

# On Toeplitz and Kronecker Structured Covariance Matrix Estimation

(Invited Paper)

Petter Wirfält\*, Magnus Jansson

ACCESS Linnaeus Center, Signal Processing lab, Royal Institute of Technology, Stockholm, Sweden

\*Corresponding author; email: wirfalt@kth.se

**Abstract**—A number of signal processing applications require the estimation of covariance matrices. Sometimes, the particular scenario or system imparts a certain theoretical structure on the matrices that are to be estimated. Using this knowledge allows the design of algorithms exploiting such structure, resulting in more robust and accurate estimators, especially for small samples. We study a scenario with a measured covariance matrix known to be the Kronecker product of two other, possibly structured, covariance matrices that are to be estimated. Examples of scenarios in which such a problem occurs are MIMO-communications and EEG measurements. When the matrices that are to be estimated are Toeplitz structured, we show our algorithms to be able to achieve the Cramér-Rao Lower Bound already at very small sample sizes.

## I. INTRODUCTION

In some instances of wireless MIMO communications, the channel between the transmitter and receiver can be modelled according to the Kronecker-model [1]. This model arises when certain relations are assumed to exist between the different antennas. When this model holds, the wireless channel covariance matrix can be modelled as a Kronecker product of two matrices of smaller dimensions. When satisfying such conditions it is highly desirable to have algorithms tailored to this particular problem in order to enhance accuracy and minimize computational costs.

Other examples of when received data can be assumed to satisfy a Kronecker model are EEG-data [2], and in certain statistical models [3]. In general, large computational gains can be made by utilizing Kronecker structure in received data.

In some of these applications, we might have additional structure in the problem: each of the factor matrices in the Kronecker product might have a certain structure. In MIMO communications for example, such structure typically appears if the antennas are symmetrically arranged: when using a uniform linear array, and the channel matrix can be assumed to be constant during a certain transmission time, the factor matrices can in some instances acquire a Toeplitz structure. Similarly, if relaxing the uniform array assumption, we can at least have persymmetric factor matrices (note that a Toeplitz structured matrix is per definition also persymmetric). Thus it is desirable to design estimators that can use such structure as well.

The research leading to these results has received funding from the European Research Council under the European Community's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement #228044.

The idea in this paper is to revisit a maximum likelihood (ML) based iterative algorithm which is designed to find the factor matrices of a Kronecker product. The algorithm is called the flip-flop algorithm [3], [4], and in [5] it was enhanced to capture persymmetric structure in the factor matrices. A big advantage of the method is that it in only one iteration finds an asymptotically (in the number of samples) efficient estimate. It also exhibits very good small sample properties and quickly produces estimates as good as the Cramér-Rao Lower Bound (CRB) [4].

However, one drawback of the method is that it fails to reproduce possible Toeplitz structure of the factor matrices it estimates. Since such a structure is prevalent in applications, it would be desirable to be able to capture this structure while retaining the small sample properties.

In the following we state our solution to this problem, and we then compare our new algorithm to the previous state-of-the-art.

## II. PROBLEM FORMULATION

We assume that we receive zero mean vector valued discrete time measurements  $\mathbf{x}(t)$ , which have a stochastic complex Gaussian distribution with covariance matrix

$$E[\mathbf{x}(t)\mathbf{x}(\tau)^*] = \mathbf{R}_0\delta(t, \tau). \quad (1)$$

We further assume that the underlying process is satisfying the Kronecker model, which means that  $\mathbf{R}_0$  in (1) can be written

$$\mathbf{R}_0 = \mathbf{A}_0 \otimes \mathbf{B}_0. \quad (2)$$

In (2),  $\mathbf{A}_0$  is  $m \times m$ ,  $\mathbf{B}_0$  is  $n \times n$ , and the matrices are both, at least, Hermitian and positive definite.

Note that since  $\mathbf{R}_0 = \mathbf{A}_0 \otimes \mathbf{B}_0 = (\gamma\mathbf{A}_0) \otimes (\gamma^{-1}\mathbf{B}_0)$ ,  $\mathbf{A}_0$  and  $\mathbf{B}_0$  are actually unidentifiable; thus we can only find them up to a scale factor.

In some instances, we will assume a persymmetric structure on  $\mathbf{A}_0$  and  $\mathbf{B}_0$ . This means that the matrices can be written as

$$\mathbf{A}_0 = \mathbf{J}\mathbf{A}_0^T\mathbf{J} \quad (3)$$

$$\mathbf{B}_0 = \mathbf{J}\mathbf{B}_0^T\mathbf{J} \quad (4)$$

where  $\mathbf{J}$  is a mirrored identity matrix. In addition to the Hermitian symmetry constraint, the matrices  $\mathbf{A}_0$  and  $\mathbf{B}_0$  are hence also symmetric with respect to the anti-diagonal.

Thus the number of independent parameters in each matrix is significantly reduced.

The factor matrices  $\mathbf{A}_0$  and  $\mathbf{B}_0$  will also in some instances be assumed to have a Toeplitz structure. This means that the number of independent parameters is reduced to only  $2m - 1$  and  $2n - 1$ , respectively. Note that a Hermitian Toeplitz matrix is persymmetric by definition.

### III. ML ESTIMATION

In this section we formulate the likelihood function, and present the ML estimators based on this function[3], [4]. We also incorporate some of the structure of the problem into the likelihood function.

The negative log-likelihood function of the problem can be written

$$l(\mathbf{A}, \mathbf{B}) = n \log |\mathbf{A}| + m \log |\mathbf{B}| + \text{tr}\{\hat{\mathbf{R}}(\mathbf{A}^{-1} \otimes \mathbf{B}^{-1})\}, \quad (5)$$

if only including parameter dependent terms. Here,

$$\hat{\mathbf{R}} = \frac{1}{N} \sum_{t=0}^{N-1} \mathbf{x}(t)\mathbf{x}^*(t) \quad (6)$$

is the sample covariance matrix,  $|\cdot|$  is the determinant function,  $\text{tr}(\cdot)$  is the trace-operator, and  $\hat{\cdot}$  denotes a quantity estimated from data.

It is well known that  $\log |\mathbf{X}| + \text{tr}\{\mathbf{R}\mathbf{X}^{-1}\}$  is minimized by  $\mathbf{X} = \mathbf{R}$ . Thus, write

$$\text{tr}\{\hat{\mathbf{R}}(\mathbf{A}^{-1} \otimes \mathbf{B}^{-1})\} = \text{tr}\left\{\sum_{k=1}^m \sum_{l=1}^m \hat{\mathbf{R}}^{kl}(\mathbf{A}^{-1})_{lk} \mathbf{B}^{-1}\right\}, \quad (7)$$

and it is seen that (5) is minimized by

$$\hat{\mathbf{B}} = \frac{1}{m} \sum_{k=1}^m \sum_{l=1}^m \hat{\mathbf{R}}^{kl}(\mathbf{A}^{-1})_{lk}, \quad (8)$$

if we keep  $\mathbf{A}$  fixed. In the above equations,  $\hat{\mathbf{R}}^{kl}$  denotes the  $k, l$ th block of size  $n \times n$  in  $\hat{\mathbf{R}}$ , and  $(\mathbf{A}^{-1})_{lk}$  is element  $l, k$  of  $\mathbf{A}^{-1}$ .

Likewise, in order to find an estimate of  $\mathbf{A}$  for a fixed  $\mathbf{B}$ , we write

$$\begin{aligned} \text{tr}\{\hat{\mathbf{R}}(\mathbf{A}^{-1} \otimes \mathbf{B}^{-1})\} &= \text{tr}\{\bar{\mathbf{R}}(\mathbf{B}^{-1} \otimes \mathbf{A}^{-1})\} \\ &= \text{tr}\left\{\sum_{k=1}^m \sum_{l=1}^m \bar{\mathbf{R}}^{kl}(\mathbf{B}^{-1})_{lk} \mathbf{A}^{-1}\right\}, \end{aligned} \quad (9)$$

and we find

$$\hat{\mathbf{A}} = \frac{1}{n} \sum_{k=1}^m \sum_{l=1}^m \bar{\mathbf{R}}^{kl}(\mathbf{B}^{-1})_{lk}. \quad (10)$$

Here,  $\bar{\mathbf{R}}^{kl}$  denotes the  $k, l$ th block of size  $m \times m$  in  $\bar{\mathbf{R}}$ , which we define according to

$$\bar{\mathbf{R}} = \mathbf{P}_{m,n} \hat{\mathbf{R}} \mathbf{P}_{m,n}^T. \quad (11)$$

In (11) we have introduced  $\mathbf{P}_{m,n} = \mathbf{P}$  which is a permutation matrix defined such that

$$\mathbf{P}(\mathbf{X} \otimes \mathbf{Y}) = (\mathbf{Y} \otimes \mathbf{X})\mathbf{P}, \quad (12)$$

for matrices of size  $m \times m$  and  $n \times n$ , respectively. Also note that  $\mathbf{P}^T = \mathbf{P}^{-1}$ .

Given these sequential estimates, a cyclic minimization algorithm is intuitively developed. There is however the issue of initialization, but in [4] it was shown that the estimates are asymptotically efficient, regardless of the initialization matrix, as long as it is positive definite; hence, we can e.g. use an identity matrix as initial guess. The following algorithm is thus arrived at:

- 1) Choose an initial guess for  $\mathbf{A}$ , i.e.  $\mathbf{A}_{\text{INIT}} = \mathbf{I}$ .
- 2) Find  $\mathbf{B}$  based on  $\mathbf{A}_{\text{INIT}}$ , i.e.  $\hat{\mathbf{B}}^{(1)} = \hat{\mathbf{B}}(\mathbf{A}_{\text{INIT}})$  using (8).
- 3) Find  $\mathbf{A}$  based on the previous estimate  $\hat{\mathbf{B}}^{(1)}$  through (10), giving  $\hat{\mathbf{A}}^{(2)}$ .
- 4) Finally, find  $\hat{\mathbf{B}}^{(3)}$  based on  $\hat{\mathbf{A}}^{(2)}$  again using (8).

Of course, in step 1 one could instead choose  $\mathbf{B}_{\text{INIT}} = \mathbf{I}$ , then the roles of  $\mathbf{A}$  and  $\mathbf{B}$  are simply interchanged in the algorithm. In the analysis of the next section, we will assume that step 1 is according to the above. We will refer to this algorithm as the Non-Iterative Flip Flop-algorithm (NIFF) [4].

### IV. STRUCTURED FACTOR MATRICES

If there is a certain structure in the factor matrices  $\mathbf{A}_0$  and  $\mathbf{B}_0$ , we desire to be able to incorporate this a-priori information in the estimation.

#### A. Persymmetric factor matrices

In [5], it was shown how to incorporate persymmetric structure of the factor matrices into NIFF; this was done through creating a structured data covariance matrix according to

$$\hat{\mathbf{R}}_{\text{PS}} \triangleq \frac{1}{2}(\hat{\mathbf{R}} + \mathbf{J}\hat{\mathbf{R}}^T\mathbf{J}), \quad (13)$$

and then using  $\hat{\mathbf{R}}_{\text{PS}}$  instead of  $\hat{\mathbf{R}}$  in (8) and (10). It can be shown that the estimates produced in each step are also persymmetric – hence, we call this method NIFF-PS.

#### B. Toeplitz structured factor matrices

Assuming  $\mathbf{A}_0$  and  $\mathbf{B}_0$  to be Toeplitz-structured, the estimates found based on (13) cannot fully recreate this structure: an extension is required in order to force the estimates to be Toeplitz. If we can modify the output of the NIFF algorithm described in Section III to take this into account, we can expect to improve our estimate.

Consider the situation when it is known a-priori that  $\mathbf{X}$  is Toeplitz. An estimate  $\hat{\mathbf{X}}$  of  $\mathbf{X}$  might not embody that structure. Using the method presented in [6], we can find  $\hat{\mathbf{X}}_T$  that is Toeplitz structured, based on  $\hat{\mathbf{X}}$  and the covariance of  $\text{vec}(\hat{\mathbf{X}})$ , where  $\text{vec}(\mathbf{X})$  is the vectorization operator which stacks the columns of  $\mathbf{X}$  on top of each other.

We can integrate this into our current framework by, in each step of the algorithm in Section III, forcing the estimates to be Toeplitz structured. To do so we need an expression for the covariance matrix of the estimate in each step.

Start by introducing

$$\text{vec}(\hat{\mathbf{A}}^{(1)}) = \hat{\mathbf{a}}^{(1)} \quad (14)$$

where the superscript <sup>(1)</sup> denotes the estimate order in the cyclic algorithm. Further, let  $\sim$  denote the error in a given quantity. Then, we can express the covariance of the estimates  $\hat{\mathbf{A}}^{(d)}$  as (momentarily ignoring the superscripts)

$$\text{cov}(\hat{\mathbf{a}}) = \mathbb{E}[(\hat{\mathbf{a}} - \mathbf{a})(\hat{\mathbf{a}} - \mathbf{a})^*] = \mathbb{E}[\tilde{\mathbf{a}}\tilde{\mathbf{a}}^*]. \quad (15)$$

where  $\mathbf{a} = \mathbb{E}(\hat{\mathbf{a}}) = \mathbf{a}_0$ , with proper scaling. Using the vec-operator on (10) gives

$$\hat{\mathbf{a}} = \frac{1}{n} \sum_{k,l=1}^n \hat{\mathbf{r}}^{kl} \hat{b}_{lk}^i, \quad (16)$$

where  $\hat{\mathbf{r}}^{kl} = \text{vec}(\hat{\mathbf{R}}^{kl})$  and  $\hat{b}_{lk}^i = (\hat{\mathbf{B}}_0^{-1})_{lk}$ . Then we can, to a first order approximation, write

$$\tilde{\mathbf{a}} = \frac{1}{n} \sum_{k,l=1}^n \tilde{\mathbf{r}}^{kl} b_{lk}^i + \frac{1}{n} \sum_{k,l=1}^n \tilde{\mathbf{r}}^{kl} \tilde{b}_{lk}^i \quad (17)$$

Using (17) in (15), we get

$$\begin{aligned} \mathbb{E}[\tilde{\mathbf{a}}\tilde{\mathbf{a}}^*] &= \frac{1}{n^2} \sum_{k,l,i,j=1}^n (\mathbb{E}[\tilde{\mathbf{r}}^{kl} b_{lk}^i (\tilde{\mathbf{r}}^{ij} b_{ji}^i)^*] \\ &\quad + \mathbb{E}[\tilde{\mathbf{r}}^{kl} b_{lk}^i (\tilde{\mathbf{r}}^{ij} \tilde{b}_{ji}^i)^*] + \mathbb{E}[\tilde{\mathbf{r}}^{kl} \tilde{b}_{lk}^i (\tilde{\mathbf{r}}^{ij} b_{ji}^i)^*] \\ &\quad + \mathbb{E}[\tilde{\mathbf{r}}^{kl} \tilde{b}_{lk}^i (\tilde{\mathbf{r}}^{ij} \tilde{b}_{ji}^i)^*]). \end{aligned} \quad (18)$$

In (18), all quantities without  $\sim$  are constant with respect to the expectation operator, which significantly simplifies the evaluation.

It is well known (see, e.g., [6]) that

$$\mathbb{E}[\tilde{\mathbf{r}}\tilde{\mathbf{r}}^*] = \frac{1}{N} (\mathbf{R}_0^T \otimes \mathbf{R}_0). \quad (19)$$

Thus, we wish to reformulate (18) such that the dependencies on the data is on the form of (19). To this end, we will need to rewrite some of the components to a more suitable form. First note that

$$\begin{aligned} \tilde{\mathbf{r}} &= \text{vec}(\tilde{\mathbf{R}}) = \text{vec}(\mathbf{P}\mathbf{R}_0\mathbf{P}^T) = (\mathbf{P} \otimes \mathbf{P}) \text{vec}(\mathbf{R}_0) \\ &= (\mathbf{P} \otimes \mathbf{P}) \mathbf{r} \end{aligned} \quad (20)$$

in which we have used the identity

$$\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A}) \text{vec}(\mathbf{B}) \quad (21)$$

for any matrices  $\mathbf{A}, \mathbf{B}$  and  $\mathbf{C}$  of suitable dimensions. We can also write

$$\tilde{\mathbf{r}}^{kl} = \mathbf{J}_{kl}^A \tilde{\mathbf{r}} = \mathbf{J}_{kl}^A (\mathbf{P} \otimes \mathbf{P}) \mathbf{r} = \mathbf{J}_{kl}^{\bar{A}} \mathbf{r}, \quad (22)$$

where  $\mathbf{J}_{kl}^A$  is a selection matrix picking the elements of  $\tilde{\mathbf{r}}^{kl}$  out of  $\tilde{\mathbf{r}}$ , and in  $\mathbf{J}_{kl}^{\bar{A}}$  we have also integrated the permutation properties. In the same manner, we define

$$\mathbf{r}^{ij} = \mathbf{J}_{ij}^B \mathbf{r}. \quad (23)$$

Looking at the structure of these selection matrices, we can see that  $\mathbf{J}_{kl}^A$  (or  $\mathbf{J}_{kl}^B$ ) contains  $m$  ( $n$ ) blocks of identity matrices of dimension  $m$  (or  $n$ ). In addition, in  $\mathbf{J}_{kl}^{\bar{A}}$  this structure is slightly permuted.

In order to find a more tractable expression for  $\tilde{b}_{lk}^i$ , we note that if  $\tilde{\mathbf{B}}$  is small compared to  $\mathbf{B}_0$  (with proper scaling, such that  $\mathbb{E}[\tilde{\mathbf{B}}] = \mathbf{B}_0$ ), we can write  $\tilde{b}_{lk}^i = -(\mathbf{B}_0^{-1} \tilde{\mathbf{B}} \mathbf{B}_0^{-1})_{lk}$  to a first order approximation. Again using (21), and noting that  $(\mathbf{B}_0^{-1} \tilde{\mathbf{B}} \mathbf{B}_0^{-1})_{lk} = \mathbf{e}_l^T (\mathbf{B}_0^{-1} \tilde{\mathbf{B}} \mathbf{B}_0^{-1}) \mathbf{e}_k$ , we get

$$\tilde{b}_{lk}^i = -\text{vec}^T(\mathbf{e}_l \mathbf{e}_k^T) (\mathbf{B}_0^{-T} \otimes \mathbf{B}_0^{-1}) \text{vec}(\tilde{\mathbf{B}}). \quad (24)$$

In the above  $\mathbf{e}_k$  is the unit vector of appropriate length (here:  $n$ ) with a single non-zero element in position  $k$ .

Equation (18) is a sum of four different terms. Evaluating it for all the steps of the NIFF is too space consuming for the present article, and will be left to a sequel. However, some of the steps are instructive to show. Looking at the terms separately according to

$$\mathbb{E}[\tilde{\mathbf{a}}\tilde{\mathbf{a}}^*] = \frac{1}{n^2} \sum_{k,l,i,j=1}^n (\mathbf{E}_1 + \mathbf{E}_2 + \mathbf{E}_3 + \mathbf{E}_4) \quad (25)$$

facilitates the presentation.

In the results stated below, we will assume that we are investigating the covariance for  $\tilde{\mathbf{a}}^{(2)}$ ; we then have a quite simple analytic expression for  $\tilde{b}_{lk}^i$ . For  $\tilde{\mathbf{b}}^{(3)}$ , the explicit formulas for the terms  $\mathbf{E}_2$ ,  $\mathbf{E}_3$ , and  $\mathbf{E}_4$  in (25) becomes more involved, but easily follows from the examples below.

The first term in (25) (cf. (18)) can be found to be

$$\begin{aligned} \mathbf{E}_1 &= \frac{1}{N} b_{lk}^i (b_{ji}^i)^c \mathbf{J}_{kl}^{\bar{A}} (\mathbf{R}^T \otimes \mathbf{R}) (\mathbf{J}_{ij}^{\bar{A}})^T \\ &= \frac{1}{N} b_{lk}^i (b_{ji}^i)^c [\mathbf{B}]_{jl} [\mathbf{B}]_{ki} (\mathbf{A}^T \otimes \mathbf{A}), \end{aligned} \quad (26)$$

where  $^c$  denotes complex conjugate and where we can note that  $(b_{ji}^i)^c = b_{ij}^i$ . Further, since we in (25) are summing over all the indices  $k, l, i$ , and  $j$ , we can replace (26) with

$$\frac{1}{n^2} \sum_{k,l,i,j=1}^n \mathbf{E}_1 = \frac{1}{nN} (\mathbf{A}_0^T \otimes \mathbf{A}_0). \quad (27)$$

The second term  $\mathbf{E}_2$  can be found as

$$\begin{aligned} \mathbf{E}_2 &= -\frac{1}{mN} [\mathbf{B}]_{ij}^c (b_{lk}^i) \sum_q^m \mathbf{J}_{kl}^{\bar{A}} (\mathbf{R}^T \otimes \mathbf{R}) (\mathbf{J}_{qq}^B)^T \\ &\quad \cdot (\mathbf{B}^{-T} \otimes \mathbf{B}^{-1}) \text{vec}(\mathbf{e}_j \mathbf{e}_i^T) \mathbf{a}^*, \end{aligned} \quad (28)$$

and by looking at (18) it can be seen that, since we are summing over the entire range of the dummy indices,  $\mathbf{E}_2 = \mathbf{E}_3^*$ . However, the terms are not Hermitian.

Finally, we have the term  $\mathbf{E}_4$ , which can be expressed as

$$\begin{aligned} \mathbf{E}_4 &= \frac{1}{m^2 N} [\mathbf{B}]_{kl} \mathbf{a} \text{vec}^T(\mathbf{e}_l \mathbf{e}_k^T) (\mathbf{B}^{-T} \otimes \mathbf{B}^{-1}) \\ &\quad \cdot \sum_{p,q}^m \mathbf{J}_{qq}^B (\mathbf{R}^T \otimes \mathbf{R}) (\mathbf{J}_{pp}^B)^T \\ &\quad \cdot (\mathbf{B}^{-T} \otimes \mathbf{B}^{-1}) \text{vec}(\mathbf{e}_j \mathbf{e}_i^T) [\mathbf{B}]_{ij}^c \mathbf{a}^* \\ &= \frac{1}{m^2 N} \text{tr}(\mathbf{AA}) [\mathbf{B}]_{kl} [\mathbf{B}]_{ij}^c b_{lj}^i b_{ik}^i \mathbf{a} \mathbf{a}^*, \end{aligned} \quad (29)$$

which we in analogy to (27) can rewrite as

$$\frac{1}{n^2} \sum_{k,l,i,j=1}^n \mathbf{E}_4 = \frac{1}{m^2 n N} \text{tr}(\mathbf{A}\mathbf{A}) \mathbf{a}\mathbf{a}^*. \quad (30)$$

Using these results in (25) we now have an expression for  $\text{cov}(\hat{\mathbf{a}})$  and we thus implement our algorithm according to

- 1) using NIFF-PS to find  $\hat{\mathbf{B}}^{(1)}$  according to Section III,
- 2) refining this estimate by minimizing

$$\hat{\mathbf{b}}_T^{(1)} = \arg \min \left[ \hat{\mathbf{b}}^{(1)} - \mathbf{b}_T^{(1)} \right] \mathbf{C}^{-1} \left[ \hat{\mathbf{b}}^{(1)} - \mathbf{b}_T^{(1)} \right]^* \quad (31)$$

which gives  $\hat{\mathbf{b}}_T^{(1)}$ , which is Toeplitz structured,

- 3) subsequently finding  $\hat{\mathbf{A}}^{(2)}$  and  $\hat{\mathbf{B}}^{(3)}$  in the same way.

In (31),  $\mathbf{C}$  is the covariance matrix of the estimate error (25). However, during our investigations we found that we achieved the same accuracy in our estimates by only using the first term of (25), which can be expressed as

$$\mathbf{C} = \left( \hat{\mathbf{A}}^T \otimes \hat{\mathbf{A}} \right) \quad (32)$$

when minimizing with respect to  $\mathbf{A}$ , and

$$\mathbf{C} = \left( \hat{\mathbf{B}}^T \otimes \hat{\mathbf{B}} \right) \quad (33)$$

when minimizing with respect to  $\mathbf{B}$ , since a scalar factor does not affect (31).

## V. NUMERICAL SIMULATIONS

We perform numerical simulations in order to evaluate the performance of our proposed estimator. In these simulations we compare the results to previously published estimators.

We generate random complex Hermitian Toeplitz matrices  $\mathbf{A}_0$  and  $\mathbf{B}_0$ , and then find  $\mathbf{R}_0$  from these.  $\mathbf{R}_0$  is kept fixed during the different Monte Carlo simulations; different realizations of  $\mathbf{R}_0$  all show the same results as the one presented. Then, in each simulation iteration we generate  $N$  samples from a complex Gaussian distribution with covariance  $\mathbf{R}_0$ , and from these samples we find  $\hat{\mathbf{R}}$ , (6). This is the data available to the respective algorithms.

Our performance measure is the normalized root-mean square error (NRMSE) of the estimates, according to

$$\text{NRMSE} = \sqrt{\frac{1}{L} \sum_{k=1}^L \frac{\|\hat{\mathbf{R}}_k - \mathbf{R}_0\|_F^2}{\|\mathbf{R}_0\|_F^2}}, \quad (34)$$

where  $L$  is the number of Monte Carlo simulations,  $\mathbf{R}_0$  is the true covariance matrix,  $\hat{\mathbf{R}}_k$  is the estimate from Monte Carlo simulation  $k$  and  $\|\cdot\|_F^2$  is the Frobenius norm according to  $\|\mathbf{A}\|_F^2 = \sum_{i,j} |\mathbf{A}_{ij}|^2$ .

We investigate three estimators in Fig. 1: one is termed Covariance matching, and is described in [4]; one is the NIFF-PS [5], which assumes persymmetric structure only; and then we also study the performance of the estimator described in Section IV-B.

In Fig. 1 we can see that the proposed method is performing better than the other algorithms. It consistently has a lower NRMSE, and it reaches the CRB at very small sample sizes.

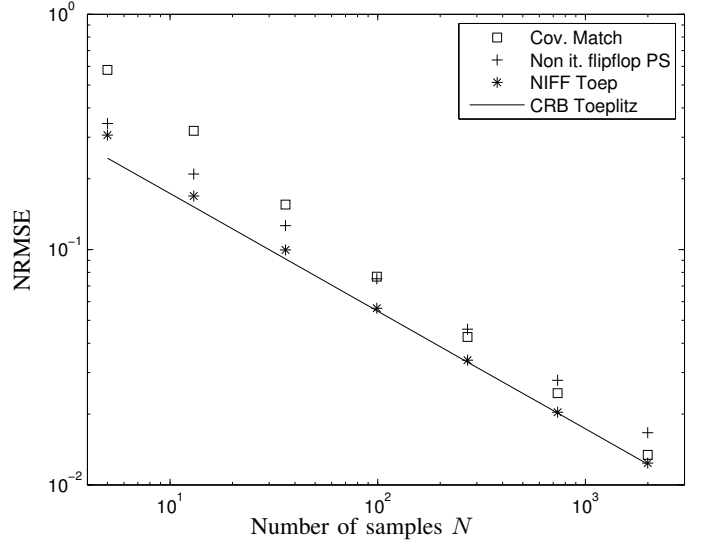


Fig. 1. Normalized RMSE of three estimators as a function of number of samples.  $\mathbf{A}_0$  is  $6 \times 6$ ,  $\mathbf{B}_0$  is  $7 \times 7$ ; both are Toeplitz structured. Averages taken over  $L = 1000$  Monte Carlo simulations for each sample size.

We can also see that the NIFF-PS algorithm is better than the Covariance matching one for small sample sizes, but as we increase the amount of data it lacks in accuracy as it is not able to take the (known) Toeplitz structure into account.

## VI. CONCLUSION

We have improved an existing iterative ML-based method by, in each iteration step, forcing the estimates to be Toeplitz structured through a covariance matching approach. This new combined approach makes the set of problems for which the ML-based algorithm is optimal larger.

The new method inherits both the low sample efficiency of the ML-method, together with the asymptotic accuracy given by the covariance matching. As our simulations show, we reach the CRB already at very low snapshot numbers.

## REFERENCES

- [1] J. Kermoal, L. Schumacher, K. Pedersen, P. Mogensen, and F. Frederiksen, "A stochastic mimo radio channel model with experimental validation," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 6, pp. 1211 – 1226, Aug 2002.
- [2] J. de Munck, H. Huizenga, L. Waldorp, and R. Heethaar, "Estimating stationary dipoles from meg/eeg data contaminated with spatially and temporally correlated background noise," *IEEE Trans. Signal Process.*, vol. 50, no. 7, pp. 1565 – 1572, Jul 2002.
- [3] N. Lu and D. L. Zimmerman, "The likelihood ratio test for a separable covariance matrix," *Statistics and Probability Letters*, vol. 73, no. 4, pp. 449 – 457, 2005.
- [4] K. Werner, M. Jansson, and P. Stoica, "On estimation of covariance matrices with Kronecker product structure," *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 478–491, Feb. 2008.
- [5] M. Jansson, P. Wirfält, K. Werner, and B. Ottersten, "ML estimation of covariance matrices with Kronecker and persymmetric structure," in *Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop, 2009. DSP/SPE 2009. IEEE 13th*, 4-7 2009, pp. 298 – 301.
- [6] H. Li, P. Stoica, and J. Li, "Computationally efficient maximum likelihood estimation of structured covariance matrices," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1314–23, May 1999.