example 1: Constrained Minimum Variance Response Filter Consider first an example of a constrained minimum variance response filter (see e.g., [23]). The input x_k and output z_k of this filter are related by $z_k = \theta^*x_k$, where θ is chosen to minimize the average output power (variance):

$$\min_{\theta}\quad P_z = \sum_{k=r}^{n}|z_k|^2 = \theta^*X_nX_n^*\theta \quad \text{s.t.}\quad A\theta = B$$

where

$$X_n = [\mathbf{x}_r, \mathbf{x}_{r+1}, \ldots, \mathbf{x}_n], \quad \mathbf{x}_i = [x_{i-r+1}, x_{i-r+2}, \ldots, x_i]^*$$
$$A = [\mathbf{s}_1, \ldots, \mathbf{s}_m, 0, \ldots, 0]^*, \quad 0 \leq m \leq r$$
$$B = [b_1, \ldots, b_m, 0, \ldots, 0]^*, \quad 0 \leq m \leq r$$
$$\mathbf{s}_i = [1, e^{j\omega_i'}, \ldots, e^{j\omega_i'(r-1)}]^*$$

This filter has a gain of b_i for component with frequency ω_i' of the input x_k if $b_i \neq 0$ while suppressing the other frequency components, in particular, frequencies ω_j' with $b_j = 0$.

Clearly, this problem corresponds to the LSE problem of (1) subject to (3) with $Y_n = 0$.

In the simulation, the input x_k's were generated by

$$x_k = \sum_{i=1}^{8} c_i e^{j\omega_i k} + e_k$$

where e is a standard Gaussian white sequence, $\omega_{1,2} = \pm\pi/2$, $\omega_{3,4} = \pm 11\pi/12$, $\omega_{5,6} = \pm\pi/4$, and $\omega_{7,8} = \pm\pi/3$, $c_{1,2} = 0.5$, $c_{3,4} = 1$, $c_{5,6} = 0.75$, and $c_{7,8} = 0.9$. The filter has order 12 and $m = 6$. It provides a (theoretically) distortionless response to $\omega_{1,2}' = \pm\pi/2$ and $\omega_{5,6}' = \pm\pi/4$ frequency components (i.e., $b_1 = b_2 = b_5 = b_6 = 1$) while suppressing the other components, especially $\omega_{3,4}' = \pm 11\pi/12$, and $\omega_{7,8}' = \pm\pi/3$ (i.e., $b_3 = b_4 = b_7 = b_8 = 0$).

The results presented are based on 10 arbitrarily chosen Monte Carlo runs for the following three filters:

- The unique batch LS (BLS) solution of (17), denoted by $\bar{\theta}_n$, where the unique minimum-norm solution of (15) was chosen before $\binom{A}{X_n^*}$ becomes full rank.
- The exact RLS solution θ_n of (18)–(20), which was initialized by the batch solution $\bar{\theta}_m$ at exactly the first time when $\binom{A}{X_m^*}$ becomes full rank.
- The RLS (IRLS) algorithm $\tilde{\theta}_n$ of (18)–(20), initialized at $n = 0$ by, according to (27) and (26) with $Y_0 = 0$,

$$\tilde{\theta}_0 = (I - \alpha^2 P)A^+B, \quad P_0 = (\alpha P)^+, \quad \alpha = 10^{-4}$$

Note that time $n = 1$ is the first time we have $x_1, \ldots, x_r$ such that $\mathbf{x}_r$ is available.

Fig. 1 shows the Euclidean distances $\|\theta_n - \tilde{\theta}_n\|$ and $\|\bar{\theta}_n - \tilde{\theta}_n\|$ as functions of time n. The distances $\|\bar{\theta}_n - \theta_n\|$ were also obtained (but not shown here). They are all smaller than 10^{-9}, which indicates that the RLS solution θ_n is quite robust numerically even when the robust version (22) was not used. The power spectra of the output of the three *adaptive* filters with time-varying tap weights $\bar{\theta}_n$, θ_n and $\tilde{\theta}_n$, respectively, are plotted in Fig. 2. The three curves overlap exactly, which indicates that the RLS algorithm converges to the exact solution. Table 1 gives their computational complexities. The following observations can be made. The recursive solution has a much lower computational burden than the batch solution; the RLS algorithm with the simple initialization converges to the exact solution quickly; and the constrained minimum variance response filter worked well, as expected. Note that this example was implemented in Matlab, which is particularly efficient for matrix computation

(and thus for the batch solution) but not for recursive computation (i.e., the RLS algorithm).
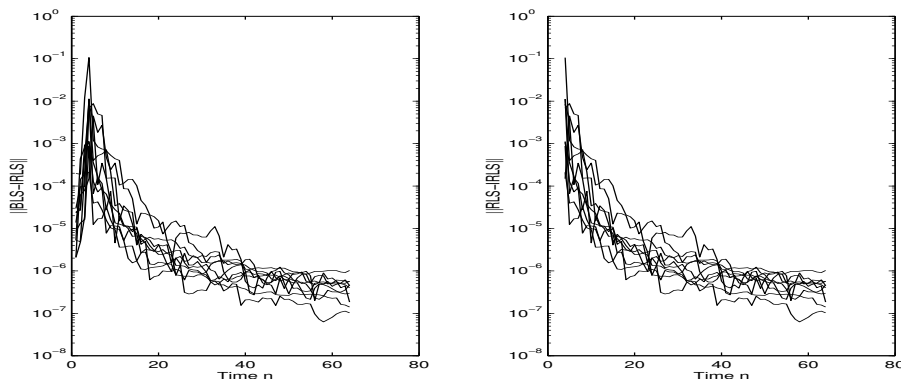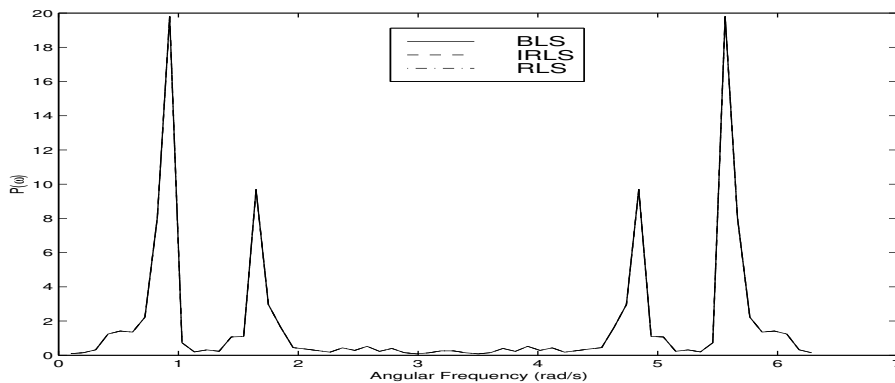

FIG. 1. *Distances among the LS solutions.*

FIG. 2. *Output power spectra of the three adaptive filters (they overlap).*

TABLE 1

Computational complexities of three LS algorithms for $n = 64$.

	Batch LS $\bar{\theta}_n$	Exact RLS θ_n	Inexact RLS $\tilde{\theta}_n$
FLOPs (in 10^6)	31.4	1.65	1.15
CPU Time (sec.)	0.64	0.12	0.038

Example 2: Convergence of Inequality-Constrained RLS Algorithm In this example, we show the convergence of the RLS algorithm with linear-inequality constraints no matter whether the true parameter θ satisfies $A\theta \geq B$ or not. The data $\mathbf{x}_n$ and y_n are generated by the following linear system

$$y_n = \theta^* \mathbf{x}_n + e_n, \quad n = 1, 2, \dots \, ,$$

The following were used: $r = 3$, $\theta^{(1)} = (1.5, \ -1, \ 0.1)'$ for the first case such that $A\theta^{(1)} = (6.6, \ 4.2)' \geq B$ and $\theta^{(2)} = (-3, \ 2 \ 2)'$ for the second case such that $A\theta^{(2)} = (-11, \ -4)' < B$, where $A = \begin{pmatrix} 5 & 1 & 1 \\ 2 & -1 & 2 \end{pmatrix}$, $B = (5, \ 1)'$. The sequences $\mathbf{x}_n$ and e_n are still i.i.d. zero-mean Gaussian processes as in the previous examples. Hence, assumptions **A1** and **A2** of Theorem 4 hold, as explained before.

Figs. 3 and 4 show the results for the two cases of true parameters $\theta^{(1)}$ and $\theta^{(2)}$, respectively. The three different lines in the left subplots represent the three elements of the vector $\tilde{\theta}_n$, respectively, which can be seen to converge quickly to the true parameters 1.5, -1, 0.1, respectively, for the first case $\theta = \theta^{(1)}$. In the right subplots, the solid and dashed lines are the two elements of vector $A\tilde{\theta}_n$, respectively. They are always above the two elements of vector B (i.e., the dotted lines $y = 1$ and $y = 5$), respectively. This indicates that the inequality constraints $A\tilde{\theta}_n \geq B$ are always satisfied.
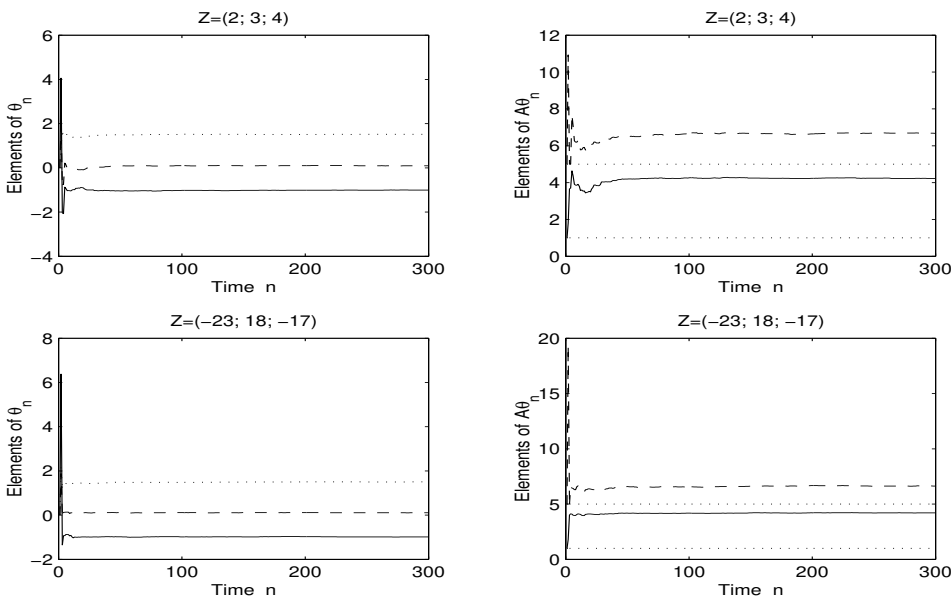

FIG. 3. *Convergence and robustness of inequality-constrained RLS algorithm with $A\theta \geq B$.*

In Fig. 4, although the true parameter $\theta^{(2)}$ does not satisfy $A\theta^{(2)} \geq B$, the algorithm $\tilde{\theta}_n$ still converge to the LS solution θ_n subject to the constraints $A\theta \geq B$, which however, does not converge to the true parameter $(-3, 2, 2)$ because the true parameter does not satisfy the constraints imposed. Since the two elements of $A\tilde{\theta}_n$ in Fig. 4 are never below the two elements (5 and 1) of B, respectively, the inequality constraints are always satisfied. In fact, one of the dotted lines $y = 5$ and a dashed line overlap completely (i.e., the solution reach the boundary — an equality constraint is satisfied), which indicates for this example that the LS cost function would be reduced
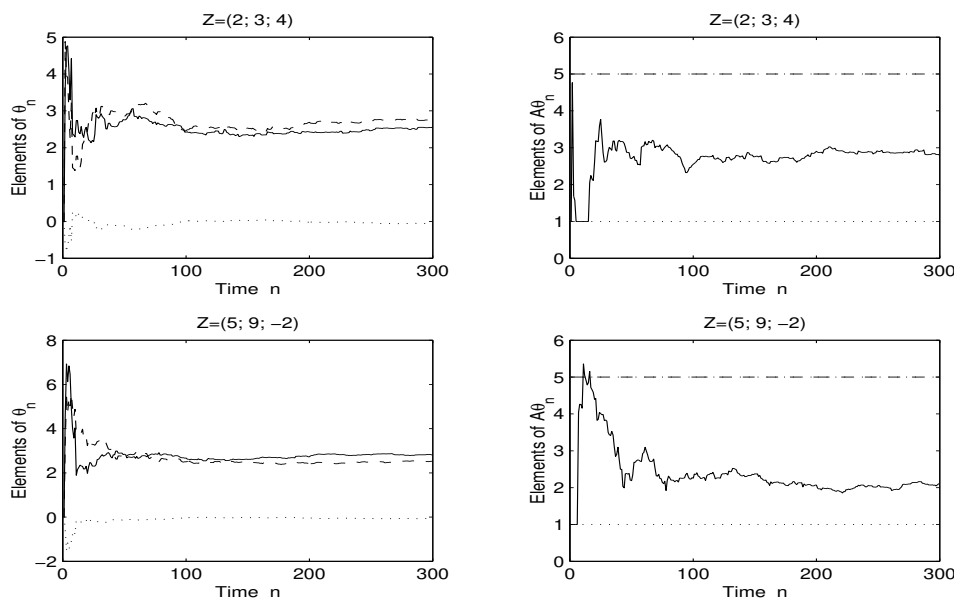
if this inequality constraint were relaxed.

FIG. 4. *Convergence of and robustness inequality-constrained RLS algorithm with $A\theta \not\geq B$.*

Thus, it is demonstrated that the inequality constrained RLS algorithm developed in this paper for different initial values is convergent and robust in the presence of computational errors.

10. Extensions. All the results presented above would also be valid if y_i is vector-valued and/or B in the linear constraints is a matrix rather than a vector.

It is also clear that the RLS solution to the LSI problem also works for the LS problems with both equality and inequality constraints.

Since the RLS solution to the LSE problem has the same recursion as that of the unconstrained RLS solution and $PX_nX_n^*P$ has similar properties (e.g., near Toeplitz) as those of $X_nX_n^*$, it can be expected that at least some of the fast algorithms (computationally efficient algorithms) developed for the unconstrained RLS solution are also valid for the RLS solution to the LSE problem. As a result, numerically robust fast algorithms (with $O(r)$ computational complexity) for the linearly-constrained LS problems may be readily available.

For the weighted LS (WLS) problems in which (1) is replaced by the following

$$\min_{\theta} \quad S_n(\theta) = (Y_n - \theta^* X_n)W(Y_n - \theta^* X_n)^* = (Y_nC - \theta^* X_nC)(Y_nC - \theta^* X_nC)^*,$$

where $W = CC^*$ is a positive definite weighting matrix, all the results of this paper are valid with Y_n and X_n replaced by Y_nC and X_nC, respectively. In this case, the mean-square error matrix of the WLS solution to the LSE problem is also minimized

by the choice $W = R^{-1}$ if the true parameter also satisfies the constraints and the residual $Y_n - \theta^* X_n$ have zero mean.

11. Conclusions. In this work, we have shown that the unique solution of the LS problem with linear-equality constraints (i.e., the LSE problem) can always be calculated by a recursion that is identical to the one for the unconstrained LS problem. The set of all solutions to the LSE problem was derived first, from which the unique solution was obtained, along with a necessary and sufficient condition for its existence. Then a recursive algorithm for the solution to the LS problem with linear-equality constraints (i.e., the LSI problem) was developed based on the recursive solutions to the LSE and the unconstrained problems. The RLS algorithm can be made numerically robust which guarantees that the recursively calculated solution satisfies the linear constraints no matter how large the numerical errors are. Also presented are a simple initialization of the RLS algorithm for the LSE and LSI problems, along with the convergence of this RLS algorithm to the exact LS solution, and the convergence of both the RLS algorithm and the exact RLS solution to the true parameter if it satisfies the constraints. Some numerical examples were given. They all support the above analytical results well.

Appendix. Moore-Penrose generalized inverse. At the repeated requests of a reviewer, we provide the following statements about the MP inverse. There are different generalizations of the inverse of matrix A for the case where A is singular or rectangular. The Moore-Penrose generalized inverse, or MP inverse for short, is the most popular one. For any $m \times n$ matrix A it always exists and is the unique $n \times m$ matrix X satisfying all four Penrose conditions:

$$AXA = A$$
$$XAX = X$$
$$(AX)^* = AX$$
$$(XA)^* = XA$$

Compared with matrix inverse, the MP inverse of $A_{m \times n}$ has properties that are (a) *identical*, e.g., $(A^{-1})^{-1} = A \leftrightarrow (A^+)^+ = A$; (b) *similar*, e.g., $AA^{-1} = A^{-1}A = I$ while $AA^+ = P_m, A^+A = P_n$, where P is an orthogonal projection matrix; (c) *restricted extensions*, e.g., $(AB)^{-1} = B^{-1}A^{-1}$ when A and B are nonsingular, while $(AB)^+ \neq B^+A^+$ but $(AB)^+ = B^+A^+$ if A is full column rank and B is full row rank; or (d) *different* when zero matrix is involved, e.g., $O^+ = O'$.

MP inverse is related to the inverse:

$$A^+ = A^{-1} \text{ if } A^{-1} \text{ exists}$$

$$A^+ = A_L^{-1} \text{ (left inverse)} = (A^H A)^{-1} A^H \quad \text{if } (A^H A)^{-1} \text{ exists}$$

$$A^+ = A_R^{-1} \text{ (right inverse)} = A^H (A A^H)^{-1} \quad \text{if } (A A^H)^{-1} \text{ exists}$$

Proof of Theorem 1. By the basic properties of the M-P inverse (see, e.g., [3]), we have

$$(A.1) \qquad X_n^* P (P X_n X_n^* P)^+ P X_n X_n^* P = X_n^* P (X_n^* P)^+ (P X_n)^+ P X_n X_n^* P = X_n^* P,$$

Substituting it and (13)–(14) into (1) yields

$$
\begin{aligned}
S_n &= \text{tr}\{[Y_n - (B^* A^{+*} + \eta^* P) X_n][Y_n - (B^* A^{+*} + \eta^* P) X_n]^*\} \\
&= \text{tr}\{[\eta^* - (Y_n - B^* A^{+*} X_n) X_n^* P (P X_n X_n^* P)^+] P X_n X_n^* P \\
&\qquad [\eta^* - (Y_n - B^* A^{+*} X_n) X_n^* P (P X_n X_n^* P)^+]^* \\
&\qquad + (Y_n - B^* A^{+*} X_n)(Y_n - B^* A^{+*} X_n)^* \\
&\qquad - (Y_n - B^* A^{+*} X_n) X_n^* P (P X_n X_n^* P)^+ P X_n (Y_n - B^* A^{+*} X_n)^*\}.
\end{aligned}
$$
(A.2)

where the second equation can be checked by a straightforward calculation of each term in view of (A.1). Clearly, S_n is minimized by the solution, if exists, of the following equation

$$\text{tr}\{[\eta^* - (Y_n - B^* A^{+*} X_n) X_n^* P (P X_n X_n^* P)^+] P X_n X_n^* P$$

$$[\eta^* - (Y_n - B^* A^{+*} X_n) X_n^* P (P X_n X_n^* P)^+]^*\} = 0.$$

It is well known that the general solution of the above equation is given by

$$(A.3) \qquad \eta = (P X_n X_n^* P)^+ P X_n (Y_n^* - X_n^* A^+ B) + \xi, \ \forall \xi : X_n^* P \xi = 0.$$

It follows from (13),(14) and (A.3) that

$$(A.4) \qquad \theta_n = A^+ B + P (P X_n X_n^* P)^+ P X_n (Y_n^* - X_n^* A^+ B) + P \xi, \ \forall \xi \text{ as in (A.3)}.$$

Without loss of generality we assume $P X_n \neq 0$, otherwise, $\theta_n = A^+ B + P \xi$ satisfying (15). Let $L(D)$ be a vector subspace spanned by the column vectors of matrix D, the column vectors of matrix V_1 be an orthonormal basis of $L(A^+)$, and the column vectors of matrix V_2 be an orthonormal basis of $L(P X_n)$.

Since $A^{+*} P = A^{+*} - A^{+*} A^+ A = 0$, we have

$$(A.5) \qquad\qquad\qquad\qquad V_1^* V_2 = 0.$$

Therefore, when $[V_1 \ V_2]$ is not a unitary matrix, there must exist a matrix V_3 such that $[V_1 V_2 V_3]$ is a unitary matrix. By the definition of V_2, there exists the following full-rank decomposition of $P X_n$

$$(A.6) \qquad\qquad PX_n = V_2 K, \ KK^* \text{ is nonsingular.}$$

Hence,

$$(A.7) \qquad PX_n X_n^* P = V_2 KK^* V_2^*, (PX_n X_n^* P)^+ = V_2 (KK^*)^{-1} V_2^*.$$

Similarly,

$$(A.8) \qquad\qquad A^+ = V_1 H, \ HH^* \text{ is nonsingular,}$$

and

$$(A.9) \qquad\qquad A = H^*(HH^*)^{-1} V_1^*, \ A^+ A = V_1 V_1^*.$$

Using (14), (A.9) and the definition of $[V_1 V_2 V_3]$, we have

$$(A.10) \qquad P = [V_1 V_2 V_3][V_1 V_2 V_3]^* - V_1 V_1^* = V_2 V_2^* + V_3 V_3^*$$

It follows from the definition of V_3, (A.7) and (A.10) that, since P is a projection ($P^2 = P$),

$$(A.11) \qquad (PX_n X_n^* P)^+ P = P(PX_n X_n^* P)^+ = (PX_n X_n^* P)^+.$$

Obviously, when $[V_1 V_2]$ is unitary, (A.11) still holds. Substituting (A.11) into (A.4) yields (15). □

Proof of Corollary 1. Note first that only θ_n of (15) with $P\xi = 0$ can be a unique solution, otherwise θ_n of (15) with $\alpha P\xi$ for any real number $\alpha \neq 0$ would be a distinct solution since $\alpha X_n^* P\xi = 0$. When $P\xi = 0$, (15) gives a unique solution due to the uniqueness of the M-P inverse. Thus the LSE problem has a unique solution iff $P\xi = 0$. A necessary and sufficient condition for $P\xi = 0$ such that $X_n^* P\xi = 0$ is that the vector $P\xi$ is in the row space of X_n^*. Since $P = I - A^+ A$ is a projector onto the orthogonal complement of the row space of A, the above necessary and sufficient condition holds iff the row space of X_n^* is the orthogonal complement of the row space (i.e., subspace spanned by the row vectors) of A, which is equivalent to $\binom{A}{X_n^*}$ having full rank r. □

Proof of Theorem 2. Denote

$$(A.12) \qquad \overline{\theta}_n = \theta_n - A^+ B, \overline{Y}_n = Y_n - B^* A^{+*} X_n, \overline{y}_n = y_n - B^* A^{+*} \mathbf{x}_n.$$

From (17) and (A.12), we have

$$(A.13) \qquad\qquad \overline{\theta}_n = (PX_n X_n^* P)^+ X_n \overline{Y}_n^*.$$

Therefore, similar to the unconstrained LS problem, if we can prove that the second equality of (20) holds, it is easy to prove that

$$(A.14) \qquad \bar{\theta}_{n+1} = \bar{\theta}_n + K_{n+1}(\bar{y}_{n+1} - \bar{\theta}_n^* \mathbf{x}_{n+1})^*$$

and (19) are true, then using (A.12) and (A.14) we have (18), in other words, it is sufficient to prove the second equality in (20).

Let $[V_1 \ W]$ be a unitary matrix. By (A.9) we have

$$(A.15) \qquad P = I - A^+ A = [V_1 \ W][V_1 \ W]^* - V_1 V_1^* = W W^*,$$

and

$$(A.16) \qquad P X_n X_n^* P = W W^* X_n X_n^* W W^*.$$

Note that $AW = 0$, which follows from (A.9) and the orthogonality of V_1 and W. Hence

$$W^*[A^* \ X_n] \begin{bmatrix} A \\ X_n^* \end{bmatrix} W = [0 \ W^* X_n] \begin{bmatrix} 0 \\ X_n^* W \end{bmatrix} = W^* X_n X_n^* W$$

From the assumption that $\begin{pmatrix} A \\ X_n^* \end{pmatrix}$ has full column rank, it is thus clear that $W^* X_n X_n^* W$ is nonsingular since W has full rank and thus

$$(A.17) \qquad (P X_n X_n^* P)^+ = W(W^* X_n X_n^* W)^{-1} W^*.$$

Using the first equality of (20), (A.17) and (9), we have

$$
\begin{aligned}
P_{n+1} &= W(W^* X_{n+1} X_{n+1}^* W)^{-1} W^* \\
&= W(I - (W^* X_n X_n^* W)^{-1} W^* \frac{\mathbf{x}_{n+1} \mathbf{x}_{n+1}^* W}{1 + \mathbf{x}_{n+1}^* W(W^* X_n X_n^* W)^{-1} W^* \mathbf{x}_{n+1}}) \\
&\quad \times (W^* X_n X_n^* W)^{-1} W^* \\
&= (I - P_n \mathbf{x}_{n+1} \mathbf{x}_{n+1}^* / (1 + \mathbf{x}_{n+1}^* P_n \mathbf{x}_{n+1})) P_n,
\end{aligned}
$$

which is the second equality of (20). $\qquad \square$

Proof of Theorem 3. Substituting (37) into (17) and using (13), (A.15), (A.17) and (38) give

$$
\begin{aligned}
\theta_n &= A^+ B + (P X_n X_n^* P)^+ \left[X_n X_n^* (\theta - A^+ B) + \sum_{i=1}^n \mathbf{x}_i e_i^* \right] \\
&= A^+ B + W(W^* X_n X_n^* W)^{-1} W^* \left(X_n X_n^* W W^* \xi + \sum_{i=1}^n \mathbf{x}_i e_i^* \right) \\
&= \theta + (\frac{1}{n} P X_n X_n^* P)^+ \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i e_i^*.
\end{aligned}
$$

□

Proof of Theorem 4. According to the analysis in Sec. 7, the difference between $\tilde{\theta}_n$ and θ_n converges to zero, under Assumption A1, as n increases. It thus suffices to show the convergence of θ_n.

Note first that the true parameter θ satisfying the inequality constraints means that either it is in the region bounded by the constraints or it satisfies some equality constraints. Under assumptions A1 and A2, it is clear from Theorem 3 that for the recursive algorithm θ_n^i of Section 6, the unconstrained solution will converge to θ if θ is in the region bounded by the constraints, or in the case when θ satisfies some equality constraints in (24) the solution subject to the corresponding equality constraints will converge. It is thus clear that the possible cause for the algorithm not to converge to θ may only come from those solutions for which θ does not satisfies the corresponding equality constraints (or not in the region bounded by them) assumed.

Suppose that the algorithm does not converge to θ. There must be an equality constraint set (or the region bounded by the constraints) S_j for which $\theta \notin S_j$ but the corresponding θ_n^j is chosen as the RLS solution by the algorithm infinitely many times. By A3, there must exist a subsequence $\{\theta_{n_k}^j\}$ of $\{\theta_n^j\}$ such that it converges to $\bar{\theta} \neq \theta$ and its elements are all chosen as the solutions (at different times) by the algorithm. Furthermore, assume $\theta \in S_i$ and denote by θ_n^i the corresponding solution that converges to θ. For the difference between the corresponding two sequences $\{\theta_n^i\}$ and $\{\theta_n^j\}$ we have under A1 and A2,

$$
\begin{aligned}
(\text{A.18}) \quad & \frac{1}{n_k}[S_{n_k}(\theta_{n_k}^i) - S_{n_k}(\theta_{n_k}^j)] \\
& = \frac{1}{n_k}[(\theta_{n_k}^i - \theta_{n_k}^j)^* X_{n_k} X_{n_k}^* \theta_{n_k}^i + \theta_{n_k}^{j*} X_{n_k} X_{n_k}^* (\theta_{n_k}^i - \theta_{n_k}^j) \\
& \quad - 2(\theta_{n_k}^i - \theta_{n_k}^j)^* X_{n_k} (X_{n_k}^* \theta + E_{n_k}^*)] \\
& = \frac{1}{n_k}[(\theta_{n_k}^i - \theta_{n_k}^j)^* X_{n_k} X_{n_k}^* (\theta_{n_k}^i - \theta) + (\theta_{n_k}^i - \theta_{n_k}^j)^* X_{n_k} X_{n_k}^* (\theta_{n_k}^j - \theta)] \\
& \quad - \frac{2}{n_k}(\theta_{n_k}^i - \theta_{n_K}^j)^* X_{n_k} E_{n_k}^* \\
& \geq \frac{c}{2}\|\theta - \bar{\theta}\|^2, \quad \forall n_k > k_0 \text{ for some } k_0.
\end{aligned}
$$

On the other hand, we know that (a) the cost function of the exact unconstrained LS solution θ_{n_k} is never smaller than that of any constrained LS solution and (b) $\theta_{n_k}^j$ has a cost function not greater than that of $\theta_{n_k}^i$ since $\theta_{n_k}^j$ is chosen as the solution by the algorithm for time n_k. In other words, we have

$$
(\text{A.19}) \qquad \frac{1}{n_k} S_{n_k}(\theta_{n_k}) \leq \frac{1}{n_k} S_{n_k}(\theta_{n_k}^j) \leq \frac{1}{n_k} S_{n_k}(\theta_{n_k}^i),
$$

Since θ_{n_k} and $\theta_{n_k}^i$ both converge to the true parameter $\theta \in S_i$, the difference between the third term and the first in (A.19) converges to zero. As a result, the difference

between the second and third terms in (A.19) must converge to zero as well. But this contradicts (A.19). It thus follows that the theorem holds.   □

<div align="center">REFERENCES</div>

[1] A. A. ANDA AND H. PARK, *Self-Scaling Fast Rotations for Stiff and Equality-Constrained Linear Least Squares Problems*,   Linear Algebra and Its Applications, 234(1996), pp. 137–161.

[2] J. L. BARLOW, N. K. NICHOLS, AND R. J. PLEMMONS,   *Iterative Methods for Equality-Constrained Least Squares Problems*, SIAM J. Sci. Stat. Comput., 9:5(1988), pp. 892–906.

[3] A. BEN-ISRAEL AND T. N. E. GREVILLE, *Generalized Inverses: Theory and Applications*, John Wiley & Sons, New York, 1974.

[4] A. BJORCK, *A General Updating Algorithm for Constrained Linear Least Squares Problems*, SIAM J. Sci. Stat. Comput., 5:2(1984), pp. 394–402.

[5] A. BJORCK, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, PA, 1996.

[6] P. E. CAINES, *Linear Stochastic Systems*, Wiley, New York, 1988.

[7] H. F. CHEN, *Recursive Estimation and Control for Stochastic Systems*, Wiley, New York, 1985.

[8] J. M. CIOFFI, *The Fast Adaptive ROTOR's RLS Algorithm*,   IEEE Trans. Acoustic, Speech Signal Processing, ASSP-38(1990), pp. 631–653.

[9] J. M. CIOFFI, *The Fast Householder RLS Adaptive Filter*, in: Proc. ICASSP, Albuquerque, NM, Apr. 1990.

[10] J. M. CIOFFI AND T. KAILATH, *Fast, Recursive-Least-Squares Transversal Filters for Adaptive Filtering*,   IEEE Trans. Acoustic, Speech Signal Processing, ASSP-32(1984), pp. 304–337.

[11] A. DAX, *On Row Relaxation Methods for Large Constrained Least Squares Problems*, SIAM J. Sci. Comput., 14:3(1993), pp. 570–584.

[12] L. ELDEN, *Perturbation Theory for the Least Squares Problem with Linear Equality Constraints*, SIAM J. Numer. Anal., 17(1980), pp. 338–350.

[13] R. W. FAREBROTHER, *Linear Least Squares Computations*, Marcel Dekker, New York, 1988.

[14] O. L. FROST, *An Algorithm for Linearly Constrained Adaptive Array Processing*, IEEE Proc., 60(1972), pp. 926–935.

[15] L. C. GODARA, *Constrained Beamforming and Adaptive Algorithms*, in: N. K. Bose and C. R. Rao, editors, Handbook of Statistics, vol. 10, volume 10, pages 269–354. Elsevier Science, New York, 1996.

[16] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, Johns Hopkins University, 1996.

[17] G. C. GOODWIN AND R.E. PAYNE, *Dynamic System Identification: Experimental Design and Data Analysis*, Academic Press, New York, 1977.

[18] M. GULLIKSSON, *Iterative Refinement for Constrained and Weighted Least Squares*. BIT, 34(1994), pp. 239–253.

[19] M. GULLIKSSON AND P.-A. WEDIN, *Modifying the QR-Decomposition to Constrained and Weighted Least Squares*, SIAM J. Matrix Anal. Appl., 13:4(1992), pp. 1298–1313.

[20] R. J. HANSON, *Linear Least Squares with Bounds and Linear Constraints*, Mathematical Programming, 21(1981), pp. 98–118.

[21] R. J. HANSON AND C. L. LAWSON, *Extensions and Applications of Householder Algorithm for Solving Linear Least Squares Problems*, Math. Comp., 23:108(1969), pp. 787–812.

[22] K. H. HASKELL AND R. J. HANSON, *An Algorithm for Linear Least Squares Problems with Equality and Nonnegativity Constraints*, Mathematical Programming, 21(1981), pp. 98–

118.

[23] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Englewood Cliffs, NJ, 1996.

[24] D. James, *Implicit Nullspace Iterative Methods for Constrained Least Squares Problems*, SIAM
      J. Matrix Anal. Appl., 13:3(1992), pp. 962–978.

[25] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*, SIAM, Philadelphia, PA,
      1995.

[26] Lev-Ari, Chang, and Kailath, *Constrained-Input Constrained-Output Stability for Adaptive
      RLS Lattice Filters*, IEEE Trans. on Circuit and Systems, 38:12(1991).

[27] L. Ljung, *System Identification: Theory for the User*, Prentice Hall, Englewood Cliffs, NJ,
      1987.

[28] L. Ljung, M. Morf, and D. Falconer, *Fast Calculation of Gain Matrices for Recursive
      Estimation Schemes*, Int. J. Control, 27(1978), pp. 1–19.

[29] R. E. Plackett, *Some Theorems in Least Squares*, Biometrika, 37(1950), pp. 149.

[30] I. K. Proudler, J. G. McWhirter, and T. J. Shepard, *Fast QRD-Based Algorithms for
      Least-Squares Linear Prediction*, in: IMA Conf. Proc., Warwick, England, 1988.

[31] K. Schittkowski and J. Stoer, *A Factorization Method for the Solution of Constrained Lin-
      ear Least Squares Problems Allowing Subsequent Data changes*, Numerische Mathematik,
      31(1979), pp. 431–463.

[32] J. Stoer, *On the Numerical Solution of Constrained Least Squares Problems*, SIAM J. Numer.
      Anal., 8(1971), pp. 382–411.

[33] R. L. Streit, *Solution of Systems of Complex Linear Equations in the L_∞ Norm with Con-
      straints on the Unknowns*, SIAM J. Sci. Stat. Comput., 7:1(1986), pp. 132–149.

[34] B. D. Van Veen and K. M. Buckley, *Beamforming: A Versatile Approach to Spatial Filter-
      ing*, IEEE ASSP Magzine, pages 4–25, Apr. 1988.

[35] D. A. Wismer and R. Chattergy, *Introduction to Nonlinear Optimization*. Elsevier North-
      Holland, New York, 1978.

[36] M. Woodbury, *Inverting Modified Matrices*, 1950. Mem. Rep. 42, Statistical Research Group.

[37] F. F. Wu, *Power System State Estimation: A Survey*, Int. J. Elec. Power & Energe Syst.,
      12:2(1990), pp. 80–87.