Microphone Array Position Calibration by Basis-Point Classical Multidimensional Scaling

Stanley T. Birchfield, *Member, IEEE,* and Amarnag Subramanya

***Abstract*—Classical multidimensional scaling (MDS) is a global, noniterative technique for finding coordinates of points given their interpoint distances. We describe the algorithm and show how it yields a simple, inexpensive method for calibrating an array of microphones with a tape measure (or similar measuring device). We present an extension to the basic algorithm, called *basis-point classical MDS (BCMDS)*, which handles the case when many of the distances are unavailable, thus yielding a technique that is practical for microphone arrays with a large number of microphones. We also show that BCMDS, when combined with a calibration target consisting of four synchronized sound sources, can be used for automatic calibration via time-delay estimation. We evaluate the accuracy of both classical MDS and BCMDS, investigating the sensitivity of the algorithms to noise and to the design parameters to yield insight as to the choice of those parameters. Our results validate the practical applicability of the algorithms, showing that errors on the order of 10–20 mm can be achieved in real scenarios.**

***Index Terms*—Calibration, Euclidean coordinates, localization, microphones.**

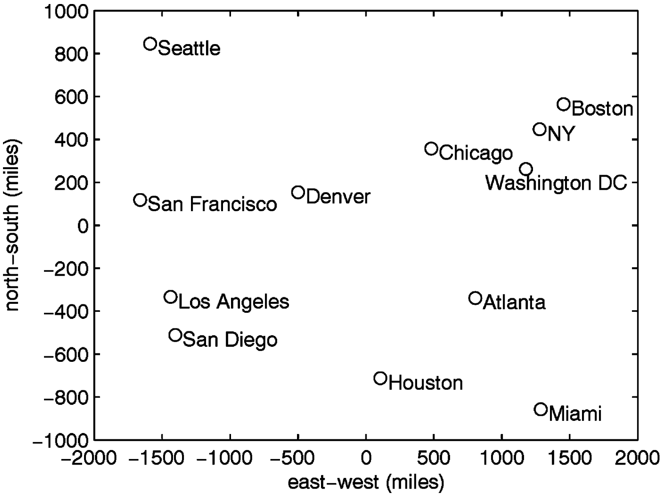


Fig. 1. Map of the United States computed by classical MDS using the driving distances between major cities published in a road atlas.

I. Introduction

APPLICATIONS using microphone arrays, such as determining the location of the current speaker (acoustic localization) or improving a speaker's sound quality by combining the microphone signals (beamforming), require the locations of the microphones to be known. For such applications, unless one is able to place the microphones at known locations during setup, one must calibrate their locations after they have been placed in the environment. This problem, known as geometric microphone array calibration, has traditionally been solved by nonlinear optimization techniques that require initial estimates and are subject to local minima.

Multidimensional scaling (MDS) is a field of study concerned with embedding a set of points in a low-dimensional space so that the distances between the points resemble as closely as possible a given set of dissimilarities between objects that they represent. These dissimilarities may be measured in a variety of ways, e.g., the difference between colors as perceived by human subjects. For decades, MDS has been a popular technique for analyzing experimental data in the physical, biological, and behavioral sciences [3], [7].

In *metric MDS*, the dissimilarities are themselves distances in a metric space. More precisely, the metric MDS problem is as follows: Given noisy distances between a set of points in a Euclidean space, estimate the coordinates of those points. *Classical MDS* is a simple, global, noniterative technique developed by Young and Householder [16] and made popular by Torgerson [13] for solving the metric MDS problem by decomposing a "squared-distance matrix" containing the squares of all the interpoint distances [3]. To illustrate, Fig. 1 shows a map of the United States computed from the driving distances between major cities published in a road atlas, using classical MDS. Even though the driving distance between two cities is a poor approximation to their actual distance, the resulting map is accurate.

In this paper we investigate several ways to use MDS to determine the locations of microphones in a microphone array. For relatively small microphone arrays (say, fewer than 10 microphones), a simple, cost-effective approach is to measure the distances between each pair of microphones using a tape measure (or similar measuring device), and then to apply the classical MDS algorithm, which is presented in Section II. For situations in which a small array needs to be calibrated one time, this method is a refreshing alternative to nonlinear optimization or to techniques requiring an expensive calibration target.

If measuring all the inter-microphone distances is impractical (e.g., when the number of microphones in the array is large), one may instead use an extension to the classical MDS algorithm that we call *basis-point classical MDS (BCMDS)*, derived in Section III. By measuring only the distances between each microphone and a small number of basis points, the entire squared-distance matrix can be constructed (because it is rank-deficient) and then fed to the classical MDS algorithm as

Manuscript received March 10, 2004; revised June 3, 2004. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Futoshi Asano.

The authors are with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC 29634 USA (e-mail: stb@clemson.edu; asubram@clemson.edu).

Digital Object Identifier 10.1109/TSA.2005.851893

To compute the locations of n microphones in a p-dimensional space from their noisy pairwise distances δ_{ij},

1. Construct the squared-distance matrix D whose entries are δ_{ij}^2.
2. Compute the inner product matrix $B = -\frac{1}{2}JDJ$, where $J = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the double-centering matrix and $\mathbf{1}$ is a vector of all ones.
3. Decompose B as $B = V\Lambda V^T$, where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$, the diagonal matrix of eigenvalues of B, and $V = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$, the matrix of corresponding unit eigenvectors. Sort the eigenvalues in non-increasing order: $\lambda_1 \geq \ldots \geq \lambda_n \geq 0$.
4. Extract the first p eigenvalues $\Lambda_p = \text{diag}(\lambda_1, \ldots, \lambda_p)$ and corresponding eigenvectors $V_p = [\mathbf{v}_1, \ldots, \mathbf{v}_p]$.
5. The microphone coordinates are now located in the $n \times p$ matrix $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T = V_p\Lambda_p^{\frac{1}{2}}$.

Fig. 2. Classical multidimensional scaling algorithm ($\mathcal{A}_0$).

before. We demonstrate in Section IV that, with common microphone array configurations, it is not difficult to select a set of basis points so that BCMDS produces accuracy essentially the same as that obtained with the entire set of measurements of classical MDS, with much less effort.

In some scenarios it may be desirable, or even necessary, to automatically calibrate an array rather than manually measuring distances with a tape measure. In Section V, we show how to determine automatically the microphone locations using a calibration target with four sound sources (synchronized with each other and with the microphones) which are treated as the basis points. By cross-correlating the received signals with the reference signals, the distances between the microphones and the sound sources can be computed automatically via time-delay estimation, from which BCMDS then computes the microphone locations. This technique of automatic calibration is especially useful for very large microphone arrays (such as the Huge Microphone Array [12]) or for arrays that need to be recalibrated frequently.

This work can be viewed as an extension of the work by Raykar *et al.* [9], [10], in which classical MDS is applied to the geometric calibration problem by placing a speaker next to each microphone and using time-delay estimation to measure the distance between speakers and microphones. Our work provides algorithms that are applicable, not only when a speaker is attached to each microphone, but also when just four speakers are placed anywhere in the environment or when no speaker is available. This paper also provides a detailed analysis of the accuracy one can expect from these algorithms, along with insight into the choice of parameters, thus providing researchers with necessary information to make the best use of these algorithms in real scenarios.

In contrast to MDS, traditional approaches to geometric calibration involve minimizing a nonlinear functional [1], [8], [11], [14], which generally requires an initial estimate and an iterative algorithm. Moreover, some techniques, being designed for applications in sonar, radar, or radio astronomy, assume narrow-band sources and planar arrays of sensors [8], [14], thus making them inapplicable to the present problem. Our work, and MDS in general, can be seen as either a simpler alternative to these nonlinear optimizations in that no initial estimate or iterations are required (and therefore local minima are completely avoided), or as a way to provide an initial estimate for further nonlinear estimation, as done in [9], [10].

After presenting the classical MDS algorithm in the next section, we derive the BCMDS algorithm in Section III and analyze both algorithms extensively in the simulations of Section IV. Section V then describes and analyzes the method of automatic calibration using a synchronized calibration target, followed by a conclusion.

II. Classical Multidimensional Scaling

Suppose we have n microphones in a p-dimensional space (usually $p = 3$). The classical multidimensional scaling algorithm works as follows [2], [3]. First construct a squared-distance matrix D such that each entry d_{ij} is the squared distance between microphones i and j: $d_{ij} = \delta_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j)$, where δ_{ij} is the distance between microphones i and j. From D compute the inner product matrix $B = -(1/2)JDJ$, where $J = I - (1/n)\mathbf{1}\mathbf{1}^T$ is the double-centering matrix and $\mathbf{1}$ is a vector of all ones.

Without noise, $B = XX^T$, where $X = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T$ is the $n \times p$ matrix of coordinates, and hence $rank(B) = rank(XX^T) = rank(X) = p$. Since B is symmetric positive semi-definite, it may be decomposed as $B = V\Lambda V^T$, where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$, the diagonal matrix of eigenvalues of B, and $V = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$, the matrix of corresponding unit eigenvectors. For convenience the eigenvalues are labeled so that $\lambda_1 \geq \ldots \geq \lambda_n \geq 0$. Since B is of rank p, it has p nonzero eigenvalues and $n - p$ zero eigenvalues and hence may be written as $B = V_p\Lambda_p V_p^T$, where $\Lambda_p = \text{diag}(\lambda_1, \ldots, \lambda_p)$ and $V_p = [\mathbf{v}_1, \ldots, \mathbf{v}_p]$. The coordinate matrix X is then given by $X = V_p\Lambda_p^{1/2}$.

With noise (i.e., the measured distances are imperfect), B will not be of rank p but rather will be full rank in practice: $rank(B) = n$. Conveniently, $X = V_p\Lambda_p^{1/2}$ still yields the optimal estimate of the coordinates in the space of dimensionality p, in the sense that $\sum_{i=1}^{n}\sum_{j=1}^{n}(\delta_{ij}^2 - \hat{\delta}_{ij}^2)^2$ is minimized, where $\hat{\delta}_{ij}$ is the estimated distance between the two microphones. In the terminology of principal components analysis (PCA), V_p contains the p eigenvectors that capture the most significant variation in the data [6]. In fact, the classical MDS algorithm is identical to the PCA algorithm. For brevity we will refer to the classical MDS algorithm, which is summarized in Fig. 2, as $\mathcal{A}_o$.

It is important to notice that, because the algorithm receives as input only the distances between points, the resulting coordi-

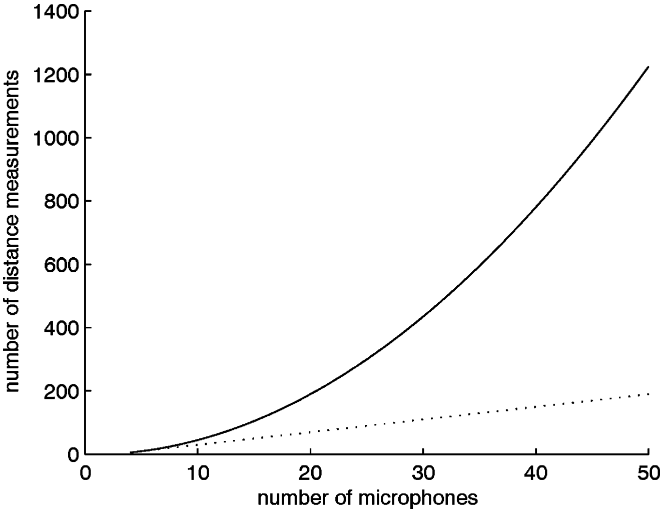


Fig. 3. Number of measurements needed for all pairwise distances (solid line) and for distances to basis points (dashed line), in three dimensions.

nates are unique only up to an arbitrary translation, rotation, and reflection. In other words, for any normalized orthogonal matrix A, $(VA)\Lambda(VA)^T = V\Lambda V^T = B$. Translation is handled by first shifting all the points so that their centroid coincides with the origin. Then the root-mean-square (rms) error in the solution compared with ground truth is computed as

$$\text{rms} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} ||A^T\mathbf{x_i} - \mathbf{y_i}||^2} \tag{1}$$

where $\mathbf{x_i^T}$ is the ith row of X, $\mathbf{y_i^T}$ is the ith row of the ground-truth matrix Y, and $|| \cdot ||$ denotes the Euclidean norm. A is the normalized orthogonal matrix that best aligns the points by rotating and reflecting them [3]

$$A = (X^TYY^TX)^{\frac{1}{2}}(Y^TX)^{-1}.$$

Equation (1) is similar to Procrustes' analysis [3] except that it does not allow for scale changes between the point sets.

III. MDS Extension to Handle Missing Data

Since without noise B is of rank p, there is much redundancy in B and D. In fact, in this section we show that in a p-dimensional space all the $n(n-1)/2$ distances can be calculated given only the distances between each point and a set of $p+1$ basis points. Assuming the set of basis points is a subset of the microphones, then we need to measure just the $p(p+1)/2$ distances between all basis points and the $(p+1)(n-p-1)$ distances between each basis point and each of the $n-p-1$ nonbasis-point microphones. The number of measurements becomes linear in the number of microphones, as opposed to quadratic, as shown in Fig. 3. The savings can be significant: 70 measurements instead of 190 for $n = 20$, or 190 measurements instead of 1225 for $n = 50$. Below we derive a basis for the three-dimensional world (3-D) by first doing so for one–dimensional (1-D) and two–dimensional (2-D). This derivation is based on the equations of Young and Cliff [15].

A. Deriving a 1-D Basis

Suppose we measure the distance δ_{AB} between two points A and B. If we are allowed to translate and rotate the coordinate axes freely, we may place these two points on the x axis (i.e., $y_A = z_A = y_B = z_B = 0$) and set their midpoint as the origin, thus obtaining the following coordinates for the points:

$$x_A = -x_B = -\frac{1}{2}\delta_{AB}$$

where we have arbitrarily chosen the sign.

A and B define a basis for a 1-D space. Given the distances δ_{AQ} and δ_{BQ} between A and B and some point Q, the definition of Euclidean distance gives

$$\begin{aligned}\delta_{AQ}^2 &= \left(-\frac{1}{2}\delta_{AB} - x_Q\right)^2 + y_Q^2 + z_Q^2 \\ \delta_{BQ}^2 &= \left(\frac{1}{2}\delta_{AB} - x_Q\right)^2 + y_Q^2 + z_Q^2.\end{aligned} \tag{2}$$

Combining these equations and solving for x_Q yields

$$x_Q = \frac{\delta_{AQ}^2 - \delta_{BQ}^2}{2\delta_{AB}}. \tag{3}$$

Thus, (3) provides the x-coordinate of an arbitrary point Q given the distances between it and the basis points A and B.

B. Deriving a 2-D Basis

If Q does not lie on the line connecting A and B (the x axis), then its residual distance to that line will be nonzero. The square of the residual distance from Q to the x-axis can be computed from (2) and (3)

$$\begin{aligned}y_Q^2 + z_Q^2 =& \delta_{AQ}^2 - \left(-\frac{1}{2}\delta_{AB} - x_Q\right)^2 \\ =& \frac{1}{2}\delta_{AQ}^2 - \frac{1}{4}\delta_{AB}^2 + \frac{1}{2}\delta_{BQ}^2 - x_Q^2.\end{aligned} \tag{4}$$

Rearranging terms yields a convenient formula that we will use again

$$x_Q^2 + y_Q^2 + z_Q^2 = \frac{1}{2}\delta_{AQ}^2 - \frac{1}{4}\delta_{AB}^2 + \frac{1}{2}\delta_{BQ}^2. \tag{5}$$

Let us orient the axes so that a third point c lies in the xy-plane (i.e., $z_C = 0$). By substituting C for Q, x_C is computed using (3) and y_C is found from (4)

$$y_C = \sqrt{\frac{1}{2}\delta_{AC}^2 - \frac{1}{4}\delta_{AB}^2 + \frac{1}{2}\delta_{BC}^2 - x_C^2}$$

where we have arbitrarily chosen the positive square root.

A, B, and C now provide a basis for a 2-D space. Given the distances δ_{AQ}, δ_{BQ}, and δ_{CQ} for some point Q, the definition of Euclidean distance gives

$$\delta_{CQ}^2 = (x_C - x_Q)^2 + (y_C - y_Q)^2 + (z_C - z_Q)^2.$$

To compute the microphone locations in three dimensions ($p = 3$),

1. Compute the coordinates of the four basis points from their pairwise distances:

$$\begin{aligned}
A &: \left(-\frac{1}{2}\delta_{AB},\ 0,\ 0\right)\\
B &: \left(\frac{1}{2}\delta_{AB},\ 0,\ 0\right)\\
C &: \left(x_C,\ \sqrt{\frac{1}{2}\delta_{AC}^2-\frac{1}{4}\delta_{AB}^2+\frac{1}{2}\delta_{BC}^2-x_C^2},\ 0\right)\\
D &: \left(x_D,\ y_D,\ \sqrt{\frac{1}{2}\delta_{AD}^2-\frac{1}{4}\delta_{AB}^2+\frac{1}{2}\delta_{BD}^2-x_D^2-y_D^2}\right).
\end{aligned}$$

2. Compute the coordinates of each microphone Q given its distance to the basis points:

$$\begin{aligned}
x_Q &= \frac{\delta_{AQ}^2-\delta_{BQ}^2}{2\delta_{AB}}\\
y_Q &= \frac{\delta_{AC}^2-\delta_{AB}^2+\delta_{BC}^2+\delta_{AQ}^2+\delta_{BQ}^2-2\delta_{CQ}^2-4x_Cx_Q}{4y_C}\\
z_Q &= \frac{1}{4z_D}\left(\delta_{AD}^2+\delta_{BD}^2+\delta_{AQ}^2+\delta_{BQ}^2-\delta_{AB}^2\right.\\
&\qquad \left.-2\delta_{DQ}^2-4x_Dx_Q-4y_Dy_Q\right).
\end{aligned}$$

(Note that x_C, x_D, and y_D above are found by substituting C or D for Q.)

3. Build the squared-distance matrix D using these coordinates, and run the classical MDS algorithm ($\mathcal{A}_o$).

Fig. 4. Basis-point classical MDS algorithm ($\mathcal{A}_b$).

Expanding the squares, rearranging terms, substituting (5), and solving for y_Q yields

$$y_Q = \frac{\delta_{AC}^2-\delta_{AB}^2+\delta_{BC}^2+\delta_{AQ}^2+\delta_{BQ}^2-2\delta_{CQ}^2-4x_Cx_Q}{4y_C}. \tag{6}$$

Thus, (3) and (6) provide the x- and y-coordinates of an arbitrary point Q given the distances between it and the basis points A, B, and C.

C. Deriving a 3-D Basis

If a point does not lie in the xy-plane, then its residual distance to the plane will be nonzero. Let us select such a point as a fourth basis point D, where x_D and y_D are found using (3) and (6), respectively, and z_D is computed by applying (5) to D, yielding

$$z_D = \sqrt{\frac{1}{2}\delta_{AD}^2-\frac{1}{4}\delta_{AB}^2+\frac{1}{2}\delta_{BD}^2-x_D^2-y_D^2}$$

where we have again arbitrarily chosen the positive square root.

The points A, B, C, and D now provide a basis for a 3-D space. Given the distances δ_{AQ}, δ_{BQ}, δ_{CQ}, and δ_{DQ} for some point Q, the definition of distance for δ_{DQ}^2 gives, after substituting (5) and solving for z_Q

$$\begin{aligned}
z_Q = \frac{1}{4z_D}&\left(\delta_{AD}^2+\delta_{BD}^2+\delta_{AQ}^2+\delta_{BQ}^2-\delta_{AB}^2\right.\\
&\left.-2\delta_{DQ}^2-4x_Dx_Q-4y_Dy_Q\right). \qquad (7)
\end{aligned}$$

Thus, (3), (6), and (7) provide the x-, y-, and z-coordinates of an arbitrary point Q given the distances between it and the basis points A, B, C, and D. These equations can be combined with classical MDS, as summarized in Fig. 4. We call the resulting algorithm *basis-point classical MDS (BCMDS)*, and refer to it throughout the rest of this paper as $\mathcal{A}_b$.

IV. Simulations

In this section, we analyze the performance of the algorithms $\mathcal{A}_o$ and $\mathcal{A}_b$ using simulations in Matlab in which we vary the microphone array geometries, noise types and levels, and choice of basis points. We also perform an analysis regarding the choice of dimensionality for a nearly-planar array.

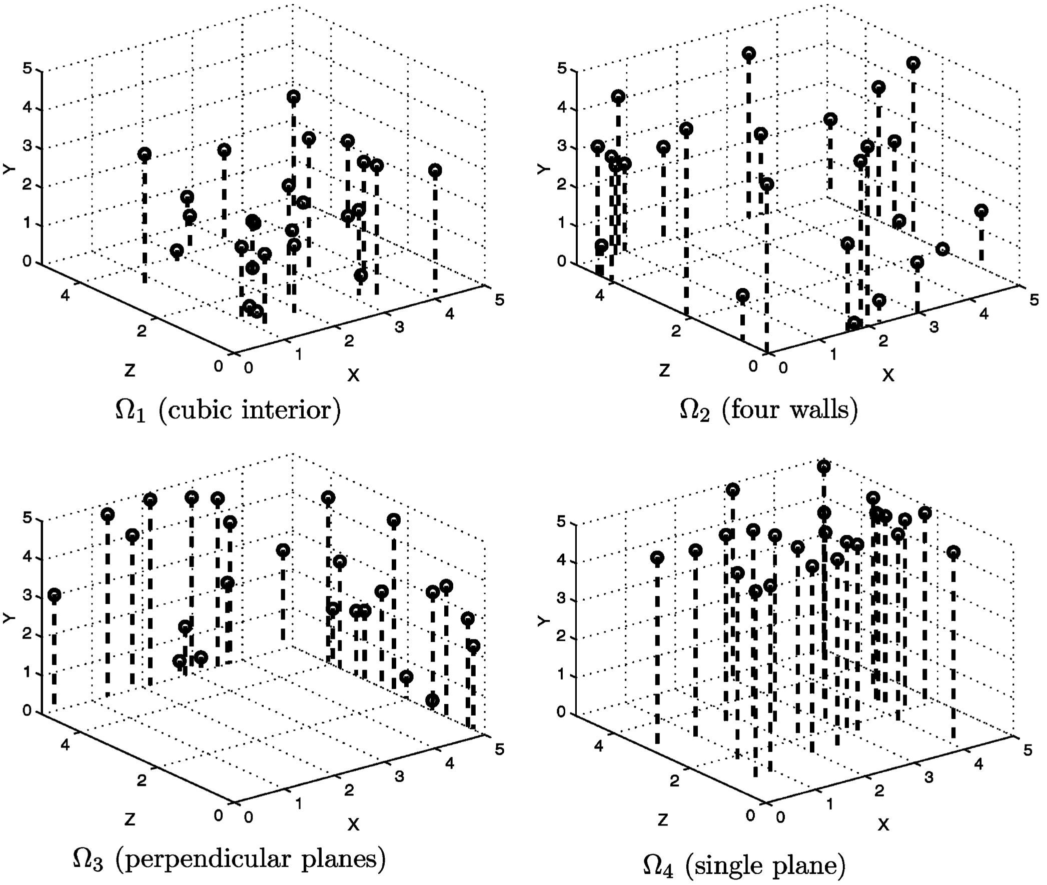
The four microphone array geometries we used in the simulations are shown in Fig. 5. The microphones were distributed, using a uniform random distribution

1) throughout the interior of a cube;
2) on the surface of four sides of the cube;
3) on the surface of two perpendicular planes sharing a common edge;
4) on the surface of a single plane.

We will refer to these four configurations as Ω_1, Ω_2, Ω_3, Ω_4, respectively. We chose the length of a side in all cases to be 5 m, although other sizes yield similar results. All experiments for the first three configurations were conducted with $p = 3$, while those for Ω_4 used $p = 2$.

A. Perturbing Distances With Gaussian Noise

For each of the configurations, we ran $\mathcal{A}_o$ after corrupting the distances with additive, independent Gaussian noise with $\sigma = 5\ \mathrm{mm}$, evaluating the output using the error metric given in (1). The number of microphones ranged from 4 to 20. The results are displayed in Fig. 6, where each data point represents the average over 1000 trials. For realistic measurement error possible with even an inexpensive measuring device such as a tape measure, the rms error remains less than 12 mm for all configurations and



Ω_1 (cubic interior)

Ω_2 (four walls)

Ω_3 (perpendicular planes)

Ω_4 (single plane)

Fig. 5. Four microphone array configurations.

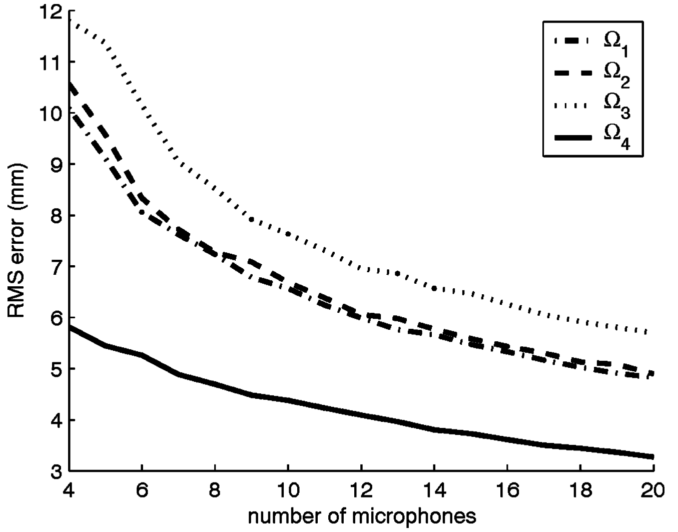


Fig. 6. Classical MDS $(\mathcal{A}_o)$ is accurate and robust with respect to Gaussian measurement noise.

all values of n. In addition to the results shown in the figure, the rms varies linearly with the amount of measurement noise. As expected, the best results were obtained with Ω_4 because of the additional constraint afforded by the planar geometry, while the maximum error was obtained with Ω_3, the 3-D array composed of the smallest number of planes.

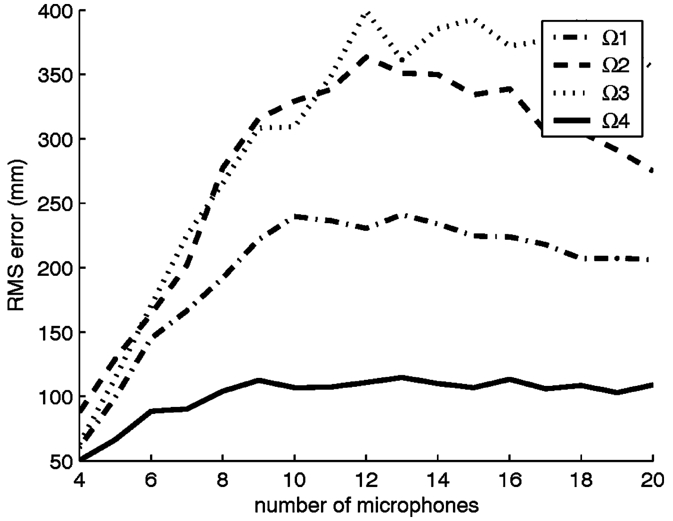
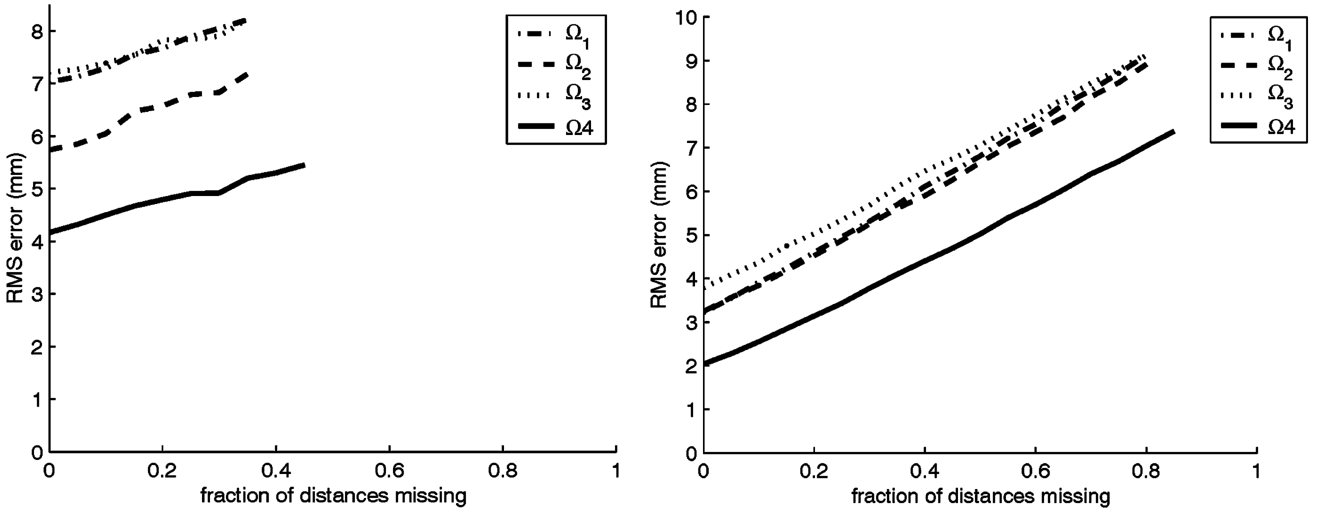


Fig. 7. Classical MDS $(\mathcal{A}_o)$ is sensitive to impulse measurement noise (e.g., missing data).

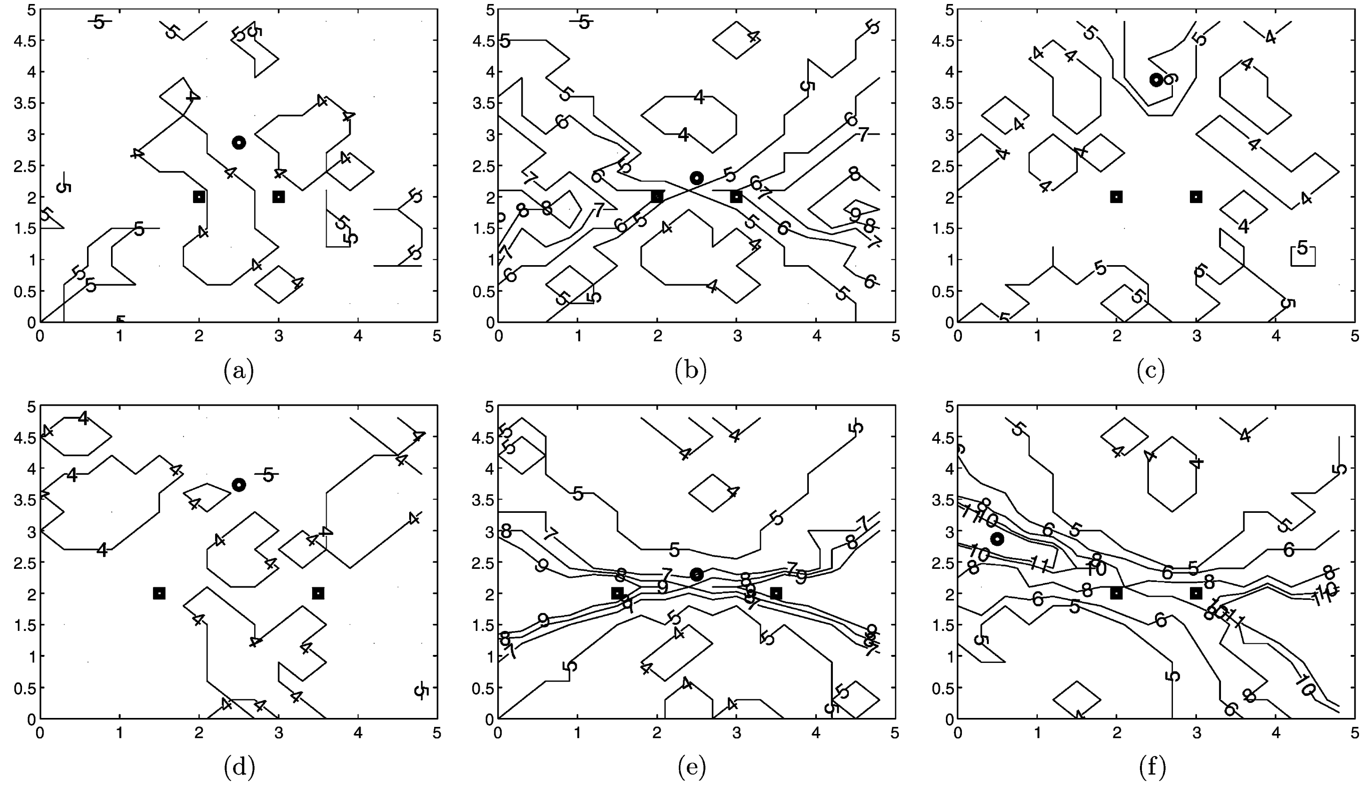
B. Handling Missing Data

Classical MDS is not so forgiving with impulse noise, as shown in Fig. 7 where 1% of the distances were set to zero (similar results were achieved with other values) to simulate missing data. Each data point again represents the average of 1000 trials. With even a small fraction of the distances missing, the rms error is in the hundreds of millimeters, with the error even in the best



(a) (b)

Fig. 8. BDMDS ($\mathcal{A}_b$) robustly handles missing data. Although the error increases as fewer measurements are available, the error remains below 9 mm throughout. (a) $n = 10$. (b) $n = 50$.



(a) (b) (c)

(d) (e) (f)

Fig. 9. Contour plot of rms error (in millimeters) for different arrangements of 1-D basis points in a 5 × 5 m room. The two squares are the basis points A and B, while the circle is the basis point C. The areas of the triangles, in square meters, are (a) 0.43, (b) 0.10, (c) 0.93, (d) 1.73, (e) 0.20, and (f) 0.43.

case (Ω_4 with $n = 20$) greater than 100 mm. Clearly, when some data are missing we cannot simply ignore that fact.

To test the ability of $\mathcal{A}_b$ to handle missing data, we selected as a basis set the four microphones enclosing the maximum volume, as explained in the next subsection. We varied the fraction of missing entries in D from 0 to the maximum allowable $((n-p-1)/n)^2$, filling these entries using the equations of Section III. Remaining entries were computed using inter-microphone distances corrupted by Gaussian noise ($\sigma = 5$ mm), as before. Plotted in Fig. 8 are the results, averaged over 100 trials per data point. The rms error remains below 9 mm for $n = 10$, even when nearly 40% of the measurements are unavailable, and also below 9 mm for $n = 50$, even when nearly 80% of the measurements are unavailable. These results confirm the statements of previous researchers that only about 25%–33% of the measurements are needed in practice [4], [5].

C. Choice of Basis Points

The accuracy of $\mathcal{A}_b$ depends upon the choice of basis points. Because an analytic sensitivity analysis of the algorithm is intractable, we instead measured its sensitivity to the basis point locations by running $\mathcal{A}_b$ on several different basis-point arrangements, as shown in Fig. 9. Starting with an equilateral

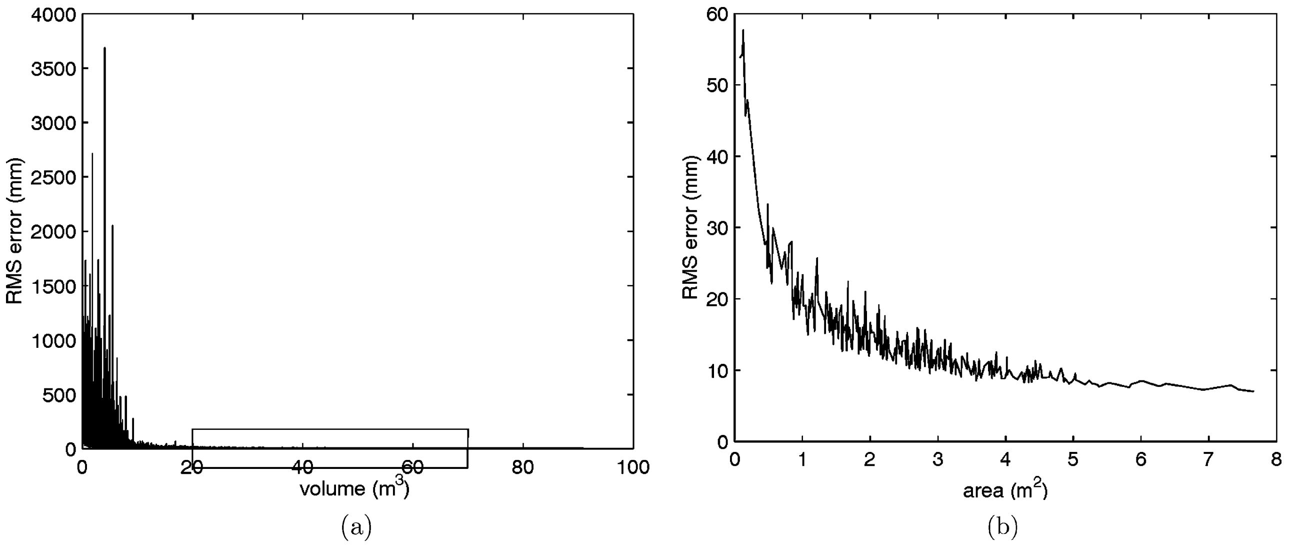


Fig. 10. In general, rms error decreases as the volume or area enclosed by the basis points increases. Rms error from $\mathcal{A}_b$ versus (a) volume for Ω_1 and (b) area for Ω_4.

triangle with adjacent points 1 m apart as the first arrangement, we moved the points in various directions to construct the other arrangements. For each arrangement, we ran $\mathcal{A}_b$ to compute the locations of all possible points in a 5 m × 5 m room, sampled on a grid with 0.2 m between adjacent points. As before, all the distances were perturbed with Gaussian noise ($\sigma = 5\ \mathrm{mm}$), and each result is the average of 100 trials. The resulting contour plots, displayed in the figure, show that the least error is achieved with the equilateral triangles [Fig. 9(a) and (d)], the larger one yielding slightly better results. Not surprisingly, the error increases as the points become nearly collinear [Fig. 9(b), (e), and (f)]; and as the distance between C and the line joining A and B increases, the error increases in the direction of the move and decreases in the perpendicular direction [Fig. 9(c)].

From these contour plots we learn three conditions that improve the results of the algorithm: (1) A large distance between A and B, (2) a large distance between C and the line joining A and B, and (3) C nearly equidistant from A and B. By similar argumentation, the results in 3-D are improved if there is also a large distance between D and the plane containing A, B, and C. Although these heuristics cannot, to our knowledge, be captured in a single, compact formula, a convenient and reasonable approximation is the volume (or area in 1-D) enclosed by the basis points. Recall that the volume enclosed by four points in 3-D is given by $(1/6)\mathrm{abs}(|\mathbf{x}'_A\ \mathbf{x}'_B\ \mathbf{x}'_C\ \mathbf{x}'_D|)$, where the prime ($'$) denotes homogeneous coordinates and the vertical bars ($|\cdot|$) denote the matrix determinant; similarly the area enclosed by three points in 2-D is given by $(1/2)\mathrm{abs}(|\mathbf{x}'_A\ \mathbf{x}'_B\ \mathbf{x}'_C|)$. Shown in Fig. 10 is a plot of rms error versus volume enclosed by the basis points of Ω_1, and versus area enclosed by the basis points of Ω_4, both with $n = 25$. The plots include only those arrangements for which the angle between the segments connecting any two pairs of basis points is at least 40° (for Ω_1) or 20° (for Ω_2). Generally speaking, the error decreases as the volume or area increases, and for any reasonable choice of basis points the error is below 16 mm. Only when the basis points enclose a volume less than 20% of the size of the room, or an area less than 35% of the size of the plane, does the error increase above this level.

D. *Choosing p for a Nearly Planar Array*

It is clear that p should be set to two when all the microphones are coplanar (2-D), while it should be set to three when they are noncoplanar (3-D). A question remains, however: If the microphones are nearly, but not exactly, coplanar, what will be the result of choosing the wrong value for p?

To answer this question, we perturbed the planar microphones of Ω_4 in the direction perpendicular to the plane with different Gaussian distributions (the greater the variance in the distribution, the less coplanarity in the microphones). Shown in Fig. 11 are the results of running $\mathcal{A}_o$ and $\mathcal{A}_b$ on the resulting microphone positions for both $p = 2$ and $p = 3$. As expected, the error for $p = 2$ increases as the microphone positions deviate from the plane, while the error for $p = 3$ decreases. The two choices yield approximately the same error with a perturbation of about 150 mm for $\mathcal{A}_o$ or 350 mm for $\mathcal{A}_b$, the results for $\mathcal{A}_b$ being significantly worse than those for $\mathcal{A}_o$. These experiments show that caution should be exercised in using either of the algorithms for an array that is fairly flat but not exactly planar, because there exist geometries for which the resulting error may be unacceptable. In such a case it is advised to augment the measurements using additional points in the environment to provide better 3-D information, and then to set $p = 3$. The solid line in the plots shows a vast improvement by simply adding a single point located at (2.5, 0, 2.5) m, that is, centered with the array but 5 m away in the direction perpendicular to the plane.

V. Automatic Calibration Using a Calibration Target

Up to now, the basis points for $\mathcal{A}_b$ have been a subset of the microphones, and the interpoint distances have been measured presumably with a tape measure, or similar device. Since $\mathcal{A}_b$ works for any basis points in the space, however, an alternative is to use a calibration target consisting of four speakers rigidly attached to one another, with the four speakers being the basis points. By measuring the $p(p+1)/2 = 6$ inter-speaker distances

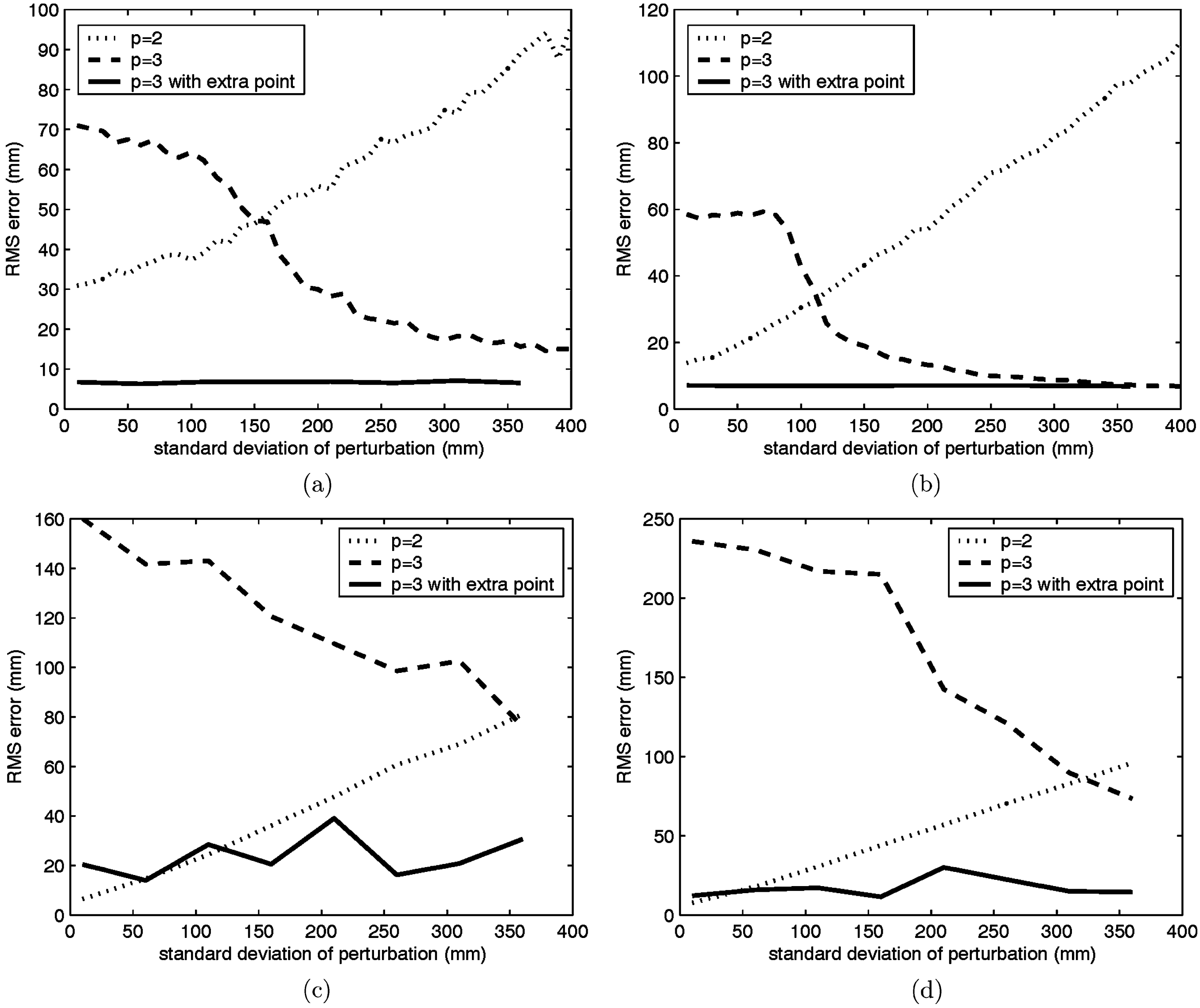


Fig. 11. For a nearly planar array, the rms error of the algorithms may be unacceptable no matter the choice of p, especially for $\mathcal{A}_b$. Results are significantly improved, though, by using an additional point in the environment that is outside the plane (solid line). (a) $\mathcal{A}_o$, $n = 8$; (b) $\mathcal{A}_o$, $n = 25$; (c) $\mathcal{A}_b$, $n = 8$; (d) $\mathcal{A}_b$, $n = 25$.

with a tape measure and the $n(p+1)$ distances between microphones and speakers using time-delay estimation, $\mathcal{A}_b$ yields an automatic, global, noniterative technique for computing the microphone locations.

Measuring the distances between speakers and microphones requires that we have access to the reference signal emitted by each speaker, synchronized with the signals received by the microphones. One way to achieve this prerequisite is to synchronize the speaker emission with the microphone digitizers and to use the waveform sent to the speaker as the reference. This synchronization can be achieved by sending and receiving from the same computer with real-time control of the timing of the two activities, or by modulating a radio-frequency (RF) signal that travels instantaneously compared with the sound wave [9]. Another way is to place an extra microphone as closely as possible to each speaker to capture the sound as soon as it is emitted [10]. This extra microphone, whose digitizer must be synchronized with the other microphones' digitizers, provides the reference signal. Either way, time-delay estimation cross-correlates the reference signal with the signals received by the microphones to estimate the time it took sound to travel from the source to the microphone. For a good description of the difficulties and details encountered using a calibration target, including the complications arising from changes in the speed of sound, please consult Sachar *et al.* [11].

For our purposes, we assume that these problems are solved, and that the distances between speakers and microphones can be measured. In this subsection we wish to evaluate the sensitivity of $\mathcal{A}_b$ to the location and size of the calibration target and to gain insight toward the choice of these parameters. To maximize the accuracy of the algorithm the shape of the calibration target was chosen to be a right, regular, triangle-based pyramid, which maximizes the volume enclosed by the speakers, as explained in the previous section.

For all four configurations $\Omega_1 \cdots \Omega_4$, we placed the simulated pyramidal target in the center of a $5 \times 5 \times 5$ m simulated room,

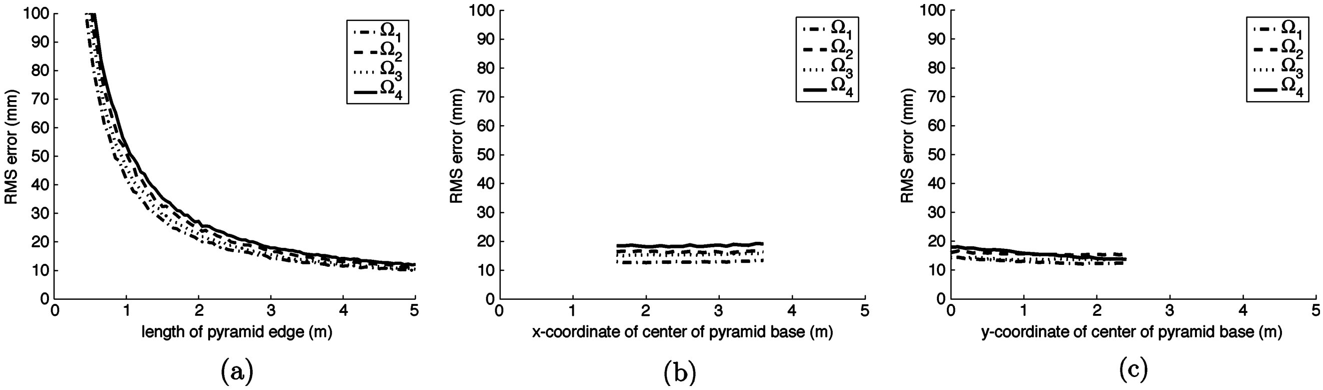


Fig. 12. Rms error for $\mathcal{A}_b$ versus size and position of the calibration target. With sufficient size, the target achieves small error. (a) Size. (b) x-coordinate. (c) y-coordinate.

resting on the floor, so that the coordinates of the center of its base were (2.5, 0, 2.5) m. For each configuration we first uniformly scaled the target, varying the pyramidal edge length from 0.1 to 5 m in increments of 0.1 m. Then, setting the length to 3 m, we varied the target's x-coordinate, sliding the target from one end of the room to the other. Finally, setting the x-coordinate to 2.5 m, we varied the y-coordinate in a similar manner. Due to the symmetry between x and z, we kept the z-coordinate constant throughout. In all simulations we required the calibration target to remain inside the room, and all distance measurements were corrupted by zero-mean Gaussian noise with $\sigma = 5$ mm.

Several conclusions can be drawn from these results, which are displayed in Fig. 12. As expected, the error of the 3-D arrays $\Omega_1 \cdots \Omega_3$ was generally less than that of the planar array Ω_4 because the computation was performed with $p = 3$. Also, the error decreased with the size of the calibration target, reaching 20 mm when the pyramidal edge was approximately 3 m. Position had less effect on the results, but in all cases the least error occurred when the distance from the target to the farthest microphone was minimized. This goal was generally achieved by keeping the target in the center of the room, except in the case of Ω_4, where moving the target toward the array minimized this distance. Keep in mind that these simulations modeled the speakers as point sources and thus ignored near-field effects that would occur in a real environment as the speakers approach the microphones.

VI. Conclusion

The positions of microphones in an array can be computed using a global, noniterative algorithm derived many years ago called *classical MDS*. The algorithm requires as input only the distances between microphones, which can be measured with a tape measure or similar device. We have extended the basic algorithm to compute these positions using only the distances between the microphones and a small set of basis points. The resulting technique, which we call *BCMDS*, is applicable in situations where it is not feasible to measure all the inter-microphone distances, such as with an array containing many microphones. We have also shown that BCMDS facilitates automatic calibration using a calibration target, where speakers that are rigidly attached to the target are the basis points.

Both classical MDS and BCMDS can be viewed as either simpler alternatives to the more traditional nonlinear optimization approaches because they require no initial solution, require no iterations, and are not subject to local minima; or they can be viewed as a way to provide an initial estimate for such nonlinear optimization. Either way, some care must be exercised in order to achieve good results: 1) for BCMDS, the basis points should be selected so that they maximize the enclosed volume (or area in 2-D); 2) with a calibration target, the speakers should be placed as far away from each other as possible; and 3) for arrays that are nearly but not exactly planar one should use additional points in the scene, for both algorithms, to reduce the error. We have shown that, when this advice is heeded, accuracy on the order of 10–20 mm can be achieved with both classical MDS and BCMDS on a variety of microphone array geometries, thus demonstrating their applicability in realistic scenarios.

References

[1] M. Berger and H. F. Silverman, "Microphone array optimization by stochastic region contraction (SRC)," *IEEE Trans. Signal Process.*, vol. 39, no. 11, pp. 2377–2386, Nov. 1991.

[2] S. T. Birchfield, "Geometric microphone array calibration by multidimensional scaling," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2003.

[3] T. F. Cox and M. A. A. Cox, *Multidimensional Scaling*. London, U.K.: Chapman and Hall, 1994.

[4] R. A. Girard and N. Cliff, "A Monte Carlo evaluation of interactive multidimensional scaling," *Psychometrika*, vol. 41, no. 1, pp. 43–64, Mar. 1976.

[5] R. S. Green and P. M. Bentler, "Improving the efficiency and effectiveness of interactively selected MDS data designs," *Psychometrika*, vol. 44, no. 1, pp. 115–119, Mar. 1979.

[6] A. Leon-Garcia, *Probability and Random Processes for Electrical Engineering*. Reading, MA: Addison–Wesley, 1994.

[7] J. Meulman, *A Distance Approach to Nonlinear Multivariate Analysis*. Leiden: DSWO Press, 1986.

[8] B. C. Ng and C. M. S. See, "Sensor-array calibration using a maximum-likelihood approach," *IEEE Trans. Antennas Propag.*, vol. 44, no. 6, pp. 827–835, 1996.

[9] V. Raykar, I. Kozintsev, and R. Lienhart, "Position calibration of audio sensors and actuators in a distributed computing platform," in *Proc. 11th ACM Int. Conf. Multimedia*, Nov. 2003, pp. 572–581.

[10] V. C. Raykar and R. Duraiswami, "Automatic position calibration of multiple microphones," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Montreal, QC, Canada, May 2004.

[11] J. M. Sachar, H. F. Silverman, and W. R. Patterson III, "Position calibration of large-aperture microphone arrays," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2002.

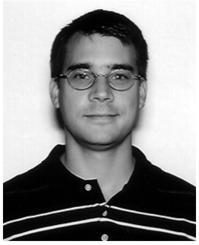
[12] H. F. Silverman, W. R. Patterson, J. L. Flanagan, and D. Rabinkin, "A digital processing system for source location and sound capture by large microphone arrays," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1997.

[13] W. S. Torgerson, "Multidimensional scaling: I. theory and method," *Psychometrika*, vol. 17, no. 4, pp. 401–419, Dec. 1952.

[14] A. J. Weiss and B. Friedlander, "Array shape calibration using sources in unknown locations: a maximum likelihood approach," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 12, pp. 1958–1966, Dec. 1989.

[15] F. W. Young and N. Cliff, "Interactive scaling with individual subjects," *Psychometrika*, vol. 37, no. 4, pp. 385–415, Dec. 1972.

[16] G. Young and A. S. Householder, "Discussion of a set of points in terms of their mutual distances," *Psychometrika*, vol. 3, no. 1, pp. 19–22, Mar. 1938.



Stanley T. Birchfield (M'00) received the B.S. degree from Clemson University, Clemson, SC, in 1993 and the M.S. and Ph.D. degrees from Stanford University, Stanford, CA, in 1996 and 1999, respectively, all in electrical engineering.

He is an assistant professor with the Electrical and Computer Engineering Department, Clemson University. His graduate work was supported by a National Science Foundation Graduate Research Fellowship. Prior to his current position, he was a research engineer with Quindi Corporation, Palo Alto, CA. His research interests include computer vision, visual correspondence and tracking, mobile robot navigation, and acoustic localization.

Amarnag Subramanya, photograph and biography not available at the time of publication.