Weight- and Layout-Optimized Sparse Arrays

Sverre Holm, Bjørnar Elgetun and Geir Dahl
Department of Informatics, University of Oslo
P. O. Box 1080, N-0316 Oslo, Norway
sverre.holm@ifi.uio.no

Abstract— **Theory for random arrays predicts a mean sidelobe level given by the inverse of the number of elements. In this paper[1] two optimization methods for thinned arrays are given: one is for optimizing the weights of each element, and the other one optimizes both the layout and the weights. The weight optimization algorithm is based on linear programming and minimizes the peak sidelobe level for a given beamwidth. It is used to investigate the conditions for finding thinned arrays with peak sidelobe level at or below the inverse of the number of elements. With optimization of the weights of a randomly thinned array it is possible to come quite close and even below this value, especially for 1D arrays. Even for 2D sparse arrays a large reduction in peak sidelobe level is achieved. Even better solutions are found when the thinning pattern is optimized also. This requires an algorithm that uses mixed integer linear programming. In this case solutions with lower peak sidelobe level than the inverse number of elements can be found both in the 1D and the 2D cases.**

I. Introduction

The topic of this paper is the study of the beam pattern of 1D and 2D arrays. Typical applications are medical ultrasound imaging [1] and high-resolution sonar imaging [2]. 2D arrays in these fields represent a technological challenge not the least because of the high channel count [3], [4]. For this reason sparse 2D arrays, where elements are removed by thinning, are considered to be necessary.

The starting point of our study are arrays that are regularly sampled with sample distance equal to half the wavelength. This is the spatial equivalent of the Nyquist criterion. The arrays may be 1-dimensional or 2-dimensional. In both cases it is assumed that the sampling is regular, ie. along a line or on a square grid (as opposed to [5] where arbitrary positions are allowed). A sparse array has elements removed by thinning. This removal may be random or it may be found from some sort of optimal algorithm. The trivial thinning of just keeping the full central part of the aperture is avoided. In this way the aperture is maintained and thus the resolution. The remaining elements may be weighted or they may be unweighted.

Steinberg [6] has given a comprehensive theory for the unweighted randomly thinned array. The main results for the far-field continuous wave (CW) beampattern are:

- The probability distribution of the elements' position determines the main lobe's shape and the nearby sidelobes in exactly the same way as they are determined by the weighting in a full array.
- The sidelobe level can be described in a statistical sense and away from the main lobe, the ratio of the mean sidelobe power to the main lobe peak power is $1/K$ where K is the number of remaining elements. This result is independent of the statistical distribution of the elements.
- The sidelobe amplitude away from the main lobe is Rayleigh distributed and unaffected by the element distribution. The peak sidelobe may be as high as 10 dB above the average sidelobe level.

[1]This work was partly sponsored by the ESPRIT program of the European Union under contract number EP 22 982.

A. Sparse array optimization

There is a long history in the radar litterature for analysis of beampatterns for sparse arrays for the far-field single-frequency case (analysis of the one-way beampattern). In ultrasound imaging, this was the approach used in [7] where it was partially confirmed that Steinberg's results for average one-way sidelobe levels can be squared to estimate the levels for the two-way beampattern for pulsed 2-D arrays. In [8] an optimization of element placement was reported. The optimization criterion was to find the best approximation to the full array's two-way CW beampattern. The solution was an optimal thinning pattern with random-like appearance. This work is an attempt to find the properties of the random-like thinning patterns. It is based on optimization of the one-way response by either changing the element weight or the element positions or both.

There have been previous attempts to weight sparse arrays. This has been reported to have no effect, but this was because regular weighting functions were sampled [7]. In [9] we showed that it is possible to find element weight functions for a given thinning pattern that give the beampattern optimal properties. An important point is that these functions have little or no resemblance with the corresponding full array's weight function. In [10] this approach was extended from 1D to 2D arrays and improved results were reported. By using the linear programming algorithm for optimization it was possible to optimize the whole sidelobe region. Due to the properties of 2D array elements in ultrasound (high impedance, low sensitivity) it is often undesirable to weight the elements of a 2D transducer array. The goal of this work is therefore not primarily to propose practical weighting functions, but rather the optimization methods are used to find properties of the beampattern of such arrays. Of special interest is to determine the minimum peak sidelobe level and compare it with the predictions from random theory. Finally a method is also described for optimizing the element positions of a random-like sparse array. This optimization gives results that are more directly useful in an array design. Other related work on joint optimization of thinning pattern and weights has been reported in the context of sonar arrays in [11] and

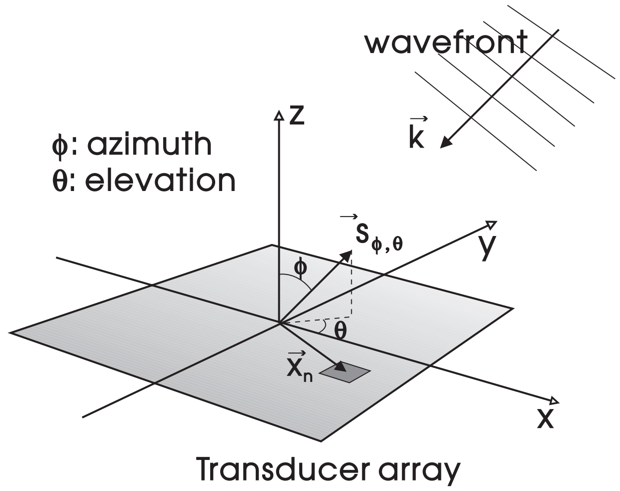

Fig. 1. A 2D planar array with coordinate system.

[12].

The optimization criterion is usually a minimization of the maximum sidelobe. This is a criterion which is related to imaging of a strong reflecting point target in a non-reflecting background containing other point targets. An alternative criterion is to minimize the integrated sidelobe energy. In a medical imaging system, this is related to imaging of a non-reflecting area like a cyst or a ventricle in a background of reflecting tissue. Some results on weight optimization for 1D arrays using this criterion and quadratic optimization have been reported in [13]. In this paper we find the properties of arrays based on minimization of the peak sidelobe, because this has been the most common criterion till now and it is straightforward to formulate optimization algorithms for it.

B. The beam pattern of a planar array

The far-field continuous wave (CW) beampattern of an array with an even number, $L = 2N$, of omnidirectional elements is given as [14]:

$$W(\vec{k}) = \sum_{n=1}^{2N} w_n e^{j\vec{k}\cdot\vec{x}_n} \quad (1)$$

where the array element locations are $\vec{x}_n \in \mathbb{R}^3$ with the corresponding weights $w_n \in \mathbb{R}$. The wavenumber vector $\vec{k} \in \mathbb{R}^3$ has amplitude $|\vec{k}| = 2\pi/\lambda$ where λ is the wavelength.

Let the unit direction vector be $\vec{s}_{\phi,\theta} = (\sin\phi\cos\theta, \sin\phi\sin\theta, \cos\phi)$ in rectangular coordinates, see Fig. 1. Then the wavenumber vector is $\vec{k} = 2\pi\vec{s}_{\phi,\theta}/\lambda$.

The elements of a 2D planar array are located in the xy-plane with element n at $\vec{x}_n = (x_n, y_n, 0)$. Thus the beampattern is:

$$W(k_x, k_y) = \sum_{n=1}^{2N} w_n e^{j(k_x\cdot x_n + k_y\cdot y_n)} \quad (2)$$

The beampattern has the following properties:

- For real weights, the beampattern is conjugate symmetric, i.e. $W(k_x, k_y) = W^*(-k_x, -k_y)$.
- Symmetric arrays with symmetric weights give a real beampattern.

When the two properties are combined, the beampattern for an array with an even number of elements is real and equal to:

$$W(\phi,\theta) = 2\sum_{n=1}^{N} w_n \cos\left(\frac{2\pi}{\lambda}\sin\phi(x_n\cos\theta + y_n\sin\theta)\right) \quad (3)$$

which gives the array response to a monochromatic wave from direction (ϕ, θ). A similar expression for an odd number of elements can easily be found.

Using matrix notation one gets

$$W(\phi,\theta) = \mathbf{v}(\phi,\theta)\mathbf{w} \quad (4)$$

where $\mathbf{w} = [w_1 \ \cdots \ w_N]^T$ are the element weights and the kernel row vector is given as:

$$\mathbf{v}(\phi,\theta) = \left[2\cos(\frac{2\pi}{\lambda}\vec{s}_{\phi,\theta}\cdot\vec{x}_1)\ , \cdots,\ 2\cos(\frac{2\pi}{\lambda}\vec{s}_{\phi,\theta}\cdot\vec{x}_N)\right]. \quad (5)$$

II. Optimization of beampattern

Two optimization problems will be formulated as linear programming problems. The first is a minimization of the maximum sidelobe level by varying element weights. The second problem gives rise to a mixed integer linear programming problem which is considerably harder to solve. It is a minimization of the number of active elements and an optimization of the weights in order to achieve a specific maximum sidelobe level.

A. Optimization of element weights

The objective is to minimize the sidelobe level in a continuous region $\mathcal{R}$ of the $\phi\theta$-plane, which is the equivalent of a filter's stopband. Since the beampattern is symmetric about the θ-axis; $W(-\phi,\theta) = W(\phi,\theta)$, only the right half-plane is necessary for the optimization. The passband in this case is minimal in the sense that it consists only of the θ-axis.

The element weight optimization problem is to minimize the beampattern in the stopband by varying the element weight vector subject to the constraint of having a normalized mainlobe. This problem may be formulated and solved as a linear programming problem as discussed next.

A linear programming (LP) problem is the minimization of a linear objective function subject to a (finite) set of linear inequalities and linear equations [15]. In matrix form an LP problem may be written as

$$\begin{array}{ll} \text{minimize} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} \le \mathbf{b} \end{array} \quad (6)$$

where $\mathbf{x}$ is a vector of n variables, and the data is given by the $m \times n$-matrix A and the vectors $\mathbf{c}$ and $\mathbf{b}$. Today large-scale LP problems can be solved efficiently on standard computers with good algorithms and implementations.

The stopband region $\mathcal{R}$ is discretized into a set of M gridpoints $R = \{(\phi_m, \theta_m) : m = 1, \ldots, M\}$. Introduce the $M \times N$ matrix V with the mth row being the N-dimensional row vector $\mathbf{v}(\phi_m, \theta_m)$ given in (5), or $\mathbf{v}_m$ for short. For a given element weight vector $\mathbf{w}$ the maximum sidelobe level δ_{sl} on the discrete set R is defined as

$$\begin{array}{l} \delta_{sl}(\mathbf{w}) = \max\{|\mathbf{v}_m \mathbf{w}| : 1 \leq m \leq M\} = \\ \max\{|\sum_{n=1}^{N} v_{m,n} w_n| : 1 \leq m \leq M\}. \end{array} \quad (7)$$

Thus the element weight optimization problem is to minimize δ subject to $|\mathbf{v}_m \mathbf{w}| \leq \delta$ for $1 \leq m \leq M$, and a normalization corresponding to a unit response for zero azimuth angle, $\mathbf{v}_0 \mathbf{w} = 1$, where $\mathbf{v}_0 = [1, \ldots, 1]$. This is a nonlinear optimization problem, but a standard reformulation [15], may be used to turn it into an LP problem. Consider the LP problem

$$\begin{array}{lll} \text{minimize} & \delta & \\ \text{subject to} & & \\ \text{(i)} & \mathbf{v}_0 \mathbf{w} = 1; & \\ \text{(ii)} & \mathbf{v}_m \mathbf{w} \leq \delta & \text{for } m \leq M; \\ \text{(iii)} & -\mathbf{v}_m \mathbf{w} \leq \delta & \text{for } m \leq M. \end{array} \quad (8)$$

The variables are $\mathbf{w}$ and δ. This problem is of the form (6) with

$$\mathbf{x} = \begin{bmatrix} \mathbf{w} & \delta \end{bmatrix}^T, \qquad \mathbf{c} = \begin{bmatrix} \mathbf{0} & 1 \end{bmatrix}^T,$$

$$A = \begin{bmatrix} \mathbf{v}_0 & 0 \\ -\mathbf{v}_0 & 0 \\ V & -1 \\ -V & -1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ -1 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

where $\mathbf{0}$ is a vector of all ones.

In an optimal solution of (8) the variable δ equals the minimum value of $\delta_{sl}(\mathbf{w})$ in (7). This problem is therefore a min-max problem. Thus the element weight optimization problem may be solved as the LP problem (8).

A very fast and reliable LP solver, CPLEX [16], was used. This is an optimization library for solving linear programming and integer linear programming problems. The problem (8) may be solved for problems corresponding to 2D arrays with thousands of elements where several hundreds of them are active [18]. Thus, the main work of the implementation is to generate the correct entries in the coefficient matrix and vectors. The simplex algorithm (a part of CPLEX) was used for solving the problem.

B. Optimization of element layout and weights

The simultaneous weighting and thinning problem is a natural extension of the element weighting problem in the previous section. Since the objective is to minimize the number of array elements, a binary variable $x_n \in \{0, 1\}$ is introduced for each element. The purpose is to let $x_n = 1$ indicate that the element is present, and $x_n = 0$ indicates that the element is removed by thinning.

Note that the objective is now to minimize the number of array elements, rather than minimizing the sidelobe level δ. The sidelobe level is consequently a fixed parameter $\bar{\delta}$ in this problem as in [11]. Thus one considers the element weighting and thinning optimization problem: minimize the number of array elements subject to constraints assuring a normalized mainlobe and fixed sidelobe level. Consider the following problem

$$\begin{array}{lll} \text{minimize} & \sum_n x_n & \\ \text{subject to} & & \\ \text{(i)} & \mathbf{v}_0 \mathbf{w} = 1; & \\ \text{(ii)} & \begin{bmatrix} V \\ -V \end{bmatrix} \mathbf{w} \leq \bar{\delta}\mathbf{1}; & \\ \text{(iii)} & \gamma_1 x_n \leq w_n \leq \gamma_2 x_n & \text{for } n \leq N; \\ \text{(iv)} & x_n \in \{0, 1\} & \text{for } n \leq N. \end{array} \quad (9)$$

Here the constraints (i) and (ii) are as before except that $\bar{\delta}$ is given, while (iii) gives a logical link between the layout variable x_n and the weight variable w_n. In order to vary the weight w_n between the two bounds γ_1 and γ_2 one has to set x_n to 1. The actual values of the parameters γ_1 and γ_2 may be set depending on the specific problem studied. For instance, interesting choices are $\gamma_1 = 0$ (nonnegative weights), $-\gamma_1 = \gamma_2 > 0$ (symmetric bounds), or positive weights with γ_2/γ_1 restricted to the maximum allowed dynamic range of the weighting hardware.

The problem (9) is a mixed integer linear programming problem, i.e., a linear programming problem where some or all variables are required to be integral. This particular problem may be written in matrix form similarly to what was shown in the previous section. In general, mixed integer LP problems are computationally very difficult optimization problems. Even this particular problem is difficult, i.e., to find an optimal solution seems hard also for moderately sized problems. This is mainly due to the complex structure of the matrix V which comes in combination with the integrality constraints on the layout variables x_n. In practise it turns out that it is only realistic to solve problems of size corresponding to 1D arrays so far. At present only simplified heuristic methods may be used to solve for the larger problems ([8], [12]). One important use of the mixed integer linear programming algorithm is that it may be used to compare the quality of different simplified heuristic methods for the same problem.

Small-scale problems may be solved by the branch and bound method in CPLEX [16]. This is a general method for solving mixed integer linear programming problems in which the feasible region is gradually divided into finer subregions for which a linear programming problem is solved. To (hopefully) control the combinatorial explosion of these subdivisions, one cuts off in subdivisions that can not lead to a further improvement of the current best solution.

For larger problems CPLEX will run “forever”, but even early in this process it may find good solutions satisfying (9), that may be of interest. The problem, however, is to prove that these solutions are the optimal ones.

III. Examples

A. 1D sparse array

A 3.5 MHz array with half wavelength spacing between elements, 64 elements and gaussian thinning to 48 elements

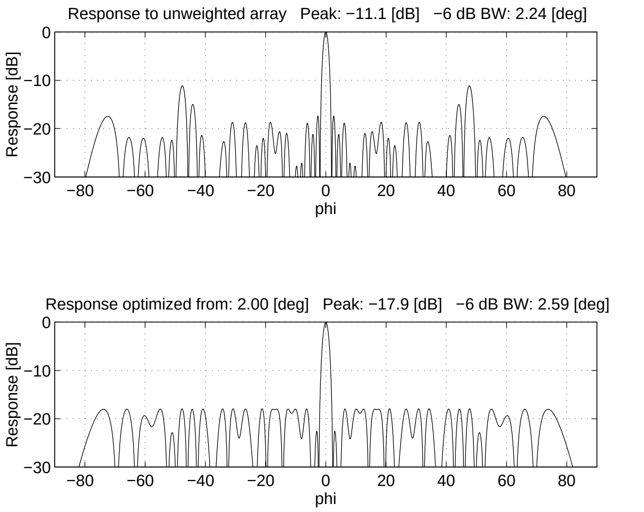

Fig. 2. Beampattern before and after optimization for 64 element array randomly thinned to 48 elements. Thinning and weights are shown in the center panel of Fig. 4.

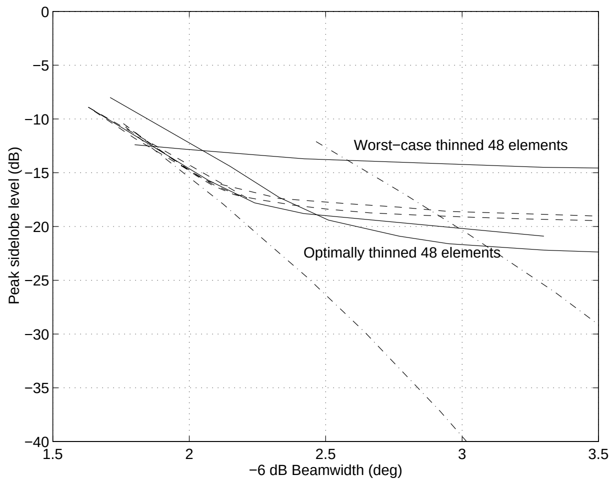

Fig. 3. Sidelobe level as a function of beamwidth for uniform sidelobe level 64 element and 48 element full arrays (dash-dot lines), for two realizations of random 25% thinning of the 64 element array (dashed lines), and for worst-case and optimally 25% thinned arrays (solid lines).

was optimized. An example of the beam patterns before and after optimization are shown in Fig. 2.

Several such optimizations were performed for various beamwidths and thinning patterns. For each thinning pattern, the start angle ϕ_1 (the boundary between the mainlobe and sidelobe regions) was varied and an optimization was performed. The resulting peak sidelobe and −6 dB beamwidth is plotted in Fig. 3. Each curve is the result of between 5 and 18 such optimizations. The figure shows two dash-dot lines which are the results of optimizing the weights to give uniform sidelobe levels for the full arrays. The left-hand one is the performance for a full 64 element array, and the right-hand one for a full 48 element array. Only thinned array with performance better than the 48-element curve are of interest. All the remaining curves are for a 64 element array thinned to 48 elements. The upper solid line shows performance for the worst symmetric thinning that could be found, giving a minimum sidelobe level of about −13 dB.

The two dashed lines are two realizations of random gaussian thinning. Both of them start leveling off at −17 to −18 dB sidelobe level. This is in the vicinity of the mean sidelobe level predicted for a random array given as the inverse of the number of elements which is −16.8 dB. However with the optimization used here this value is achieved as a peak value instead.

Finally the two lower solid curves are the results from optimizing the weights for two near-optimal thinning patterns. They were obtained with the combined weight and layout optimization algorithm with sidelobe targets of −18 and −19.5 dB. The other values in their curves were obtained by keeping the layout and then optimizing the weights only for different values of start-angles in the optimization. With such thinnings the peak sidelobe level can be improved down to the range −17 to −20 dB.

All the thinning patterns are shown in Table 1. Examples

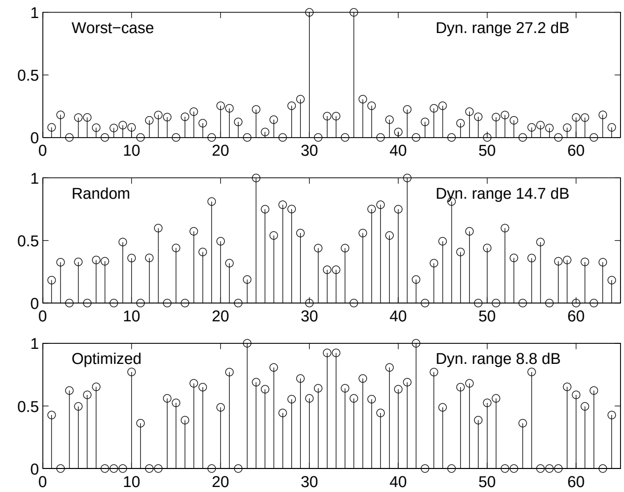

Fig. 4. Weights found after optimization from 2 degrees for three different element layouts. The beampattern of the random layout is shown in the lower panel of Fig. 2.

of the weights required are shown in Fig. 4. They are quite different from the much smoother weight functions that are obtained for full arrays (see the Dolph-Chebyshev weights of Figs. 46-49 of [17]).

B. 2D sparse arrays

A 2D array for 3.5 MHz with 12 by 12 elements with half wavelength spacing in both dimensions was then considered. The array is inscribed in a circle giving 112 elements. Random thinning to 64 elements (57%) and optimization of the weighting gives a beampattern with a sidelobe level of −12 to −15 dB. The procedure for finding the optimal thinning and weighting was then used with a sidelobe target of −19.5 dB. The optimized layout was then input with

Elements enabled	Comment
11011101110111011101110111011101	Worst-case symmetric array
11010110110110101111101111111011	Random 1 (upper dashed curve)
11011011011111111001010111110111	Random 2 (lower dashed curve)
10111100011001111101101111111111	Optimized 1, (-18 dB) (upper solid curve)
00101001111101111011101111111111	Optimized 2, (-19.5 dB) (lower solid curve)

TABLE I

Left-hand part (32 elements) of symmetric 64 element arrays. All references to relative position are to the right-hand part of curves in Fig. 3.

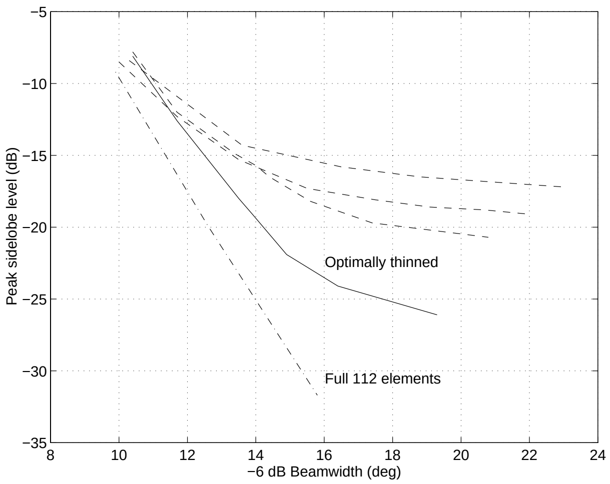

Fig. 5. Sidelobe level as a function of beamwidth for several weight-optimized uniform sidelobe cases: 112 element full array (dash-dot line) and three realizations of random thinning to 64 elements (dashed lines). The best result is obtained for a layout-optimized 62 element thinning (solid line).

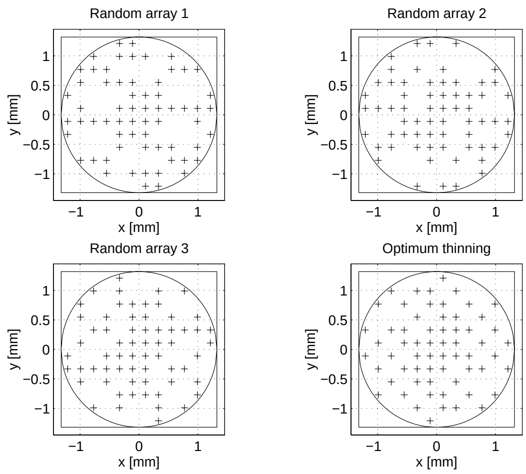

Fig. 6. Element layouts for 112 element full array thinned to three different random 64 element layouts and a 62 element optimized layout. The random arrays are sorted according to the peak sidelobe level in Fig. 5 with Random 1 having the highest peak sidelobe level for large beamwidths.

varying start-angles in the weight optimization algorithm. The peak sidelobe level can now be reduced down to −20 to −22 dB (Fig. 5). Each curve is the result of between 5 and 8 optimizations with different start values for the azimuth angles. The sidelobe value should be compared to the value predicted for mean sidelobe level of $1/64 = -18.1$ dB, and shows that there is a potential of getting a peak value which is 3 dB lower than that predicted for the mean if optimized thinning patterns can be found. This is about the largest array size where optimized element layouts can be found with reasonable use of computer resources (less than about 4 hours CPU time and some hundreds of Mbytes of RAM). The four element layouts are shown in Fig. 6.

Further examples may be found in [18].

IV. Conclusion

A method based on linear programming for finding the optimum weights for minimum peak sidelobe level and a method using mixed integer linear programming for finding both the weights and the element layout have been presented. They have been used to find properties of sparse arrays with random thinning and arrays with optimized thinning. The properties are found by trading off sidelobe level for beamwidth.

Theory for random arrays predicts a mean sidelobe level given by the inverse of the number of elements. In practise the sidelobe level fluctuates much around this mean. With optimization of the weights it is possible for the peak value to be at or even below the predicted value for the mean, especially for 1D arrays. Even for 2D sparse arrays a large reduction in peak sidelobe level is achieved. However, when the thinning pattern is optimized also, solutions which have lower peak sidelobe level than the inverse number of elements can be found both in the 1D and the 2D cases.

References

[1] B. A. J. Angelsen, H. Torp, S. Holm, K. Kristoffersen and T. A. Whittingham, "Which transducer array is best?" *Eur. Journ. Ultrasound,* vol 2, pp 151-164, 1995.

[2] D. J. O'Neill, "Element placement in thinned arrays using genetic algorithms," in Proc. Oceans '94, pp II 301-306, 1994.

[3] S. W. Smith, H. G. Pavy, O. T. von Ramm, "High-speed ultrasound volumetric imaging system. Part I: Transducer design and beam steering," IEEE Trans. Ultrason. Ferroelec. Freq. Contr. vol 38, no 2, pp 100-108, 1991.

[4] R. E. Davidsen, J. A. Jensen, S. W. Smith, "Two-dimensional random arrays for real time volumetric imaging", Ultrasonic Imaging, Vol. 16, pp. 143-163, 1994.

[5] H. Schjær-Jacobsen and K. Madsen, “Synthesis of nonuniformly spaced arrays using a general nonlinear minimax optimization method,” IEEE Trans. Antennas and Propagat., pp 501-506, July 1976.

[6] B. Steinberg, *“Principles of aperture and array system design,”* J. Wiley & Sons, 1976.

[7] D. H. Turnbull and F. S. Foster, “Beam steering with pulsed two-dimensional transducer arrays,” IEEE Trans. Ultrason. Ferroelec. Freq. Contr. vol 38, no 4, pp 320-333, 1991.

[8] P. K. Weber, R. M. Schmidt. B. D. Tylkowski, J. Steck, “Optimization of random sparse 2-D transducer arrays for 3-D electronic beam steering and focusing”, Proc. 1994 IEEE Symp. Ultrasonics, pp 1503-1506, Cannes, France.

[9] J.O. Erstad, S. Holm, “An approach to the design of sparse array systems”, Proc. 1994 IEEE Symp. Ultrasonics, pp 1507-1510, Cannes, France.

[10] S. Holm, B. Elgetun, “Optimization of the beampattern of 2D sparse arrays by weighting,” Proc. 1995 IEEE Symp. Ultrasonics, pp 1345-1348, Seattle, Washington.

[11] R. M. Leahy and B. D. Jeffs, “On the design of maximally sparse beamforming arrays,” IEEE Trans. Antennas Propagat., vol AP-39, No. 8, Aug. 1991, pp. 1178 - 1187.

[12] V. Murino, A. Trucco, and C. S. Regazzoni, “Synthesis of unequally spaced arrays by simulated annealing,” IEEE Trans. Sign. Proc., vol. 44, No. 1, pp 119-123, Jan. 1996.

[13] S. Holm, “Maximum sidelobe energy versus minimum peak sidelobe level for sparse array optimization,” Proc. 1996 IEEE Nordic Signal Processing Symposium (NORSIG 96), Espoo, Finland, pp 227-230, Sept. 1996.

[14] D. H. Johnson, D. E. Dudgeon *“Array signal processing”,* Prentice Hall, 1993.

[15] V. Chvátal, *“Linear programming,”* W. H. Freeman and Company, 1983.

[16] *“Using the CPLEX callable library”*, CPLEX Optimization, 1994.

[17] F. J. Harris, “On the use of windows for harmonic analysis with the discrete Fourier transform,” Proc. IEEE, vol. 66, pp. 61-83, Jan. 1978.

[18] S. Holm, B. Elgetun. G. Dahl, “Properties of the beampattern of weight- and layout-optimized sparse arrays, ” accepted for publication in IEEE Trans. Ultrason. Ferroelec. Freq. Contr., 1997.