SLOWLY CONVERGING PARAFAC
SEQUENCES: SWAMPS AND TWO-FACTOR
DEGENERACIES

Ben Mitchell
Don Burdick
Institute of Statistics & Decision Sciences, Duke University

DP #93-A04
Slowly Converging PARAFAC Sequences:
Swamps and Two-Factor Degeneracies

Ben C. Mitchell
Donald S. Burdick
Institute of Statistics and Decision Sciences
Duke University
Durham, NC 27708

February 16, 1993
Abstract

The alternating least squares PARAFAC algorithm is a useful tool for resolving trilinear three-way data arrays. Occasionally, however, it becomes bogged down for many iterations in the vicinity of a poor quality resolution before moving on to a much superior optimum fit. We investigate this behavior in a simulation study and suggest ways of overcoming the obstacles it presents.

KEY WORDS: Trilinear models, Tensorial Resolution, PARAFAC, Swamps, Two-Factor Degeneracy
1 Introduction

There are two main approaches to the trilinear analysis of three-way data arrays. One approach is to resolve the array by performing a generalized eigenanalysis. Although they vary in details, several procedures based on this approach have been proposed in the chemometrics literature [2, 18, 10, 11]. We will call such procedures EBPs for eigenanalysis based procedures. EBPs work well when the signal-to-noise ratio is high.

The other main approach utilizes alternating least squares in an iterative procedure which exploits the conditional linearity of the trilinear model. Its iterative nature means that starting values are required, but it is guaranteed to improve the least squares fit of the model to the data at each iteration. This approach is the one commonly used by psychometricians working in three mode factor analysis [6, 5]. Its prototype is the PARAFAC algorithm developed and popularized by Richard Harshman. Chemists may be more familiar with this approach from the paper by Appelof and Davidson [1]. We will use the term PARAFAC for procedures following this approach because of its widespread use in the field of psychometrics.

We have argued elsewhere [17] the merits of using PARAFAC, with starting values obtained via an EBP, when the noise level is moderate. In this article we use simulated data to investigate some of the convergence properties of PARAFAC. In particular, we describe a phenomenon called "swamps" in which a PARAFAC sequence spends a long time in the vicinity of an inferior resolution before emerging from the swamp to converge to an acceptable resolution. We find that swamps are associated with another phenomenon which Kruskal et al. [9] have called a two-factor degeneracy. We introduce a test for two-factor degeneracy and show how it can be used to avoid swamps and speed up the PARAFAC convergence process.

A total of 64 different $40 \times 40 \times 4$ data arrays were simulated according to the trilinear PARAFAC model. In general, the trilinear PARAFAC model for a three-way data array $A = (a_{ijk})$ is given by

$$a_{ijk} = \sum_{r=1}^{R} x_{ir} y_{jr} z_{kr} + n_{ijk}$$

for $i = 1, 2, \ldots, I$, $j = 1, 2, \ldots, J$, and $k = 1, 2, \ldots, K$. Using Kruskal's triple product, this becomes

$$A = \bigotimes(X, Y, Z) + N$$
where $X$ is $I \times R$, $Y$ is $J \times R$, and $Z$ is $K \times R$ and $N$ is an array of random discrepancies. The signal array $S = \otimes (X, Y, Z)$ is called the triple product of the three factor matrices $X$, $Y$, and $Z$ [7]. Each factor matrix has $R$ linearly independent columns where $R$ is the rank of the signal. Each column of a factor matrix is called a profile, e.g. the second column of the factor matrix $Y$ is the second Mode $Y$ profile.

Each of the 64 signal arrays used in the simulation had $I = 40$, $J = 40$, $K = 4$, and $R = 4$. The Mode $X$, Mode $Y$, and Mode $Z$ profiles were chosen to mimic profiles of actual chemical compounds, and then artificial compounds were constructed from these profiles in a $4^3$ factorial design. Further details can be found in Section 4.

The noise level was 25% throughout. That is, each element $n_{ijk}$ of the noise array was obtained as the product

$$n_{ijk} = s_{ijk}z_{ijk}$$

where $s_{ijk}$ is the corresponding element of the signal array and $z_{ijk}$ is an independently generated normal deviate with mean $\mu = 0$ and standard deviation $\sigma = 0.25$.

The dimension and noise parameters were selected with the goal of obtaining arrays which were neither too easy nor too difficult to resolve into components.

An EBP requires two slabs or linear combinations of slabs as input for the eigenanalysis. Following Leurgans et al. [11], we use the sum of the four Mode $Z$ slabs and a single Mode $Z$ slab as input to the eigenanalysis. Varying the selection of the single slab can yield as many as four starting values per data array for the PARAFAC algorithm. Occasionally the eigenanalysis produces complex roots, so that less than four starts may actually occur for some arrays. All 64 arrays did have at least one usable starting value.

## 2 The Quantities: $\Delta$ and $\Upsilon$

In this section we define two quantities which are useful in assessing the amount of change that results from a PARAFAC iteration as well the degree of closeness between two triple product factorizations. Both quantities are defined in terms of the uncorrected correlation coefficient (UCC).
The uncorrected correlation coefficient for a pair of vectors is defined by

\[ \text{UCC}(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\sqrt{(\vec{x} \cdot \vec{x})(\vec{y} \cdot \vec{y})}} = \cos \theta. \]  

(1)

where \( \theta \) is the angle between the vectors \( \vec{x} \) and \( \vec{y} \) and \( \vec{x} \cdot \vec{y} \) is the sum of the products of the corresponding elements of \( \vec{x} \) and \( \vec{y} \). The UCC between a pair of arrays is similarly calculated. The UCC is a closeness measure for vectors that is unaffected by scalar multiplication.

We next define a criterion for comparing resolutions. Suppose we have the two sets of factor matrices each with \( R \) columns,

\[ \{X_A, Y_A, Z_A\} \quad \text{and} \quad \{X_B, Y_B, Z_B\}. \]

The columns correspond to the components of the resolutions but the correspondence may not be the same for both sets. For example, the first columns of the matrices in factor set \( A \) may correspond to the third columns of the matrices in factor set \( B \).

Define a matching to be one of the \( R! \) possible ways to correspond the columns in factor set \( A \) with the columns in factor set \( B \), and consider all possible matchings. Calculate the UCC between corresponding profiles, and average these \( 3R \) values. The criterion \( \Upsilon \) is the highest of the \( R! \) scores obtained by this process, i.e.

\[ \Upsilon = \max[\text{ave}(\text{UCC})] \]

where the maximum is taken over the \( R! \) possible matchings.

This procedure can be applied to any two sets of commensurate factor matrices. \( \Upsilon \) is a measure of closeness which achieves its maximum of 1 when the two sets are identical. If one set is a resolution and the other the signal, the \( \Upsilon \) score measures the quality of that resolution.

The quantity \( \Delta \) is defined by

\[ \Delta_n = 1 - \Upsilon_{(n,n-1)} \]

where \( \Upsilon_{(n,n-1)} \) is the measure of closeness between the current and previous resolutions in a PARAFAC sequence. A small value of \( \Delta_n \) indicates that the current PARAFAC iteration has produced little change because successive resolutions are nearly identical. It is natural therefore to terminate the PARAFAC iterations when \( \Delta \) becomes less than some preset value \( \Delta_0 \).
3 Swamps and Two-Factor Degeneracies

3.1 An Example

It occasionally happens that a PARAFAC sequence will slow down and appear to converge well before it has reached the optimum fit. An example of this behavior, taken from the simulation, is presented in Figures 1 through 5.

Figure 1 shows $\Delta_n$ and the base 10 logarithm of $\Delta_n$ plotted against the iteration number $n$. Notice that PARAFAC iterations are yielding very small changes after about 200 iterations, and after about 350 iterations the correlation between successive resolutions as measured by $\gamma$ is above $1 - 10^{-9}$. 
Figure 2: The first and second resolved factors from Example 3.1 obtained by setting $\log \Delta_0 = -9$ (solid curves) and $\log \Delta_0 = -10$ (dashed curves). The left column displays the profiles from the first factor while the right displays those from the second. The top row is Mode X, the middle Mode Y, and the bottom Mode Z.
Figure 3: The third and fourth resolved factors from Example 3.1 obtained by setting $\log \Delta_0 = -9$ (solid curves) and $\log \Delta_0 = -10$ (dashed curves). The left column displays the profiles from the third factor while the right displays those from the fourth. The top row is Mode X, the middle Mode Y, and the bottom Mode Z.
Figure 4: The first and second resolved factors from Example 3.1 obtained by setting $\log \Delta_0 = -10$ (dashed curves) and the signal (solid curves). The left column displays the profiles from the first factor while the right displays those from the second. The top row is Mode X, the middle Mode Y, and the bottom Mode Z.
Figure 5: The third and fourth resolved factors from Example 3.1 obtained by setting log $\Delta_0 = -10$ (dashed curves) and the signal (solid curves). The left column displays the profiles from the third factor while the right displays those from the fourth. The top row is Mode X, the middle Mode Y, and the bottom Mode Z.
However, if the PARAFAC algorithm is continued beyond 400 iterations, the step size increases until it reaches a peak before declining again. This peak shows up as a huge spike between the 1000th and 1100th iteration in the graph of $\Delta$.

We have found repeated occurrences of this type of behavior in which the PARAFAC algorithm encounters long stretches of miniscule change per iteration followed by a period of comparatively rapid change. We use the term *swamp* to describe such a situation. To be more precise, a swamp is any region of a PARAFAC sequence where the graph of $\log \Delta$ is bounded on the left by a local min and the right by a local max. We shall use the notation $S(m, M, n)$ to denote a swamp where $-m$ is the left bounding local min, $-M$ the right bounding local max, and $n$ is the number of iterations between the two.

In the example, the smallest value of $\log \Delta$ between the two spikes is roughly $-9.2$ and occurs at iteration 400, and the value of the right bounding spike is $-4.7$ and occurs at iteration 1056, so this swamp is denoted by $S(9.2, 4.7, 656)$. By our definition there is another minor swamp $S(6.21, 5.7, 79)$ at the very beginning of the sequence, but it is less serious and therefore less interesting than the $S(9.2, 4.7, 656)$ swamp.

To illustrate the seriousness of the $S(9.2, 4.7, 656)$ swamp in the example, consider the two resolutions obtained by setting $\log \Delta_0$ to $-9$ and $-10$, respectively. The former requires 350 iterations, the latter 1754. The two are displayed together for comparative purposes in Figures 2 and 3. The first two factors are displayed in Figure 2 and the last two in Figure 3. Notice that there are substantial differences between the two resolutions in the X and Z modes of the first and third factors and in the Y mode of the second factor. The two resolutions agree rather well for the fourth factor in all three modes.

The clear difference between the two resolutions does not by itself establish that the post-swamp resolution is better. That inference can be drawn from an examination of Figures 4 and 5 in which the post-swamp resolution is displayed together with the signal. The agreement here is quite good and clearly better than the agreement between the signal and the pre-swamp resolution.

How then should one deal with swamps? One approach might be to set $\Delta_0$ low enough to enable the sequence to pass beyond all of the swamps. Determining just how low $\Delta_0$ would have to be is one problem with this
approach. Another problem is its inefficiency for swampless sequences. A much less stringent value of \( \Delta_0 \) would suffice for sequences which contain no swamps.

It would be very helpful, therefore, if we could identify a characteristic of intermediate PARAFAC resolutions that would indicate that a swamp may lay ahead. Two-factor degeneracy (2FD), a phenomenon reported by Kruskal et al [9], appears to be just such a characteristic. A 2FD occurs when two factors are highly correlated in all three modes, but in such a manner that the product of their respective UCCs is near \(-1\). This phenomenon is clearly observed in Example 3.1. Upon reinspection of Figures 2 and 3 it is clear that the first and third Mode X profiles of the pre-swamp resolution are highly negatively correlated, and that the corresponding profiles in both Mode Y and Mode Z are nearly identical. The respective UCCs between these profiles are: \(-98.17\%\), \(99.93\%\), and \(99.80\%\). Their product is \(-97.90\%\), which is certainly near \(-1\). The post-swamp resolution, on the other hand, does not suffer from a 2FD.

3.2 Simulation Results

To investigate the relationship between 2FDs and the convergence properties of the PARAFAC algorithm, we conducted a simulation study in which intermediate resolutions produced by PARAFAC were tested for 2FDs. The test was based on the product of the UCCs between the respective profiles in all three modes of the two factors. A result less than \(-0.5\) indicated the presence of a 2FD. The test was performed for every pair of factors in the resolution.

A total of 64 data arrays were used in the study. A detailed description of the data can be found in the next section. For each array the EBP suggested by Leurgans, Ross, and Able (LRA) [10, 11] was used to generate as many as four starting resolutions for the PARAFAC algorithm. The algorithm was run with the termination criterion \( \log \Delta_0 \) set to \(-12\).

Each of the 64 arrays can be placed into one of three categories.

**Stable:** For 41 of the 64 arrays none of the PARAFAC sequences encountered any 2FDs.

**Difficult:** For 7 of the arrays, all of the sequences had a strong 2FD at the time of termination.
Discriminating: For the remaining 16 arrays each had at least one sequence that encountered a 2FD and at least one that did not.

For the stable arrays each sequence converged relatively quickly and produced nearly identical resolutions. In all cases the value of $\Upsilon$ between any two given resolutions on the same array exceeded 99.9\%. Also, a high percentage of these resolutions accurately reconstructed the signal. For 39 of the 41 arrays in this class the value of $\Upsilon$ between signal and resolution was above 97\% and 27 of these were above 99\%. None of sequences required more than 3000 iterations and only 4 arrays had sequences requiring more than 1000. More than half of the arrays required less than 300 iteration for each sequence. Compare this to the difficult arrays where 3 of the 7 required more than 18,000 iterations to converge and only one had a sequence requiring less than 1000.

In all cases the difficult arrays inaccurately reconstructed at least one of the signal components. For only 2 of the 7 was the $\Upsilon$ between signal and resolution above 97\%. This is certainly not surprising. Since all of the signal profiles were nonnegative, it is not possible for a resolution containing a 2FD to be highly accurate.

Like the stable arrays, the discriminating arrays had resolutions that agreed with each other and accurately reconstructed the signal a high percentage of the time. For 14 of the 16 arrays in this class the value of $\Upsilon$ between signal and resolution was above 97\% and 10 of these were above 99\%. However, for many of the arrays there was a wide discrepancy in the number of iterations required to reach termination for sequences which began at different EBP starting values. Sequences which encountered a 2FD were much more likely to experience swamps and generally took longer to reach the $-12$ termination criterion. These results are summarized in Table 1.

Table 1 displays for each of the 16 discriminating arrays the average number of iterations to termination calculated separately depending on whether or not the sequence encountered a 2FD. Notice that the sequences which contained 2FDs tended to be longer. In the five most acute cases (IDs 123, 233, 323, 334, and 344) the 2FD was present at the outset in the initial EBP resolution.

The message is clear. It is possible to reduce substantially the number of iterations required for an optimal resolution by making multiple EBP starts and incorporating periodic checks for the presence of 2FDs into the
Table 1: The table below lists the arrays that are classified as Discriminating. An identifier ID of these arrays are given in the left column. In the center column is the average number of iterations required for \( \log \Delta \) to reach \(-12\) for the sequences that were free of 2FDs. The right column indicates the iterations required for those that did encounter 2FDs to reach \(-12\) from the point that the 2FD was first encountered.

<table>
<thead>
<tr>
<th>ID</th>
<th>No</th>
<th>2FD</th>
<th>2FD</th>
<th>ID</th>
<th>No</th>
<th>2FD</th>
<th>2FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>121</td>
<td>174</td>
<td>368</td>
<td></td>
<td>321</td>
<td>433</td>
<td>890</td>
<td></td>
</tr>
<tr>
<td>123</td>
<td>818</td>
<td>2134</td>
<td></td>
<td>322</td>
<td>323</td>
<td>488</td>
<td></td>
</tr>
<tr>
<td>131</td>
<td>193</td>
<td>252</td>
<td></td>
<td>323</td>
<td>417</td>
<td>4707</td>
<td></td>
</tr>
<tr>
<td>132</td>
<td>248</td>
<td>286</td>
<td></td>
<td>332</td>
<td>420</td>
<td>494</td>
<td></td>
</tr>
<tr>
<td>231</td>
<td>123</td>
<td>133</td>
<td></td>
<td>334</td>
<td>1254</td>
<td>2156</td>
<td></td>
</tr>
<tr>
<td>232</td>
<td>141</td>
<td>150</td>
<td></td>
<td>344</td>
<td>589</td>
<td>1639</td>
<td></td>
</tr>
<tr>
<td>233</td>
<td>943</td>
<td>2020</td>
<td></td>
<td>422</td>
<td>256</td>
<td>357</td>
<td></td>
</tr>
<tr>
<td>311</td>
<td>231</td>
<td>213</td>
<td></td>
<td>431</td>
<td>375</td>
<td>337</td>
<td></td>
</tr>
</tbody>
</table>
PARAFAC algorithm. Since performing the test requires a mere fraction of the time required to perform a single PARAFAC iteration, we incorporate it into the PARAFAC routine and perform it after each iteration. If a 2FD is encountered, the sequence is discarded in favor of another.

4 The Data Arrays

We now describe in detail how the data arrays used in the empirical study and the example were constructed. The Excitation and Emission profiles were constructed to resemble actual observed Excitation and Emission Profiles found in the Appendix of Standards in Fluorescence Spectrometry [16]. The profiles used can be found in Tables 2 and 3. The Frequency Profiles were generated by specifying lifetimes and modulation frequencies, then applying Equation 2.

\[ z_{kr} = \frac{\omega_k \tau_r}{1 + \omega_k^2 \tau_r^2} \]  

(2)

To construct the 64 data arrays used in the investigation, four Mode X factor matrices, four Mode Y factor matrices, and four Mode Z factor matrices were constructed. These were combined in a full $4^3$ factorial design to generate 64 signal arrays. The Mode X and Mode Y factor matrices were constructed using the Profiles listed in Table 4. The four Mode Z factor matrices were constructed using the lifetimes and frequencies listed in Tables 5 and 6 and then applying Equation 2. As an example, the (2, 3) element of $Z_2$ is given by

\[ Z_2(2, 3) = \frac{15 \cdot 0.100}{1 + (15 \cdot 0.100)^2} \]

The three digit variable ID is used to identify the combination taken. For example, setting ID to 224 yields the signal

\[ S(224) = \bigotimes(X_2, Y_2, Z_4). \]

To reproduce the data arrays, normalize the Mode X and Mode Y profiles, create the Mode Z profiles using Equation 2, calculate $S(ID)$ as the indicated triple product, and generate noise arrays via the matlab code:
## Table 2: The Mode X Signal Profiles

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
<th>$x_8$</th>
<th>$x_9$</th>
<th>$x_{10}$</th>
<th>$x_{11}$</th>
<th>$x_{12}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
</tr>
<tr>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>1.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>1.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
</tr>
<tr>
<td>2.0</td>
<td>3.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>2.0</td>
<td>3.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>4.0</td>
<td>6.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>5.0</td>
<td>11.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>7.0</td>
<td>3.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>12.0</td>
<td>17.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>25.0</td>
<td>21.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>30.0</td>
<td>9.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>30.0</td>
<td>9.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>45.0</td>
<td>13.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>45.0</td>
<td>13.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>60.0</td>
<td>18.0</td>
<td>0.0</td>
<td>2.0</td>
<td>3.0</td>
<td>4.0</td>
<td>5.0</td>
<td>6.0</td>
<td>7.0</td>
<td>8.0</td>
<td>9.0</td>
<td>10.0</td>
</tr>
</tbody>
</table>
### Table 3: The Mode Y Signal Profiles

<table>
<thead>
<tr>
<th>y1</th>
<th>y2</th>
<th>y3</th>
<th>y4</th>
<th>y5</th>
<th>y6</th>
<th>y7</th>
<th>y8</th>
<th>y9</th>
<th>y10</th>
<th>y11</th>
<th>y12</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.1</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.2</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
Table 4: The Mode X and Mode Y Factor Matrices. Each profile should be normalized.

<table>
<thead>
<tr>
<th>Factor Matrix</th>
<th>Mode X Profiles</th>
<th>Factor Matrix</th>
<th>Mode Y Profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_1$</td>
<td>$x_6 \ x_{10} \ x_3 \ x_1$</td>
<td>$Y_1$</td>
<td>$y_{10} \ y_5 \ y_2 \ y_7$</td>
</tr>
<tr>
<td>$X_2$</td>
<td>$x_3 \ x_2 \ x_{10} \ x_5$</td>
<td>$Y_2$</td>
<td>$y_2 \ y_9 \ y_7 \ y_{12}$</td>
</tr>
<tr>
<td>$X_3$</td>
<td>$x_4 \ x_8 \ x_7 \ x_{11}$</td>
<td>$Y_3$</td>
<td>$y_3 \ y_{12} \ y_{10} \ y_{11}$</td>
</tr>
<tr>
<td>$X_4$</td>
<td>$x_9 \ x_{12} \ x_{11} \ x_3$</td>
<td>$Y_4$</td>
<td>$y_6 \ y_9 \ y_4 \ y_8$</td>
</tr>
</tbody>
</table>

Table 5: The Lifetimes used to construct the four Mode Z factor matrices. The Lifetime units are $10^{-6}$ seconds.

<table>
<thead>
<tr>
<th>$T_1$</th>
<th>0.001 0.010 0.100 1.000</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_2$</td>
<td>0.005 0.010 0.100 0.500</td>
</tr>
<tr>
<td>$T_3$</td>
<td>0.010 0.005 0.050 0.100</td>
</tr>
<tr>
<td>$T_4$</td>
<td>0.050 0.070 0.200 0.500</td>
</tr>
</tbody>
</table>
Table 6: The Frequencies in MHz used to construct each Mode Z profile.

<table>
<thead>
<tr>
<th>$\omega$</th>
<th>5</th>
<th>15</th>
<th>60</th>
<th>120</th>
</tr>
</thead>
</table>

rand('normal')
rand('seed',ID)
N = rand(2,2);
N = rand(40,160);

The simulated data arrays are then given by

$$A = S \odot (1 + 0.25N)$$

where 1 is an array of all ones and $\odot$ denotes term by term multiplication. In matlab:

$$A = S + 0.25 * (S .* N);$$

The array used in Example 3.1 was generated by setting ID equal to 123. The initial resolution was obtained using an EBP suggested by Leurgans, Ross, and Abel [11], where the third Mode Z slab was chosen along with the sum of the slabs
References


