ROBUST ITERATIVE FITTING OF MULTILINEAR MODELS BASED ON LINEAR PROGRAMMING

Sergiy A. Vorobyov[†] *Yue Rong*[†] *Nicholas D. Sidiropoulos*[‡] *Alex B. Gershman*[†]

[†]Dept. of Communication Systems, University of Duisburg-Essen, Germany
[‡]Dept. of Electronic and Computer Engineering, Technical University of Crete, Chania, Greece

ABSTRACT

PARAllel FACtor (PARAFAC) analysis is an extension of low-rank matrix decomposition to higher-way arrays. It decomposes a given array in a sum of multilinear terms. PARAFAC analysis generalizes and unifies common array processing models (like joint diagonalization and ESPRIT); it has found numerous applications from blind multiuser detection and multi-dimensional harmonic retrieval, to clustering and nuclear magnetic resonance. The prevailing fitting algorithm in all these applications is based on alternating least squares (ALS) optimization, which is matched to Gaussian noise. In many cases, however, measurement errors are far from being Gaussian. In this paper, we develop an iterative algorithm for least absolute error fitting of general multilinear models, based on efficient interior point methods for Linear Programming (LP). We also benchmark its performance in Laplacian, Cauchy, and Gaussian noise environments, versus the respective CRBs and the commonly used ALS algorithm.

1. INTRODUCTION

The PARAFAC model is a useful data analysis tool that has recently found applications in array signal processing and communications [1], [2]. When generalizing the concept of low-rank decomposition to higher-way arrays, PARAFAC is instrumental in the analysis of data arrays indexed by three or more independent variables, just like Singular Value Decomposition (SVD) is instrumental in ordinary matrix (two-way array) analysis. In most applications of PARAFAC analysis, the ALS regression procedure is used to fit the model parameters [1], [2]. Least Squares (LS) regression is optimal (in the maximum likelihood sense) when measurement errors are additive i.i.d. Gaussian. Gaussianity is an often-made assumption, due to the central limit theorem, but also for tractability considerations. However, in many applications the measurement errors are far from being Gaussian random variables [3].

The Least Absolute Error (LAE) criterion is often used as a robust alternative to LS. LAE regression is optimal (in the maximum likelihood sense) when measurement errors are additive i.i.d. Laplacian. The Laplacian distribution is more heavy-tailed than the Gaussian one; therefore, it is better suited to model impulsive noise and outliers. Another distribution commonly used for modelling impulsive noise is the Cauchy, and, more generally, the class of α-stable distributions [3]. It is therefore of interest to develop PARAFAC regression procedures that optimize the LAE fitting criterion.

In this paper we develop such an iterative procedure that makes use of LP. The performance of the proposed algorithm is illustrated by means of simulations and compared to the pertinent Cramér-Rao bounds (CRBs) and Trilinear ALS (TALS) procedure [1].

2. PARALLEL FACTOR ANALYSIS

We introduce notation that will be useful in the sequel. Consider an $I \times J \times K$ *three-way array* $\underline{\boldsymbol{X}}$ with typical element $x_{i,j,k}$ and the F*-component trilinear decomposition*

$$x_{i,j,k} = \sum_{f=1}^{F} a_{i,f} b_{j,f} c_{k,f} \tag{1}$$

for all $i = 1, \ldots, I$, $j = 1, \ldots, J$ and $k = 1, \ldots, K$. Here $a_{i,f}$ stands for the (i,f)-th element of $I \times F$ matrix $\boldsymbol{A}$, and similarly $b_{j,f}$ and $c_{k,f}$ stand for (j,f)-th and (k,f)-th elements of $J \times F$ and $K \times F$ matrices $\boldsymbol{B}$ and $\boldsymbol{C}$, respectively. Matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are in general complex-valued. Equation (1) expresses $x_{i,j,k}$ as a sum of F rank-one triple products; it is known as trilinear decomposition, or *PARAFAC analysis* model of $x_{i,j,k}$.

Let $\boldsymbol{A}_i = \mathcal{D}_i(\boldsymbol{A})$ denote the operator which takes the i-th row of matrix $\boldsymbol{A}$ and produces a diagonal matrix by placing this row on the main diagonal. Then by "slicing" the three-dimensional array $\underline{\boldsymbol{X}}$ in a series of "slabs" (two-dimensional arrays), we obtain

$$\boldsymbol{X}_i = \boldsymbol{B}\boldsymbol{A}_i\boldsymbol{C}^T, \qquad i = 1, \ldots, I \tag{2}$$

A. B. Gershman is on leave from the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ontario, Canada.

Here such a slicing is made perpendicular to the ith dimension, i.e., $\boldsymbol{X}_i := [x_{i,:,:}]$ is the $J \times K$ two-dimensional slice of $\underline{\boldsymbol{X}}$ corresponding to the given index i. Two other types of slicing of $\underline{\boldsymbol{X}}$ are useful in understanding the algorithm that will be developed in the next section. They are given by

$$\boldsymbol{Y}_j = \boldsymbol{C}\boldsymbol{B}_j\boldsymbol{A}^T, \qquad j = 1, \ldots, J \tag{3}$$

$$\boldsymbol{Z}_k = \boldsymbol{A}\boldsymbol{C}_k\boldsymbol{B}^T, \qquad k = 1, \ldots, K \tag{4}$$

where $\boldsymbol{B}_j = \mathcal{D}_j(\boldsymbol{B})$, $\boldsymbol{C}_k = \mathcal{D}_k(\boldsymbol{C})$, while the $K \times I$ matrix $\boldsymbol{Y}_j$ and $I \times J$ matrix $\boldsymbol{Z}_k$ are defined as $\boldsymbol{Y}_j := [x_{:,j,:}]$ and $\boldsymbol{Z}_k := [x_{:,:,k}]$, respectively.

3. TRILINEAR ALTERNATING LAE REGRESSION BASED ON LINEAR PROGRAMMING (TALAE-LP)

In practice, the three-way array will contain measurement noise, i.e. $\underline{\widetilde{\boldsymbol{X}}} = \underline{\boldsymbol{X}} + \underline{\boldsymbol{V}}$ where the (i, j, k)th element of $\underline{\widetilde{\boldsymbol{X}}}$ can be written as

$$\tilde{x}_{i,j,k} = x_{i,j,k} + v_{i,j,k} \tag{5}$$

and $v_{i,j,k}$ denotes the additive complex i.i.d. zero-mean measurement noise with statistically independent real and imaginary parts.

The PARAFAC fitting problem is then formulated as follows. We are given the noisy data $\underline{\widetilde{\boldsymbol{X}}}$ and wish to estimate $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$. Let us introduce the tall matrix

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_I \end{bmatrix}_{JI \times K} = \begin{bmatrix} \boldsymbol{B}\boldsymbol{A}_1 \\ \vdots \\ \boldsymbol{B}\boldsymbol{A}_I \end{bmatrix} \boldsymbol{C}^T = (\boldsymbol{A} \odot \boldsymbol{B})\boldsymbol{C}^T \tag{6}$$

where $\odot$ stands for the Khatri-Rao matrix product. Similarly, we introduce the matrix of noisy data

$$\widetilde{\boldsymbol{X}} = \begin{bmatrix} \widetilde{\boldsymbol{X}}_1 \\ \vdots \\ \widetilde{\boldsymbol{X}}_I \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}_1 \\ \vdots \\ \boldsymbol{X}_I \end{bmatrix} + \begin{bmatrix} \boldsymbol{V}_1 \\ \vdots \\ \boldsymbol{V}_I \end{bmatrix} \tag{7}$$

Then the conditional Maximum Likelihood (ML) estimation problem for the matrix $\boldsymbol{C}$ given matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ and assuming i.i.d. Gaussian measurement noise is the LS fitting problem $\min_{\boldsymbol{C}} \|\widetilde{\boldsymbol{X}} - (\boldsymbol{A} \odot \boldsymbol{B})\boldsymbol{C}^T\|_F^2$ where $\| \cdot \|_F^2$ denotes the Frobenius matrix norm.

If the measurement noise is i.i.d. Laplacian (with i.i.d. Laplacian-distributed real and imaginary parts in the complex case), then ML estimation is equivalent to LAE regression. Some manipulations are necessary in order to express the absolute error criterion in the form of a convenient vector ℓ_1 norm. Towards this end, introduce the operator $\mathcal{F}(\cdot)$

$$\boldsymbol{s} = \mathcal{F}(\boldsymbol{S}) = \begin{bmatrix} \breve{\boldsymbol{S}}_{:,1} \\ \vdots \\ \breve{\boldsymbol{S}}_{:,L} \end{bmatrix}, \quad \breve{\boldsymbol{S}}_{:,l} = \begin{bmatrix} \mathrm{Re}\{\boldsymbol{S}_{:,l}\} \\ \mathrm{Im}\{\boldsymbol{S}_{:,l}\} \end{bmatrix} \tag{8}$$

where $\boldsymbol{S}$ is a complex-valued $M \times L$ matrix, and $\boldsymbol{S}_{:,l}$ denotes its lth column. The following property holds:

$$\mathcal{F}\{\boldsymbol{D}\boldsymbol{F}\} = (\boldsymbol{I} \otimes \mathcal{G}\{\boldsymbol{D}\})\mathcal{F}\{\boldsymbol{F}\} \tag{9}$$

where $\boldsymbol{I}$ is the identity matrix of a commensurate dimension, $\boldsymbol{D}$ and $\boldsymbol{F}$ are any complex-valued matrices of commensurate dimensions, $\otimes$ denotes the Kronecker matrix product, and $\mathcal{G}\{\boldsymbol{D}\}$ stands for the operator

$$\mathcal{G}\{\boldsymbol{D}\} = \begin{bmatrix} \mathrm{Re}\{\boldsymbol{D}\} & -\mathrm{Im}\{\boldsymbol{D}\} \\ \mathrm{Im}\{\boldsymbol{D}\} & \mathrm{Re}\{\boldsymbol{D}\} \end{bmatrix} \tag{10}$$

Using property (9), we find that the absolute error model fitting criterion can be written as

$$\|\tilde{\boldsymbol{x}} - (\boldsymbol{I}_K \otimes \mathcal{G}\{\boldsymbol{A} \odot \boldsymbol{B}\})\boldsymbol{c}\|_1 \tag{11}$$

i.e., through the ℓ_1 norm of a real-valued vector. Here, $\tilde{\boldsymbol{x}} = \mathcal{F}(\widetilde{\boldsymbol{X}})$, $\boldsymbol{c} = \mathcal{F}(\boldsymbol{C})$, and the dimension of the identity matrix is clarified by means of the subscript K.

Using the other two ways of slicing the array $\underline{\boldsymbol{X}}$, we introduce the matrices $\boldsymbol{Y} = (\boldsymbol{B} \odot \boldsymbol{C})\boldsymbol{A}^T$ and $\boldsymbol{Z} = (\boldsymbol{C} \odot \boldsymbol{A})\boldsymbol{B}^T$. Furthermore, we introduce

$$\widetilde{\boldsymbol{Y}} = \begin{bmatrix} \widetilde{\boldsymbol{Y}}_1 \\ \vdots \\ \widetilde{\boldsymbol{Y}}_J \end{bmatrix}_{KJ \times I}, \qquad \widetilde{\boldsymbol{Z}} = \begin{bmatrix} \widetilde{\boldsymbol{Z}}_1 \\ \vdots \\ \widetilde{\boldsymbol{Z}}_K \end{bmatrix}_{IK \times J} \tag{12}$$

where $\widetilde{\boldsymbol{Y}}_j$, $j = 1, \ldots, J$, and $\widetilde{\boldsymbol{Z}}_k$, $k = 1, \ldots, K$ are the noisy slabs of $\underline{\widetilde{\boldsymbol{X}}}$ along corresponding dimensions.

Now we have all notations necessary to explain the new fitting algorithm.

The idea behind this algorithm is similar to that of TALS regression for Gaussian noise [2] and is as follows: each time, update a subset of parameters using the LAE criterion, conditioned on previously obtained estimates of the remaining parameters; proceed to update another subset of parameters; repeat until convergence.

In more detail, we first initialize matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ randomly or by single-invariance ESPRIT when applicable [1], [2]. Then, given the matrix $\widetilde{\boldsymbol{X}}$, and these initial estimates of $\boldsymbol{A}$ and $\boldsymbol{B}$ (which we denote hereafter as $\widehat{\boldsymbol{A}}$ and $\widehat{\boldsymbol{B}}$), our purpose is to find the estimate of the matrix $\boldsymbol{C}$ which minimizes the norm (11). Specifically, we have to find the estimate of $\boldsymbol{C}$ by solving the following optimization problem:

$$\widehat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c}} \|\tilde{\boldsymbol{x}} - (\boldsymbol{I}_K \otimes \mathcal{G}\{\widehat{\boldsymbol{A}} \odot \widehat{\boldsymbol{B}}\})\boldsymbol{c}\|_1$$

$$\widehat{\boldsymbol{C}} = (\mathcal{F}^{-1}\{\widehat{\boldsymbol{c}}\})^T \tag{13}$$

for given $\tilde{\boldsymbol{x}}$, $\widehat{\boldsymbol{A}}$ and $\widehat{\boldsymbol{B}}$. In (13), $\mathcal{F}^{-1}\{\cdot\}$ denotes the inverse operator to $\mathcal{F}\{\cdot\}$ of (8). Introducing the vector $\boldsymbol{e} = [1, 1, \ldots, 1]^T$ and the vector of slack variables $\boldsymbol{q}_1$ (both of

commensurate dimensions), we can equivalently write the problem (13) as

$$\min_{\boldsymbol{c},\,\boldsymbol{q}_1} \boldsymbol{e}^T\boldsymbol{q}_1 \quad \text{subject to} \quad \tilde{\boldsymbol{x}} - (\boldsymbol{I}_K \otimes \mathcal{G}\{\widehat{\boldsymbol{A}} \odot \widehat{\boldsymbol{B}}\})\boldsymbol{c} \preceq \boldsymbol{q}_1$$
$$\tilde{\boldsymbol{x}} - (\boldsymbol{I}_K \otimes \mathcal{G}\{\widehat{\boldsymbol{A}} \odot \widehat{\boldsymbol{B}}\})\boldsymbol{c} \succeq -\boldsymbol{q}_1$$

where $\succeq$ denotes the usual *pointwise* ordering. This optimization problem is an LP problem that can be very efficiently solved using interior-point methods [4].

Using the second way of slicing the three-dimensional array (i.e., working with the data $\tilde{\boldsymbol{y}} = \mathcal{F}(\widetilde{\boldsymbol{Y}})$) and exploiting the property (9), we obtain that the estimate of $\boldsymbol{A}$ can be found by solving the following optimization problem:

$$\begin{aligned} \widehat{\boldsymbol{a}} &= \arg\min_{\boldsymbol{a}} \|\tilde{\boldsymbol{y}} - (\boldsymbol{I}_I \otimes \mathcal{G}\{\widehat{\boldsymbol{B}} \odot \widehat{\boldsymbol{C}}\})\boldsymbol{a}\|_1 \\ \widehat{\boldsymbol{A}} &= (\mathcal{F}^{-1}\{\widehat{\boldsymbol{a}}\})^T \end{aligned} \tag{14}$$

with given $\tilde{\boldsymbol{y}}$ and previously estimated $\widehat{\boldsymbol{B}}$ and $\widehat{\boldsymbol{C}}$. This problem can be rewritten as the following LP problem:

$$\min_{\boldsymbol{a},\,\boldsymbol{q}_2} \boldsymbol{e}^T\boldsymbol{q}_2 \quad \text{subject to} \quad \tilde{\boldsymbol{y}} - (\boldsymbol{I}_I \otimes \mathcal{G}\{\widehat{\boldsymbol{B}} \odot \widehat{\boldsymbol{C}}\})\boldsymbol{a} \preceq \boldsymbol{q}_2$$
$$\tilde{\boldsymbol{y}} - (\boldsymbol{I}_I \otimes \mathcal{G}\{\widehat{\boldsymbol{B}} \odot \widehat{\boldsymbol{C}}\})\boldsymbol{a} \succeq -\boldsymbol{q}_2$$

where $\boldsymbol{q}_2$ is the vector of slack variables of commensurate dimension.

Finally, using the third way of slicing the three-dimensional array and applying the property (9), we can find the estimate of $\boldsymbol{B}$ by solving the following optimization problem:

$$\begin{aligned} \widehat{\boldsymbol{b}} &= \arg\min_{\boldsymbol{b}} \|\tilde{\boldsymbol{z}} - (\boldsymbol{I}_J \otimes \mathcal{G}\{\widehat{\boldsymbol{C}} \odot \widehat{\boldsymbol{A}}\})\boldsymbol{b}\|_1 \\ \widehat{\boldsymbol{B}} &= (\mathcal{F}^{-1}\{\widehat{\boldsymbol{b}}\})^T \end{aligned} \tag{15}$$

with given $\tilde{\boldsymbol{z}}$ and previously estimated $\widehat{\boldsymbol{A}}$ and $\widehat{\boldsymbol{C}}$. This problem is equivalent to the following LP problem:

$$\min_{\boldsymbol{b},\,\boldsymbol{q}_3} \boldsymbol{e}^T\boldsymbol{q}_3 \quad \text{subject to} \quad \tilde{\boldsymbol{z}} - (\boldsymbol{I}_J \otimes \mathcal{G}\{\widehat{\boldsymbol{C}} \odot \widehat{\boldsymbol{A}}\})\boldsymbol{b} \preceq \boldsymbol{q}_3$$
$$\tilde{\boldsymbol{z}} - (\boldsymbol{I}_J \otimes \mathcal{G}\{\widehat{\boldsymbol{C}} \odot \widehat{\boldsymbol{A}}\})\boldsymbol{b} \succeq -\boldsymbol{q}_3$$

where $\boldsymbol{q}_3$ is the vector of slack variables of commensurate dimension.

Fitting proceeds by updating one matrix at a time, conditioned on interim estimates of the other two, in a round-robin fashion. Note that the conditional update of any given matrix may either improve or maintain but cannot worsen the current fit. Monotone convergence of the fit (but not necessarily to the global minimum) follows directly from this observation. The per-iteration complexity of TALAE-LP is equal to the cost of solving LP problems [4]. This is of the same order of complexity as computing a matrix pseudo-inverse in the TALS method [1] and can be estimated as $\mathcal{O}(F^3 + FIJK)$. Overall complexity depends on the number of iterations, which varies depending on problem-specific parameters and the given batch of data.

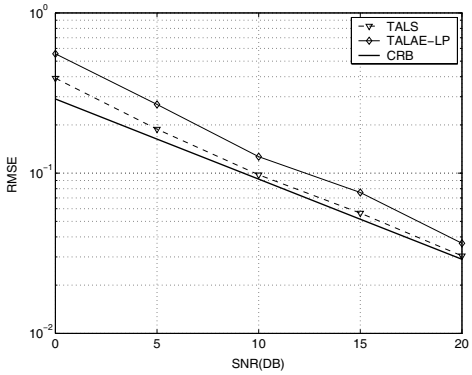

Figure 1: RMSEs versus SNR. Gaussian channel noise.

4. SIMULATIONS

Let us compare the performance of the proposed TALAE-LP algorithm with that of the conventional TALS method, and against the pertinent CRB. The example of blind PARAFAC multiuser detection for a Direct-Sequence Code Division Multiple Access (DS-CDMA) communication system [1] is simulated. For the DS-CDMA application, the elements of the matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ have the following meanings: $a_{i,f}$ is fading/gain between user f and antenna element i; $b_{j,f}$ is j-th chip of the spreading code of user f; $c_{k,f}$ is k-th symbol transmitted by user f. Correspondingly, matrix $\boldsymbol{A}$ is the channel matrix, $\boldsymbol{B}$ is the spreading code matrix, and $\boldsymbol{C}$ is the user symbol matrix, all unknown to the receiver. Here, F is the number of users, I is the total number of antenna elements in the array, J is the number of Intersymbol Interference (ISI)-free chips per symbol, and K is the length of the transmitted sequence of symbols.

The data $\underline{\boldsymbol{X}}$ are contaminated by channel noise. Three models of the channel noise are used. One is Gaussian noise, while the other two are Laplacian and Cauchy noise.

For LS fitting, we use the COMFAC algorithm [1] which is a fast implementation of TALS. The MOSEK convex optimization MATLAB toolbox [5] is used to solve LP problems associated with our TALAE-LP algorithm. Scale and permutation ambiguities are inherent to this blind separation problem [1]; the scale ambiguity manifests itself as a complex constant that multiplies each individual row of $\boldsymbol{C}$. For constant-modulus transmissions, this ambiguity can be removed via Automatic Gain Control (AGC) and differential encoding/decoding. We assume differentially-encoded user signals. For the purpose of performance evaluation only, the permutation ambiguity is resolved using a greedy LS $(\boldsymbol{C}, \widehat{\boldsymbol{C}})$ column-matching algorithm.

We present Monte Carlo simulations that are designed to assess the Root Mean Square Error (RMSE) performance of the aforementioned algorithms. The parameters used in the simulations are as follows: N = number of Monte Carlo tri-

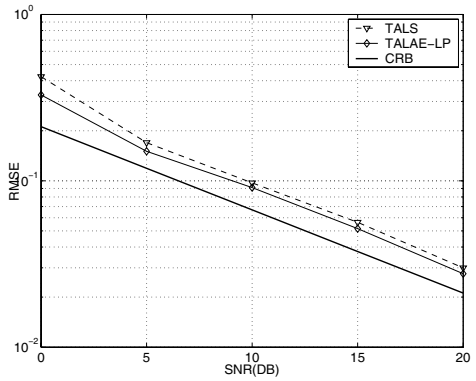

Figure 2: RMSEs versus SNR. Laplacian channel noise.

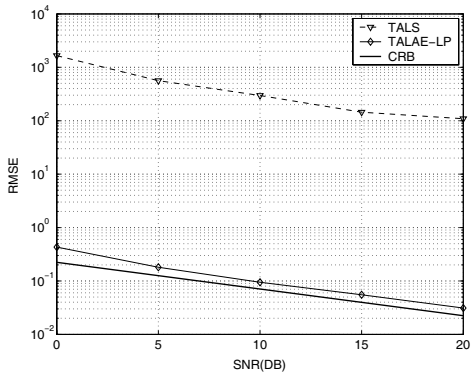

Figure 3: RMSEs versus SNR. Cauchy channel noise.

als = 100; $F = 2$; $I = 8$; $J = 8$; $K = 20$; and $\alpha = 1$. Here, α is the characteristic exponent which determines the heaviness of the tail of the symmetric α-stable distribution used in our third example ($\alpha = 1$ yields the Cauchy distribution). The associated symmetric α-stable characteristic function is given by $\phi(\omega) = \exp\{-\gamma|\omega|^\alpha\}$, where γ is a positive constant related to the scale of the distribution. The *geometric* Signal-to-Noise Ratio (SNR) in this case is defined as [6] $\mathrm{SNR}_{\mathrm{Cauchy}} = \frac{A^2}{4C_{\mathrm{g}}\gamma^2}$, where $C_{\mathrm{g}} = e^{C_{\mathrm{e}}} \approx 1.78$ and A is the magnitude of the noise-free signal. For the Gaussian case, the geometric SNR is equivalent to the standard SNR, and in the Laplacian case we use the standard SNR, since Laplacian distribution does not belong to the class of symmetric α-stable distributions.

Throughout the simulations, we assume that the noise power is normalized to be equal to 1. User signals are redrawn from an i.i.d. Bernoulli distribution and differentially encoded for each Monte Carlo trial. BPSK modulation is used for all user signals. The gains of the channel matrix $\boldsymbol{A}$ and the elements of the spreading code matrix $\boldsymbol{B}$ are generated as i.i.d. Gaussian unit variance random variables and are fixed in each Monte Carlo trial, while re-drawn from one trial to another.

Figures 1, 2 and 3 display the performance of the aforementioned algorithms in terms of RMSE versus the SNR for the case of Gaussian, Laplacian and Cauchy noise, respectively, and compare the performance with the corresponding CRBs. Figures 1 and 2 demonstrate that in the case of Gaussian noise, the TALS method performs slightly better than the proposed robust TALAE-LP algorithm, while in the case of Laplacian noise, the TALAE-LP algorithm has slightly better performance as compared to the TALS method. In the case of Cauchy noise (Fig. 3), the TALS method breaks down, while the TALAE-LP algorithm is not affected and is close to the CRB. The degradation in performance relative to TALS in the Gaussian case can be considered as a moderate price for greatly improved robustness against heavy-tailed Cauchy noise.

5. CONCLUSIONS

An iterative algorithm for robust fitting of trilinear PARAFAC models has been proposed. The algorithm relies on alternating optimization using LP. The proposed algorithm outperforms the popular alternative LS PARAFAC fitting procedure under heavy-tailed noise. Even though our algorithm is matched to the Laplacian distribution, it performs very well under Cauchy noise. Furthermore, its performance degrades only moderately under Gaussian noise.

6. REFERENCES

[1] N. D. Sidiropoulos, G. B. Giannakis and R. Bro, "Blind PARAFAC receivers for DS-CDMA systems," *IEEE Trans. Signal Processing*, vol. 48, pp. 810-823, Mar. 2000.

[2] N. D. Sidiropoulos, R. Bro and G. B. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE Trans. Signal Processing*, vol. 48, pp. 2377-2388, Aug. 2000.

[3] P. Tsakalides and C.L. Nikias, "Maximum likelihood localization of sources in noise modeled as a stable process," *IEEE Trans. Signal Processing*, vol. 43, pp. 2700-2713, Nov. 1995.

[4] Y. Nesterov and A. Nemirovski, *Interior Point Polynomial Algorithms in Convex Programming*. Philadelphia, PA: SIAM, 1994.

[5] *The MOSEK optimization tools version 2.0 (Build 19). User's manual and reference,* EKA Consulting ApS, Denmark, 2001. `http://www.mosek.com`

[6] J. G. Gonzalez, "Robust techniques for wireless communications in non-Gaussian environments," Ph.D. dissertation, University of Delaware, 1997.