Optimization and Beamforming of a Two Dimensional Sparse Array

Mandar A. Chitre
Acoustic Research Laboratory
National University of Singapore
10 Kent Ridge Crescent, Singapore 119260
email: mandar@arl.nus.edu.sg

John R. Potter
Acoustic Research Laboratory
National University of Singapore
10 Kent Ridge Crescent, Singapore 119260
email: johnp@arl.nus.edu.sg

Abstract

A large number of elements are required at half-wavelength spacing in two dimensional arrays to achieve good angular resolution. This complicates the construction of the array and requires immense computational power for beamforming. A sparse array reduces the number of elements at the cost of introducing grating lobes in the beampattern. Aperiodic sparse arrays can reduce the sensitivity of grating lobes. However, the optimal methods for the design of two dimensional aperiodic sparse arrays are computationally too intensive to be used for synthesis of large arrays. This paper investigates the application of Simulated Annealing and Genetic Algorithms to the aperiodic array optimization problem. It also demonstrates that frequency-domain multistage beamforming can further reduce the beamforming computational requirements by allowing frequency dependent beam resolution and field of view.

1. Introduction

A two dimensional aperture has application in many systems in the radio, microwave, visible and x-ray bands of the electromagnetic spectrum as well as the 10 Hz (underwater acoustics) to 10 MHz (medical imaging) bands in the acoustic spectrum. Such an aperture has to be spatially sampled by a two dimensional array antenna (planar array). In certain applications, the antenna may be curved in the third dimension. To achieve a good angular resolution, the physical size of the array aperture has to be large. At the same time, to prevent aliasing effects of spatial undersampling, a half-wavelength spacing between the array elements is desirable. This results in an array with a very large number of elements. For an array with 1 deg angular resolution, approximately 10,000 sensor elements are required. The human eye has a resolution of 0.006 deg, which would require 3×10^8 sensor elements. The construction of such a fully-populated array is complicated, the cost is prohibitive and the processing power needed for beamforming is immense.

The number of elements in an array can be reduced by making the array sparse (more than half-wavelength spacing) at the cost of array performance. A sparse array cannot resolve spatial directions completely due to aliasing. Sensitive lobes (called grating lobes) appear in the beampattern in directions other than the intended beamforming direction. The negative effect on performance can be minimized by reducing the sensitivity of the grating lobes using aperiodic spacing between the array elements. Optimal methods for synthesis of such aperiodic sparse arrays have been designed [1,2]. However, these methods can not be used in the synthesis of large arrays (due to immense computational requirements) and are capable of generating symmetric arrays only. A stochastic method using simulated annealing overcomes some of these limitations [3].

2. The problem

The array design and beamforming problem and the suggested solution can best be illustrated by using an example. Suppose a broadband array is to be designed with the specifications as shown in table 1.

Table 1. System Specifications

Parameter	Value
Maximum Wavelength	75 mm
Minimum Wavelength	15 mm
Field of View	17 deg $\times$ 8.5 deg
Best Angular Resolution	0.7 deg $\times$ 0.7 deg

A fully-populated array with these specifications would need an aperture of 1.3 m and element spacing of 7.5 mm. A circular array with these parameters would require more than 23,600 elements. The wavelength range assumed in this example is the microwave range for electromagnetic radiation or the common acoustic range for underwater acoustics. The processing speeds needed for beamforming in the electromagnetic applications are very large, so we shall assume an underwater acoustic application where the wavelengths correspond to the 20 kHz to 100 kHz range. For such a underwater acoustic system, each element needs to be sampled at 250 kHz to avoid temporal aliasing. For 8 bit sampling, a data transfer rate of 47 Gbits/s is required. To form 24×12 (i.e. 288) beams in real-time using time-domain beamforming and a 4 tap interpolation filter, the computational load would be 14 Tflops. This huge data transfer rate and computational power is not achievable with current high-performance computing devices.

3. Sparse array

To reduce the number of elements in the array, either the size of the array will have to be reduced or the spacing between elements will have to be increased. Reducing the size of the array will reduce the aperture size and thus the angular resolution would be poorer than required. Increasing the spacing between elements causes spatial aliasing and thus introduces grating lobes in the beampattern of the array.

Instead of using omni-directional sensor elements, elements with directional beampattern can be used to reduce the sensitivity of grating lobes. In fact, if the elements used are such that they fill the entire surface of the array without any gaps, the grating lobes of the array beampattern fall exactly in the nulls of the element beampattern [4]. The beampattern of the system is the product of the array beampattern and the beampattern of each individual element. Thus the system beampattern will have no grating lobes as seen in figure 1.

If 50 mm square elements are used to compose an approximately circular array with a diameter of 1.3 m, around 530 elements will be needed. This will yield an array which is sparse in both dimensions by a commonly used ratio of 1:7. The data transfer rate will thus reduce to 1 Gbit/s and the computational load to form 288 beams would reduce to 305 Gflops.

The array beam can be electronically steered to look in a required direction by introducing appropriate time delays to the output of each element. However the beampattern of an individual element cannot be steered without physically moving the element as the beam is a result of the physical size of the element. The exact cancellation of the grating lobes occurs only during broadside beamforming. When the array looks in other directions in the field of view, the grating lobes move out of the nulls and become significant as seen in figure 2.

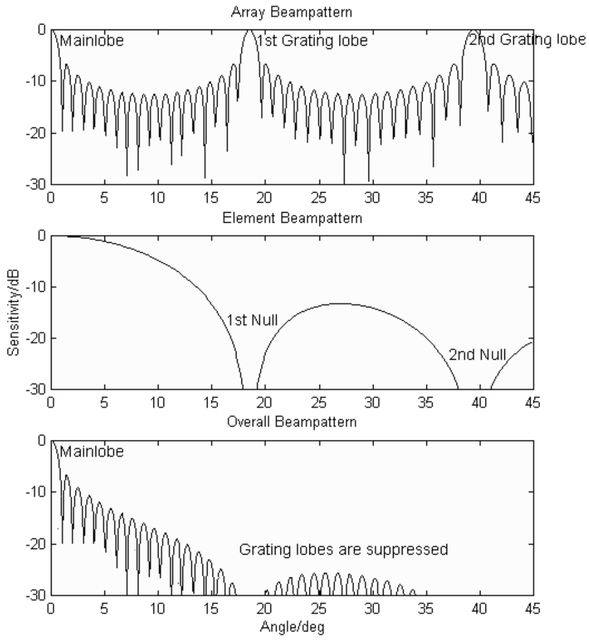

Figure 1. Cancellation of grating lobes

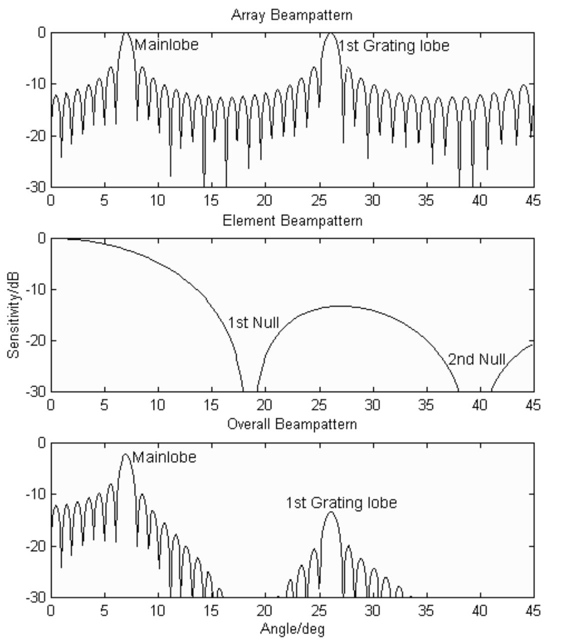

Figure 2. Grating lobes in non-broadside beams

4. Aperiodic sparse array design

When the array is sparse but the elements are periodically spaced, the spatial aliasing arising due to the inter-element spacing of each pair of elements occurs at the same angle. Thus the grating lobe at that angle is as sensitive as the main lobe. If the elements are aperiodically spaced, the aliasing occurs at different angles and thus the grating lobes are wider but less sensitive. This suggests that the non-broadside beamforming performance of the array may be improved by offsetting every element relative to others in the array.

4.1. Principle component analysis

Several arrays with different shapes and offsets of elements were designed. The performance of each array was simulated. Many parameters of the physical array design which were expected to affect the performance of the array were measured.

A principle component analysis was performed to find the important factors that affected the performance of the array. The broadside performance of the array seemed to be controlled by a single parameter, the normalized compactness. The normalized compactness is defined as the mean distance of the center of each element from the center of the array normalized by both, the square root of the number of elements in the array and the size of each element. This indicated that good broadside performance can be achieved by making the array as circular as possible and not leaving any gaps in the interior of the array.

For non-broadside performance, the second most important criterion is the number of different offsets of the element positions relative to other elements. A good array should have as many different offsets as possible. Unfortunately geometrical considerations constrain the options to either row or column offsets, but not both at one time without leaving gaps in the interior of the array. The compactness criterion also imposes a limit on the number of different offsets that can be introduced before the array performance degradation due to loss of compactness outweighs the gains due to offset diversity.

4.2. Simulated annealing

An automated optimization of the positions of the elements in the array is desirable. A full search of the parameter space is impractical, considering that there are 530 elements, each having 2 coordinates as parameters. A simulated annealing search was therefore implemented and tested to find a good array design.

The simulated annealing algorithm mimics the behavior of the molecules of a pure substance during the slow cooling that results in the formation of a perfect crystal [5]. The use of this technique to solve other types of problems is based on the analogy between the state of each molecule and the state of the set of parameters that affects the performance of the system to be optimized. An energy function is defined for the system, which reduces with improved performance of the system. The algorithm is iterative. At each iteration, a small random perturbation is induced in the state. If the new state causes the energy function to reduce, it is accepted. Otherwise the state is accepted with a probability dependent on the temperature of the system, in accordance with the Boltzmann distribution.

$$p(accept) = 1 \qquad \text{if } E_{new} < E_{old}$$

$$p(accept) = e^{\frac{E_{old} - E_{new}}{kT}} \qquad \text{otherwise}$$

Here k is the Boltzmann constant and T is the temperature of the system. The temperature T is gradually lowered following the reciprocal of the logarithm of the iteration number until the system freezes in a final state.

To optimize the array design, the x and y coordinates of each of the elements were used as the state variables. The ratio of the energy within the 6 dB main lobe to the energy from all other directions was used as the performance criteria. The negative performance criteria was used as the energy function. The state was perturbed at every iteration by moving each element by a random distance in the x and y directions. If this caused the element to overlap with another element, the old position was retained.

The evaluation of the energy function is computationally intensive. The evaluation of the array performance at each look direction requires 13×10^9 trigonometric operations and 26×10^9 floating point operations. The array has to be evaluated for at least 9 look directions for a reasonable estimate of array performance. Thus a total of 4×10^{11} operations are required for the calculation of the energy of a given state. Because the energy function has to be evaluated at every iteration, the number of iterations that could be practically performed was limited to a few thousand. The system converged very fast from the random initial stage. However, as the elements came closer together, most movements of each element would result in an overlap and would have to be discarded. Thus the system would converge very slowly once the elements were packed closely. To allow full optimization, a prohibitively large number of iterations would be required.

However, the partial results from the simulated annealing showed that the elements were indeed packed closely together and were offset relative to each other. The results also indicated that half-wavelength offsets were preferred. Based on this observation, new designs with half-wavelength offsets were tested and found to perform better than previous designs.

4.3. Genetic algorithms

Genetic algorithms are a promising technique for locating qausi-optimal solutions to non-linear optimization problems. The design of the sparse array is a non-linear optimization problem. The main problem in using genetic algorithms for the optimization of the array design is the encoding of the array design into genes.

Several encoding techniques of the position of the elements into genes were tested and found to converge very slowly. The fitness function i.e. the performance criteria of the array, is computationally expensive to evaluate. Every evaluation of the fitness function requires 4×10^{11} operations. The fitness function has to be evaluated for each member of the generation. Evaluating a large number of generations is thus impractical due to the computational time requirements. Finding an encoding which may converge faster still remains an unsolved problem. Although promising, genetic algorithms have not been useful for the optimizing of the array due to computational limitations.

5. Beamforming

The standard beamforming technique calculates the time delays for each element for the required look direction. The signals from each of the element are time delayed by the calculated amount using an interpolation filter and then summed. The resulting signal would have most of the energy coming from the chosen look direction. The exact distribution of the energy from different directions in the final signal is described by the beampattern of the array. This is the time-domain beamforming technique as shown in figure 3.

If N is the number of elements and M is the number of beams to be formed using a T tap filter at a rate of R samples per second, the total number of operations needed for time-domain beamforming is approximately 2NMTR.

In time-domain beamforming, all the beams are formed at all frequencies at once. The angular resolution at lower frequencies is lower and therefore fewer beams are needed to cover the field of view. Even so, time-domain beamforming has to form all the beams even at lower frequencies. Thus it would seem that implementing beamforming in frequency domain may allow different resolutions at different frequencies and thus reduce the computational load at the cost of increased software complexity.

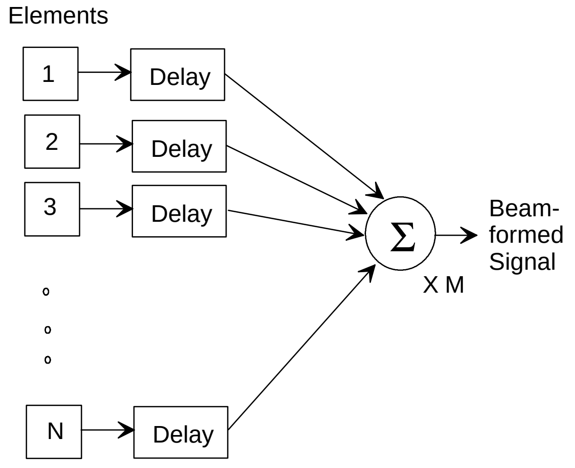

Figure 3. Time-domain beamforming

5.1. Frequency-domain beamforming

A time delay in the time domain is a phase shift in the frequency domain. Thus beamforming can be implemented by fourier transforming the signal from each element, phase-shifting the transformed signal based on the look directions, and adding the phase-shifted signal together in the frequency domain. It is then possible to form a different number of beams at different frequencies.

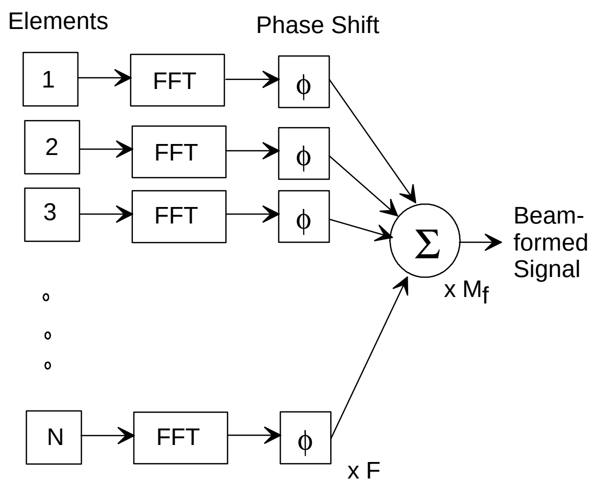

Figure 4. Frequency-domain beamforming

The frequency-domain beamforming technique is illustrated in figure 4. If a short-time fourier transform with F points is used to transform the signal, F/2 frequency bins will be available to different number of beams. If M_f

is the number of beams formed using frequency bin f, N is the number of elements and R is the number of samples per second, the total of $\frac{NR}{F}\sum M_f$ phase shifts and complex additions are required. In addition, a total of N F-point fast fourier transforms (FFT) are required. A complex phase shift can be implemented in 6 floating point operations and a complex addition requires 2 floating point operations. A F-point FFT can be implemented in $2F\log_2 F$ operations. Thus a total of $2NF\log_2 F + \frac{8NR}{F}\sum M_f$ operations are needed for the frequency-domain beamforming.

5.2. Multi-resolution beamforming

Different number of beams can be formed at different frequencies with frequency-domain beamforming. The exact number of beams required at each frequency can be determined based on the angular resolution at that frequency and the field of view. The field of view can also be made frequency-dependent if desired.

Table 2 shows the beams needed at the center frequencies of each frequency bin using a 16-point FFT for the example array under consideration.

Table 2. Frequency-dependent resolution

FOV = 17 deg x 8.5 deg

Frequency Bin	3 dB Beamwidth	No. of Beams
0 kHz	-	$0 \times 0 = 0$
18 kHz	3.9 deg	$4 \times 2 = 8$
36 kHz	2.0 deg	$9 \times 5 = 45$
54 kHz	1.3 deg	$13 \times 7 = 91$
71 kHz	1.0 deg	$17 \times 9 = 153$
89 kHz	0.8 deg	$21 \times 11 = 231$
107 kHz	0.7 deg	$24 \times 12 = 288$
125 kHz	0.6 deg	$0 \times 0 = 0$
Total number of beams		816

The computational load using frequency-domain multi-resolution beamforming reduces to 54 Gflops. This is 6 times less than the computational resources needed for the conventional time-domain beamforming.

5.3. Multistage beamforming

A computational load of 54 Gflops, even though smaller than 14 Tflops, is a fairly high requirement for current technology. In order to reduce this load further, the number of elements used for beamforming can be reduced to half the number, by combining sets of 2 elements into super-elements. The beamforming is then performed on the output of these super-elements.

When the signals from the 2 elements (placed edge to edge) in each super-element are added together, the super-element acts like a large element, double the size of the individual elements. This increases the sparseness of the array in one axis and thus degrades the non-broadside array performance in that axis. At low frequencies, the angular separation between the main lobe and the grating lobes is large and therefore the performance degradation is relatively small and therefore acceptable.

At high frequencies the performance degradation can be substantial. As the high frequency grating lobes occur at angles comparable with the field of view, non-broadside beams are severely affected. The main lobe of these beams moves out of the sensitive region of the beampattern of the super-elements and the grating lobes move out of the nulls in the beampattern. The grating lobe suppression reduces and the main lobe becomes less sensitive. In extreme cases, grating lobes become more sensitive than the main lobe.

The key to maintaining good performance is to keep the angle between the look direction of the super-elements and the look direction of the array small as compared to the angular separation of the main lobe and the grating lobes. This condition is satisfied at low frequencies if the signals from the two elements in the super-element are added together, thus forming a broadside beam for the super-element. At high frequencies, multiple beams spanning the field of view are required to keep the maximum angular separation between the look directions of the super-element and the array small. Appropriate pre-formed beams can then be selected and used for the main beamforming depending on the desired look direction for the array.

This not only distributes the beamforming load among multiple processors at each super-element, but also reduces the total beamforming load. The actual number of pre-formed beams can be varied with frequency. At low frequency, a single broadside pre-formed beam will yield good performance. At higher frequencies, 2 beams (say at ±5 deg elevation) are needed. At even higher frequencies 3 beams (say broadside and ±5 deg elevation) are needed. While forming the final beam, the appropriate pre-formed beam is used thus moving the grating lobe as close to the null in the super-element beampattern as possible.

If a single beam is used for the lowest 3 frequency bins, 2 beams are used for the next 2 frequency bins and 3 beams for the top frequency bin, a total of 10 beams need to be formed at each preprocessor. The computational load at each preprocessor would then be 2.5 Mflops and the computational load at the main beamforming processor would reduce to 26 Gflops. The total computational load therefore is approximately 27 Gflops.

6. Final array design

The results from the principle component analysis and the simulated annealing indicate that the array should be as compact and circular as possible and should have large number of element offsets in multiples of half-wavelength relative to other elements. The element offset requirement can only be achieved in one direction without creating gaps in the array. Because the desired field of view is wider in the azimuthal direction than in elevation, the non-broadside performance in the azimuthal direction is more important. Therefore the offsets should be introduced between rows rather than columns.

Multistage beamforming suggests that the computational load can be reduced further by combining every two elements into a single super-element. Because the elevation performance is less important (as the field of view is narrower), the signals from every two elements placed vertically should be combined. The array design should thus include approximately 265 super-elements, each made of two elements placed vertically.

The final array designed with all the above considerations is shown in figure 5. The array has 254 super-elements (508 elements). 89 % of the surface area of the circumscribing circle is occupied by elements. The radius of the circumscribing circle is 0.65 m and therefore the aperture of the array is around 1.3 m. The normalized compactness of the array is 0.51.

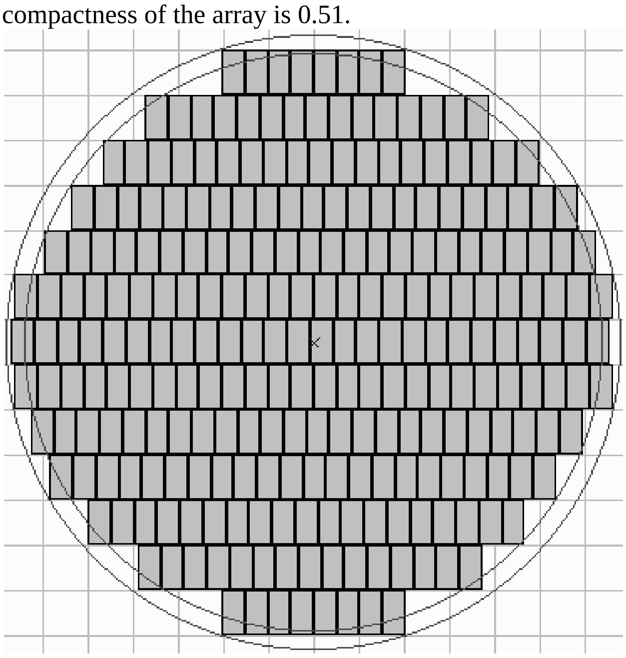

Figure 5. Final physical array design

7. Beamforming performance

The performance of the array shown in figure 5 was simulated for each look direction and frequency. The array gave an acceptable performance with 6 dB signal-to-noise ratio of better than 3 dB.

Some of the performance simulation results are presented here. Figure 6 shows the best (broadside) and worst (edge of field of view) performance of the array at low frequency (wavelength = 75 mm). Figure 7 shows the performance of the array at high frequency (wavelength = 15 mm).

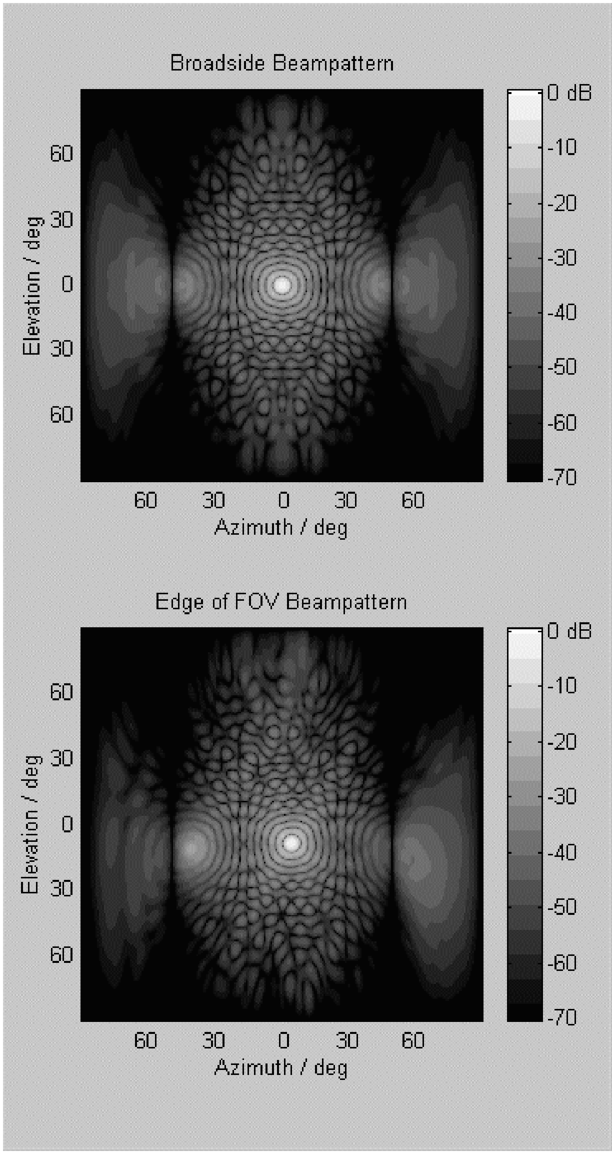

Figure 6. Low frequency array performance

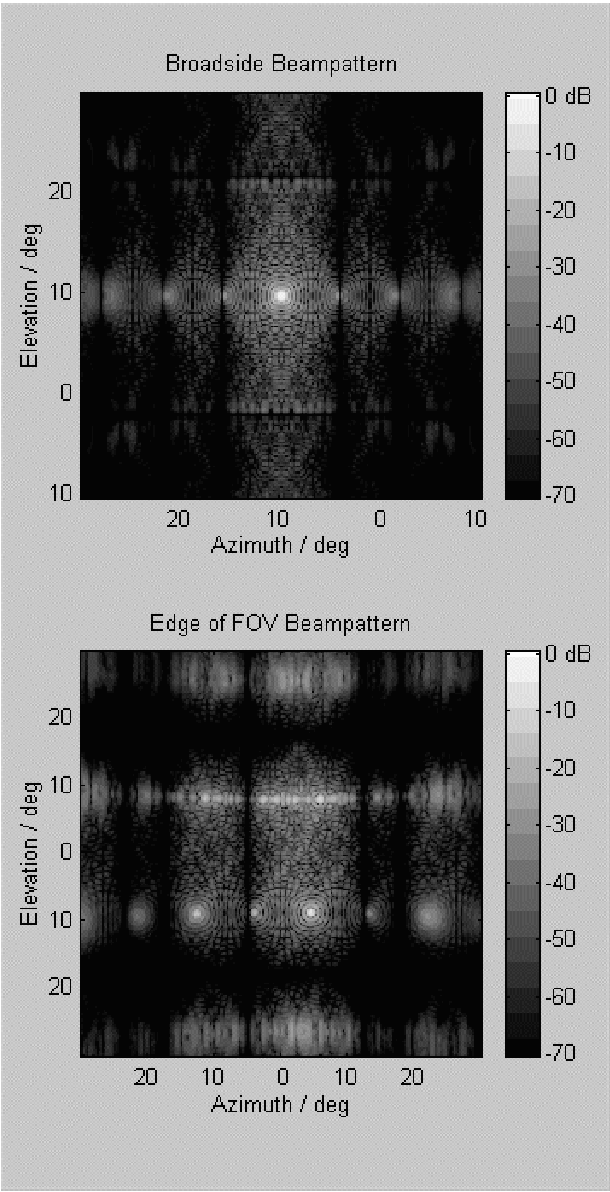

Figure 7. High frequency array performance

The suppression of grating lobes due to the beampattern of the super-elements can be clearly seen in the broadside beampattern at both high and low frequencies in figures 6 and 7. In the beampatterns at the edge of the field of view, the grating lobes can be seen clearly. The sensitivity of the grating lobes is much higher in the high frequency case in figure 7.

8. Summary

A fully-populated array with the example specifications is not feasible due to the huge beamforming computational requirements of 14 Tflops and data transfer requirements of 47 Gbits/s. By making the array sparse, grating lobe problems are introduced. These problems are solved by using directional sensor elements in the array and placing them aperiodically and compactly. With the reduction in number of elements, the data transfer requirements reduce to an achievable 1 Gbit/s. However, the computational requirements remain too high at 305 Gflops. By combining the elements into super-elements and using multi-resolution frequency-domain beamforming, this computational load can be brought down to an achievable 27 Gflops. Furthermore, some of the load can be distributed to preprocessors at each sensor element.

The optimization of the positions of each element in the array still remains a manual process. With simple state variable encoding, both simulated annealing and genetic algorithms require too long to converge to an acceptable solution with current processing capabilities. However, there is scope for research in developing new encoding techniques which may make simulated annealing or genetic algorithm optimization feasible. With the increasing computational capacity of new computers, these techniques hold promise for the future of array optimization.

9. Conclusions

Good angular resolution can be achieved only by large aperture arrays. Fully-populated large arrays are prohibitively expensive to build and are often impossible due to current technological limits. Sparse aperiodic arrays can be designed to replace the fully-populated arrays with little loss in performance, but a large gain in terms of complexity and computational requirements.

Frequency-domain beamforming allows beam resolution and field of view to vary with frequency. By optimizing the beams at different frequencies, the computational requirements for beamforming can be further reduced.

Multistage beamforming can further reduce the computational requirements and distribute the computational load. When used with frequency-domain beamforming, multistage beamforming can be frequency dependent and thus yield better computational savings.

References

[1] Leahy R.M. & Jeffs B.D., "On the Design of Maximally Sparse Beamforming Arrays", IEEE Tr. Anten. Prop., v39, p1178-1187, 1991.

[2] Hohn S., Elgetun B. & Dahl G., "Properties of the Beampattern of Weight- and Layout-Optimized Sparse Arrays", IEEE Tr. UFFC, v44, 1997.

[3] Trucco A., "Synthesis of Aperiodic Planar Arrays by a Stochastic Approach", IEEE Oceanic Eng. Soc. Newsletter, Winter 1998.

[4] Potter J.R., "A new Ambient Noise Imaging system for ANI, passive and bistatic active acoustic imaging in shallow water", High Frequency Acoustics in Shallow Water, 1997.

[5] Kirkpatrick S., Gellatt C.D.Jr. & Vecchi M.P., "Optimization by Simulated Annealing", Science, v220, no. 4598, p671-680, 1983.