BeBeC-2016-S2


# CROSS SPECTRAL MATRIX DIAGONAL OPTIMIZATION

Robert P. Dougherty[1]
[1]OptiNav, Inc. and the University of Washington
1414 127th PL NE #106, 98004, Bellevue, WA, USA

## ABSTRACT

Frequency domain beamforming relies on the array Cross Spectral Matrix (CSM). In many cases, typified by windy aeroacoustics tests, the CSM is contaminated by interfering noise on the diagonal. The noise in these matrix elements be can excluded from the beamforming results by not using those values, with or without compensating by adjusting the algorithm. Some algorithms, such as Functional Beamforming however, are best suited to a compete CSM. A new method is given here for estimating the diagonal elements of a CSM from the off-diagonal elements on the basis that the corrected diagonal elements should be minimal, subject to the constraint that the adjusted CSM must be nonnegative. The algorithm is simple and practical. It does not require knowledge of the array design or steering vectors. It uses multiple calls to an eigenvalue code and a linear programming code during iteration. An example of the performance of the method for a case with 24 microphone array, a single loudspeaker source, and severe wind noise is given.

## 1 INTRODUCTION

Beamforming tests in aeroacoustics are often performed in windy locations. The signal for each microphone can be contaminated by the effects of turbulent pressure fluctuations near the microphone. The resulting microphone “self noise” increases the diagonal elements of the array Cross Spectral Matrix (CSM). In extreme cases, such as in-flow arrays in wind tunnel tests, the self noise can 20 dB or more higher than the level of the acoustic sources of interest. [1]. If conventional Frequency Domain Beamforming (FDBF) is performed with the self noise included in the CSM, then the resulting beamform maps have compressed dynamic range and unreliable peak levels. One approach to self noise is to reduce it physically by recessing the array behind a fabric screen or foam rubber sheet to create a gap between the boundary layer and the and microphones [2-4]. This can be effective, but can also be difficult to implement, does not fully remove the self noise, and complicates the acoustic propagation. Another possibility is to measure the background CSM, perhaps with the wind tunnel model removed or placed in a quiet configuration, and subtract it from the data CSM [5,6].

With without physical wind noise abatement or measurement, the compact location of self noise on the diagonal of the CSM creates a temptation to try to remove its effect mathematically. The simplest approach is to ignore the diagonal elements of the CSM in beamforming [1]. In the case of FDBF, deleting the diagonal of the CSM can substantially improve the appearance of the beamform maps, but it causes several problems, including some negative powers in the source maps and questions about the accuracy of the peak levels. Some advanced beamforming algorithms such as Functional Beamforming [7,8], depend on the complete CSM, and diagonal deletion can create particularly unreliable results. It is possible to modify deconvolution processing algorithms to operate without the diagonal CSM elements, and even estimate the self noise separately from the acoustic sources as part of a deconvolution method if desired [9-12]. This approach is can be computationally intensive and depends on knowledge of the steering vectors or equivalent information for all of the important acoustic sources. A new alternative for modifying the CSM directly, without knowledge of the steering vectors is presented below. It reduces the diagonal elements less drastically than replacing them by zero. The modified CSM is constrained to be nonnegative, which is a theoretical requirement for a CSM.

## 1.1 Structure of the Cross Spectral Matrix

Suppose there are $N$ microphones and $M$ mutually incoherent acoustic sources with narrowband source functions $q_j(t),\ j = 1, \dots, M$. Each micropone, $i$, measures the sum of the acoustic the acoustic waves, and, in addition, self noise $n_i(t)$. The model for the narrowband array pressure $N$-vector, $\boldsymbol{p}(t)$ is

$$\boldsymbol{p}(t) = \sum_{j=1}^{M} q_j(t)\boldsymbol{g}_j \ + \ \ \boldsymbol{n}(t) \tag{1}$$

where the $\boldsymbol{g}_j$ are the normalized array steering vectors for the sources. The strength of source $j$ is defined as the expected value $s_j = E\left[\left|q_j\right|^2\right], j = 1, \dots, M$. The mutual incoherence of the sources is expressed as $E\left[q_j^* q_k\right] = 0, k \neq j$. The self noise functions are assumed to be mutually incoherent and also incoherent with the acoustic sources. The self noise power for microphone $i$ is $\gamma_i = E[|n_i|^2], i = 1, \dots, N$.

The $N \times N$ CSM is $\mathbf{B} = \mathrm{E}[\boldsymbol{p}\boldsymbol{p}']$. The model assumptions give

$$\mathbf{B} = \sum_{j=1}^{M} s_j \boldsymbol{g}_j \boldsymbol{g}_j' + \sum_{i=1}^{N} \gamma_i \boldsymbol{e}_i \boldsymbol{e}_i' \tag{2}$$

where $\boldsymbol{e}_i = (0, \dots, 0, 1, 0, \dots, 0)^{\mathrm{T}}$ with the 1 in postion $i$ , $1 \leq i \leq N$.

A nonzero, complex, $N$-vector, $\boldsymbol{v}$, is said to be normalized if $\boldsymbol{v}'\boldsymbol{v} = 1$. Let $\mathbf{B}$ be a Hermitian matrix. $\mathbf{B}$ is said to be nonnegative ($\mathbf{B} \geq 0$) if its minimum eigenvalue, $\lambda_{\min}(\mathbf{B})$, is nonnegative. The minimum eigenvalue can be expressed as the Rayleigh quotient,

$$\lambda_{\min}(\mathbf{B}) = \min_{\boldsymbol{v}'\boldsymbol{v} = 1} \boldsymbol{v}'\mathbf{B}\boldsymbol{v} \tag{3}$$

By construction, $\mathbf{B}$ is Hermitian. For any normalized $\boldsymbol{v}$,

$$\boldsymbol{v}'\mathbf{B}\boldsymbol{v} = \sum_{j=1}^{M} s_j \left|\boldsymbol{g}_j'\boldsymbol{v}\right|^2 + \sum_{i=1}^{N} \gamma_i |v_i|^2 \tag{4}$$

Since all of the terms are nonnegative, this shows $\mathbf{B} \geq 0$.

### 1.2 Mathematical goal

Let $\mathbf{B} \geq 0$ be a Hermitian matrix. The mathematical problem is to partition $\mathbf{B}$ into two parts:

$$\mathbf{B} = \mathbf{C} + \mathrm{diag}(\boldsymbol{x}) \tag{5}$$

where $\boldsymbol{x} = (x_1, \ldots, x_N)^T$ is a vector whose elements are real and nonnegative and $\mathbf{C}$ is Hermitian. The basis of the partition is that $x_1 + \cdots + x_N = \mathrm{tr}[\mathrm{diag}(\boldsymbol{x})]$ is maximized subject to the constraint $\mathbf{C} \geq 0$.

### 1.3 Relationship of the mathematical solution to the CSM

The measured CSM is $\mathbf{B} = \mathbf{C}_s + \mathrm{diag}(\boldsymbol{\gamma})$ where $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_N)^T$ and

$$\mathbf{C}_s = \sum_{j=1}^{M} s_j \boldsymbol{g}_j \boldsymbol{g}_j' \tag{6}$$

If the mathematical partition is applied to $\mathbf{B}$, then $\mathbf{C} + \mathrm{diag}(\boldsymbol{x}) = \mathbf{C}_s + \mathrm{diag}(\boldsymbol{\gamma})$. In the mathematical statement, $\boldsymbol{x}$ is maximized (and therefore the diagonal elements of $\mathbf{C}$ are minimized), subject to the constraint $\mathbf{C} \geq 0$.

Suppose the number of acoustic sources, $M$, is smaller than the number of microphones, $N$. Then $\{\boldsymbol{g}_j\}$ cannot span $\mathbb{C}^N$, so there must be a vector $\boldsymbol{v} \in \ker(\mathbf{C}_s)$ such that $\boldsymbol{v}'\mathbf{C}_s\boldsymbol{v} = 0$. Applying this to the two expressions for $\mathbf{B}$ and using $\mathbf{C} \geq 0$ gives

$$\sum_{i=1}^{N} (\gamma_i - x_i)|v_i|^2 \geq 0. \tag{7}$$

In the maximization of $\boldsymbol{x}$, the result does not exceed $\boldsymbol{\gamma}$ in the direction $(|v_1|^2, \ldots, |v_N|^2)^T$. This does not prove that $\boldsymbol{x} = \boldsymbol{\gamma}$, but it constrains the error. The larger the dimension of $\ker(\mathbf{C}_s)$, the more constraints can be derived in this way and the more accurate the partition is guaranteed to be. The question of determining the circumstances under which it can be shown that $\boldsymbol{x} = \boldsymbol{\gamma}$ is an interesting problem.

# 2 ALGORITHM

## 2.1 Outline

Consider a real $N$-vector, $\boldsymbol{x}$, with each element, $x_i$ is constrained to the interval $[0, B_{ii}]$. Here $B_{ii}$ is the $i^{\text{th}}$ diagonal element of $\mathbf{B}$. Subtracting $\boldsymbol{x}$ from the diagonal of $\mathbf{B}$ gives a candidate for $\mathbf{C} = \mathbf{B} - \text{diag}(\boldsymbol{x})$. The plan is to begin with $\boldsymbol{x} = 0$ and increase $\boldsymbol{x}$ as much as possble, subject to the constraint $\mathbf{B} - \text{diag}(\boldsymbol{x}) \geq 0$.

An iterative algorithm is used to find $\boldsymbol{x}$. An expanding set of $K$ normalized trial vectors $\{\boldsymbol{v}_k, k = 1, \dots, K\}$ is constructed, initialized with $K = N$ and $\{\boldsymbol{v}\} = \{e_1, \dots, e_N\}$. There is a tentative result $\mathbf{C}$, initilazed to $\mathbf{B}$. At each step in the iteration, each of the $\boldsymbol{v}_k$ is used to express the inequality

$$\boldsymbol{v_k}'[\mathbf{B} - \text{diag}(\boldsymbol{x})]\boldsymbol{v_k} \geq 0 \tag{8}$$

as a linear constraint on $\boldsymbol{x}$:

$$\sum_{i=1}^{N} A_{ki} x_i \leq b_k\,, \qquad \text{k} = 1, \dots, \text{K} \tag{9}$$

where $A_{ki} = |v_{ik}|^2$ ($v_{ik}$ is element $i$ of trial vector $k$), and $b_k = \boldsymbol{v_k}'\mathbf{B}\boldsymbol{v_k}$. A $K \times N$ matrix $\mathbf{A} = [A_{ki}]$ and a $K$-vector $b = (b_1, \dots, b_K)^T$ are constructed. The linear programming problem "maximize $x_1 + \cdots + x_N$ subject to $\boldsymbol{x} \geq 0$ and $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}$" is solved by the simplex method to find an updated tentative solution, $\boldsymbol{x}$. The initial vectors in $\{\boldsymbol{v}\}$ ensure that $x_i \leq B_{ii}$. The tentative matix $\mathbf{C}$ is updated by replacing its diagonal elements by $B_{ii} - x_1, i = 1, \dots, N$. The spectral decomposition of the updated $\mathbf{C}$ is computed and the eigenvectors are added to $\{\boldsymbol{v}\}$ for the next iteration. If the algorithm has not converged, then $\mathbf{C}$ is likely to have some negative eigenvalues. Including the corresponding eigenvectors in $\{\boldsymbol{v}\}$ ensures that, going forward, $\mathbf{C}$ will not have negative eigenvectors in those directions. The algorithm continues with the expanded set of constraints until a predefined number of iterations have been completed or a convergence criterion is satisfied. Suitable convergence criteria might be that $\boldsymbol{x}$ is changing slowly or that $\mathbf{C}$ has no eigenvalues smaller than a certain limit.

## 2.2 Diagonal optimization algorithm

This algorithm to partition $\mathbf{B}$ into $\mathbf{C} + \text{diag}(\boldsymbol{x})$ makes use of a linear programming solver LPsolve$(A, b, c)$ that finds $\boldsymbol{x} \geq 0$ to maximize $\mathbf{c} \cdot \boldsymbol{x}$ subject to $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}$. It also requires an eigenvalue solver for Hermitian matrices. The inputs are $\mathbf{B}$ and a predefined number of iterations, nIter.

CSM Diagonal Optimization

Input $\mathbf{B}$ and nIter;
int N = dim($\mathbf{B}$);
$\mathbf{C} \leftarrow \mathbf{B}$ ;
$\{\boldsymbol{v}\} \leftarrow \{e_1, \ldots, e_N\}$ ;
double[N] c = new double[]{1,…,1};
for(int iter = 1; iter ≤ nIter; iter++){
    int Nv = size($\{\boldsymbol{v}\}$);
    double[][] A = new double[nV][N];
    double[] b = new double[nV];
    for(int k = 1; k ≤nV; k++){
        $\boldsymbol{v}_k \leftarrow \{\boldsymbol{v}\}$[k];
        for(int i = 1; i ≤N; i++){
            A[k][i] = $|(\boldsymbol{v}_k)_i|^2$ ;
        }
        b[k] = $\boldsymbol{v}_k{}'\mathbf{B}\boldsymbol{v}_k$;
    }
    double[] x = LPsolve$(A, b, c)$;
    for(int i = 1; i ≤N; i++)$\mathbf{C}_{ii} \leftarrow \mathbf{B}_{ii} -$ x[i];
    $\{\boldsymbol{v}\} \leftarrow \{\boldsymbol{v}\} \cup$ eigenvectors($\mathbf{C}$);
}
return $\mathbf{C}$ and x;

## 3 EXAMPLE

An example test setup is shown in Fig. 1. A 24 element planar microphone array ("OptiNav Array 24 Jr.") was configured to image a loudspeaker at a distance of 1 m. A fan-driven air nozzle was arranged so that the jet impinged obliquely on the array face, as shown on in Fig. 1. The microphones in Array 24 Jr. are electret capsules that are recessed below small holes in the sheet metal plate that forms the array surface. A broadband signal generator was connected to the speaker and adjusted so the that the level from the speaker was considerably lower than the wind noise. Data were acquired for 15 seconds for each condition. Results are presented here for two conditions: speaker alone, and speaker with wind. The objective for the beamforming is to be able to measure the level from the speaker using the data set that is contaminated by wind noise. The processing was performed in 41 1/12 octave bands from 1-10 kHz.

### 3.1 Array spectra with and without wind

Spectra from all 24 microphones with the speaker on and the wind off are given in Fig. 2. There is about 6 dB of scatter between the microphones, so a CSM diagonal reconstruction method that assumed that all the microphone levels were the same would not be accurate. The corresponding plot for both the speaker and the wind operating is given in Fig. 3.

The increase in SPL due to turning on the wind is shown in Fig. 4. The amount of increase varies dramatically over the array with a 30 dB range. The common assumption that the self noise is equal in all of the channels would clearly be wrong here. Four of the microphones,

numbers 12, 17, 20, and 22 were less impacted by the wind than the others. These are all near the bottom of the array, suggesting that the difference was, in part, caused by a jet pointing error.

The legend is not shown in Figs. 2-4 because it was too large to fit, but Microphone 1 is indicated by the heavier line. The microphone is used as the reference in the subsequent analysis.

## 3.2 Beamforming results

The two data sets were analyzed in using FDBF and an improved version of Functional Beamforming which does not suffer from the reduction in peak level that is seen in the original method. The FB parameter $\nu$ was set to 100. The CSM data for the beamforming was either the raw CSM including the self noise, the CSM with the diagonal elements set to 0 (diagonal deletion), or the result of iterative diagonal optimization. The number of iterations for the optimization was fixed at 100. The total computer time to process the 41 bands in a spectrum (starting from the raw CSM, which took a few seconds to compute from the time series data) was 136 s for FDBF without diagonal optimization, 208 s for FDBF with diagonal optimization, 86 s for Functional Beamforming with out diagonal optimization, and 160 s for Functional Beamforming with diagonal optimization. Functional Beamforming is faster than FDBF in the processing code (Beamform Interactive) because it is implemented more efficiently.

### 3.2.1 Beamform maps

Beamform maps for thee selected frequencies are shown in Fig. 5 (2 kHz), 6 (4 kHz) and 7 (8 kHz). Figs. 5a, 6a, and 7a show FDBF and Figs. 5b, 6b, and 7b show FB. In each figure, the left column is for the speaker alone and the right column pertains to the speaker and wind. The three rows correspond to the treatment of the CSM diagonal: retain (raw CSM), delete, or optimize.

None of the methods had any trouble locating the speaker with no wind, although the FB plots are sharper than the FDBF case. FDBF with wind and the raw CSM (upper right plot) has very poor dynamic range, as expected. FB has better dynamic range than FDBF with the raw CSM. For both FDBF and FB, the CSM treatment that gives the sharpest images with wind is diagonal deletion. The images with wind and CSM optimization appear sharper than the raw CSM plots with wind, but not as sharp as the diagonal deletion plots with wind.

### 3.2.2 Beamform spectra

Spectral plots are shown in Figs. 8-13. Each plot shows the microphone 1 spectrum for the speaker alone and for the speaker and wind, along with a beamforming spectrum for each of those conditions. The six figures represent the six combinations of CSM treatment and beamforming algorithm. The beamforming spectra show the peak beamforming level in a small region of interest centered on the speaker location. The beamforming spectra are intended to show the level at the array due to the speaker source. Ideally, they should match the spectrum for microphone 1 for no wind, since the speaker source strength should be the same for the two datasets.

The FDBF curve using the raw CSM with no wind (Fig. 8) matches the no-wind spectrum for microphone 1 very well, as expected. The corresponding FDBF curve with wind (also in

Fig. 8) tracks 6-12 dB below the curve for microphone 1 with wind; the speaker appears not to be seen.

Looking ahead to Figs. 9-13, all of the methods show good agreement between microphone 1 with no wind and beamforming with no wind. The results that change according the methods are the spectra with the speaker and wind.

In Fig. 9, FDBF with diagonal deletion and wind falls almost consistently below the reference curve by about 3 dB. Exceptions occur at 1250 Hz and 3 kHz. Both of these are frequencies where the SNR is particularly low (Fig. 4). The spectrum for FDBF with wind and diagonal optimization tracks the expected curve without the bias of the diagonal deletion case, but, again, misses at low frequency and around 3 kHz.

The Functional Beamforming curve for wind and the raw CSM appears to follow below the microphone spectrum with noise, like the FDBF case, for the lower half of the frequency range, but is farther below it, perhaps 15 dB. In the upper half of the frequency range, the FB curve is closer to the no-wind microphone 1 curve than the microphone spectrum with wind.

Figure 12 shows that diagonal deletion in the case with wind causes the levels of the FB maps to be very unreliable. This result is the opposite of the spatial appearance if the beamform maps, which appear to be improved with diagonal deletion. It is possible that FB could be adjusted to avoid this failure.

Diagonal optimization improves the FB results from the worst case (Fig. 12) to the best (Fig. 13.)

The RMS differences between the beamforming spectra and the no-wind microphone 1 spectra are compared in Fig. 14. For all of the no-wind beamforming cases, the RMS error is close to 1.7 dB. With wind, diagonal optimization is the best CSM treatment for both FDBF and FB. Comparing beamforming methods, FB is the better one for the raw CSM and diagonal optimization. FDBF is better for diagonal deletion.

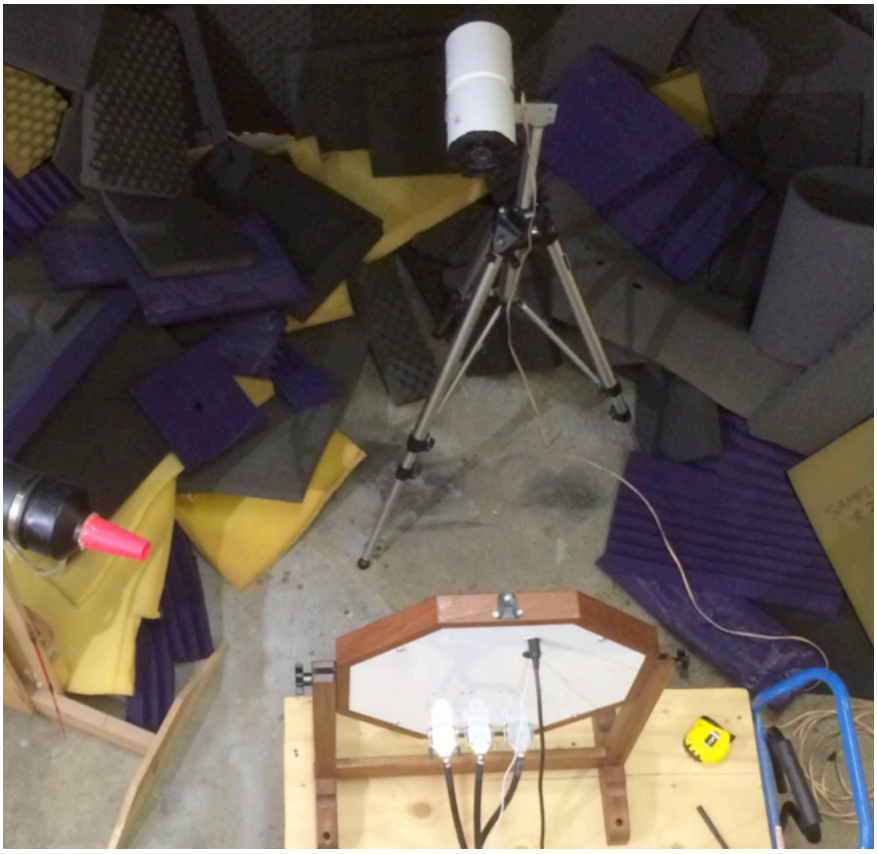

*Fig. 1. Test setup. The phased array attempts to measure a broadband speaker source while a jet of air from the nozzle on the left impinges obliquely on the array face.*

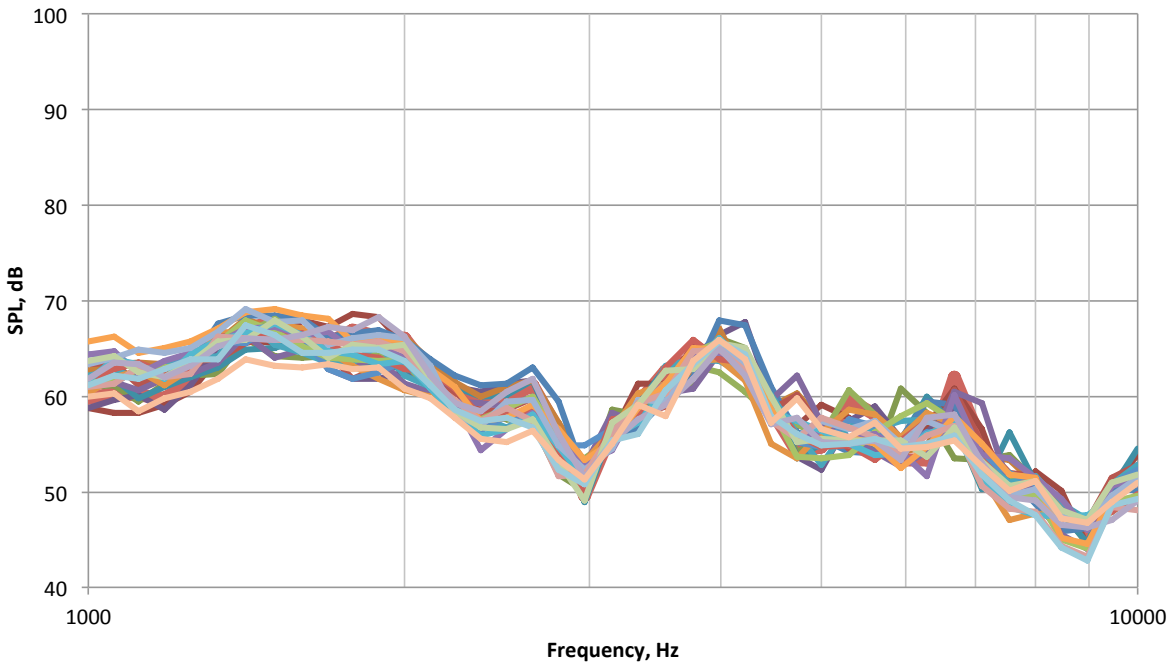


*Fig. 2. Microphone spectra for the speaker alone. The heavy curve is microphone 1, which is used as the reference for beamforming comparisons.*

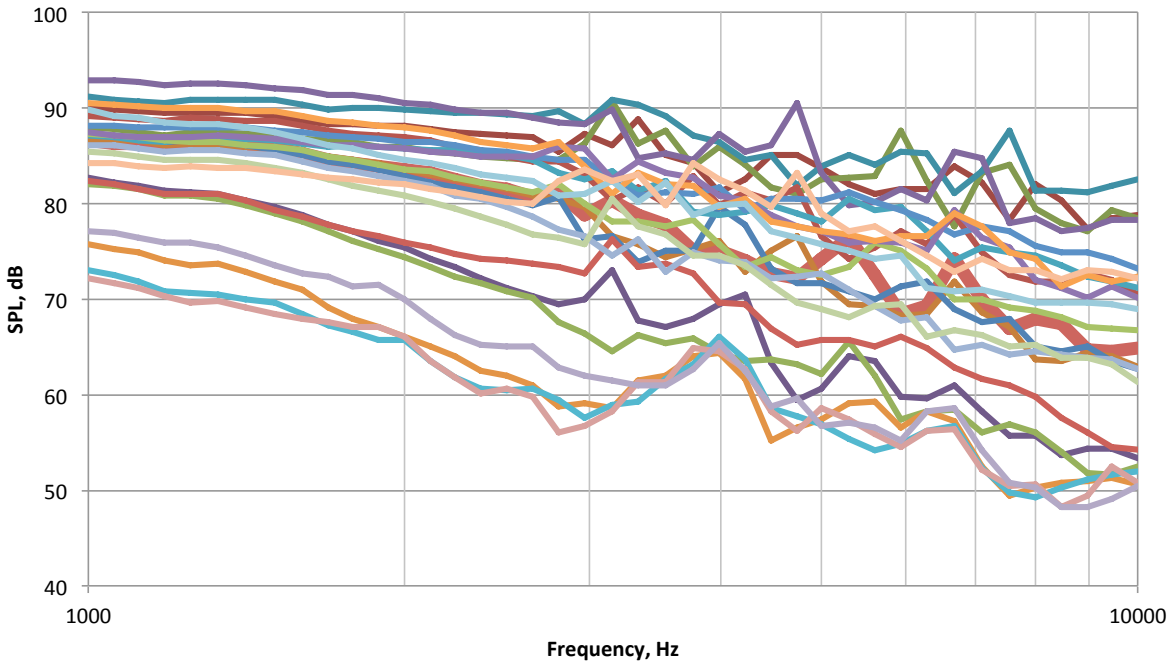


*Fig. 3. Microphone spectra for the speaker and wind.*

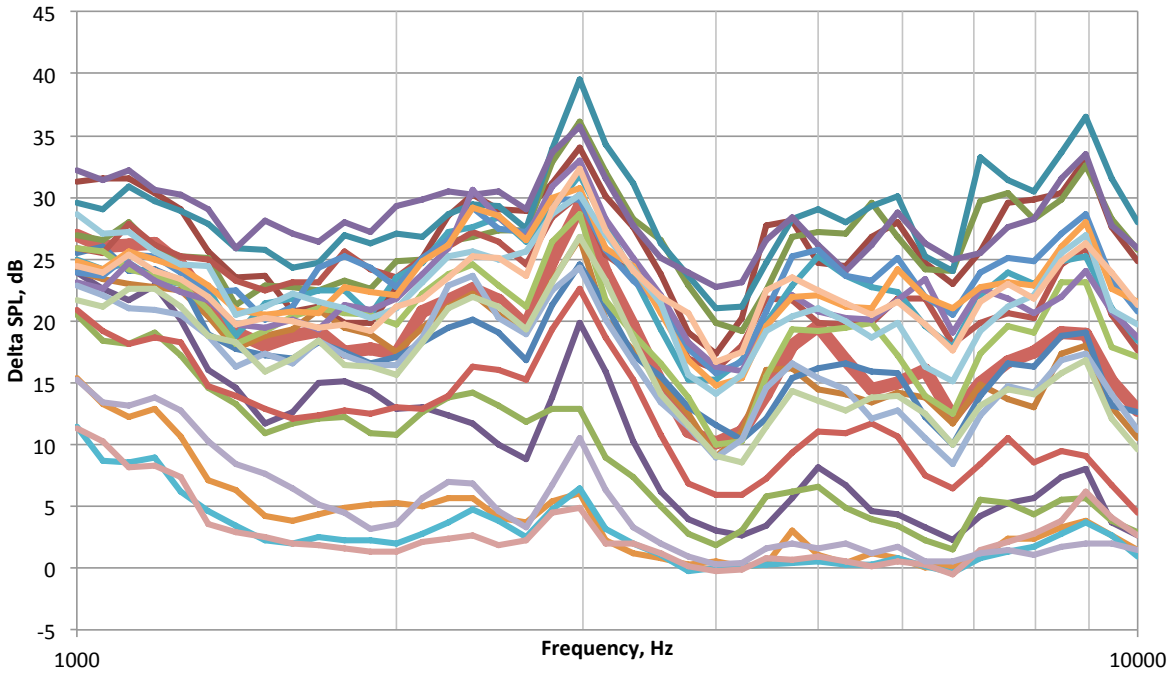


*Fig. 4. Relative microphone self noise: the difference between the levels with wind and the speaker and the speaker alone.*

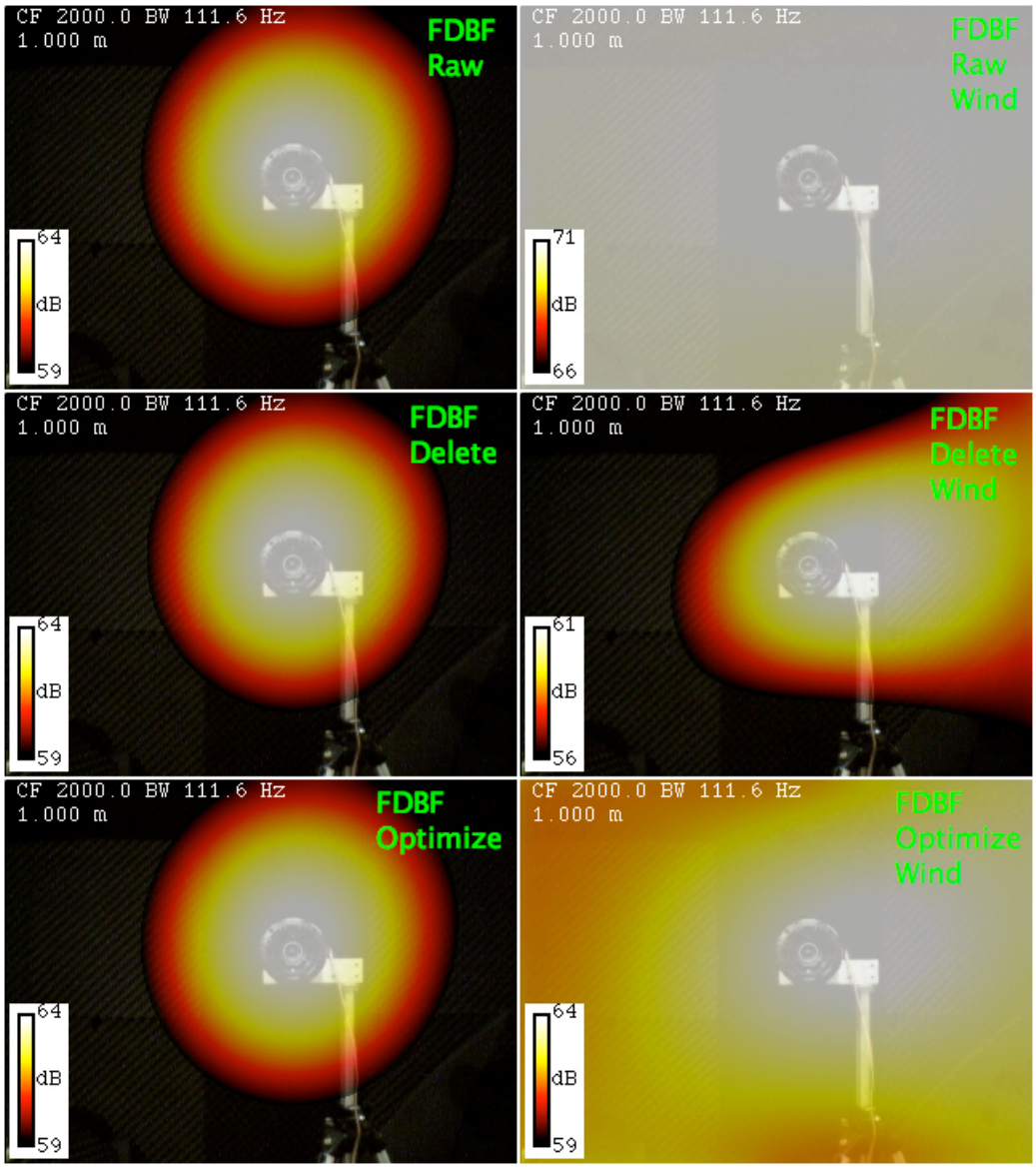


*Fig. 5a. Beamforming plots for FDBF, 2 kHz, Left column: speaker alone. Right column: speaker and wind. Top row: raw CSM. Middle row: diagonal deletion CSM. Bottom row: diagonal optimization CSM.*

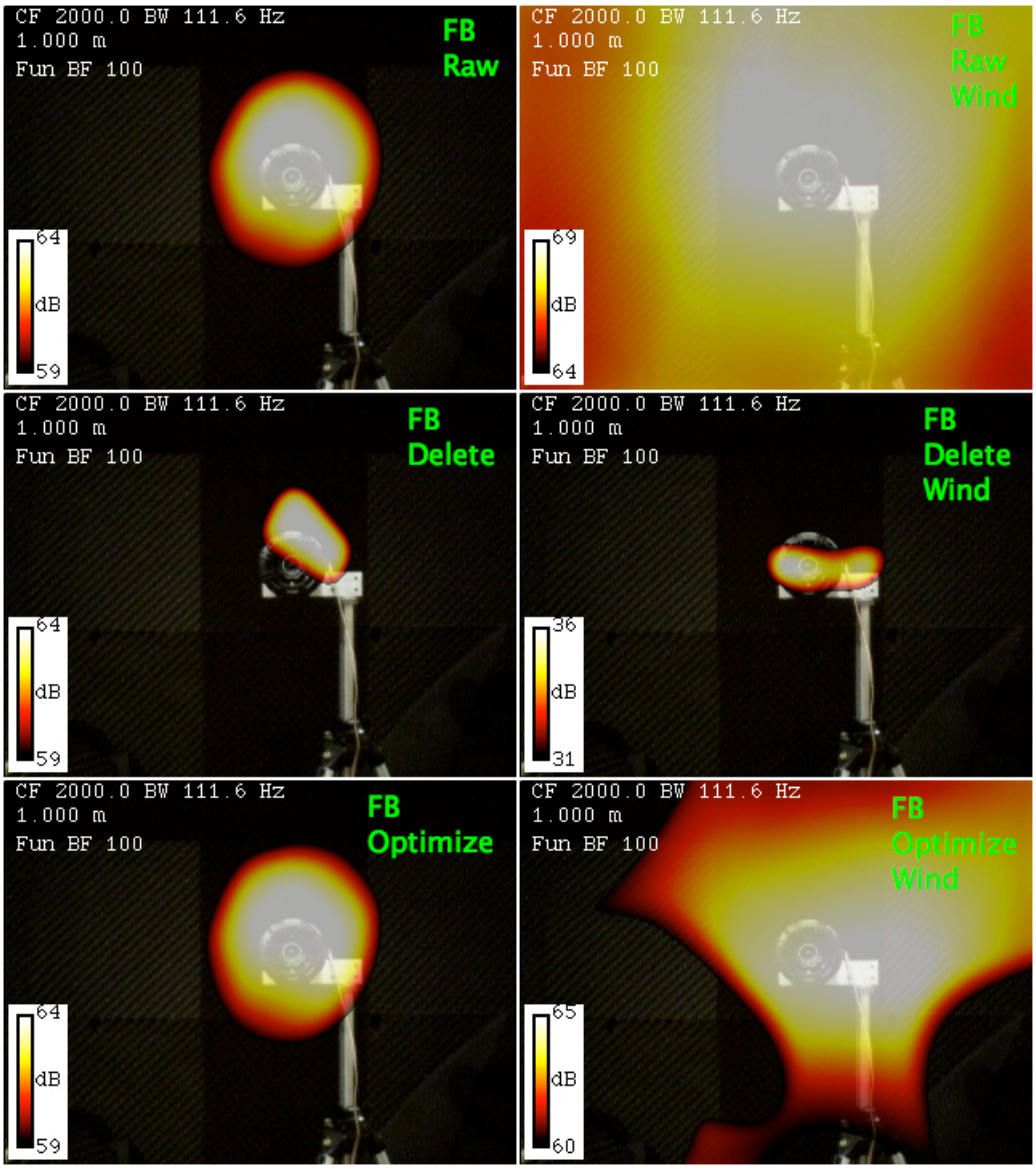


*Fig. 5b. Beamforming plots for Functional Beamforming, 2 kHz, Left column: speaker alone. Right column: speaker and wind. Top row: raw CSM. Middle row: diagonal deletion CSM. Bottom row: diagonal optimization CSM.*

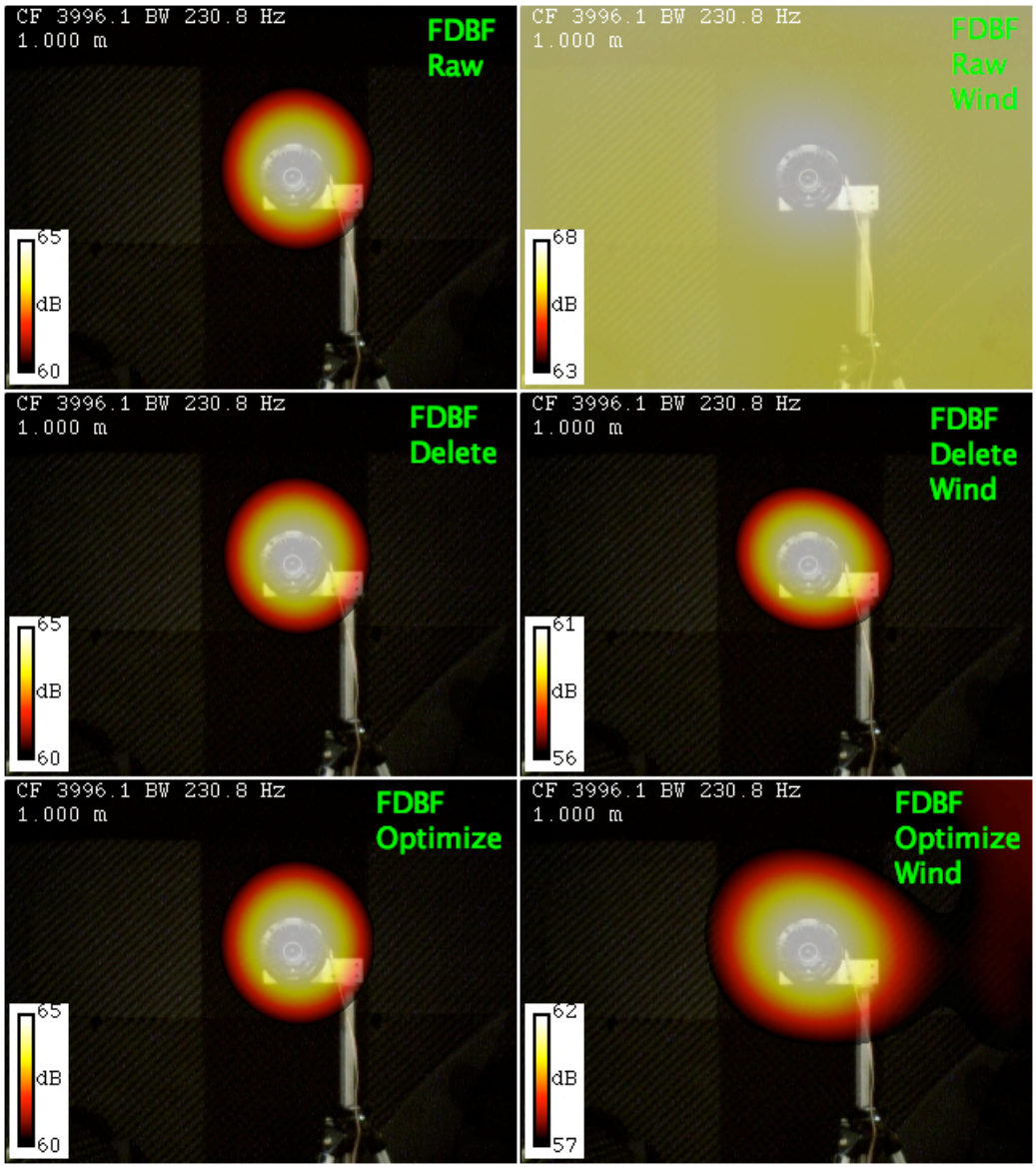


*Fig. 6a. Beamforming plots for FDBF, 4 kHz, Left column: speaker alone. Right column: speaker and wind. Top row: raw CSM. Middle row: diagonal deletion CSM. Bottom row: diagonal optimization CSM.*

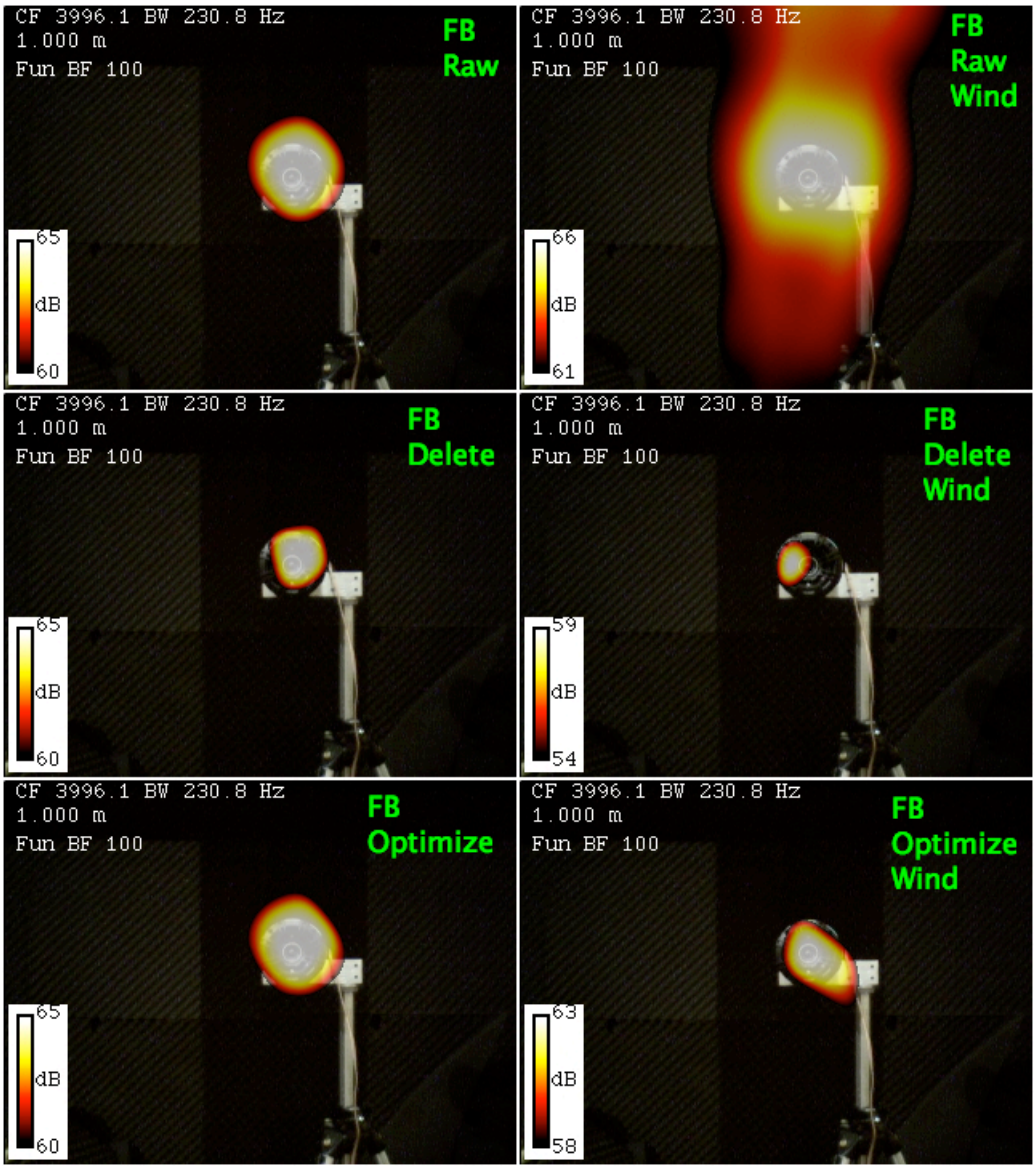


*Fig. 6b. Beamforming plots for Functional Beamforming, 4 kHz, Left column: speaker alone. Right column: speaker and wind. Top row: raw CSM. Middle row: diagonal deletion CSM. Bottom row: diagonal optimization CSM.*

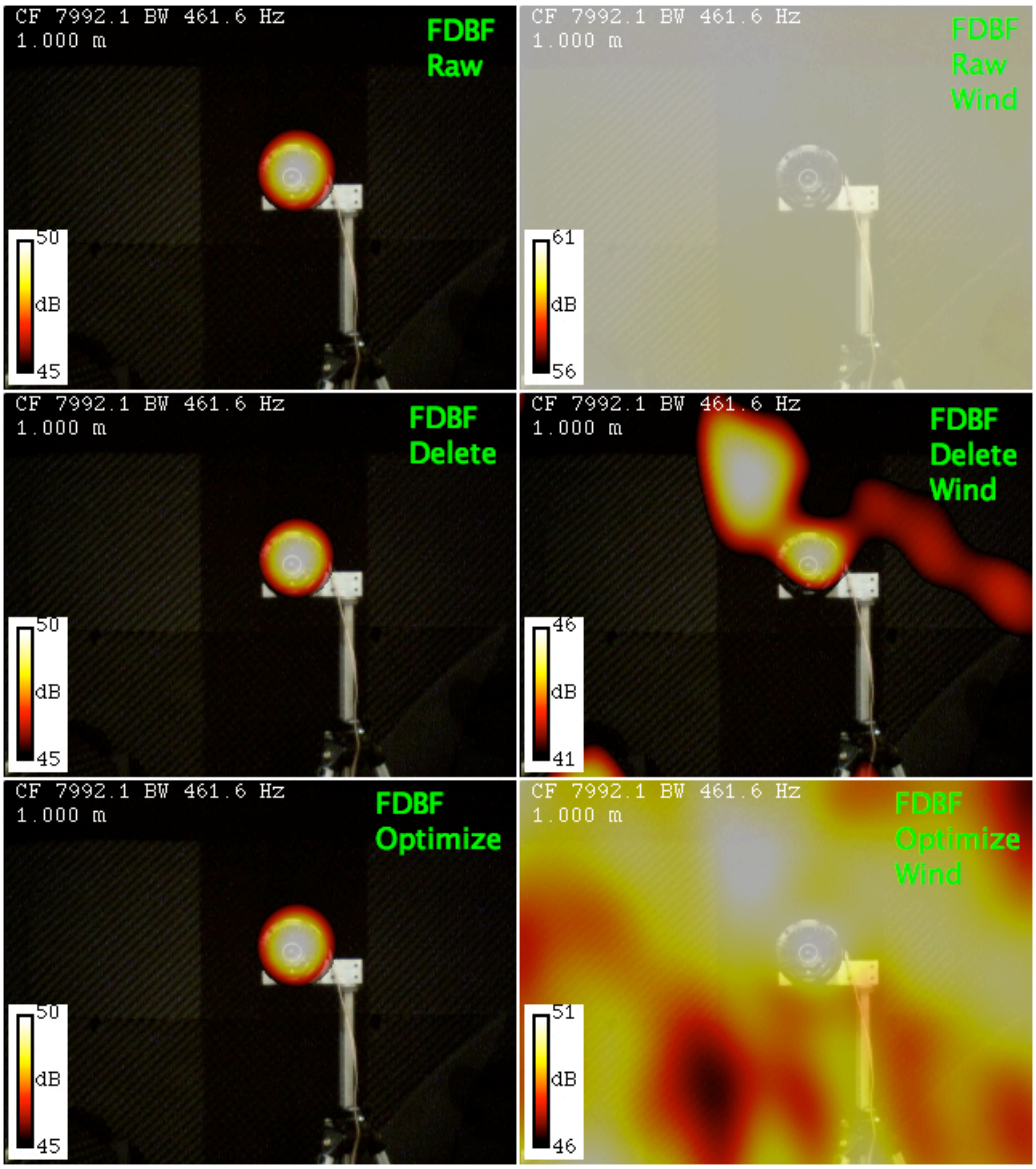


*Fig. 7a. Beamforming plots for FDBF, 8 kHz, Left column: speaker alone. Right column: speaker and wind. Top row: raw CSM. Middle row: diagonal deletion CSM. Bottom row: diagonal optimization CSM.*

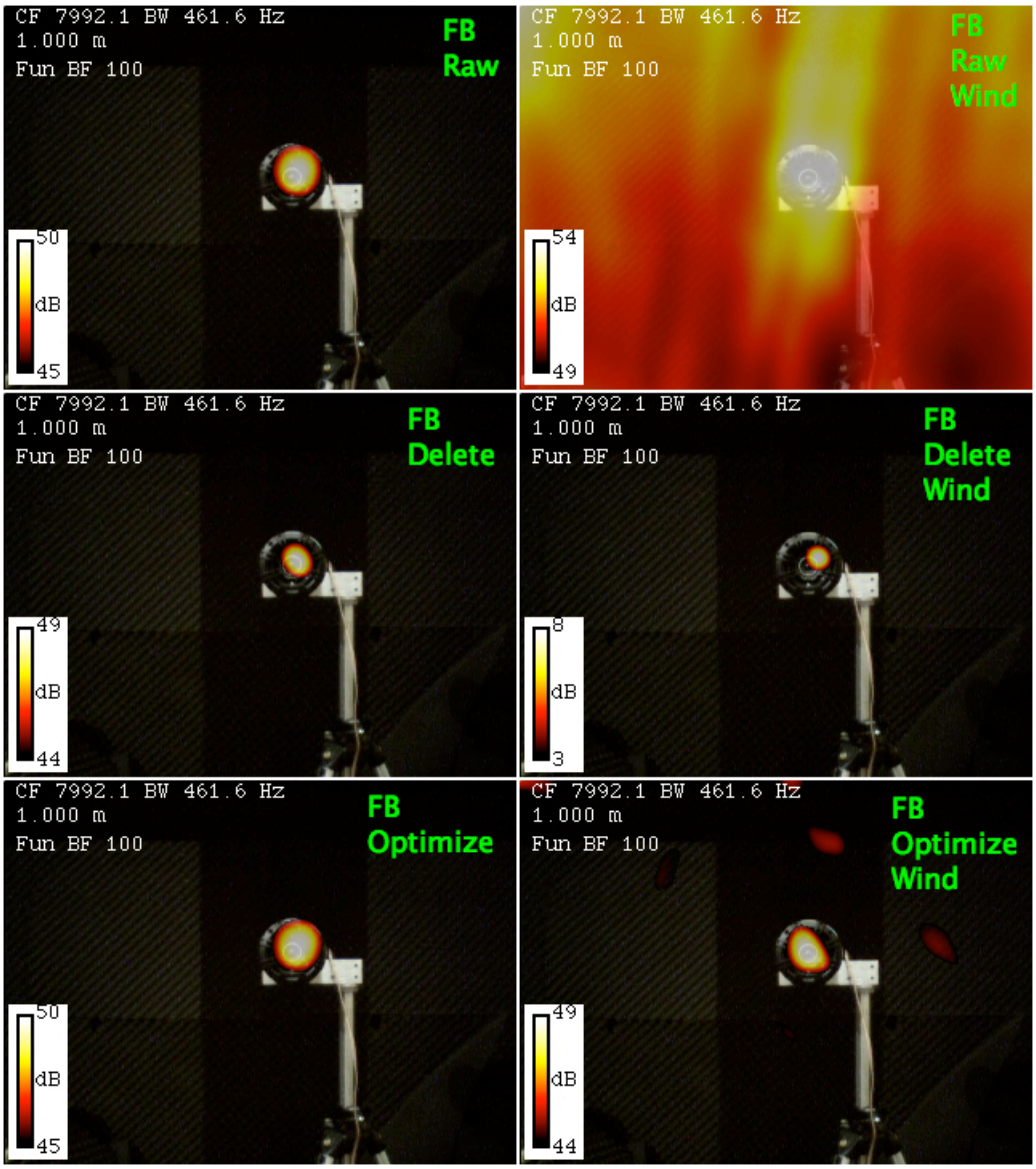


*Fig. 7b. Beamforming plots for Functional Beamforming, 8 kHz, Left column: speaker alone. Right column: speaker and wind. Top row: raw CSM. Middle row: diagonal deletion CSM. Bottom row: diagonal optimization CSM.*

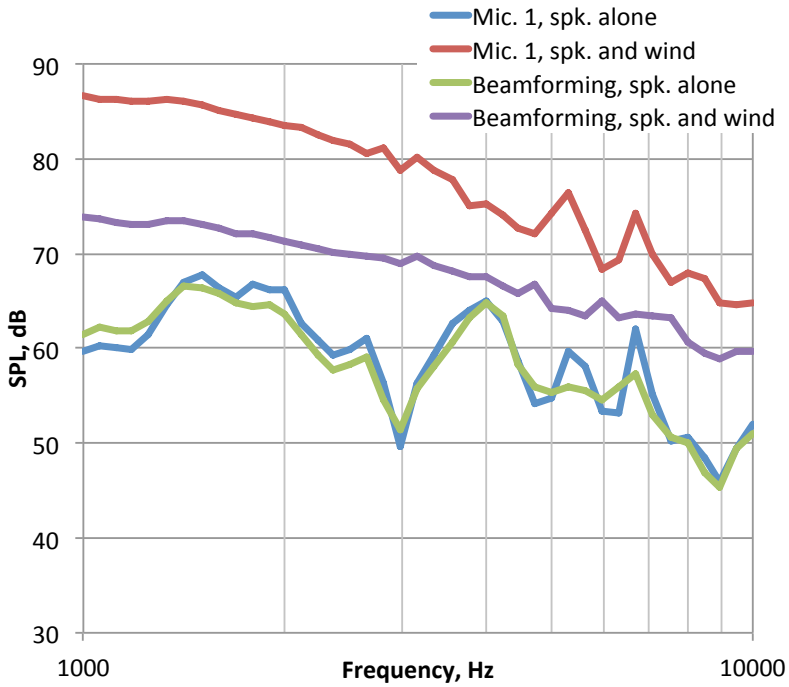


*Fig. 8. Spectra from microphone 1 and beamforming with and without wind. FDBF with the raw CSM.*

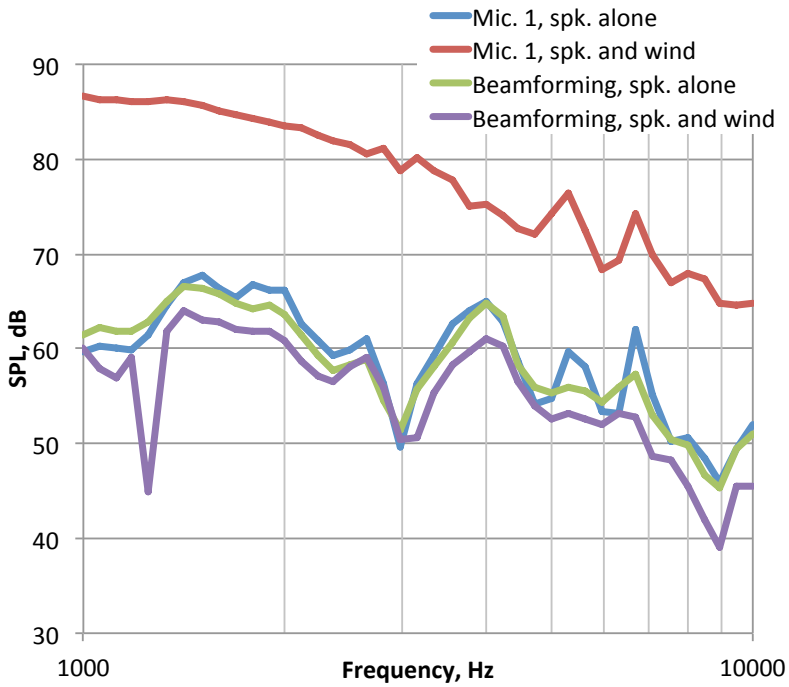


*Fig. 9. Spectra from microphone 1 and beamforming with and without wind. FDBF with diagonal deletion CSM.*

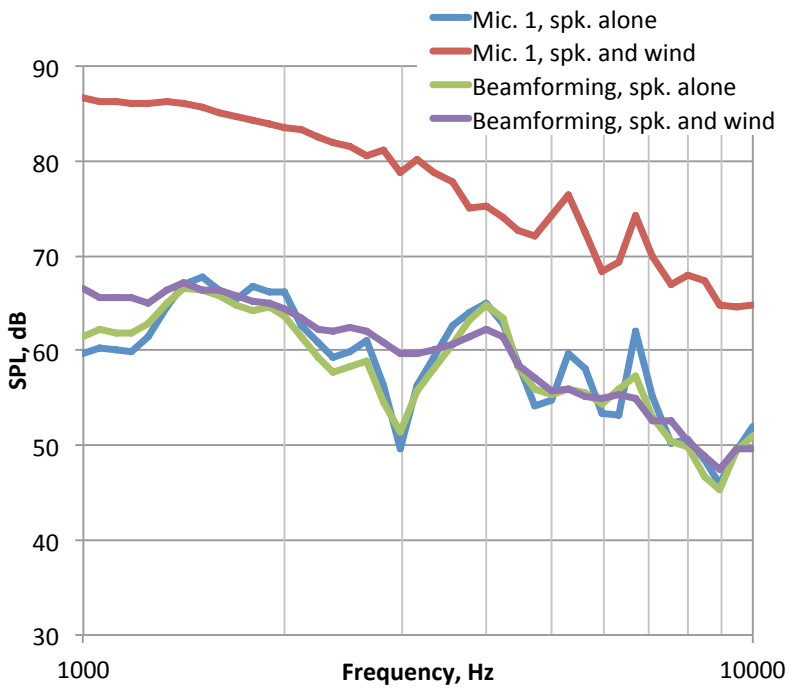


*Fig. 10. Spectra from microphone 1 and beamforming with and without wind. FDBF with optimized CSM.*

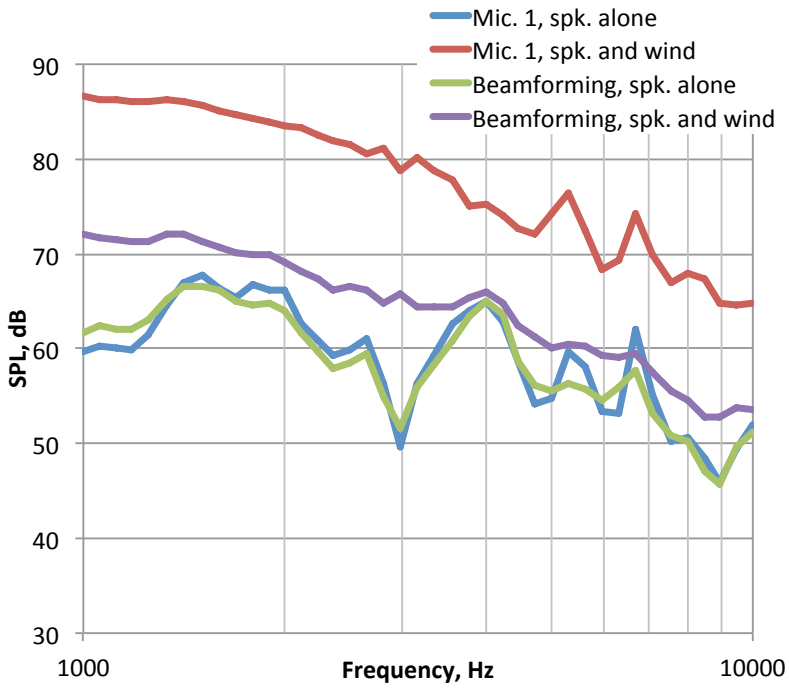


*Fig. 11. Spectra from microphone 1 and beamforming with and without wind. Functional Beamforming with the raw CSM.*

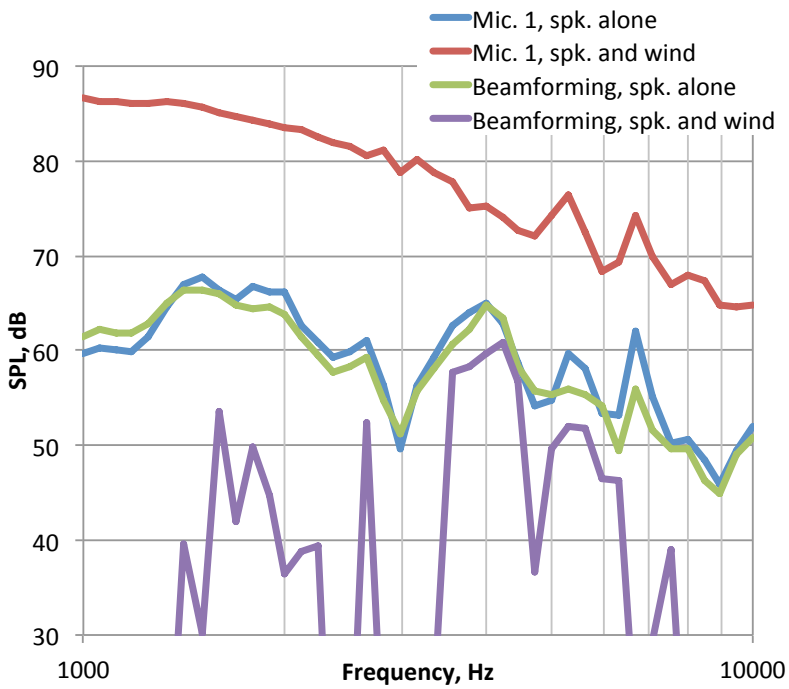


*Fig. 12. Spectra from microphone 1 and beamforming with and without wind. Functional Beamforming with diagonal deletion CSM.*

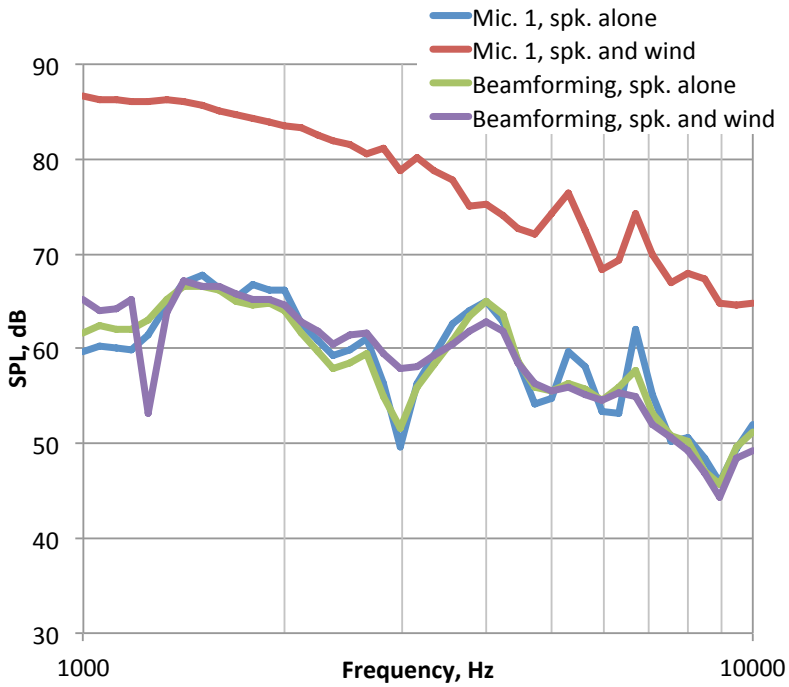


*Fig. 13. Spectra from microphone 1 and beamforming with and without wind. Functional Beamforming with optimized CSM.*

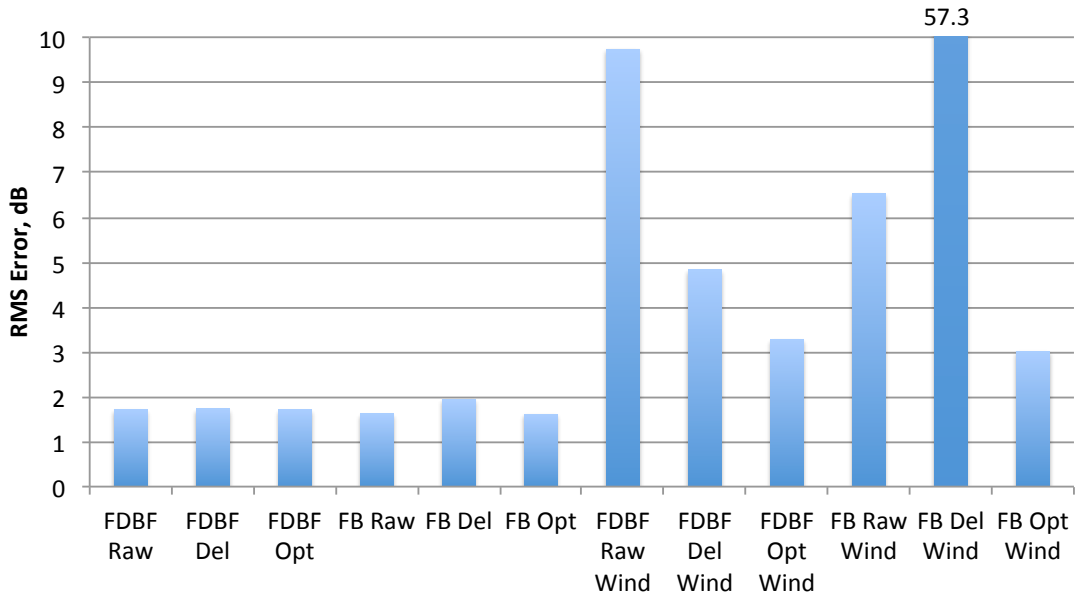


*Fig. 14. RMS difference between beamforming spectra and the spectrum from microphone 1 with no wind. Left side: beamforming with no wind data. Right side: beamforming with speaker and wind.*

## 4 CONCLUSIONS

A new CSM diagonal optimization algorithm has been presented. It offers much of the self-noise interference reduction of diagonal deletion while avoiding the potentially severe problems of negative CSM eigenvalues. It does not assume that either the acoustic microphone levels or self noise levels are uniform, and does not rely on knowledge of the array steering vectors. It is feasible and has been shown to be effective for a case of a simple source with a very high level of self noise contamination. Its formulation suggests that it works best when the number of acoustic sources is small compared with the number of microphones.

In the example test, it was found that CSM diagonal deletion applied to a case with high self noise increased the dynamic range and sharpened the peaks, but introduced errors in the peak level. The errors can were very large in the case of Functional Beamforming. Diagonal optimization significantly improved the accuracy of the levels of the beamforming peaks.

## REFERENCES

[1] R.P. Dougherty, “Beamforming in Acoustic Testing,” Aeroacoustic Measurements, Edited by T. J. Mueller, Springer, Berlin, 2002, pp. 93-96.

[2] S.M. Jaeger, W.C. Horne and C.S. Allen, “Effect of surface treatment on array microphone self-noise,” AIAA/CEAS Paper 2000-1937, 2000.

[3] P. Sijtsma and H. Holthusen, "Source location by phased array measurements in closed wind tunnel test sections," AIAA Paper No. 99-1814, 1999.

[4] R.P. Dougherty, "Directional acoustic attenuation of planar foam rubber windscreens for phased arrays," Berlin Beamforming Conference BeBeC-2012-05, Feb. 2012.

[5] D. Blacodon and G. Élias, "Level estimation of extended acoustic sources using an array of microphones," Wind Tunnels, AIAA Paper No. 2003-3199, May 2003.

[6] C.J. Bahr and W.C. Horne,"Advanced background subtraction applied to aeroacoustic wind tunnel testing", AIAA Paper No. 2015-3272, June, 2015.

[7] R.P. Dougherty, "Functional Beamforming", Berlin Beamforming Conference, BeBeC 2014-1, 2014.

[8] R.P. Dougherty, "Functional Beamforming for aeroacoustic source distributions", AIAA Paper 2014-3066, 2014.

[9] K. Ehrenfried, L. Koop, A. Henning and K. Kaepernick, "Effects of wind-tunnel noise on array measurements in closed test sections," BeBeC Paper No. 2006-07, 2007.

[10] P. Sijtsma. "CLEAN based on spatial source coherence." Int. J. Aeroacoustics, 6, 357–374, 2007.

[11] T. Brooks and W. Humphreys, "Extension of DAMAS phased array processing for spatial coherence determination (DAMAS-C)," AIAA Paper 2006-2654, May, 2006.

[12] R.P. Dougherty, R.C. Ramachandran and G.Raman, "Deconvolution of Sources in Aeroacoustic Images from Phased Microphone Arrays Using Linear Programming," , AIAA Paper 2013-2210, Berlin, Germany, 2013.