Adaptive Rank Estimation for Spherical Subspace Trackers

Aleksandar Kavčić[1]   and   Bin Yang[2]

[1] Department of Electrical
    and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213, USA
Phone: (412) 268-7103
Email: kavcic@ece.cmu.edu

[2] Department of Electrical Engineering
Ruhr University Bochum
44780 Bochum, Germany
Phone: +49 (234) 700-7387
Email: yng@sth.ruhr-uni-bochum.de

Abstract

We develop a rank tracking method for spherical subspace trackers. The method adaptively sets a threshold based on the averaged noise eigenvalue. The signal eigenvalues are then compared to the threshold to reach a decision on the subspace rank. The threshold itself is chosen to balance the error probabilities due to rank underfitting and overfitting. Simulation results show that our method performs as well as (and for very low signal to noise ratios even better than) information theoretic criteria.

1   Introduction

Applying high resolution methods like MUSIC [1] and ESPRIT [2] in direction of arrival (DOA) and frequency estimation, requires computation of the signal subspace, or its orthogonal complement, the noise subspace. Since batch eigenvalue decomposition of the sample correlation matrix is unacceptable due to its high computational complexity, alternative efficient subspace tracking methods have been sought. A large number of these efficient algorithms fall in the category of sphericalized subspace trackers [3, 4, 5, 6, 7]. Their common feature is that the computational savings are achieved by averaging the noise eigenvalues. Thus, if the problem size (correlation matrix size) is n, and the signal subspace dimension (rank) is r, these algorithms require only $O(nr^2)$ or $O(nr)$ operations per signal subspace update.

In typical tracking applications, the signal subspace dimension r will change with time. Therefore, subspace tracking is not complete unless its dimension is tracked as well. Information theoretic criteria [8, 9, 10] have been successfully applied to rank estimation in batch eigenvalue decomposition cases [11]. Modifications of these methods have also been applied to spherical subspace trackers [6]. However, information theoretic rank estimating criteria, due to their complexity, are not well adapted for efficient computation. An important feature of most spherical subspace trackers is their parallel structure resulting in real-time execution on systolic arrays. Information theoretic criteria represent bottlenecks for parallel applications [12]. It is therefore of interest to seek simpler rank estimation schemes.

The very structure of spherical subspace trackers offers a simple solution to rank estimation. Spherical subspace trackers track the average noise eigenvalue (the average of $n - r$ lowest eigenvalues of the correlation matrix). This eigenvalue is an estimate for the noise level in the system. In this paper we will show how to use the average noise eigenvalue to adaptively set a threshold that will let us determine the signal subspace rank. We will refer to the new method as adaptive threshold rank estimation. The method is designed for use with spherical subspace trackers (although it can be applied to full eigenstructure decompositions by simply averaging the lowest eigenvalues) and is much simpler than information theoretic criteria. We will also show that the performance of the adaptive rank estimation matches (and in some cases outperforms) information theoretic criteria applied to spherical subspace trackers. The following notation is used throughout the paper. Boldface and underlined characters are used to represent matrices and column vectors, respectively. The superscript $^\mathrm{H}$ denotes the Hermitian transposition. $\| \cdot \|$ represents the 2-norm of matrices and vectors.

2   Adaptive threshold rank estimation

A common way to recursively estimate a correlation matrix is to form a sample correlation matrix $\hat{\mathbf{C}}(k)$ by exponential weighting of rank-one updates:

$$\hat{\mathbf{C}}(k) = \beta \hat{\mathbf{C}}(k - 1) + \underline{x}(k)\underline{x}(k)^{\mathrm{H}}. \tag{1}$$

Here, k denotes the time index ($0 < k < \infty$), β is the exponential forgetting factor ($0 < \beta < 1$), and $\underline{x}(k)$ is the data vector (snapshot of a linear sensor array or output of a tap delay line) at the time instant k. We say that the numerical rank of $\hat{\mathbf{C}}(k)$ with respect to a tolerance tol^2 is r, if exactly r of the n eigenvalues of $\hat{\mathbf{C}}(k)$ are greater than tol^2: $l_1(k) \geq l_2(k) \geq \cdots \geq l_r(k) > tol^2 \geq l_{r+1}(k) \geq \cdots \geq l_n(k)$. Here $l_i(k)$ denote the eigenvalues of $\hat{\mathbf{C}}(k)$, and $1 \leq i \leq n$. Typically, the tolerance is determined based on the finite precision of the machine where the computation is conducted. Since in our case the precision is related to the accuracy of noisy measurements of $\underline{x}(k)$, it is clear that the tolerance needs to be somehow related to the noise power. In [13], Stewart suggests the following rule for choosing the tolerance:

$$tol^2 \geq \frac{n - r}{1 - \beta}\sigma_\varepsilon^2 = (n - r)\lambda^{\mathrm{N}}, \tag{2}$$

where σ_ε^2 is the noise variance of the independent and identically distributed additive noise component of the vector $\underline{x}(k)$, and λ^{N} is the noise eigenvalue of $\mathrm{E}\left[\hat{\mathbf{C}}(k)\right]$. [1]

Motivated by Stewart's findings, we shall define the threshold $T(k)$ (which can be viewed as an adaptively determined tolerance)

$$T^2(k) = \phi^2 \cdot \lambda^{\mathrm{N}}(k). \tag{3}$$

The time argument k indicates that the noise eigenvalue $\lambda^{\mathrm{N}}(k)$ might change (drift) with time, and therefore the threshold changes with time, too. We call ϕ the thresholding factor. While, according to Stewart, the thresholding factor should be $\phi \geq \sqrt{n - r}$ as in (2), we will constrain ϕ to $\phi > 1$, and determine its value based on some optimality criterion. We defer the discussion on the optimal thresholding factor until section III. Notice that the formulation of the threshold in (3) is well suited for spherical subspace trackers because they track the average of the $n - r$ lowest eigenvalues of $\hat{\mathbf{C}}(k)$, which we call the noise-averaged eigenvalue. The noise-averaged eigenvalue is an estimate[2] of $\lambda^{\mathrm{N}}(k)$ and can be used in (3) to adaptively estimate the threshold.

Thus far we have been concentrating on the threshold itself, without specifying how the threshold is combined with spherical subspace trackers to deliver a rank estimate. The idea is to keep track of the rank r and then, after a simple threshold comparison in each updating cycle, allow the rank to change to $r + 1$ or $r - 1$, or stay at r, depending on the result of the threshold comparison. However, the proposed strategy will not work unless we introduce a slight change to the spherical subspace tracking algorithm we are using. To see why the change is necessary, suppose that the rank is suddenly incremented by one. Accordingly, there will be an extra signal eigenvalue to be tracked. But since this extra signal eigenvalue is treated as a noise eigenvalue and therefore reaveraged in every update, it cannot follow the sudden change of rank. To solve this problem, we propose to track an extra noise eigenvalue and an extra

---

[1] If we assume a stationary signal $\underline{x}(k)$, then from (1) we have $\mathrm{E}\left[\hat{\mathbf{C}}(k)\right] = \beta\mathrm{E}\left[\hat{\mathbf{C}}(k - 1)\right] + \mathrm{E}\left[\underline{x}(k)\underline{x}(k)^{\mathrm{H}}\right]$, and for large k, $\mathrm{E}\left[\hat{\mathbf{C}}(k)\right] = \mathrm{E}\left[\hat{\mathbf{C}}(k - 1)\right] = \mathbf{C}(k) = \mathrm{E}\left[\underline{x}(k)\underline{x}(k)^{\mathrm{H}}\right]/(1 - \beta)$. Let λ^{N} denote the lowest $n - r$ eigenvalues of $\mathbf{C}(k)$ and let σ_ε^2 denote the lowest $n - r$ eigenvalues of $\mathrm{E}\left[\underline{x}(k)\underline{x}(k)^{\mathrm{H}}\right]$, then clearly, $(1 - \beta)\lambda^{\mathrm{N}} = \sigma_\varepsilon^2$.

[2] More precisely, in the case of batch eigenvalue decomposition, it is the maximum likelihood estimate [14].

noise eigenvector. This extra noise eigenvalue will not enter the reaveraging step and will therefore be free to increase abruptly in case of rank increase. Thus, if the determined rank is r, we will track $r + 1$ distinct eigenvalues of $\hat{\mathbf{C}}(k)$, $l_1(k) \geq l_2(k) \geq \cdots \geq l_r(k) \geq l_{r+1}(k)$ and the noise-averaged eigenvalue $l_A(k) = \frac{1}{n-r-1} \sum_{i=r+2}^{n} l_i(k) \leq l_{r+1}(k)$. Thereby, since $l_{r+1}(k)$ is actually a noise eigenvalue, only those eigenvectors, corresponding to the first r eigenvalues form the actual basis for the signal subspace. With this change, we formulate the rank tracking scheme:

SET $T(k) = \phi\sqrt{l_A(k)}$;

IF $\sqrt{l_r(k)} < T(k)$ **THEN SET** $r = r - 1$;

ELSEIF $\sqrt{l_{r+1}(k)} > T(k)$ **THEN SET** $r = r + 1$;

ELSE rank stays the same;

END.

The whole tracking algorithm is given in Table 1. Notice that no specific spherical subspace tracking algorithm is used in Table 1. Instead, a general algorithm is assumed, where its place can be taken by any one of the previously cited spherical subspace trackers. Also notice that the initialization step might depend on the actual algorithm used. Systolic implementation of the threshold rank updating has thus far been demonstrated only on the Jacobi-type NASVD algorithm [12].

3   Optimal thresholding factor

In this section we develop a tool to analyze the performance of the threshold comparison rank tracking criterion. Assuming that we know the true rank r, we will treat the estimated signal and noise singular values as random variables, and link them to the probabilities of a miss and false alarm. The probability of a miss P_m is defined as the probability that the rank is estimated to be one lower than the true rank r. The probability of false alarm P_f is defined as the probability of determining the rank to be one more than r.

Since calculating the probabilities P_m and P_f in real tracking applications is a rather complicated task, we shall simplify the problem. First, we assume that the signal we are tracking is stationary (or at least wide sense stationary). Second, we employ a rectangular window (instead of an exponential) of length N to determine the sample correlation matrix $\hat{\mathbf{C}}$ as

$$\hat{\mathbf{C}} = \frac{1}{N} \sum_{k=1}^{N} \left[\underline{x}(k)\underline{x}(k)^{\mathrm{H}} \right]. \tag{4}$$

We will let $N \to \infty$ in order to study the asymptotic behavior of the eigenvalues of $\hat{\mathbf{C}}$. We assume that $\underline{x}(k)$ is composed of a sum of r independent complex exponentials, each of amplitude A_i, where the A_i's $(1 \leq i \leq r)$ are independent Gaussian random variables of zero mean. We also assume that $\underline{x}(k)$ is corrupted by additive, spatially and temporally uncorrelated Gaussian noise of zero mean. Our final assumption is that the correlation matrix $\mathbf{C} = \mathrm{E}(\hat{\mathbf{C}})$ has r distinct and $n - r$ identical eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_r > \lambda_{r+1} = \cdots = \lambda_n = \lambda^{\mathrm{N}}$. With these assumptions, we may apply the asymptotic theory [15] for the characteristic roots (eigenvalues) $l_1 \geq l_2 \geq \cdots \geq l_n$ of the sample correlation matrix

$\hat{\mathbf{C}}$ of a jointly Gaussian distributed vector $\underline{x}(k)$.

Asymptotic probability of a miss Pm: Assuming that the true rank is r, the probability of a miss is defined as the probability that the estimated rank is $r - 1$. In our threshold rank estimation scheme, this happens when $\sqrt{l_r}$ is lower than the threshold $T = \phi\sqrt{l_A}$, i.e., $Pm(\phi) = \mathrm{P}\left\{\sqrt{l_r} < \phi\sqrt{l_A}\right\}$, where $l_A = \frac{1}{n-r-1}\sum\limits_{i=r+2}^{n} l_i$. Since l_A is a good estimate[3] of λ^{N}, we have

$$Pm(\phi) = \mathrm{P}\left\{l_r \le \phi^2 l_A\right\} \approx \mathrm{P}\left\{l_r \le \phi^2 \lambda^{\mathrm{N}}\right\}. \tag{5}$$

We now apply the asymptotic result [15] that l_r has an asymptotically normal distribution with mean λ_r and variance $\sigma_\lambda^2 = \dfrac{2\lambda_r^2}{N}$. Thus, we can easily find the probability of a miss as[4]

$$Pm(\phi) \quad \approx \quad \int\limits_{-\infty}^{\phi^2 \lambda^N} \frac{1}{\sqrt{2\pi\sigma_\lambda^2}}\, \mathrm{e}^{-\frac{(x-\lambda_r)^2}{2\sigma_\lambda^2}}\, \mathrm{d}x = \frac{1}{2}\,\mathrm{erfc}\left(\frac{\sqrt{N}}{2}\left[1 - \frac{\phi^2 \lambda^N}{\lambda_r}\right]\right). \tag{6}$$

If we notice that $\frac{\lambda_r}{\lambda^N} = 1 + SNR_r$, where SNR_r is the ratio of powers of the r-th signal mode and noise, equation (6) becomes

$$Pm(\phi) \approx \frac{1}{2}\,\mathrm{erfc}\left(\frac{\sqrt{N}}{2}\left[1 - \frac{\phi^2}{1 + SNR_r}\right]\right). \tag{7}$$

Thus, we see that if the SNR is smaller, in order to maintain the same Pm, we need to decrease ϕ.

Asymptotic probability of false alarm P_f: Unfortunately, we were not able to find an expression for the probability of false alarm, but at least we can derive an upper bound for P_f. The probability of false alarm is the probability that the rank is estimated one higher than the true rank r. According to our threshold comparison criterion, this occurs when the non-averaged noise eigenvalue l_{r+1} exceeds the threshold $T = \phi\sqrt{l_A}$, i.e., $P_f(\phi) = \mathrm{P}\left\{l_{r+1} > \phi^2 l_A\right\}$ (see Appendix A for further comments on P_f and its upper bound). From [15] we have that the logarithm of the likelihood function, defined as

$$L(r) = N(n - r)\ln\left(\frac{\sum\limits_{i=r+1}^{n} l_i}{n - r}\right) - N\ln\left(\prod\limits_{i=r+1}^{n} l_i\right) \tag{8}$$

is an asymptotically χ^2-distributed random variable with $\frac{1}{2}(n - r + 2)(n - r - 1)$ degrees of freedom. We also have

$$L(r) \ge N \cdot \ln \frac{\left[1 + \frac{(l_{r+1}/l_A) - 1}{n - r}\right]^{n-r}}{(l_{r+1}/l_A)} \tag{9}$$

---

[3]The maximum likelihood estimate of λ^{N} is $\hat{\lambda}_{ML}^{\mathrm{N}} = \frac{l_{r+1} + (n-r-1)l_A}{n-r}$. Certainly, for $n - r \gg 1$, we can say $l_A \approx \hat{\lambda}_{ML}^{\mathrm{N}}$.

[4]The complementary error function is defined as $\mathrm{erfc}(x) = \frac{2}{\sqrt{\pi}}\int\limits_{x}^{\infty} \mathrm{e}^{-t^2}\,\mathrm{d}t$.

because if we add $N(n - r - 1) \ln K$ to the right hand side of (9), we get $L(r)$. K is the ratio of the arithmetic and the geometric mean of the lowest $n - r - 1$ eigenvalues of $\hat{\mathbf{C}}$, and therefore $K \geq 1$. We thus have

$$P_f(\phi) \;=\; \mathrm{P}\left\{ l_{r+1} > \phi^2 l_A \right\} \tag{10}$$

$$P_f(\phi) \;=\; \mathrm{P}\left\{ N \cdot \ln \frac{\left[1 + \frac{(l_{r+1}/l_A) - 1}{n - r} \right]^{n-r}}{(l_{r+1}/l_A)} > \eta(\phi) \right\} \tag{11}$$

$$P_f(\phi) \;\leq\; \mathrm{P}\left\{ L(r) > \eta(\phi) \right\} = \mathrm{P}\left\{ \chi^2_{\frac{1}{2}(n-r+2)(n-r-1)} > \eta(\phi) \right\}. \tag{12}$$

Here, $\eta(\phi)$ is obtained as the solution of

$$\frac{\left[1 + \frac{\phi^2 - 1}{n - r} \right]^{n-r}}{\phi^2} = \mathrm{e}^{\frac{\eta}{N}}. \tag{13}$$

Notice that the left hand side of (13) is a monotonically increasing function of ϕ (for $\phi \geq 1$ and $n - r > 1$), which allowed us to go from (10) to (11).

Asymptotically optimal thresholding factor ϕ_{opt}: Now that we have a way of determining P_m and P_f (actually just an upper bound for P_f) from ϕ, we can make a plot of $1 - P_m$ over P_f, for different values of ϕ. This is similar to a receiver operating characteristic, typically drawn for receivers when we are trying to detect signals based on the Neyman-Pearson strategy and a binary hypothesis, see [16] for example. We draw a curve $1 - P_m$ versus P_f (actually $1 - P_m$ versus the upper bound of P_f) for different values of r in the range $0 \leq r \leq r_{max}$, where r_{max} is the maximal anticipated signal subspace rank. Examples of these plots are shown in Figure 1.b. Notice that the plot for $r = 0$ does not make sense, since for $r = 0$, $P_m = 0$. Nevertheless, we plotted this curve according to equations (7), (12) and (13) just to establish what the dependence on the value of r is. Figure 1.a shows the plot of ϕ for different values of P_f. Thus, from Figure 1.a, we find the P_f that corresponds to a given ϕ and r, and then, from Figure 1.b, we find the quantity for P_m. Then we choose that value ϕ that best satisfies our criteria for how large P_f and P_m should be. For example, we may choose the optimal value ϕ_{opt} as that value ϕ, for which $P_m(\phi) = P_f(\phi)$. The plot of the line $P_m = P_f$ is shown in Figure 1.b. Intersections with the plots of $1 - P_m$ versus P_f represent optimal points for different values of r. Tracing now back to Figure 1.a, we find the optimal values of ϕ. They are: $\phi = 1.828$ for $r = 0$, $\phi = 1.719$ for $r = 2$, and $\phi = 1.613$ for $r = r_{max} = 4$. Since we want to make ϕ a constant (independent of r), we choose that value ϕ to allow for optimal detection of the largest anticipated number of signal components, that is, we choose our ϕ_{opt} as $\phi_{opt}(r_{max}) = 1.613$. An alternative method would be to let ϕ depend on the currently estimated r, but that complicates slightly the criterion.

In figures 2.a and 2.b, we illustrate how the optimal value ϕ_{opt} depends on N, r, and SNR_r. In Figure 2.a, we fixed SNR_r and n, while we changed N and plotted ϕ_{opt} for different values of r, ranging between 0 and $r_{max} = 4$. The plot reveals that ϕ_{opt} decreases as both N and r (or equivalently r_{max}) increase. The circled points in Figure 2 correspond to the circled points in Figure 1. Figure 2.b reveals that ϕ_{opt} increases as the signal to noise ratio SNR_r

increases.

Optimal thresholding factor for real tracking environments: Since a real tracking environment differs from the asymptotic environment for which we derived the expressions for P_m and P_f, one may ask how useful is the asymptotic value ϕ_{opt} in real tracking applications when we made so many assumptions. As a reminder, in our asymptotic curves, we assumed stationary signals, rectangular window of size $N \rightarrow \infty$, $n - r \gg 1$, $P_f(\phi) \leq$ P $\{L(r) > \eta(\phi)\}$ and $l_A \approx \lambda^N$. In addition, in real tracking applications, the system designer cannot count on knowing the value for SNR_r. This is because, among other parameters such as the real SNR (defined as the power of a source over the power of noise) and the problem dimension n, SNR_r also depends on the relative distances between the tracked frequencies (or DOAs) of the r signal components, and these are not known a priori. Also, all the asymptotic probability distributions were derived for the batch eigenvalue decomposition case, and these are not necessarily the same for recursive updating schemes. Therefore, *quantitatively*, the asymptotic results from the previous subsection are not accurate since they do not correspond to real subspace tracking conditions. A better value for ϕ_{opt} needs to be reached by running a few tracking experiments for a fixed n, r_{max}, β and a minimum anticipated real SNR, and then choosing that ϕ_{opt} that provides the best balance between P_m and P_f.

By running these experiments, we found out that, *qualitatively*, the value for ϕ_{opt} in real tracking applications obeys laws similar to those in the asymptotic case. That is,

1. As n increases, ϕ_{opt} increases.

2. As r increases (or if r_{max} increases to provide for the worst case), ϕ_{opt} decreases.

3. As the SNR increases, ϕ_{opt} increases.

4. As β increases (or as the equivalent window length $N(\beta) = \dfrac{1}{1 - \beta}$ increases), ϕ_{opt} decreases.

Hence, the asymptotic theory provides qualitative guidelines on how to choose ϕ_{opt}. Based on these observations, we fine-tune ϕ until the rank estimation performance reaches a satisfactory level. This means that we need to experiment and choose that ϕ that works best for the maximum anticipated r. Note that items 1 and 2 are consistent with Stewart's findings (2). We demonstrate these dependencies in section IV, where we present subspace and dimension tracking results.

4   Tracking results

In this section we compare rank tracking performance of the threshold comparison criterion to that of the MDL [9, 10] information criterion applied to spherical subspace trackers [6]. To demonstrate that the performance of the adaptive threshold rank tracking method is independent of the actual spherical subspace tracker used, we demonstrate results for the TQR-SVD tracker [7] of computational complexity $O(nr^2)$ and the Jacobi-type NASVD tracker [5] of complexity $O(nr)$.

In Figure 3, we follow the DOAs of six targets. The targets are allowed to appear and disappear abruptly within

a time window of 1000 updates. Because of this high nonstationarity, we set the forgetting factor fairly low at $\beta = 0.9$. We choose to put the signals of the targets in a very noisy environment (additive, spatially and temporally uncorrelated Gaussian noise) to study the limits of rank tracking performance. The signal to noise ratio (SNR) for each target is given in Figure 3.a. We used the ESPRIT algorithm to extract DOAs from the updated signal subspace. Below each DOA graph, is a plot of the tracked rank. Notice that at these low SNRs, the adaptive threshold rank estimation delivers better results than the MDL criterion, as long as the thresholding factor is set around 2.0 (precisely $\phi = 2.05$). This suggests that information theoretic criteria are not optimal for use with spherical subspace trackers. Not surprisingly, the adaptive threshold rank tracking method performs better because it was designed specifically for spherical subspace trackers.

In Figure 4, we follow slowly moving targets, resulting in a DOA change of not more than $20°$ per 1000 updates. Contrary to the previous example, the targets cannot disappear abruptly, so we can increase the forgetting factor to $\beta = 0.97$ to allow for a more distant past to have an influence on the current subspace estimate. Notice that here the performance of MDL and the threshold comparison are almost the same, although even here we can observe that the threshold comparison method performs slightly better. We see this from the shorter duration of the period where the rank is underestimated to be $r = 2$. Also, notice that due to the larger forgetting factor, the optimal thresholding factor dropped to $\phi = 1.75$, which is consistent with the observations made in section III. If we were to keep the original thresholding factor from Figure 3, we would observe many errors due to rank underfitting, particularly in the signal whose SNR is -2dB.

5   Conclusion

We have presented a novel rank tracking method designed specifically for spherical subspace trackers. The method uses the noise-averaged eigenvalue that all spherical subspace trackers track, to adaptively set a threshold. Comparing signal eigenvalues to the so determined threshold delivers an estimate for the rank of the tracked subspace. The optimal thresholding factor is chosen such that a balance between the probabilities of a miss and false alarm is achieved. This means that the system designer does need to spend some time to establish what the optimal thresholding factor for his/her application is. On the other hand, once the optimal thresholding factor is established, we have a method that (contrary to the popular information theoretic criteria) is simple, easily implemented in systolic architecture and delivers results that, in cases of severe SNRs, outperform information theoretic criteria.

A   On the probability of false alarm

Recently, an approximate probability density function for the sum of the lowest $n - r$ eigenvalues $\sum_{i=r+1}^{n} l_i$ was derived [17]. This could suggest that possibly a more accurate expression for the probability of false alarm P_f can be

found. That would certainly be the case if we would calculate P_f as $\mathrm{P}\left\{\frac{1}{n-r}\sum_{i=r+1}^{n}l_i>\tau\right\}$, where τ is some threshold. However, our probability of false alarm is defined as $P_f=\mathrm{P}\left\{l_{r+1}>\phi^2 l_A\right\}$, where $l_A=\frac{1}{n-r-1}\sum_{i=r+2}^{n}l_i$, so the result in [17] cannot be applied. Also, by using the average of the lowest $n-r$ eigenvalues in the comparison against a threshold τ we are loosing information. For example, if the $(r+1)$-st eigenvalue starts to rise gradually, this rise will not be observed due to the reaveraging of noise eigenvalues. Off course, this might not be relevant if the threshold τ is fixed, but τ has to be adaptively determined based on the observation of $\frac{1}{r}\sum_{i=r+1}^{n}l_i$ in each update. As a result of not detecting the gradually increasing eigenvalue l_{r+1}, the threshold τ drifts and the change in rank stays undetected. We found out through simulation that the criterion $l_{r+1}>\phi^2 l_A$ works much better than $\frac{1}{n-r}\sum_{i=r+1}^{n}l_i>\tau$ in detection of gradually increasing eigenvalues. This is the reason why we used $l_{r+1}>\phi^2 l_A$ as the rank inflation criterion even though it only provides us with an expression for the upper bound of P_f.

References

[1] R. O. Schmidt, "Multiple emiter location and signal parameter estimation," in *Proc. RADC Spectrum Estimation Workshop*, pp. 243–258, October 1979.

[2] R. Roy and T. Kailath, "ESPRIT – estimation of signal parameters via rotational invariance techniques," *IEEE Trans. ASSP*, vol. 37, 1989.

[3] I. Karasalo, "Estimating the covariance matrix by signal subspace averaging," *IEEE Trans. ASSP*, vol. 34, pp. 8–12, February 1986.

[4] R. D. DeGroat, "Noniterative subspace tracking," *IEEE Trans. Signal Processing*, vol. 4, pp. 571–577, March 1992.

[5] A. Kavčić and B. Yang, "A new efficient subspace tracking algorithm based on singular value decomposition," in *Proceedings IEEE ICASSP'94*, (Adelaide), pp. IV485–IV488, April 1994.

[6] B. Yang and F. Gersemsky, "An adaptive algorithm of linear computational complexity for both rank and subspace tracking," in *Proc. IEEE ICASSP'94*, (Adelaide), pp. IV33–IV36, April 1994.

[7] E. M. Dowling, L. P. Amman, and R. D. DeGroat, "A TQR-iteration based adaptive SVD for real time angle and frequency tracking," *IEEE Trans. Signal Processing*, vol. 42, pp. 914–926, April 1994.

[8] A. Akaike, "A new look at the statistical model identification," *IEEE Trans. Automatic Control*, vol. 19, pp. 716–723, 1974.

[9] J. Rissanen, "Modelling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.

[10] G. Schwartz, "Estimating the dimension of a model," *Ann. Stat.*, vol. 6, pp. 461–464, 1978.

[11] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Trans. ASSP*, vol. 33, pp. 387–392, 1985.

[12] A. Kavčić and B. Yang, "Simultaneous subspace tracking and rank estimation," in *Proc. SPIE Advanced Signal Processing Algorithms*, vol. 2563, (San Diego), pp. 206–217, July 1995.

[13] G. W. Stewart, "An updating algorithm for subspace tracking," *IEEE Trans. Signal Processing*, vol. 40, pp. 1535–1541, June 1992.

[14] L. L. Scharf, *Statistical Signal Processing*. New York: Addison-Wesley Publishing Company, 1991.

[15] T. W. Anderson, *An Introduction to Multivariate Statistical Analysis*. New York: John Wiley and Sons, 2nd ed., 1984.

[16] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*, vol. I. New York: John Wiley and Sons, 1968.

[17] A. A. Shah and D. W. Tufts, "Determination of the dimension of a signal subspace," in *Proc. IEEE 27th Asilomar Conference on Signals, Systems, and Computers*, vol. 2, (Pacific Grove, CA), pp. 1072–1077, Nov. 1993.

Initialization

1. Obtain the user supplied thresholding factor ϕ and the forgetting factor β.

2. Initialize r. Initialize eigenvalues $l_1(0), l_2(0), \ldots, l_r(0), l_{r+1}(0)$.
   Initialize noise-averaged eigenvalue $l_A(0)$. Initialize threshold $T(0) = \phi\sqrt{l_A(0)}$.
   Initialize matrix of eigenvectors $\mathbf{V}(0) = \left[\underline{v}_1(0), \underline{v}_2(0), \ldots, \underline{v}_r(0), \underline{v}_{r+1}(0)\right]$.

Loop for $k = 1, \ldots, \infty$

1. Input $\underline{x}(k)$.

2. Calculate noise eigenvector: $\underline{v}_A(k-1) = \left[\mathbf{I} - \mathbf{V}(k-1)\mathbf{V}(k-1)^{\mathrm{H}}\right] \underline{x}(k)$.

3. Normalize noise eigenvector $\underline{v}_A(k-1) = \dfrac{\underline{v}_A(k-1)}{\|\underline{v}_A(k-1)\|}$.

4. Update eigenvalues and eigenvectors. The actual structure of this step depends largely on the spherical subspace tracker used.

$$\left[\begin{array}{cc} \mathbf{V}(k-1) & \underline{v}_A(k-1) \end{array}\right] \overset{\text{update}}{\longrightarrow} \left[\begin{array}{cc} \mathbf{V}(k) & \underline{\tilde{v}}_A(k) \end{array}\right]$$

$$[l_1(k-1), l_2(k-1), \ldots, l_r(k-1), l_{r+1}(k-1), l_A(k-1)] \overset{\text{update}}{\longrightarrow}$$

$$\overset{\text{update}}{\longrightarrow} [l_1(k), l_2(k), \ldots, l_r(k), l_{r+1}(k), l'_A(k)]$$

5. Reaverage noise-averaged eigenvalue to maintain noise subspace sphericity:
   $$l_A(k) = \frac{l'_A(k) + (n-r-2)\cdot\beta\cdot l_A(k-1)}{n-r-1}.$$

6. Dimension updating by threshold comparison:
   IF $\sqrt{l_r(k)} < T(k-1)$ perform rank deflation:
   - Set $l_A(k) := \dfrac{l_{r+1}(k) + (n-r-1)l_A(k)}{n-r}$.
   - Delete $l_{r+1}(k)$ and the rightmost column of $\mathbf{V}(k)$.
   - Set $r := r - 1$.

   ELSEIF $\sqrt{l_{r+1}(k)} > T(k-1)$ perform rank inflation:
   - Append an extra column to the right of $\mathbf{V}(k)$ by setting $\mathbf{V}(k) := \left[\begin{array}{cc} \mathbf{V}(k) & \underline{\tilde{v}}_A(k) \end{array}\right]$.
   - Set $l_{r+2}(k) := l_A(k)$.
   - Set $r := r + 1$.

   ELSE rank stays the same.

7. Update the threshold value: $T(k) := \phi\sqrt{l_A(k)}$.

end

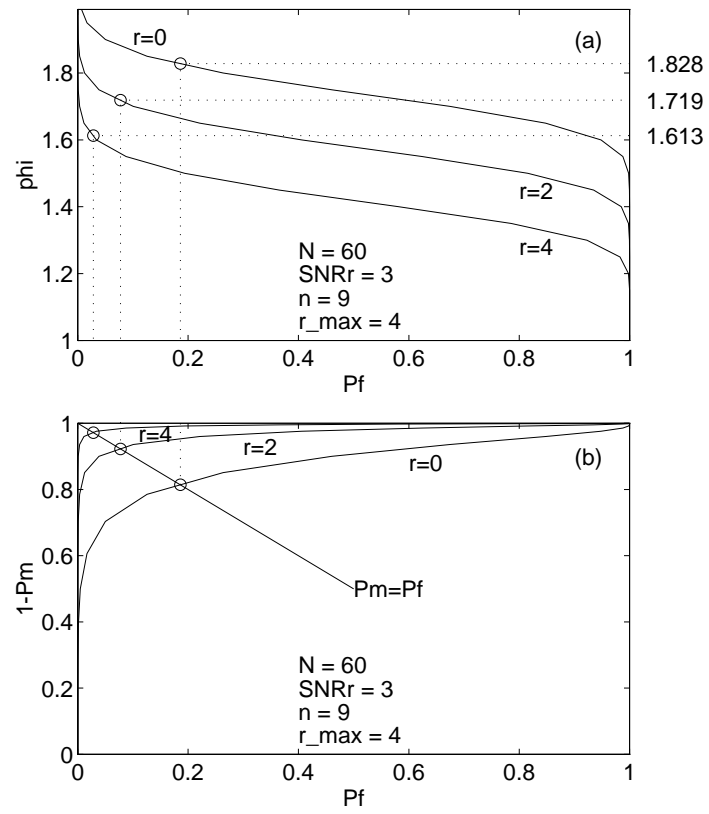Table 1: Spherical subspace tracking and adaptive threshold rank estimation

Figure 1: (a): Thresholding factor ϕ versus the probability of false alarm P_f for different values of r. (b): $1 - P_m$ versus P_f for different values of r. Intersections with the line $P_m = P_f$ determine the optimal values of ϕ.
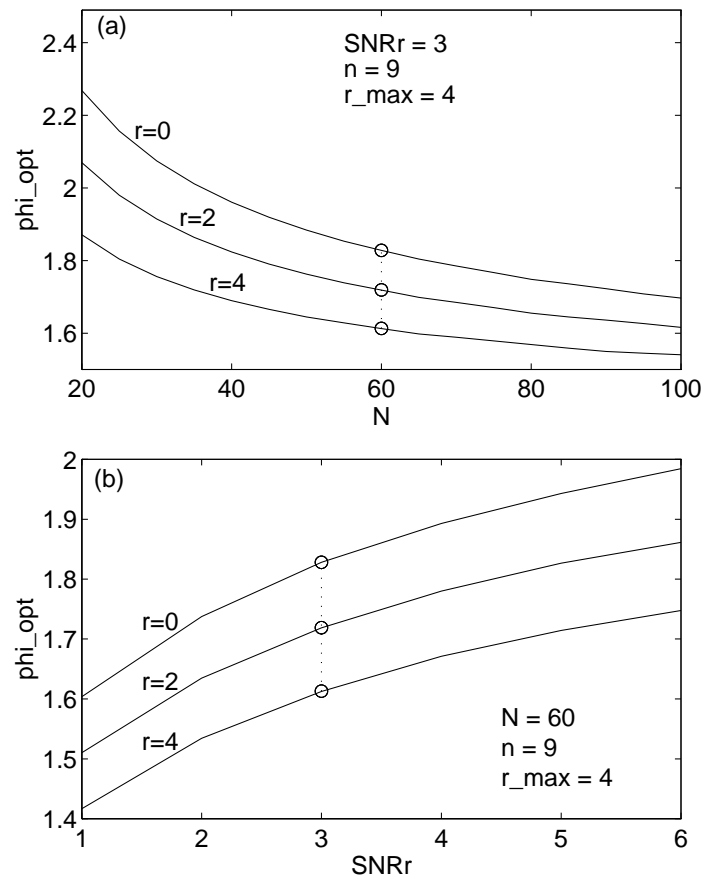
Figure 2: (a): Asymptotically optimal thresholding factor ϕ versus the size of the rectangular window N for different values of the true rank r. (b): Asymptotically optimal thresholding factor ϕ versus the signal to noise ratio SNR_r for different values of the true rank r.
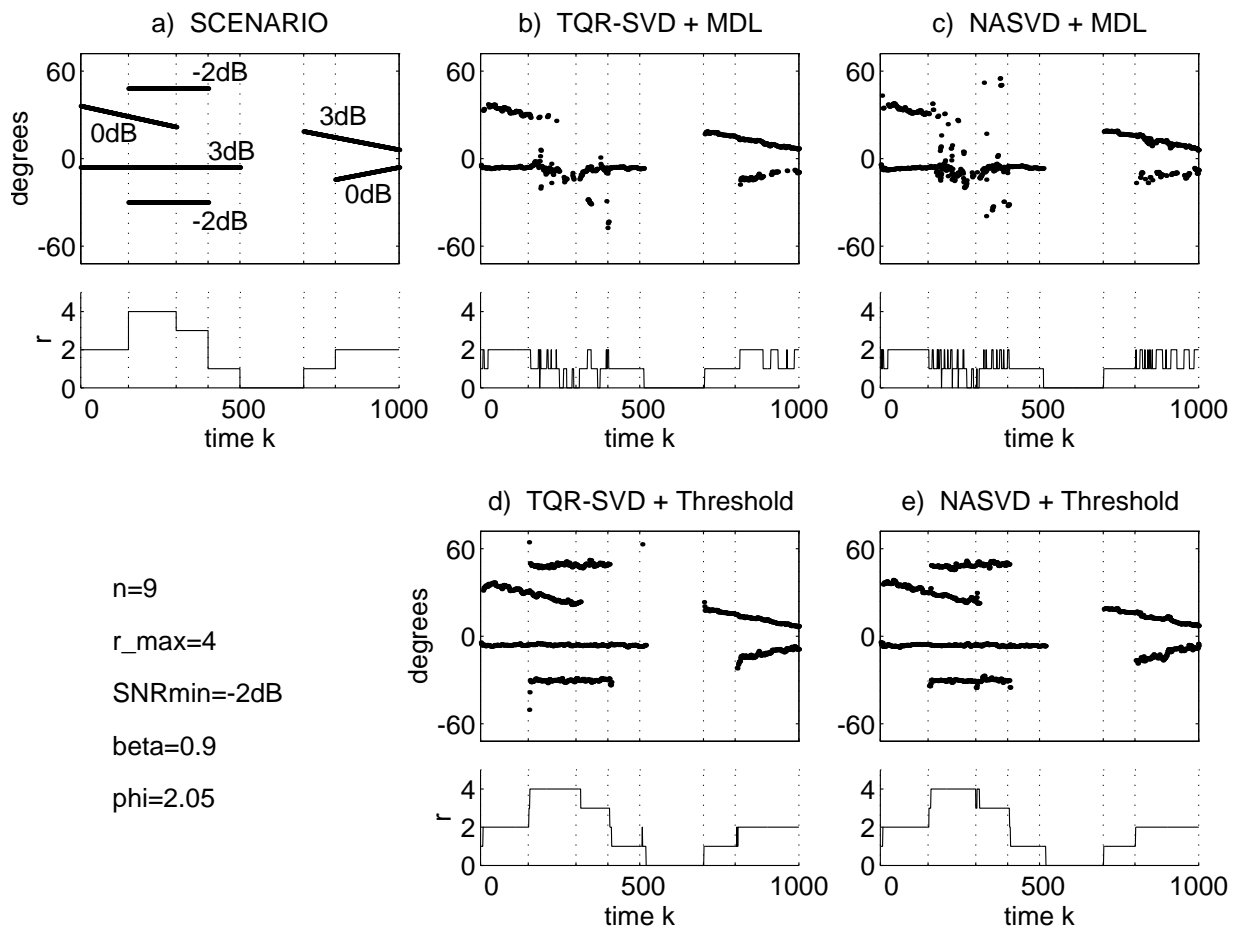
Figure 3: Comparison of different spherical subspace and rank tracking techniques in highly non-stationary signal environments with low signal-to-noise ratios.
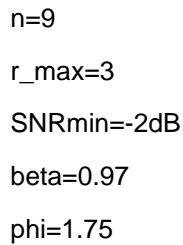
Figure 4: Comparison of different spherical subspace and rank tracking techniques in slowly varying signal environments with low signal-to-noise ratios. The scenario of crossing target paths tests the ability of the algorithms to resolve closely separated signal eigenvectors.