Katholieke Universiteit Leuven

Departement Elektrotechniek ESAT-SISTA

TR 96-25

STRUCTURED TOTAL LEAST SQUARES METHODS IN SIGNAL PROCESSING [1]

Philippe Lemmerling[2], Sabine Van Huffel[3] and Bart De Moor[4]

May 1996
(*to appear in the Proceedings of EUSIPCO96, Trieste, Italy, September 10-13, 1996*)

[1]This report is available by anonymous ftp from *ftp.esat.kuleuven.ac.be* in the directory *pub/SISTA/lemmerli/reports/int9625.ps.Z*

[2]ESAT - Katholieke Universiteit Leuven, K. Mercierlaan 94, 3001 Leuven-Heverlee, Belgium, tel 32/16/32 17 96, fax 32/16/32 19 86, email: *philippe.lemmerling@esat.kuleuven.ac.be*, URL: *http://www.esat.kuleuven.ac.be/sista/members/lemmerli/*. Work supported by the EU Programme "Human Capital and Mobility, Networks", project CHRX-CT94–432, the Flemish Government (Concerted Research Action MIPS: Model-based Information Processing Systems), the Belgian Government (Interuniversity Attraction Poles IUAP-17: Modelling and Control of Dynamical Systems, IUAP-50: Automation in Design and Production) and the European Commission (Human Capital and Mobility Network SIMONET: System Identification and Modelling Network). The First author is a Ph.D. student funded by the I.W.T. (Flemish Institute for the Promotion of Scientific-Technological Research in Industry).

[3]Sabine Van Huffel is a Research Associate with the Belgian N.F.W.O. (National Fund for Scientific Research).

[4]Bart De Moor is an associate professor at the Katholieke Universiteit Leuven and is a Senior Research Associate with the National Fund for Scientific Research.

STRUCTURED TOTAL LEAST SQUARES METHODS IN SIGNAL PROCESSING

Philippe Lemmerling * , *Sabine Van Huffel* *, *Bart De Moor* *
*ESAT Laboratory, Department of Electrical Engineering, Katholieke Universiteit Leuven, Kardinaal Mercierlaan 94, 3001 Leuven, Belgium.
Tel: +32 16 321709; fax: +32 16 321986
e-mail: philippe.lemmerling@esat.kuleuven.ac.be.

ABSTRACT

In many signal processing applications, one has to solve an overdetermined system of linear equations $Ax \approx b$, while minimizing the errors on A and b. The Total Least Squares (TLS) method calculates corrections ΔA and Δb such that $(A + \Delta A)x = b + \Delta b$ and $\|[\Delta A \;\; \Delta b]\|_F$ is minimal. The resulting parameter vector x is a Maximum Likelihood (ML) estimate when the noise on the entries of $[A \;\; b]$ is i.i.d. Gaussian noise with zero mean and equal variance. In many applications, these last conditions do not hold because of the structure present in $[\Delta A \quad \Delta b]$. Under those circumstances, the TLS will not yield a ML estimate of the parameter vector x since the SVD (which is the standard way to obtain the TLS solution) is not structure preserving. Therefore, several **structured** Total Least Squares methods have been developed in recent years: Constrained Total Least Squares (CTLS) method [1][2], the Structured Total Least Squares (STLS) method [3] and the Structured Total Least Norm (STLN) method [8][7]. As opposed to the ordinary TLS these methods yield a ML estimate of the parameter vector x, by imposing the structure of $[A \;\; b]$ to $[\Delta A \;\; \Delta b]$.

1 INTRODUCTION

The structured TLS problem can be formulated as follows:

$$\begin{aligned} &\min_{\Delta A, \Delta b, x} \|[\Delta A \;\; \Delta b]\|_F \\ &\text{such that } \;(A + \Delta A)x = (b + \Delta b) \;\text{ and} \\ &[\Delta A \;\; \Delta b] \text{ has the same structure as } [A \;\; b]. \end{aligned} \qquad (1)$$

It is the last constraint that distinguishes (1) from the ordinary TLS formulation and thereby allows us to obtain a ML estimate of the parameter vector x when $[A \;\; b]$ is structured and the noise on the different entries of $[\Delta A \;\; \Delta b]$ is i.i.d. Gaussian noise with zero mean and equal variance.
In the following section we shortly describe the formulation of the three most important approaches for structured TLS problems: the Constrained Total Least Squares (CTLS) formulation, the Structured Total Least Squares (STLS) formulation and the Structured Total Least Norm formulation (STLN).
Each of these formulations can capture problem (1), but as will be discussed in section 2, some of them allow extensions towards other norms and multiple right hand sides.
In section 3 we describe the different existing and/or improved solution methods of the three formulations.
Section 4 gives an interpretation of the three different approaches by comparing their solution methods to those of the ordinary TLS method.
Section 5 illustrates the properties of the algorithms described in section 3 by means of a simple modeling example.
In section 6 we apply the structured TLS approach to a Linear Prediction problem of more realistic dimensions than the small modeling example. By doing Matlab simulations we give an idea of the complexity of the structured TLS approach.
Finally, we end with the conclusions and further research. In the remainder of the paper we adopt a Matlab like notation: $v(i)$ indicates the ith element of vector v, $A(i,j)$ indicates the entry of A on row i and in column j and $A(:,i)$ denotes the ith column of A. We will also use the following shorthand notations: $S \equiv [A \;\; b]$ with $A \in R^{m \times n}$, $b \in R^{m \times 1}$ and $\Delta S \equiv [\Delta A \;\; \Delta b]$ with $\Delta A \in R^{m \times n}$, $\Delta b \in R^{m \times 1}$.

2 THREE FORMULATIONS

In this section we give a description of the three most important formulations of structured TLS problems. We will illustrate each of the formulations by means of the following Hankel matrix $S = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{bmatrix}^T$ and its corresponding correction matrix $\Delta S = \begin{bmatrix} \tilde{\alpha} & \tilde{\beta} & \tilde{\gamma} \\ \tilde{\beta} & \tilde{\gamma} & \tilde{\delta} \end{bmatrix}^T$.

2.1 CTLS

The CTLS formulation groups the different entries of $[\Delta A \;\; \Delta b]$ in a so called noise vector $f \in R^{k \times 1}$, with k the number of different entries in $[\Delta A \;\; \Delta b]$. The structure of the problem is represented by matrices $F_i \in R^{m \times k}$

which are defined as follows $\Delta S(:,i) = F_i f$. The CTLS formulation then becomes:

$$\min_f f^T W f \qquad (2)$$
$$\text{such that } (A + [F_1 f \dots F_n f])x = b + F_{n+1} f,$$

with W a diagonal weighting matrix. Our 3×2 example gives $f = [\tilde{\alpha}\;\; \tilde{\beta}\;\; \tilde{\gamma}\;\; \tilde{\delta}]^T$ and $F_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ and $F_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$.

2.2 STLS

This formulation can handle extended data matrices that can be written as a linear combination of basis matrices T_i:

$$S = T_0 + s(1)T_1 + \dots + s(k)T_k,$$

with $s \in R^{k\times 1}$ the different elements of S, $T_i \in R^{m\times(n+1)}, i = 1,\dots,k$ the basis matrices and k the number of different entries in $[\Delta A \;\; \Delta b]$. For our 3×2 example this results in $k = 4$ and $T_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}^T$, $T_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}^T, \dots$. The STLS formulation then is:

$$\min_{t,y} \sum_{i=1}^{k} W(i,i)(t(i) - s(i))^2$$
$$\text{such that } Ty = 0 \text{ and } y^T y = 1, \qquad (3)$$

with $t \in R^{k\times 1}$ the different elements of $S + \Delta S$, $T = T_0 + \sum_{i=1}^{k} t(i)T_i$ and W a diagonal weighting matrix.

2.3 STLN

The STLN formulation uses a vector $\alpha \in R^{q\times 1}$ to represent the different entries of ΔA and $\beta \in R^{d\times 1}$ contains the different entries of Δb which are not already contained in α. The STLN formulation is as follows:

$$\min_{\alpha,\beta,x} \frac{1}{2}(\alpha^T D_\alpha^2 \alpha + \beta^T D_\beta^2 \beta)$$
$$\text{such that } Ax + X\alpha = b + F\beta + P\alpha \qquad (4)$$

with $X \in R^{m\times q}$ defined by $X\alpha = \Delta A x$ and $F \in R^{m\times d}, P \in R^{m\times q}$ defined by $F\beta + P\alpha = \Delta b$. $D_\alpha \in R^{q\times q}$ and $D_\beta \in R^{d\times d}$ represent weighting matrices (e.g. to represent the number of occurences of the elements of α and β in ΔA and Δb). For our 3×2 example this gives $\alpha = [\tilde{\alpha}\;\; \tilde{\beta}\;\; \tilde{\gamma}]^T$, $\beta = \tilde{\delta}$, $F = [0\;\; 0\;\; 1]^T$, $P = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ and $X = \begin{bmatrix} x_1 & 0 & 0 \\ 0 & x_1 & 0 \\ 0 & 0 & x_1 \end{bmatrix}$

2.4 Differences between formulations

As can be seen from the previous subsections, the CTLS and STLN formulations are very similar. The STLS formulation ((3) and (6)-(7)) imposes different constraints on the parameter vector, but as proved in [5] this has no influence on the solution in generic cases.

All the formulations are able to capture the common problem (1). However, only the STLN formulation allows for extensions towards multiple right hand sides ($b \in R^{m\times d}$ with $d > 1$) [7] and minimization of the error vector in other norms than the L_2 norm (L_1, L_∞) [8].

3 Solution methods

This section explains the different existing algorithms as well as some of our own improvements for solving (2), (3) and (4).

3.1 CTLS

In [1] problem (2) is reformulated as an unconstrained minimization problem in x:

$$\min_x \begin{bmatrix} x \\ -1 \end{bmatrix}^T S^T (H_x W^{-1} H_x^T)^{-1} S \begin{bmatrix} x \\ -1 \end{bmatrix}, \qquad (5)$$

with $H_x = \sum_{i=1}^{n} x_i F_i - F_{n+1}$. In [1] a Newton method is proposed that uses analytically derived gradients and Hessians. However, the high number of inversions and matrix-matrix multiplications causes an important loss of accuracy. Therefore we propose to use a quasi-Newton method in which we use numerically calculated gradients. This solution method will be denoted CTLS1.

3.2 STLS

In order to solve (3), [3] derives an equivalent formulation:

find the solutions u and v according to the smallest τ, such that

$$Sv = D_v u\tau \quad u^T D_v u = 1 \qquad (6)$$
$$S^T u = D_u v\tau \quad v^T D_u v = 1. \qquad (7)$$

with $D_v = \sum_{i=1}^{k} \frac{1}{W(i,i)}(T_i v)(T_i v)^T$, $D_u = \sum_{i=1}^{k} \frac{1}{W(i,i)}(T_i^T u)(T_i^T u)^T$ and W a diagonal weighting matrix. The nice thing about this new formulation is that it states the original STLS formulation as a nonlinear SVD. The goal of the new STLS formulation is then to find the smallest singular value τ of the nonlinear SVD, as well as the corresponding right and left singular vector. The relation with the original STLS formulation is as follows:

$$y = v/\|v\|_2$$
$$t(i) = s(i) - u^T T_i v\tau, \; i = 1,\dots,k$$

In [3] an inverse iteration algorithm is derived for solving (6)-(7). This algorithm will further be denoted by

STLS1.
We present an improved method which solves the following equivalent system of equations:

$$\begin{aligned} Sy &= D_y l \\ S^T l &= D_l y \\ y^T y &= 1 \end{aligned} \tag{8}$$

We can find the t from the original STLS formulation (3) as follows $t(i) = s(i) - l^T T_i y, \ i = 1, \ldots, k$. For the solution of system (8) we simply use a Newton method. The involved Jacobians are easy to derive analytically. This algorithm will be denoted STLS2.

3.3 STLN

In [8] several methods for solving (4) are proposed. Among them we find a penalty function method that transforms the constrained optimization problem into an unconstrained optimization problem. We do not consider these methods since they can not attain a high level of accuracy [4]. Another method proposed in [8] solves (4) iteratively by approximating the constraint in each step by a linear approximation while leaving the objective function unchanged. We will call this algorithm STLN1. As shown in [6] we can improve convergence speed by including curvature terms in the Hessian of the objective function that we use in each iteration. This algorithm will be denoted by STLN2.

4 INTERPRETATION

Since the ordinary TLS approach differs from the structured TLS approach only in the last constraint of (1), we can easily compare solution methods of both approaches. Essentially, the ordinary TLS problem is also an optimization problem, so we could have solved it by using standard optimization techniques. This is the approach adopted by STLN for the structured TLS case. For the ordinary TLS approach however, it is generally known that the best way to compute the solution is by taking a SVD of $[A \quad b]$. The structured counterpart of this solution method is the nonlinear SVD (6)-(7), defined in the STLS approach. The CTLS approach tries to find the same minimal nonlinear singular value τ and the corresponding right singular vector by optimizing the nonlinear Rayleigh quotient (5). As we will see in the next section (and as we already know from the ordinary TLS), it is not such a good idea to use an (nonlinear) Eigenvalue Decomposition (EVD) instead of the corresponding (nonlinear) SVD, since the effect of squaring $[A \quad b]$ will cause a tremendous loss of accuracy.

5 MODELING EXAMPLE

In this section we compare the different algorithms proposed in section 3 for a simple modeling example: identify a first order system with impulse response [6 5 4 3 2 1]. As shown in [3], this problem

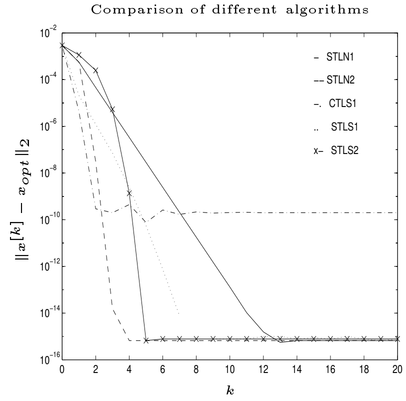

Figure 1: *In this figure we compare the different algorithms for the structured TLS problem. It shows the distance of $x^{[k]}$ (=the parameter vector at iteration k) to the exact solution x_{opt}.*

can be formulated as a structured TLS problem with $S = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 6 \end{bmatrix}^T$. In Figure 1 we compare the different algorithms. It shows the distance of $x^{[k]}$ (=the parameter vector at iteration k) to the exact solution x_{opt} (which can be calculated exactly for this problem (see [7])). We observe the following. The CTLS1 algorithm clearly suffers from the squaring effect mentioned in section 4: the accuracy is the lowest of all the implemented algorithms. If we compare the algorithms STLN1 and STLN2 we clearly see that STLN1 has only a linear convergence rate, whereas STLN2 has a superlinear convergence rate. The difference can be explained by the fact that STLN2 uses curvature information of the constraints whereas STLN1 doesn't (see section 3). The inverse iteration algorithm STLS1 has a linear convergence rate, whereas STLS2 which solves the nonlinear SVD equation by using a Newton method has a superlinear convergence rate.

6 LINEAR PREDICTION EXAMPLE

In this section we consider a more complicated problem than the one used in section 5. We construct a real signal $z(t)$ consisting of complex damped exponentials. To make the signal real we combine each exponential with an exponential that has the same damping but the opposite frequency:

$$\begin{aligned} z(t) &= \sum_{k=1}^{8} e^{(-d(k)+2\pi\sqrt{-1}f(k))t}, k = 1, \ldots, N \\ d &= [0.1 \ \ 0.2 \ \ 0.3 \ \ 0.35 \ \ 0.1 \ \ 0.2 \ 0.3 \ \ 0.35] \\ f &= [0.45 \ \ 0.4 \ \ 0.3 \ \ 0.1 \ \ -0.45 \ \ -0.4 \ \ -0.3 \ \ -0.1], \end{aligned}$$

with N the number of samples we use. We now add normally distributed zero-mean noise $n(t)$ with standard

deviation σ_n to the signal $z(t)$. The goal is to find the original signal $z(t)$ from the noisy signal $z(t) + n(t)$.
We know that 9 consecutive samples from the noiseless signal satisfy a Linear Prediction equation. This means that if we construct the system of LP equations $Ax \approx b$, the resulting extended Toeplitz data matrix $[A \quad b]$, with first column $[z(N-8) \ldots z(1)]$ and first row $[z(N-8) \ldots z(N)]$ is rank deficient. To find a ML estimate of z we arrange the noisy signal in a Toeplitz matrix and try to find a rank deficient approximation of it. This is clearly a structured TLS problem. We will use STLN1 and STLN2 in order to get an idea of the complexity of both algorithms. We do not use STLS1 and STLS2, since their behaviour is similar to that of STLN1 and STLS2 (see Figure 1). CTLS1 is excluded because of its bad numerical accuracy. Table 1 summarizes the results. As can be deduced from the previous, we have to solve a structured TLS problem with an extended data matrix $[A \quad b] \in R^{(N-8)\times 9}$. We take $N = 100$. As we can see

σ_n	iterations STLN1	flops STLN1	iterations STLN2	flops STLN2
1e-10	2	90350997	2	95378778
1e-5	3	96083678	3	110622076
1e-2	10	389535906	9	378946323

Table 1: This table summarizes the number of flops and iterations that are needed to find the ML estimate of the signal $z(t)$. σ_n is the standard deviation of the normally distributed white noise we added to the noiseless signal.

from table 1 it pays off to use the more complicated STLN2 method for those cases where STLN2 needs less iterations (STLN2 needs more flops per iteration since it uses the curvature information of the constraints).

7 CONCLUSIONS

Many problems in signal processing can be stated as structured TLS problems. If we want to obtain ML estimates, we have to use the structured TLS approach and not the ordinary TLS approach.
Several equivalent formulations can be deduced from the main structured TLS formulation (1): the CTLS formulation, the STLS formulation and the STLN formulation. We can easily relate them to a similar solution method for the ordinary TLS. The most successful approach up til now is STLN. The reason is that we do not have efficient algorithms to calculate the nonlinear SVD (STLS) or EVD (CTLS). Further research will therefore focus on algorithms for the nonlinear SVD and EVD. One could hope that this will lead to cubically converging algorithms (whereas classical optimization hardly attains quadratical convergence rates) since this is the fact for the ordinary TLS.

8 ACKNOWLEDGMENTS

The first author is a Ph.D. student funded by the IWT (Flemish Institute for Support of Scientific-Technological Research in Industry). The second and third authors are respectively Research Associate and Senior Research Associate with the Belgian N.F.W.O. (National Fund for Scientific Research). This work is supported by the Belgian Programme on Interuniversity Poles of Attraction (IUAP-nr.17 and IUAP nr. 50), initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture, by the EU Programme "Human Capital and Mobility, Networks", project CHRX-CT94–432, and by a Concerted Research Action (GOA) project of the Flemish Community, entitled "Model-based Information Processing Systems".

References

[1] T. J. Abatzoglou and J. M. Mendel, *Constrained Total Least Squares*, IEEE International Conf. on Acoustics, Speech & Signal Processing, Dallas, 1987, pp. 1485-1488.

[2] T. J. Abatzoglou, J. M. Mendel and G. A. Harada, *The Constrained Total Least Squares Technique and its Applications to Harmonic Superresolution*, IEEE Transactions on Signal Processing, 39 (1991), pp. 1070-1086.

[3] B. De Moor, *Total least squares for affinely structured matrices and the noisy realization problem*, IEEE Transactions on Signal Processing, 42 (1994), pp. 3004-3113.

[4] R. Fletcher, *Methods for Nonlinear Constraints*, in Nonlinear Optimization, Ed. M.J.D. Powell, Nato Conference Series , Cambridge, Great Britain, pp.185-211 .

[5] P. Lemmerling, B. De Moor and S. Van Huffel, *On the equivalence of constrained total least squares and structured total least squares*, ESAT/SISTA report TR 95-55, Dept. of Electr. Eng., Katholieke Universiteit Leuven, Belgium, 1995, submitted to IEEE Transactions on Signal Processing.

[6] P. Lemmerling and S. Van Huffel, *Superlinear algorithms for solving the structured total least norm problem*, ESAT/SISTA report TR 95-44, Dept. of Electr. Eng., Katholieke Universiteit Leuven, Belgium, 1995, submitted to SIAM journal on matrix analysis and applications.

[7] J.B. Rosen, H. Park, and J. Glick, *Total least norm formulation and solution for structured problems*, SIAM Journal on Matrix Anal. Appl., 1996, vol. 17, no. 1, pp. 110-126.

[8] S. Van Huffel, H. Park and J. Ben Rosen, *Formulation and Solution of Structured Total Least Norm Problems for Parameter Estimation*, ESAT/SISTA report TR 95-08, Dept. of Electr. Eng., Katholieke Universiteit Leuven, Belgium, 1995, accepted for publication in the IEEE Transactions on Signal Processing.