Weight Functions for Two-Dimensional Arrays using Regularized Deconvolution

James Alsup
AETC, Incorporated
San Diego, CA

Harper Whitehouse
Linear Measurements, Inc.
San Diego, CA

November 7, 2001

Objective

Improve Array Beamformer Performance

- for arrays with non-uniformly-spaced elements
- via application of beam-space weights

Approach

Suppress sidelobes

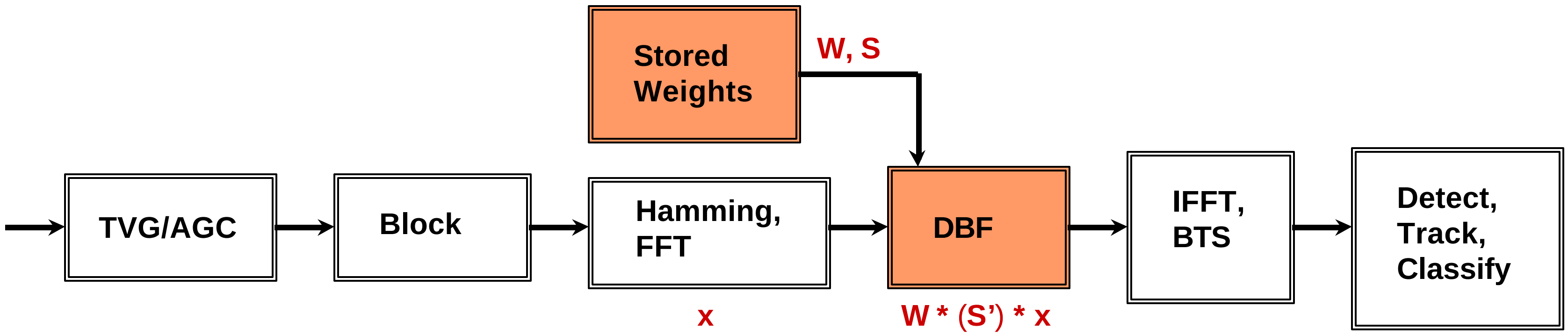
- retaining or improving mainlobe resolution
- with fixed weight matrices

Implementation

- **Formulate beamformer as a Fredholm equation**
  - **use point spread function (psf) kernel concept**
  - **kernel is look-direction dependent**

- **Use modern deconvolution methods**
  - **use pseudo-inverse of the Fredholm kernel**
    - **i.e., pseudo-inverse of the psf**
  - **regularize to achieve numerical stability**
  - **constrain to achieve robust performance**

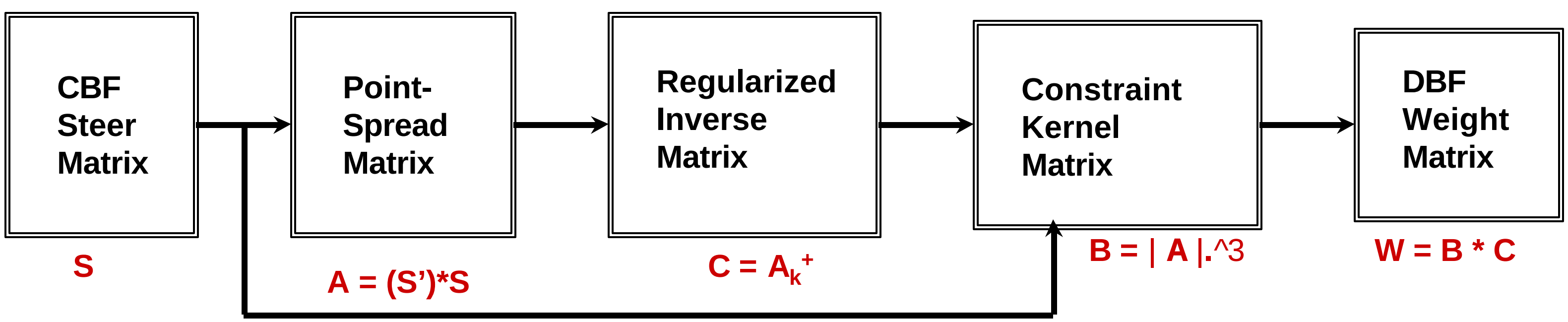
Processing Flow



TVG: time-varying gain
AGC: automatic gain control

BTS: beam time series
DBF: deconvolutional beamforming

Algorithm for Stored Weights



Algorithm Overview

1. Observed vs. “ideal” beampattern

$$\mathbf{g} = \mathbf{A} * \mathbf{f}$$

2. Observed beampattern vs. element data

$$\mathbf{g} = \mathbf{S}' * \mathbf{x}$$

3. Point-spread function vs. steering matrix

$$\mathbf{A} = \mathbf{S}' * \mathbf{S}$$

4. Solve for “ideal” beampattern

$$\mathbf{f} = (\mathbf{A}_k^{+}) * \mathbf{S}' * \mathbf{x}$$

5. Desired vs. “ideal” beampattern

$$\mathbf{h} = \mathbf{B} * \mathbf{f}$$

6. Desired beampattern vs. weight matrix

$$\mathbf{h} = \mathbf{W} * \mathbf{S}' * \mathbf{x}$$

7. Reformulate for signal phase property

$$\mathbf{h} = | \mathbf{W} * \mathbf{S}' * \mathbf{x} | .* \exp(j * \text{angle}(\mathbf{g}))$$

AETC

Algorithm - 1

LMI

1. $g = A * f$

f = "ideal" vector of target azimuths

- spanning 0-360 degrees
- e.g., 0:8:352, 45x1

g = observed beampattern vector

- spanning 0-360 degrees, e.g., 45x1
- includes mainlobe & sidelobes
  - superposed for each "target"

A = point-spread matrix

- delta-function in, beampattern out
- e.g., 45x45

2. $g = S' * x$

g = observed beampattern vector

- spanning 0-360 degrees, e.g., 45x1
- includes mainlobe & sidelobes

x = observed element data vector

- at single frequency
- complex; e.g., 40x1

S = steer matrix

- one column per look direction
- phase-only; e.g., 40x45

3. $A = S' * S$

S = steer matrix

- one column per look direction
- phase-only; e.g., 40x45

A = point-spread matrix

- one CBF beampattern per row
- single frequency; e.g., 45x45

4. Combine equations:

$$A * f = S' * x$$

Solve for f :

$$f = (A_k^+) * S' * x$$

A_k^+ = regularized inverse of A

using first k singular values

Algorithm - 2

5. $\mathbf{h} = \mathbf{B} * \mathbf{f}$

f = "ideal" vector of target azimuths

h = desired beampattern vector response

- mainlobe not too narrow
- sidelobes reduced from g

B = smoothing constraint matrix

- delta-function in, improved beampattern out
- e.g., 45x45

6. Combine equations:

$$\mathbf{h} = \mathbf{B} * (\mathbf{A}_k^{+}) * \mathbf{S}' * \mathbf{x}$$

rewrite :

$$\mathbf{h} = \mathbf{W} * \mathbf{S}' * \mathbf{x}$$

where

$$\mathbf{W} = \mathbf{B} * (\mathbf{A}_k^{+})$$

W represents stored weight matrix

- beamspace matrix
- single frequency, NxN

7. Reformulate for signal phase property :

$$\mathbf{h} = |\mathbf{h}| .* \exp(j * \text{angle}(\mathbf{g}))$$

then :

$$|\mathbf{h}| = | \mathbf{W} * \mathbf{S}' * \mathbf{x} |$$

where

$$\mathbf{W} = \mathbf{B} * (\mathbf{A}_k^{+})$$

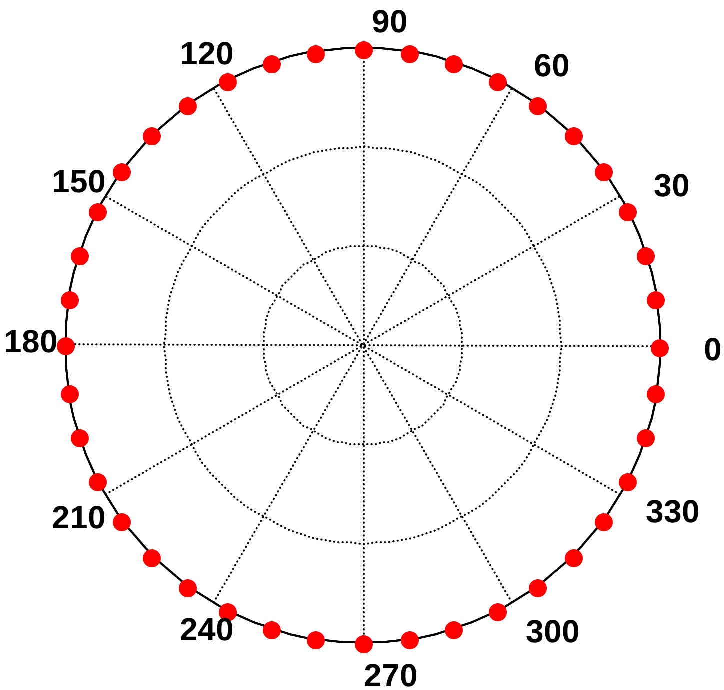
(don't want to mess with original phase of g, anyway)

Details/examples below

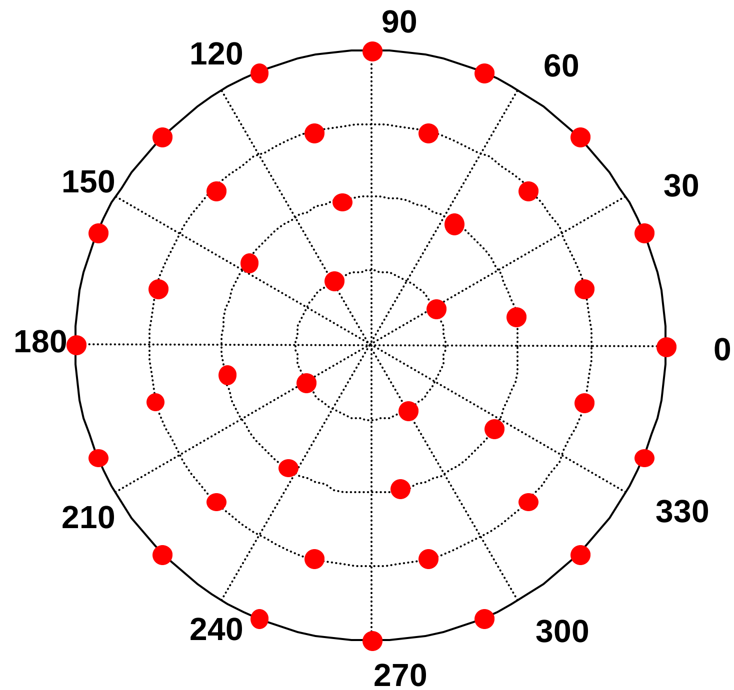
- how to get (A_k^{+})
- how to select B
- appearance vs. frequency
- sensitivity to errors

Array Examples

L M I



Horizontal Planar Array, Annular
D = 60m, M = 40

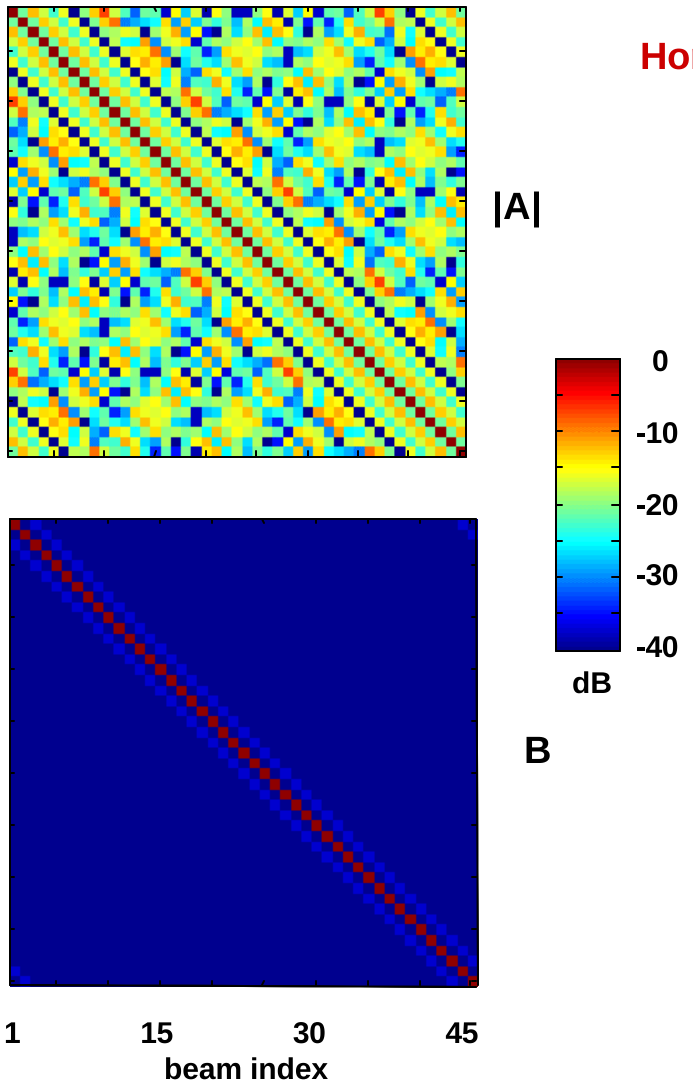


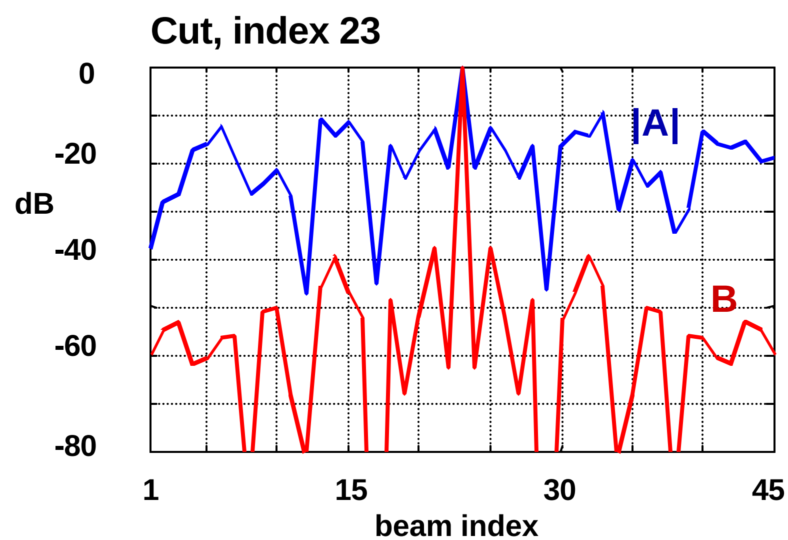
Horizontal Planar Array, Filled
D = 8m, M = 40

A, B Matrices, 400Hz

LMI

Horizontal Planar Array, Annular





Computation Details

L M I

(1) [u,s,v] = svd(A)

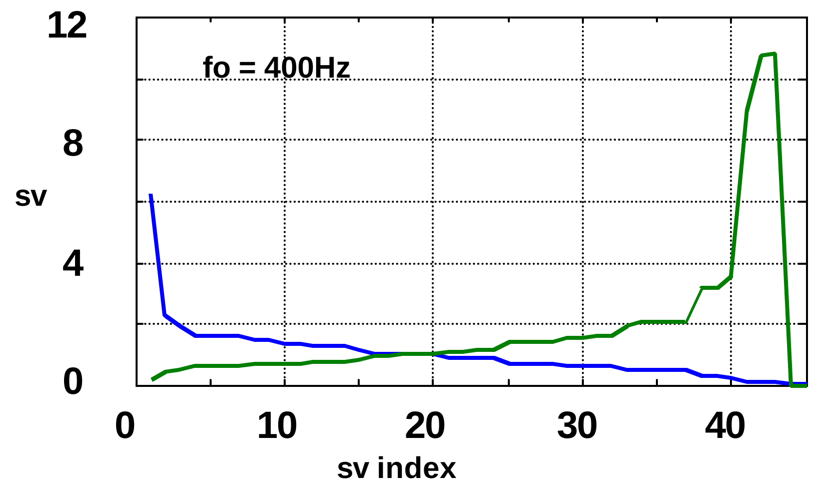
(2) si = inverse of s via Per Christian Hansen method
(keep k singular values, invert each, make remaining ones zero)

(3) A_k^+ = v * si * (u') ---- regularized inverse

(4) B = | A | .^ (3) ---- constraint/smoothing matrix

(5) W = B * (A_k^+)

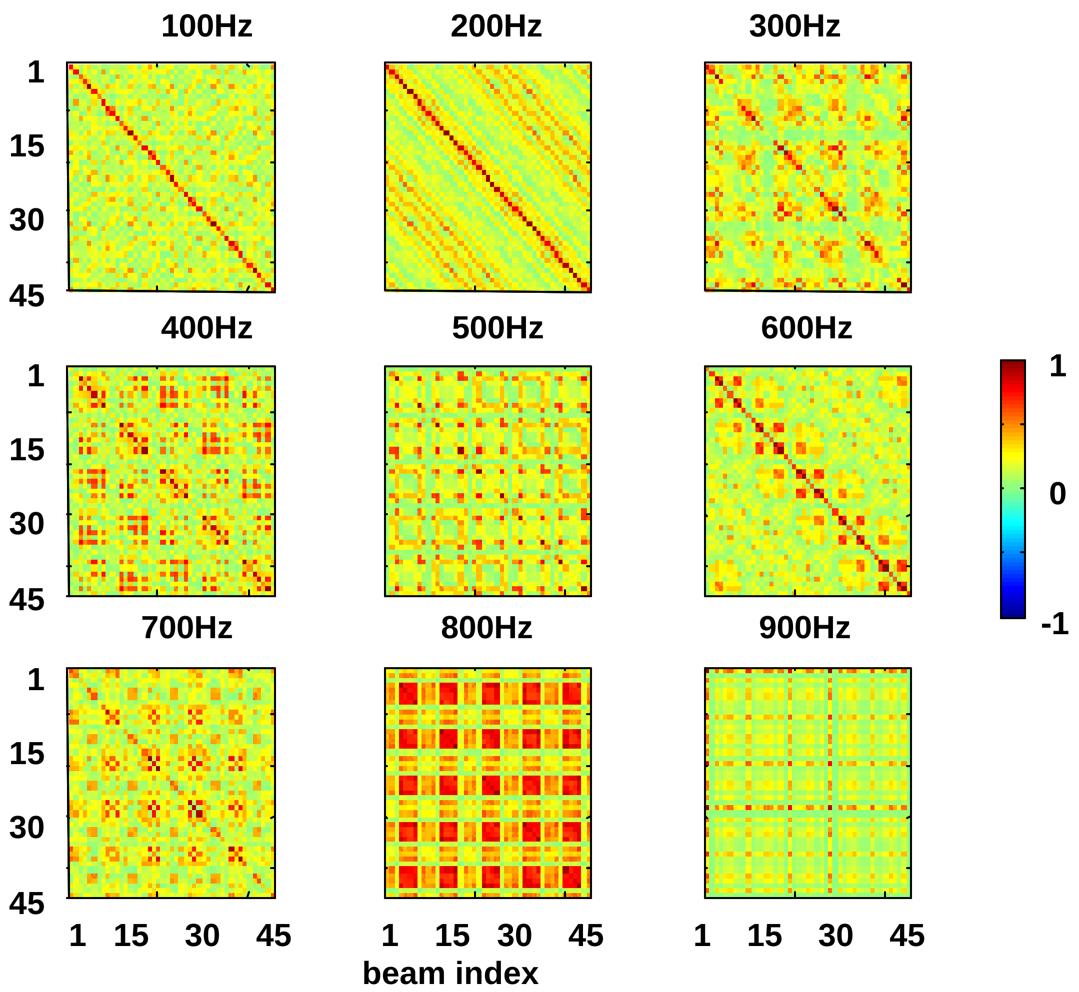
Horizontal Planar Array, Annular



A = average point-spread matrix
sv = singular value

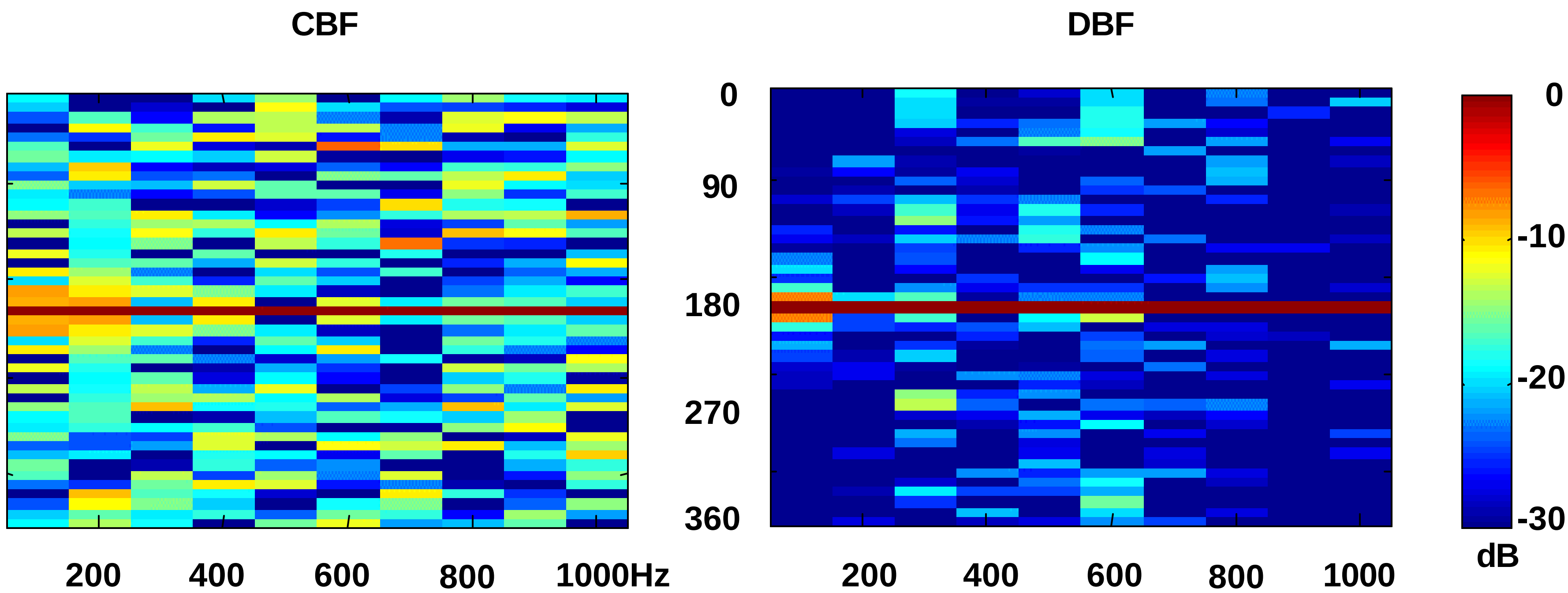
Weight Matrix W

Horizontal Planar Array, Annular



CBF & DBF vs. Frequency

Horizontal Planar Array, Annular

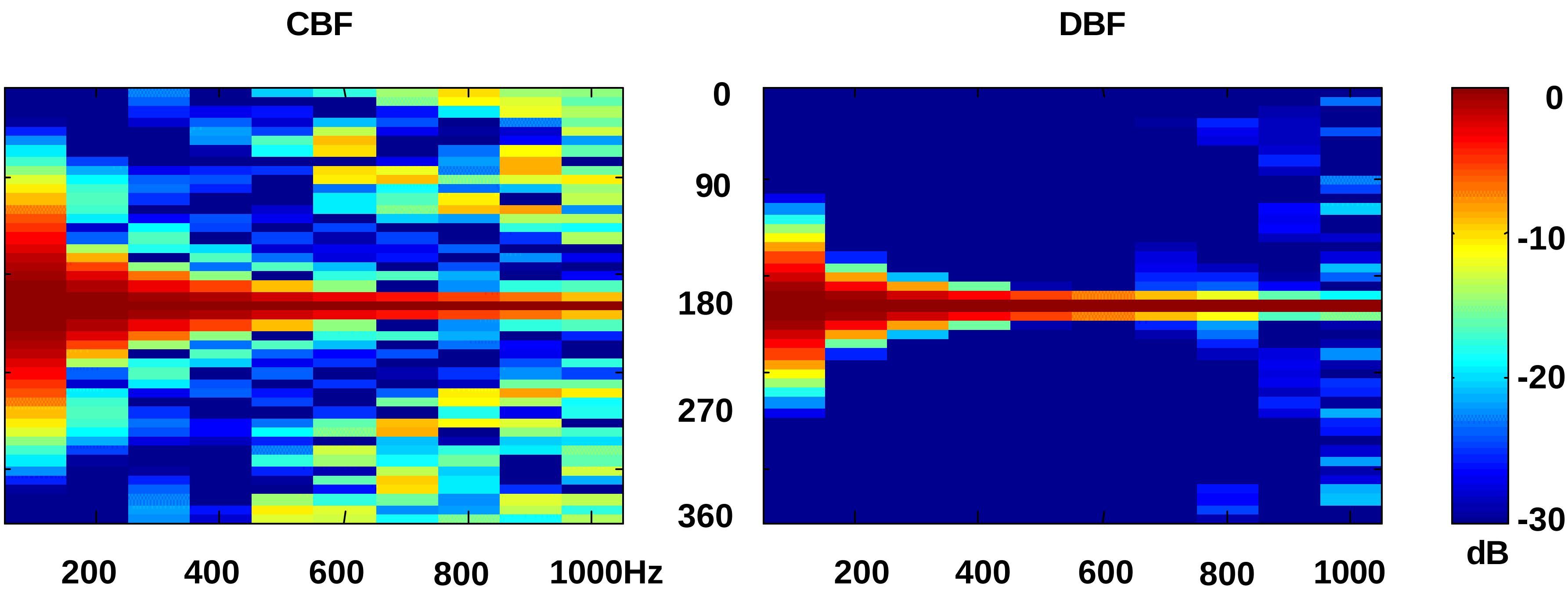


CBF = conventional beamformer

DBF = deconvolutional beamformer

CBF & DBF vs. Frequency

Horizontal Planar Array, Filled



CBF = conventional beamformer

DBF = deconvolutional beamformer

AETC

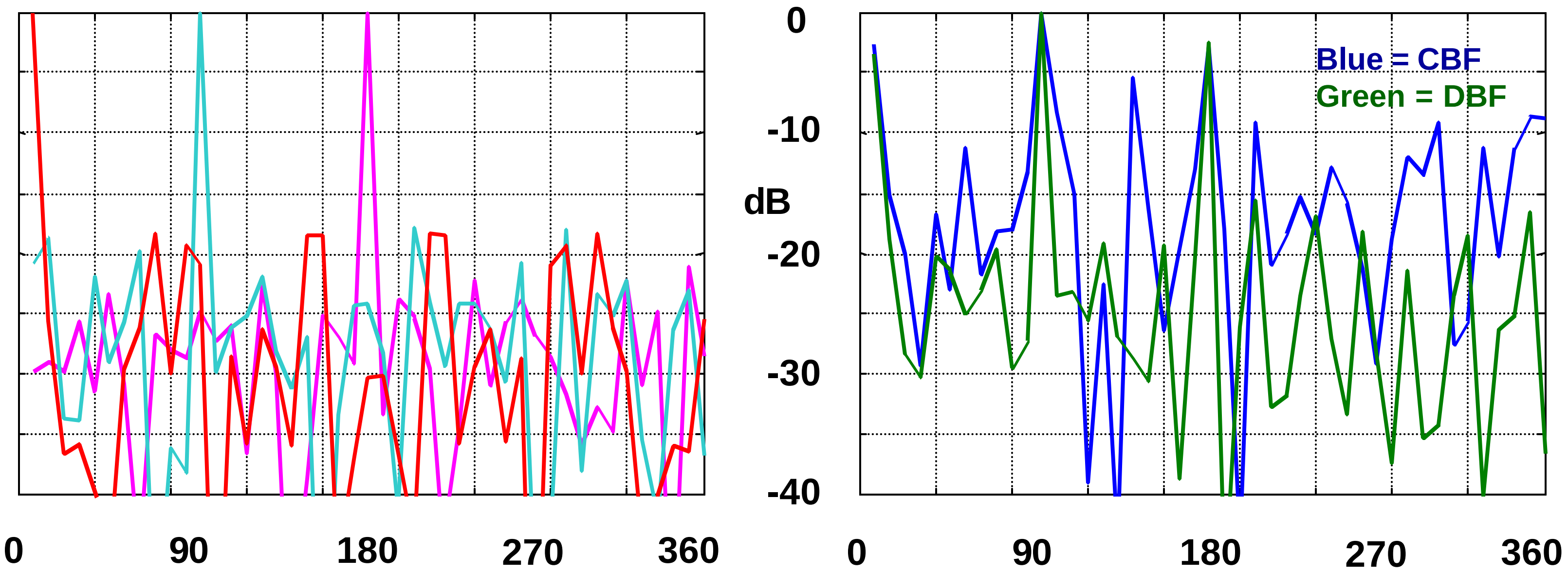
DBF Linearity Response

LMI

Horizontal Planar Array, Annular

Individual arrivals

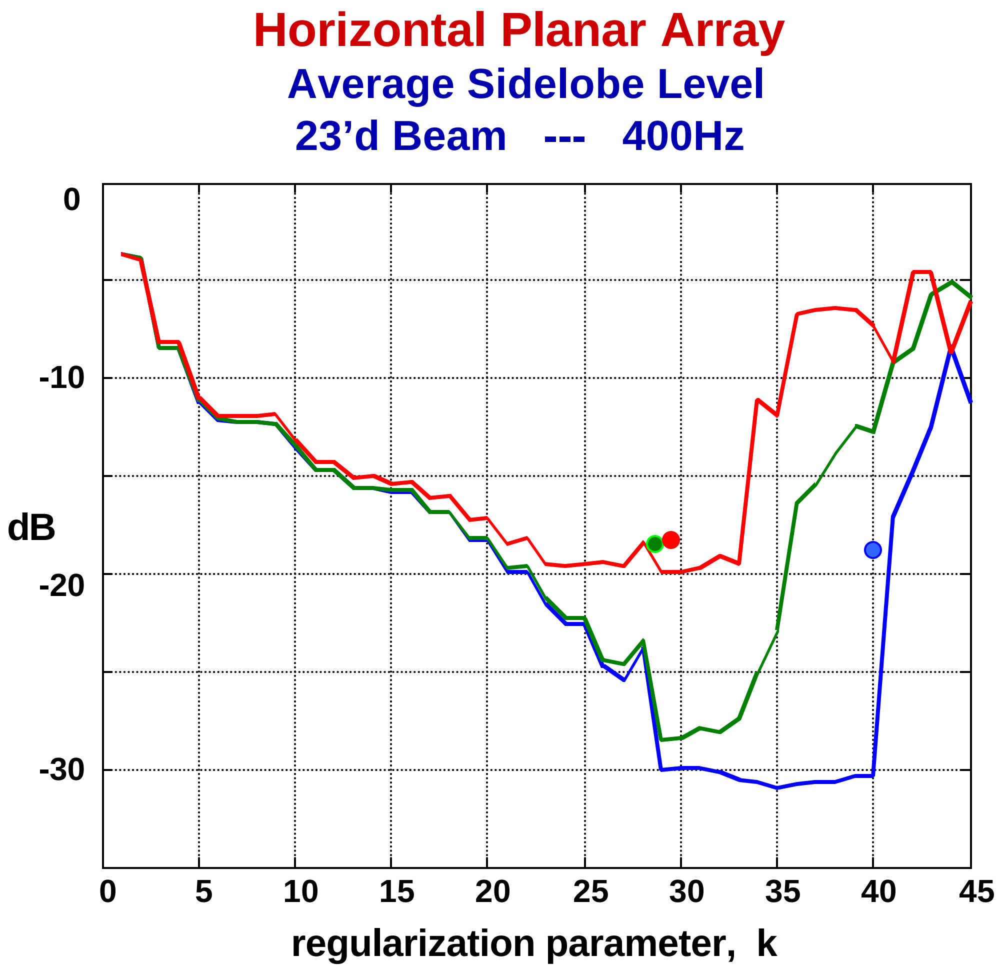
Simultaneous arrivals



CBF = conventional beamformer

DBF = deconvolutional beamformer

Sensitivity to Position Error



Error in lambda
CURVES: DBF
BLUE:
error = 0
GREEN:
error = .02
RED:
error = .10
DOTS: CBF

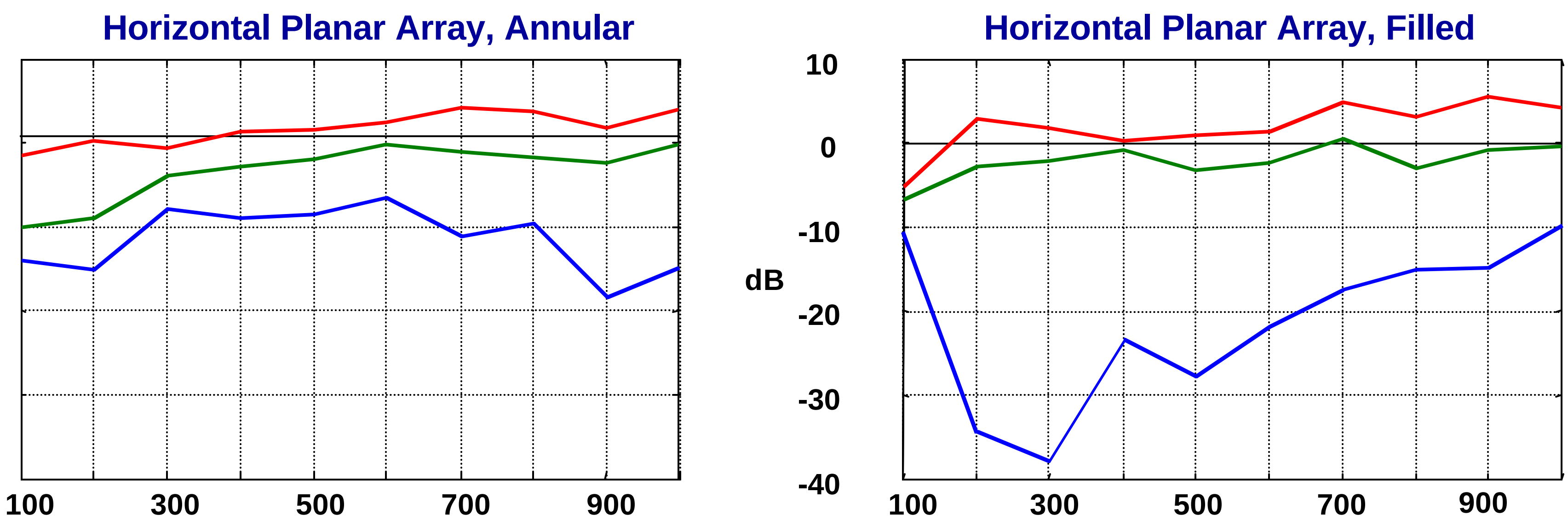
AETC

Sensitivity to Position Error

LMI

Average Sidelobe Level Difference

DBF relative to CBF, at 10 frequencies across band



BLUE: error = 0
GREEN: error = .02 λ
RED: error = .10 λ

AETC

Summary

L M I

Have shown how to construct weight matrices

- for planar, non-uniformly-spaced arrays
- one matrix per frequency to suppress sidelobes
- using regularized, constrained deconvolution
- applied in beam space
- output beamwidth retained or improved
- precomputed, stored for use (not “adaptive”)
- complementary to (compatible with) ABF
- achieve linear response for simultaneous arrivals
- can optimize vs. expected position error

ABF = adaptive beamformer

AETC

References

L M I

J. B. Abbiss, J. Allen, R. Bocker and H. J. Whitehouse, "Fast regularised deconvolution in optics and radar," in J. G. McWhirter (Editor), *Mathematics in Signal Processing III,* Clarendon Press, 1994.

P. C. Hansen, "Truncated singular value decomposition solutions to discrete ill-posed problems with ill-determined numerical rank," *SIAM J. Sci Stat. Computing*., Vol 11, No. 3, 1990, pp. 503-518.

P. C. Hansen, “Regularization Tools”, version 3.0, *Numerical Algorithms 6*, 1994, pp.1-35 .

J. M. Speiser, H. J. Whitehouse and J. C. Allen, "Fast matrix-vector multiplication using displacement rank approximation via an SVD," in R. J. Vaccaro (Editor), *SVD and Signal Processsing, II Algorithms, Analysis and Applications*, Elsevier, 1991.

J. Adams, “A New Optimal Window,” *IEEE Transactions on Signal Processing*, Vol.39, No.8, August 1991, pp.1763-1769..

Acknowledgements

This work was sponsored by the Naval Air Systems Command, PMA-264, Capt. Randy Loyer, and managed by SPAWAR Systems Center/San Diego, Dr. David Rees. The authors acknowledge the collaboration of Shelby Sullivan, Sr., Science Applications International Corporation.

B A C K U P

AETC

Background

LMI

- Program: Nonlinear Dynamical Stochastic Resonance
- Task: Non-Linear Aperture Processing
- Sponsor: NAVAIR, PMA-264, Capt. Randy Loyer
- Manager: SSC/SD, Dr. David Rees, D857
- Task Team Members:
  - James Alsup AETC
  - Harper Whitehouse LMI
  - Shelby Sullivan Sr. SAIC

AETC

James M. Alsup
AETC Incorporated
8910 University Center Lane, Suite 900
San Diego, CA 92122
(858) 450-1211 ext 228
alsup@sd.aetc.com
www.aetc.com

LMI

Harper J. Whitehouse
Linear Measurements, Incorporated
4174 Sorrento Valley Boulevard
San Diego, CA 92121
(858) 535-2172
whitehouse@linearmeasurements.com
www.linearmeasurements.com

Algorithm - 1

$$g = A * f$$

f = "ideal" vector of target azimuths
- spanning 0-360 degrees
- e.g., 0:8:352, 45x1

g = observed beampattern vector
- spanning 0-360 degrees, e.g., 45x1
- includes mainlobe & sidelobes
  - superposed for each "target"

A = point-spread function
- delta-function in, beampattern out
- e.g., 45x45

Algorithm - 2

$$g = S' * x$$

g = observed beampattern vector
- spanning 0-360 degrees, e.g., 45x1
- includes mainlobe & sidelobes

x = observed element data vector
- at single frequency
- complex; e.g., 40x1

S = steer matrix
- one column per look direction
- phase-only; e.g., 40x45

Algorithm - 3

$$A = S' * S$$

S = steer matrix
- one column per look direction
- phase-only; e.g., 40x45

A = point-spread matrix
- one CBF beampattern per row
- single frequency; e.g., 45x45

Algorithm - 4

Combine equations:

$$A * f = S' * x$$

Solve for f :

$$f = (A_k^+) * S' * x$$

A_k^+ = **regularized inverse of** A

AETC

Algorithm - 5

LMI

$$h = B * f$$

f = "ideal" vector of target azimuths

h = desired beampattern vector response

- mainlobe not too narrow
- sidelobes reduced from g

B = smoothing constraint matrix

- delta-function in, improved beampattern out
- e.g., 45x45

Algorithm - 6

Combine equations:

$$h = B * (A_k^+) * S' * x$$

Rewrite :

$$h = W * S' * x$$

where

$$W = B * (A_k^+)$$

W represents stored weight matrix

- **beamspace matrix**
- **single frequency, NxN**

Algorithm - 7

Reformulate for signal phase property :

$$\mathbf{h} = |\mathbf{h}| \; .* \; \exp(j * \text{angle}(\mathbf{g}))$$

Then :

$$|\mathbf{h}| = | \mathbf{W} * \mathbf{S'} * \mathbf{x} |$$

where

$$\mathbf{W} = \mathbf{B} * (\mathbf{A_k^+})$$

(don't want to mess with original phase of g, anyway)