PhD Dissertation

International Doctorate School in Information and Communication Technologies

DISI - University of Trento

Innovative methodologies for the synthesis of large array antennas for communications and space applications

Federico Caramanica

Advisor:
Prof. Andrea Massa
University of Trento

November 2011

Abstract

Modern communication and space systems such as satellite communication devices, radars, SAR and radio astronomy interferometers are realized with large antenna arrays since this kind of radiating systems are able to generate radiation patterns with high directivity and resolution. In such a framework conventional arrays with uniform inter-element spacing could be not satisfactory in terms of costs and dimensions. An interesting alternative is to reduce the array elements obtaining the so called "thinned arrays". Large isophoric thinned arrays have been exploited because of their advantages in terms of weight, consumption, hardware complexity, and costs over their filled counterparts.

Unfortunately, thinning large arrays reduces the control of the peak sidelobe level (PSL) and does not give automatically optimal spatial frequency coverage for correlators. First of all the state of the art methodologies used to overcome such limitations, e.g., random and algorithmic approaches, dynamic programming and stochastic optimization algorithms such as genetic algorithms, simulated annealing or particle swarm optimizers, are analyzed and described in the introduction. Successively, innovative guidelines for the synthesis of large radiating systems are proposed, and discussed in order to point out advantages and limitations. In particular, the following specific issues are addressed in this work:

1. *A new class of analytical rectangular thinned arrays with low peak sidelobe level (PSL). The proposed synthesis technique exploits binary sequences derived from McFarland difference sets to design thinned layouts on a lattice of $P \times P\,(P+2)$ positions for any prime P. The pattern features of the arising massively-thinned arrangements characterized by only $P\times(P+1)$ active elements are discussed and the results of an extensive numerical analysis are presented to assess advantages and limitations of the McFarland-based arrays.*

2. *A set of techniques is presented that is based on the exploitation of low correlation Almost Difference Sets (ADSs) sequences to design correlator arrays for radioastronomy applications. In particular three approaches are discussed with different objectives and performances. ADS-based analytical designs, GA-optimized arrangements, and PSO optimized arrays are presented and applied to the synthesis of open-ended "Y" and "Cross" array configurations to maximize the coverage $u-v$ or to minimize the peak sidelobe level (PSL). Representative numerical results are illustrated to point out the features and performances of the proposed approaches, and to assess their effectiveness in comparison with state-of-the-art design methodologies, as well. The presented analysis indicates that the proposed approaches overcome existing PSO-based correlator arrays in terms of PSL control (e.g., $> 1.0dB$ reduction) and tracking $u-v$ coverage (e.g., up to 2% enhancement), also im-*

proving the speed of convergence of the synthesis process.

3. *A genetic algorithm (GA)-enhanced almost difference set (ADS)-based methodology to design thinned planar arrays with low-peak sidelobe levels (PSLs). The method allows to overcome the limitations of the standard ADS approach in terms of flexibility and performance. The numerical validation, carried out in the far-field and for narrow-band signals, points out that with affordable computational efforts it is possible to design planar array arrangements that outperform standard ADS-based designs as well as standard GA design approaches.*

Keywords

[Planar Arrays, Thinned Arrays, Correlator Array Antenna, Difference Sets, McFarland Sequences, Almost Difference Sets, Genetic Algorithms, Particle Swarm Optimizer]

Contents

List of Tables

standard GA methodology, we obtain a reduction of PSL that goes from 1.73[dB] to 0.24[dB].

- **Table XI.** *Problem I- PSL minimisation in array synthesis:* Summary of the results obtained. Comparing the results of the new proposed ADSGA technique with the standard GA methodology, the SPSO, the HSPSO [25] and DS [21], we obtain that ADSGA is able to improve PSL performance also when $\hat{N} \neq N_{ADS}$.

- **Table XII.** *Problem I- PSL minimisation in array synthesis:* Summary of the results obtained. Comparing the results of the new proposed ADSGA technique with the standard GA methodology, the SPSO, the HSPSO [25] and DS [21], we obtain that ADSGA is able to improve PSL performance also when $\hat{N} \neq N_{ADS}$.

- **Table XIII.** *Problem II- extension of the range of ADS applicability:* Summary of the results obtained about thinning factor ν. Comparing the results of the new proposed ADSGA technique with the standard GA methodology and [18].

- **Table XIV.** *Problem II- extension of the range of ADS applicability:* Summary of the results obtained about main lobe dimension BW. Comparing the results of the new proposed ADSGA technique with the standard GA methodology and [18].

- **Table XV.** *Problem II- extension of the range of ADS applicability:* Summary of the results obtained. Comparing the results of the new proposed ADSGA technique with the standard GA methodology and [18]. We obtain with ADSGA a reduction of PSL in both examples.

- **Table XVI**. *Problem III - GA designed ADS construction technique*: Properties of the ADS sequences that have been designed by the proposed GA-based techniques. Neither of these (N, K, Λ, t)-ADS sequences can found in [61] or [65].

List of Figures

- **Figure 10.** *Statistical Arrays* - In (a) there is the computed radiation pattern of a statistically designed array using as a model the $25dB$ Taylor design but with approximately 90 per cent of the elements removed. In (b) the corresponding locations of the elements.

- **Figure 11.** *Thinned Arrays with Genetic Algorithms* - Flow chart of a genetic algorithm.

- **Figure 12.** *Thinned Arrays with Simulated Annealing* - Flow-chart of the optimization procedure.

- **Figure 13.** *Isophoric Array* - (a) Isophoric linear array power pattern. Number of elements = 32. Aperture size = 62 half-wavelengths. (b) Random linear array power pattern. Number of elements = 32. Aperture size = 62 half-wavelengths.

- **Figure 14.** *Isophoric Array* - Expected power pattern of isophoric array with $V = 63$ and $K = 32$.

- **Figure 15.** *Isophoric Array* - Expected power pattern of isophoric planar array with $V = V_x V_y = 15 \times 17$ half-waves and $K = 128$ elements. this exact pattern is realizable with "spatial hopping". Note pattern floor at $10 \log_{10} \rho = -24$dB.

- **Figure 16.** *Linear Thinned Arrays based on Almost Difference Sets* - Autocorrelation function $C_S^{ADS}(z)$ of $\underline{D}_1$ and $\underline{D}_2$ in Table I.

- **Figure 17.** *Linear Thinned Arrays based on Almost Difference Sets* - Normalized $PP(u)$ derived from the ADS derived from the ADS $\underline{D}_4$ ($\underline{D}_4 = \underline{D}_4^{(\sigma)}\Big|_{\sigma=0}$) and its cyclic shifts $\underline{D}_4^{(\sigma)}$ ($\sigma = 17$, $\sigma = 24$). Number of elements: $N = 45$-Aperture size: 22λ.

- **Figure 18.** *Linear Thinned Arrays based on Almost Difference Sets - Comparative Assessment* - Plots of the PSL bounds of the ADS-based finite arrays and of the estimator of the PSL of the random arrays (RND - random array, RNL - random lattice array) when $\nu = 0.489$ versus (a) the array dimension, N, and (c) the index η. Normalized generated from $\underline{D}_4^{opt}$ and estimated PSL values of the corresponding random sequences (b).

- **Figure 19.** *Radio Astronomy* - Conceptual sketch of a radio astronomical measurement using a correlator antenna array. The brightness distribution $I(l, m)$ in the angular domain is retrieved by the inverse Fourier transform of the samplings

of its visibility $V(u,v)$ in the spatial frequency domain. The sampling points are determined by autocorrelating the array configuration $f(x,y)$ in the spatial domain.

- **Figure 20.** *Radio Astronomy* - Relationship among antenna quantities for an incoherent field.

- **Figure 21.** *Radio Astronomy* - The geometry of an interferometer. The baseline intersects the celestial sphere at B, which has declination d and the local hour angle h. The source is at point S, with coordinates δ and H. The projection of the baseline on the intersection of the plane SOB and a plane tangent to the celestial sphere at S is $D\cos\theta$.

- **Figure 22.** *Radio Astronomy* - Basic correlator interferometer system.

- **Figure 23.** *Radio Astronomy* - Comparison between the signal processing schemes of a 2-element: (a) sum array and (b) correlator array.

- **Figure 24.** *Radio Astronomy* - (a) Original source image with the visibility specified by the Gaussian function in (3.124). (b) Image retrieved by the uniform Y-shaped array shown in Fig. $4(a)$.

- **Figure 25.** *Radio Astronomy* - (a) Configuration of the optimized 27-element Y-shaped array (Y_1) for the maximum snapshot $u-v$ coverage. (b) Snapshot $u-v$ coverage of Y has 558s sampled grids.

- **Figure 26.** *Radio Astronomy* - (a) Configuration of the optimized 27-element Y-shaped array (Y_2) for the maximum tracking $u-v$ coverage. (b) Tracking $u-v$ coverage of Y_2 has a filling ratio of 86.5%, as defined in (3.139).

- **Figure 27.** *Radio Astronomy* - (a) Configuration of the optimized 27-element Y-shaped array (Y_2) for the lowest SLL. (b) Synthesized beam of Y has a peak SLL of -20.3 dB.

- **Figure 28.** *Radio Astronomy* - Comparison between a uniform array, a power-law array ($\alpha = 1.7$) and the optimized array Y_3 for SLLs in 8-hour tracking observations with different source declinations.

- **Figure 29.** *Radio Astronomy* - (a) Original image of a Gaussian source and retrieved images by (b) array Y_1, (c) array Y_2 and (d) array Y_3 . The best image is retrieved by optimized array Y_2.

- **Figure 30.** *McFarland Rectangular Arrays* - Example of (a) a McFarland array and (b) the associated (two-level) autocorrelation function ($P = 3$).

- **Figure 31.** *GA-Based McFarland Synthesis* - Plots of (a) the PSL values of the whole set of McFarland arrays and (b) evolution of the PSL of the GA solution during the iterative (i being the iteration index) sampling of the McFarland solution space.

- **Figure 32.** *McFarland Rectangular Arrays* - Behaviour of $\Delta(\eta)$ versus P when $\eta \in \{0.7, 0.8, 0.9, 1.0\}$.

- **Figure 33.** *GA-Based McFarland Synthesis* - Evolution of the PSL of the GA solution during the iterative (i being the iteration index) sampling of the McFarland solution space when (a) $P = 5$ and (b) $P = 7$.

- **Figure 34.** *GA-Based McFarland Synthesis* - Optimal McFarland layouts (a), (c) and the corresponding power patterns (b), (d) when $P = 5$ (a), (b) and $P = 7$ (c), (d).

- **Figure 35.** *GA-Based McFarland Synthesis* - Optimal McFarland layouts (a) $P = 11$ and (b) $P = 13$.

- **Figure 36.** *GA-Based McFarland Synthesis* - Power patterns of the optimal McFarland layouts deduced for (a) $P = 11$ and (b) $P = 13$.

- **Figure 37.** *Comparison with Standard GA-Thinned Rectangular Arrays* - Optimal layout (a) and the corresponding power pattern (b) obtained by GA when $P = 7$, $Q = 63$ and $K = 56$.

- **Figure 38.** *Y-shaped Arrays* [$P = 18$, $Q = 9$, $\Lambda = 4$, $r = 13$, *Equal-unequal arms*] - Plots of the arrangement (a) and associated $S_T(u, v)$ (b) for the array Y_3 [31]; optimal ADS geometry with equal (c) or unequal (e) arms, and associated synthesized beams (d),(f).

- **Figure 39.** *Y_{ADS} Arrays* [$P = 18$, $Q = 9$, $\Lambda = 4$, $r = 13$, *Equal-unequal arms*] - Behavior of optimal (a) PSL, (c) , and (e) ν versus evaluated shift for ADS-based Y arrays, and comparison with reference designs from [31]. Plots of (b) PSL, (d) B, and (f) versus evaluated shift for ADS-based Y arrays.

- **Figure 40.** *Y_{ADS} Arrays* [$P = 18$, $Q = 9$, $\Lambda = 4$, $r = 13$, *Equal-unequal arms*] - Behavior of (a) B versus PSL, (b) ν versus PSL, and (c) ν versus for all Y_{ADS} arrays derived from $\mathbf{D}_1$, and comparison with reference designs from [31].

- **Figure 41.** *Y_{ADS} Arrays* [$P = 18$, $Q = 9$, $\Lambda = 4$, $r = 13$, *Equal-unequal arms*] - Behavior of for Ξ all Y_{ADS} arrays derived from $\mathbf{D}_1$, and comparison with reference designs from [31].

- **Figure 42.** *Problem A* [*Equal-unequal arms*, $N = 27$] - Synthesis results for the GA and ADSGA approaches: (a) behavior of the optimal PSL versus the iteration number i, and comparison with reference designs from [31], (b) optimal Y_{ADSGA} array arrangement, and (c) associated synthesized pattern.

- **Figure 43.** *Problem B* [*Equal-unequal arms*, $N = 27$] - Synthesis results for the RNDPSO and ADSPSO approaches: (a) optimal Y_{ADSPSO} array arrangement and (b) associated $u - v$ coverage function.

- **Figure 44.** *Problem C [Equal-unequal arms,* $N = 27$] - Synthesis results for the RNDPSO and ADSPSO approaches: (a) optimal array arrangement and (b) associated tracking $u - v$ coverage function.

- **Figure 45.** *Problem A* [*Equal-unequal arms*, $N = 27$] - Synthesis results for the RNDPSO and ADSPSO approaches: (a) Behavior of the optimal PSL versus the iteration number i, and comparison with reference designs from [31], (b) optimal Y_{ADSPSO} array arrangement, and (c) associated synthesized pattern.

- **Figure 46.** *Problem A* - Behavior of the optimal PSL versus the iteration number i for the RNDGA, ADSGA, RNDPSO, and ADSPSO approaches for (a) $N = 132$ (equal and unequal arms) and (b) $N = 270$ (equal arms).

- **Figure 47.** *ALMA - Problem A* [*Equal-unequal arms*, $N = 63$] - Synthesis results for the ADSPSO approach: (a) optimal array arrangement and (b) associated $S_T(u, v)$.

- **Figure 48.** *Cross arrays - Problem A* [*Equal-unequal arms*, $N = 60$] - Synthesis results for the RNDGA, ADSGA, RNDPSO and ADSPSO approaches: (a) behavior of the optimal PSL versus the iteration number i, (b) optimal ADSPSO array arrangement and (c) associated $S_T(u, v)$.

- **Figure 49**. *Example from* [23] *of Planar Array based on* $\mathbf{D}_3^{opt}$- *ADS* . Number of elements: $P \times Q = 7 \times 11$. Plots of the PSL bounds versus $\eta = \frac{t}{PQ-1}$ ($PQ = 77$, $\nu = 0.4805$) (a). Plot of the normalized array factor (b) generated from $\mathbf{D}_3^{opt}$- *ADS* array arrangement (c) (courtesy from [23]).

- **Figure 50**. *Numerical validation - Problem I - PSL minimisation in array synthesis*: Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.

- **Figure 51**. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 52**. *Numerical validation - Problem I - PSL minimisation in array synthesis*: Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.

- **Figure 53**. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 54**. *Numerical validation - Problem I - PSL minimisation in array synthesis*: Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.

- **Figure 55**. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 56**. *Numerical validation - Problem I - PSL minimisation in array synthesis*: Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.

- **Figure 57**. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 58**. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Graphical comparison of the PSL of different array configurations (the side P on the horizontal axis) for ADSGA an GA methodologies. We can observe that the PSL improvement of the ADSGA method reduces compared with standard GA as the dimension of the array increases.

- **Figure 59**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.

- **Figure 60**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 61**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.

- **Figure 62**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 63**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.

- **Figure 64**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 65**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.

- **Figure 66**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches.

(c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 67**. *Numerical validation - Problem II - PSL minimisation in array synthesis:* Graphical comparison of the PSL of different array configurations (the side P on the horizontal axis) for ADSGA an GA methodologies. We can observe that the PSL improvement of the ADSGA method reduces compared with standard GA as the dimension of the array increases.

- **Figure 68**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Graphical comparison of the PSL against the iteration i of ADSGA, GA and Haupt [18] approaches along the two main directions $\phi = 0°$ (a) and $\phi = 90°$ (b). Slices of the amplitude pattern obtained after optimization procedure along the two main directions $\phi = 0°$ (c) and $\phi = 90°$ (d).

- **Figure 69**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 70**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Graphical comparison of the PSL against the iteration i of ADSGA, GA and Haupt [18] approaches along the two main directions $\phi = 0°$ (a) and $\phi = 90°$ (b). Slices of the amplitude pattern obtained after optimization procedure along the two main directions $\phi = 0°$ (c) and $\phi = 90°$ (d).

- **Figure 71**. *Numerical validation - Problem II - extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

- **Figure 72**. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(36, 32, 28, 23)$-ADS arrangement, (c) Final 2D ADS layout.

- **Figure 73**. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(36, 32, 28, 23)$-ADS arrangement.

- **Figure 74**. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(60, 6, 0, 29)$-ADS arrangement, (c) Final 2D ADS layout.

- **Figure 75**. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(60, 6, 0, 29)$-ADS arrangement.

- **Figure 76**. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(64, 59, 54, 43)$-ADS arrangement, (c) Final 2D ADS layout.

- **Figure 77**. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(64, 59, 54, 43)$-ADS arrangement.

- **Figure 78**. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(100, 5, 0, 79)$-ADS arrangement, (c) Final 2D ADS layout.

- **Figure 79**. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(100, 5, 0, 79)$-ADS arrangement.

- **Figure 80**. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(144, 137, 130, 101)$-ADS arrangement, (c) Final 2D ADS layout.

- **Figure 81**. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(144, 137, 130, 101)$-ADS arrangement.

- **Figure 82**. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(192, 184, 176, 135)$-ADS arrangement, (c) Final 2D ADS layout.

- **Figure 83**. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(192, 184, 176, 135)$-ADS arrangement.

- **Figure 84**. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(196, 7, 0, 153)$-ADS arrangement, (c) Final 2D ADS layout.

- **Figure 85**. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(196, 7, 0, 153)$-ADS arrangement.

- **Figure 86**. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(225, 8, 0, 168)$-ADS arrangement, (c) Final 2D ADS layout.

- **Figure 87**. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(225, 8, 0, 168)$-ADS arrangement.

Chapter 1

Structure of the Thesis

This chapter describes how the Thesis is organized.

First of all, Chapter 2 presents an overview of the Thesis, pointing out the context of the thinned antenna arrays for communication and radio astronomy, the problem that have been considered and a brief analysis of the solutions proposed in literature.

Chapter 3 describes some of the most significative and relevant techniques in the state-of-the-art, to design thinned arrays for communication and radio astronomy. The aim is to present the basis and background of the work carried out in this Thesis during the research activity developed during my PhD and make a comparative assessment with methodologies proposed in this Thesis.

Chapter 4 deals with a new class of rectangular thinned arrays with low and controlled peak side lobe level (PSL). These arrays are based on McFarland Difference Sets (DSs), that likewise two-dimensional DSs exhibit a two-level autocorrelation function, and on a suitable synthesis procedure based on Genetic Algorithm (GA) optimization. GA has been exploited due to the extremely large number of admissible McFarland sequences. This methodology allows to obtain massively-thinned arrangements with a rectangular shape that exhibit different total main beam widths (TMBWs) in azimuth and elevation and low PSL.

Chapter 5. In this chapter, in order to design correlator arrays for radioastronomy applications a set of hybrid techniques is introduced and numerical validated. These hybrid techniques take advantage of the apriori information on suboptimal analytically derived

correlator arrangements. In more detail, to improve performance of correlators for radioastronomy Almost Difference Sets (ADSs) sequences, that are characterized by almost ideal autocorrelation properties, are exploited with stochastic optimization algorithms such as genetic algorithms (GAs) and particle swarm optimizers (PSOs).

Chapter 6 proposes a GA-enhanced ADS technique (ADSGA) for the synthesis of planar antenna arrays for communication applications and shows that the developed ADSGA hybrid technique allows to overcome the limitations related to the use of ADS sequences and obtain optimal performance.

Chapter 7 concludes the Thesis. In particular the main results are summarized, the open problems and future research directions in the exploitation of the proposed methodologies and techniques are outlined.

Chapter 2

Introduction

2.1 Context and Background

There are many practical ways to exploit antenna arrays. Antenna arrays are widely used both in civil and military applications. In communication and broadcast engineering they are used in TLC systems such as TV and radio transmitters, for example in AM or FM broadcast radio stations to enhance signal. Arrays are largely utilized in warships, aircraft radar systems and missile fire-control systems. Other uses are sonar, weather research and biomedical (e.g. radiotherapy) applications [1][2]. Another particular kind of framework where antenna arrays can be very useful is represented by space applications, e.g. satellite communication systems and radio astronomy. The radiating systems of these applications have some common requirements: high resolution (the term "resolution" is used in the sense of Rayleigh and is proportional to the beamwidth), high gain, low sidelobe level [3] and, for radio astronomy applications, optimal coverage in spatial frequency domain. In communication and space applications, steerable reflectors are one of the most useful kinds of antennas. Reflectors have a diameter that can be equal up to 100m but they cannot be much larger because of mechanical problems and prohibitive costs.

Figure 1. *Introduction* - Example of large reflector antenna.

For these reasons, the attention has turned to very large arrays with a number of radiating elements from two up to hundreds or thousands. For conventionally designed arrays where all elements are uniformly spaced an upper limit exists to the spacing, if the grating lobes are not permitted to appear in the visible region. In this case we have traditional *filled arrays* that have an element placed in every location of a uniform lattice with half-wavelength spacing between the lattice points. As a result the required number of elements, being proportional to the aperture dimension in wavelength, becomes astronomically large if a beamwidth on the order of minute of arc is desired [3].

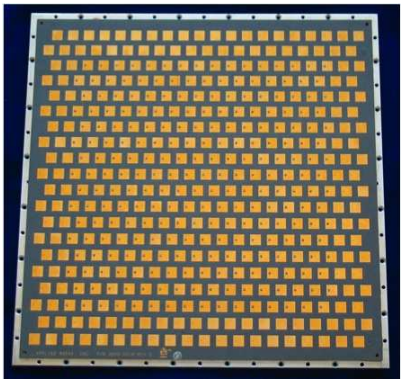


Figure 2. *Introduction* - Example of conventional filled array with patch radiating elements.

Most of the recent investigations on arrays with non-uniformly spaced elements showed the possibility of reducing the number of radiating elements and optimizing the design of arrays. An unequally spaced, *thinned array* may be used to:

1. achieve a narrow main lobe with reduced number of elements

2. achieve a wide scan angle or operate over a broad frequency band without the appearance of grating lobes

3. achieve desirable radiation patterns without amplitude taper across the aperture.

Thinning an array means turning off some elements in a uniformly spaced or periodic array to create a desired amplitude density across the aperture [4]. An element connected to the feed network is "on", and an element connected to a matched or dummy load is "off". When thinned arrays have fewer than half of the elements of their filled counterparts, they are called *massively thinned arrays*. In this research proposal we are not interested in amplitude tapering techniques since these methodologies have a higher complexity and cost [5]. We have to remember that thinning is normally accompanied by loss of sidelobe control, for this reason, thinned arrays are synthesized in according to one or more optimization criteria. For example, optimization of the beam pattern means to achieve the minimum PSL in the entire visible range or the maximum gain [3][4][6].

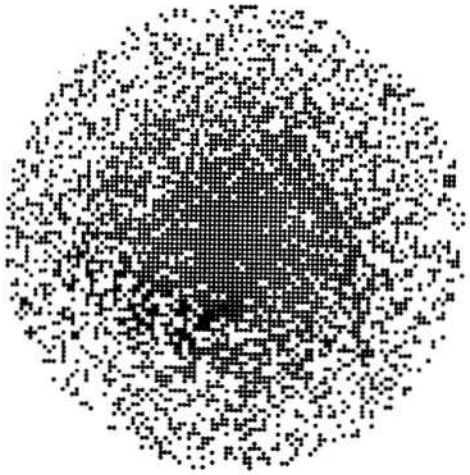


Figure 3. *Introduction* - Example of large circular thinned array.

In this scenario large thinned arrays allow us to obtain the following advantages: better performance with respect to reflector antenna, increased operational robustness, implementation cost saving and more programmatic flexibility. Each of these topics is discussed further in the following paragraphs. For larger antennas, the beam width naturally is narrower. As a result, antenna-pointing error becomes more critical. To stay within the main beam and incur minimal loss, antenna pointing has to be more precise. Yet this is difficult to achieve for larger structures. With an array configuration of smaller antennas, antenna-pointing error is not an issue. The difficulty is transferred from the mechanical to

the electronic domain. As long as the combining process is performed with minimal signal degradation, an optimal gain can be achieved. Arraying also allows an increase ineffective aperture beyond the present capability for supporting a mission at a time of need. In the past, the Voyager Mission relied on arraying to increase its data return during Uranus and Neptune encounters in the late 1980s. The Galileo Mission provides another example in which arraying was used to increase the science data return by a factor of 3. (When combined with other improvements, such as a better coding scheme, a more efficient data compression and a reduction of system noise temperature, a total improvement of a factor of 10 was actually realized) [7]. Arraying can increase system operability. Firstly, higher resource utilization can be achieved. In the case of an array the set can be partitioned into many subsets supporting different missions simultaneously, everyone tailored according to the link requirements. So doing, resource utilization can be enhanced. Secondly, arraying offers high system availability and maintenance flexibility. Let us suppose an array built with 10 percent spare elements. The regular preventive maintenance can be done on a rotating basis while allowing the system to be fully functional at all times. Thirdly, the cost of spare components would be smaller. Instead of having to supply the system with 100 percent spares in order to make it fully functional around the clock, the array offers an option of furnishing spares at a fractional level. Equally important is the operational robustness against failures. With a single resource, failure tends to bring the system down. With an array, failure in an array element degrades system performance but does not result in a service shutdown [7]. In particular, thinned arrays can be projected to have a certain amount of redundant radiating elements in order to guaranteeing PSL control in presence of one or multiple failures.

A cost saving is realized from the fact that smaller antennas, because of their weight and size, are easier to build and move. The fabrication process can be automated to reduce the cost. It is often approximated that the antenna construction cost is proportional to the antenna volume. The reception capability, however, is proportional to the antenna surface area. Note, however, that antenna construction is only a part of the overall life cycle cost for the entire system deployment and operations. To calculate the actual savings, one needs to account for the cost of the extra electronics required at multiple array elements and the cost related to the increase in system complexity [7]. One of the most important quality of thinned arrays is the reduced number of antennas: with few radiating elements we can keep under control the PSL, satisfying the technical requirements, and also increase the cost saving. Arraying offers a programmatic flexibility because additional elements can be incrementally added to increase the total aperture at the time of mission need. This option allows for a spread in required funding and minimizes the need to have all the

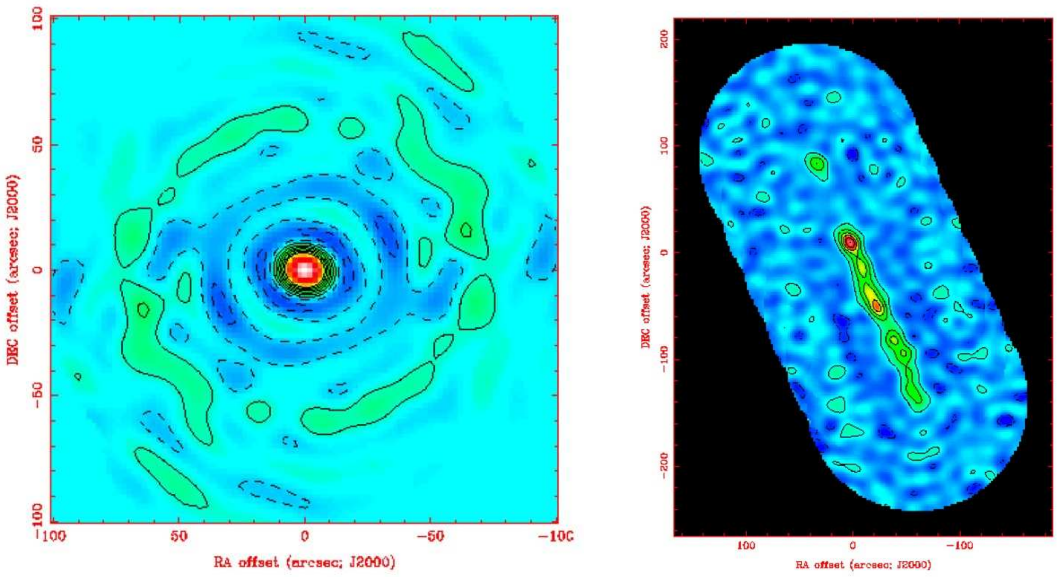
cost incurred at one time. The addition of new elements can be done with little impact to the existing facilities that support ongoing operations.

In conclusion thinned arrays seem to be suitable to satisfy the previous requirements typical of communication systems and improve performance.

Radio interferometers and synthesis arrays, which are basically ensembles of two element interferometers, are used to make measurements of the fine angular detail in the deep radio emission from the sky. The angular resolution of single radio antennas is insufficient for many astronomical purposes. Practical considerations limit the resolution to a few tens of arcseconds. For example, the beamwidth of a 100m diameter antenna at 7mm wavelength is approximately 17arcsec. In the optical range the diffraction limit of large telescopes (diameter-8 m) is about 0.015 arcsec, but the angular resolution achievable from the ground by conventional techniques is limited to about one arcsec by turbulence in the troposphere. For progress in astronomy it is particularly important to measure the positions of radio sources with sufficient accuracy to allow identification with objects detected in the optical and other parts of the electromagnetic spectrum. It is also very important to be able to measure parameters such as intensity, polarization, and frequency spectrum with similar angular resolution in both the radio and optical domains. Radio interferometry enables such studies to be made. Precise measurement of the angular positions of stars and other cosmic objects is the concern of astrometry. This includes the study of the small changes in celestial positions attributable to the parallax introduced by the earth's orbital motion, as well as those resulting from the intrinsic motions of the objects. Such measurements are an essential step in the establishment of the distance scale of the universe. Radio techniques provide an accuracy of the order of arcsec or less for the relative positions of objects closely spaced in angle.

Compared with communication systems, to obtain optimal performance, namely a high-sensitive and high-resolution measurement of radio sources, a uniform inter-element spacing of the radiating elements is not the best solution. We need not only a low PSL but also coverage of spatial frequency domain as uniform as possible. If the spatial domain is not uniformly sampled the radio source is not correctly recovered and spurious artifacts are presents. A non-uniformly spaced correlator array, as shown in [8][9], gives the possibility of reducing the PSL and optimizing the coverage.

Figure 4. *Introduction* - The VLA, an array of 27 elements, each a 25-m paraboloid, is a Y-shaped array having three equiangular linear arms of 21 km.



(a) (b)

Figure 5. *Introduction* - (a) and (b) are examples of radio maps obtained with radio astronomy correlators.

Chapter 3

State of the Art

3.1 Arrays for Communication and Radio Astronomy - Introduction to the State-of-the-Art

In the framework of arrays for communications, radar and space applications, Skolnik proposed one of the first examples of thinning large antenna arrays. In his work [4] he describes statistically designed density-tapered arrays. With the usual method for designing directive antennas with low sidelobes, the received (or radiated) energy is greater at the centre than at the edges [4]. The idea proposed in [4] is the following: the density of elements located within the aperture is made proportional to the amplitude of the aperture illumination of conventional filled arrays (designed with Taylor or Dolph methods [10][11]). In other words, the signal at each element of the array is of equal amplitude but the spacing between adjacent elements differs. The selection of the element locations is performed statistically by utilizing the amplitude illumination as the probability density function for specifying the location of elements (for this reason it is also called space taper) [4]. Statistically designed density-tapered arrays are useful when the number of elements is large and when it is not practical to employ an amplitude taper to achieve low sidelobes. A density taper has advantages over an amplitude taper in certain applications. Transmitting arrays, for example, with individual power amplifiers at each element are easier to design and to build and more efficient to operate if each amplifier delivers full rated power [4]. The density-tapered array permits the system designer to employ equal-power amplifiers at each element and still achieve low sidelobes. Receiving antennas can also benefit from density tapering. In conclusion, this technique is to be considered for the design of large array antennas where good sidelobes are important and where it is not convenient to use an amplitude taper across the aperture [4].

In [6] Steinberg derived a formula for the PSL of a thinned array where the elements are randomly located. In a random array, the location of each radiating element is a random variable drawn from a population described by a probability density function (e.g.uniform pdf). Since an a-priori description of a random array can only be given statistically, it is logical to seek an estimator of the peak sidelobe in terms of a probability or confidence level that the predicted value will not be exceeded. Steinberg obtained a probabilistic estimator of the peak sidelobe of uniform random array with equally weighted elements. This theoretical result was tested by measurement of the peak sidelobe of several hundred Monte Carlo computer-simulated random arrays [6].

During the 1960's many thinning algorithms were created. The methodologies to thin arrays fall into the following categories: algorithmic-specific aperiodic designs; random-element locations chosen at random; random removal-holes chosen at random; dynamic programming-quasi-trial-and- error. In [6], Steinberg compared algorithmic design of thinned aperiodic arrays tested by computer simulations with random arrays. The distribution is compared to that of a set of 170 random arrays [6][6]. Both distributions are found to be nearly log normal with the same average and median values. They markedly differ in their standard deviations. However, the standard deviation of the random array distribution is approximately half that of the algorithmic group. The author showed that algorithmically thinned arrays rarely offer enough control of the far radiation pattern to be superior to random arrays. Furthermore the compactness of the random distribution almost guarantees against selection of a random array with catastrophically large peak sidelobes. The only procedure that gives superior performance is dynamic programming-quasi trial-and-error method of sidelobe control, a highly constrained approach. More in detail, the first element is located at random. The second location is that which gives the best combination. The third location is that which gives the best trio based on the fixed locations of the first two elements, etc. Despite dynamic random design method is commonly considered as the reference strategy for the synthesis of thinned arrays because of its simplicity (does not require any computational procedure), its good performance (quasi trial-and-error method gives a slight improvement) and flexibility [6][6].

In order to improve performance of thinned arrays respect to random arrays, different ways have been used. The first is based on the use of optimization algorithms and the second on particular kind of combinatorial sequences.

Assuming, like in the previous methodologies, the number of radiators is a finite number and each radiator can have two values on and off (thinning may also be thought of as quantized amplitude taper where the amplitude at each element is represented by one

bit), the number of possible combinations, where Q is the number of array elements, is 2^Q. Thinning a large array for low sidelobes involves checking a rather large number of possibilities in order to find the best thinned aperture. Exhaustive checking of all possible element combinations is only practical for small arrays [13]. Optimization algorithms represent an alternative to exhaustive search. Most optimization methods (including down-hill simplex, Powell's method, and conjugate gradient) are not well suited for thinning arrays. They can only optimize a few continuous variables and get stuck in local minima [14]. Also, these methods were developed for continuous parameters, whereas the array-thinning problem involves discrete parameters. The dynamic programming method can optimize a large parameter set (many elements), but it is also vulnerable to local minima [15]. Simulated annealing and genetic algorithms (GA) [14][16][17] are optimization methods that are well suited for thinning arrays. There is no limit to the number of variables that can be optimized. Although quite slow, these algorithms can handle very large arrays. These methods are global since they have random components that test for solutions outside the current minimum, while the algorithm converges. The global nature of the algorithms and the lack of derivative information cause a very slow converge compared to other non-global methods. If the array is symmetric, then the number of possibilities is substantially smaller and the GA converges faster.

In [18], Haupt presents an example of thinning strategy based on Genetic Algorithms (GAs) used to find a thinned array that produces the lowest PSL allowing us to improve the performance of large arrays. A Genetic Algorithm is a global method for optimization inspired by the Natural Selection Principle whose main concepts are competition and adaptability [14]. The paper [18] shows that the on/off structure of the thinned array (linear or planar) is codified into the chromosomes of the GA. After encoding the parameters in binary strings called genes, GA performs the genetic operations of reproduction, crossover, natural selection, and mutation to arrive at the optimum solution. During each iteration, the trial solution provides by the GA is given in input to the fitness function. The fitness is defined in [18] as the PSL and the purpose of the GA is to find out the array configuration minimizing this function. A genetic algorithm can be used to numerically optimize both linear and planar arrays and arrives at better thinning configurations for arrays than previous optimization attempts or statistical attempts. Previous methods of array thinning used statistical methods may fail to produce an optimum thinning while the genetic algorithm searches in a smart way for the best thinning that produces low sidelobes [18].

A different approach to obtain low PSL large arrays is to use particular kind of com-

binatorial sequences. With this approach Leeper describes in [5][19] a class of massively thinned linear and planar arrays that shows well-behaved sidelobes in spite of the thinning.

The Genetic search algorithms can obtain better performance but this method is not appropriate for very large or very highly thinned arrays and the improvements that this methodology offers are difficult to predict a-priori. Rather than using a search algorithm, the approach in [5][19] attacks directly the sidelobe control problem by applying the properties of Difference Sets (DSs) [2], to the placement of antenna elements within a regular lattice. In particular Leeper uses the class of Cyclic-Difference Sets (CDS) sequences as function that describes the position of active elements in arrays [20]. The property that makes CDS an effective prescription for the design of the thinned array is that the autocorrelation of CDS (and generally all kind of DSs) is a two-valued function. It is possible to demonstrate [5] that this kind of autocorrelation allows controlling the PSL of an array built with CDS geometry. The CDS method guarantees more effective suboptimal array synthesis in terms of PSL with respect to random elements placement. 2D-CDSs have similar autocorrelation property of 1DCDSs [2][5][19][20].

The deterministic placements of DS create an isophoric array ("isophoric" means "uniform weight") with attendant uniformity of spatial coverage. The uniformity consistently produces, with no searching required, a reduction in PSL when compared to random element placement. More specifically, in any linear array of aperture half-wavelengths, the Nyquist sampling theorem shows that the array power pattern can be completely determined from uniformly spaced samples of the pattern. In an isophoric array, the even-numbered samples will necessarily be "locked" to a constant value less than $1/K$ times the main-beam peak, where K is the number of elements in the thinned array. While the odd-numbered samples are not so constrained, the net effect is to produce patterns with much lower PSL than are typical with cut-and-try random placement. Obviously, isophoric arrays can be planar as well linear [5].

In [21], Kopilovich suggests another method for synthesizing a planar aperiodic thinned array antenna with a low peak sidelobe level. Instead of using the previous CDS, Kopilovich shows the implementation of combinatorial constructions called non-Cyclic difference sets. The most important class of the non-Cyclic 2D-DSs is represented by the sets of Hadamard type (HDSs). In the same way of the previous Leeper method, Kopilovich uses the fact that when the elements of an equi-amplitude array antenna are arranged according to a DS law, its pattern takes constant value in the net of uniformly located space points in the sidelobe region, and this value is less than $1/K$, where K is the active element number. In distinction to the method using Cyclic DSs developed by Leeper [5][21], that

enables one to build planar antenna arrays only on rectangular grids with co-prime side lengths, the described method omits such a constraint. Based on such sets, rectangular and square aperiodic roughly half-filled array antennas can be built. Kopilovich uses this strategy to obtain square array antennas, with the element number in the array up to 300.

The definition of binary sequences of length with suitable autocorrelation properties, for which DSs are not available, has been carefully investigated in information theory and combinatorial mathematics. It has been found that it is often possible to determine sequences with a three-level autocorrelation function by taking into account the so-called almost difference sets (ADSs) [22][23]. ADSs are a research topic of great interest in combinatorial theory with important applications in cryptography and coding theory. Moreover, although ADS generation techniques are still subject of research, large collections of these sets are already available. In such a framework, the whole class of ADSs seem to be a good candidate for enlarging the set of admissible analytic configurations with respect to the DS case. From this viewpoint, ADSs allow to obtain low PSL and predictable results in a very effective. With respect to DSs, ADSs have the advantage of having a larger set of admissible sequences [22][23].

Finally, the last approach described to improve large arrays performance is based on merging the combinatorial and stochastic methods in order to take advantage from their good characteristics and to compensate for their drawbacks [5].

One of the first attempts to exploit this idea was developed by Caorsi et al. [24]. The ripples formation caused by CDS could be corrected in some way by GA search capabilities, while the uniform spatial coverage of CDS-optimized arrays could be helpful to speed up the convergence of the genetic procedure. One possible way of implementing this approach is to consider CDS based arrays as a-priori knowledge to be inserted in the genetic search process in order to improve its efficiency. To this end, the steps aimed at transferring good CDS-derived schemata into the GA population are the following. At the initialization step, the GA population is composed by a selected CDS D_0 and V cyclic shifts of the D_0 difference set, while the remaining chromosomes of the initial population are randomly mutated cyclic shifts. Moreover, during the iterative loop of the GA, the mutation occurs in order to introduce new unexplored solutions into the search space. In order to keep higher order CDS-derived schemata, trial solutions having binary configurations belonging to higher order schemata are mutated only in chromosome positions out of the schemata locations [24]. These mechanisms are aimed at constraining the GA to synthesize array configurations similar to CDS-based ones, but with limited

ripple amplitudes thanks to evolutionary capabilities [24].

In the same way Donelli et al. make use of a hybrid technique based on HDS and binary PSO [25][26]. PSO is a stochastic multiple agents optimization algorithm extensively applied in the framework of antenna array optimization [25][26][27]. By imitating the social behaviour of groups of insect and animals in their food searching activities, PSO is based upon the cooperation among particles. The ensemble of the particles, referred to as swarm, explores the solution space to find out the best position (i.e., the optimum of a suitably defined cost function). HSs-based arrays generate the initial trial solutions of this hybrid method that then is optimized by binary PSO. Integrating the HS-based method developed by Kopilovich [21] with PSO optimization strategy gives an important improvement in thinned array performance.

In the framework of the antenna array for space systems, we have a particular application where the previous synthesis techniques were applied. Arrays are used in radio astronomy to estimate the brilliance [9][29][30]. Astronomers are interested in designing correlator arrays that properly sample the spatial distribution they observe. The design of correlator (also known as interferometric) arrays is essentially an optimal sampling problem [9][29][30] in which the positions of the antennas are chosen in order to ensure optimal performance regarding all possible observation situations (source positions and durations of observation), scientific purposes (single field imaging, astrometry, detection, ...) and constraints (cost, ground composition and practicability, operation of the instrument, ...) [31][32]. In order to obtain such features, high performance correlator arrays have to show either a maximal coverage in the spatial frequency (or $u-v$) domain, or a minimum peak sidelobe level (PSL) in the angular (or $l-m$) domain [8][31]. Towards this end, many different design principles have been proposed, including minimum redundancy [33], pseudo-randomness [34], power laws [35], difference set arrangements [36], and minimization of the holes in the sampling [37]. Ruf in [16] uses simulated annealing to optimize low-redundancy linear arrays while Jin [31] makes use of PSO. Well-established optimization based sum-array design techniques cannot be directly applied, since, unlike in traditional sum arrays, the responses in both the $u-v$ and the $l-m$ domains have to be considered in the design procedure [31]. As a consequence, design techniques have to include the array spatial coverage evaluation, the Earth rotation effects and the $l-m$ beam calculation in the synthesis procedure.

3.2 Random Arrays [6]

3.2.1 Introduction

The cost of a large phased array which is designed primarily for high angular resolution rather than for weak signal detection may be reduced manifold through *thinning*, i.e., reducing the number of elements in the aperture below that of the *filled* array in which the inter element spacing is nominally one half-wavelength. Increasing the inter element spacing has another salutary effect; a separation of a few wavelengths reduces mutual coupling to negligible proportions. Thinning, therefore, is attractive from both points of view. But these benefits are not free of penalty. Unless the element locations are randomized or made otherwise non periodic, grating lobes appear. Also, the reduction in the number of elements reduces the designer's control of the radiation pattern in the sidelobe region, which in turn influences the level of the largest, or peak, sidelobe. In this chapter the peak sidelobe of random arrays is studied (N.B.: The **random array** ([6]) is characterized by element locations chosen by some random process. Conversely in a **statistical array** ([4]) a conventional filled array is designed and a given fraction of the elements is removed at random).

3.2.2 Linear Random Array

Consider an array of N unit, isotropic and monochromatic radiators at locations x_n. The x_n are chosen from a set of independent random variables described by some first probability density distribution, initially assumed to be uniform over the interval $[-L/2, L/2]$ where L is the array length. It is assumed that each element, irrespective of its location, is properly phased so that a main lobe of maximum strength is formed at θ_0, which is measured from the normal to the array. The reduced angular variable $u = \sin\theta - \sin\theta_0$, contains the beam steering information. The complex far-field radiation pattern $f(u)$ is the Fourier transform of the current density. Since the latter is a set of delta functions, $f(u)$ is proportional to the sum of unit vectors having phase angles kx_nu, $k = 2\pi/\lambda$ being the wavenumber associated with the wavelength λ. The array factor is the Fourier transform of the current density $i(x)$. The current density $i(x)$ of a random array of N equally excited isotropic elements is the sum of delta functions at the locations x_n and the **complex far-field radiation pattern** becomes

$$f(u) = \mathcal{F}\left\{\sum_{n=0}^{N} \delta\left(x - x_n\right)\right\} = \sum_{n=0}^{N} \exp\left(jkx_nu\right) \tag{3.1}$$

3.1 can be rewritten as

$$\begin{array}{rcc} f(u) & = & \sum_{n=0}^{N} \cos(kx_n u) + j \sum_{n=0}^{N} \sin(kx_n u) \\ & = & a(u) + b(u) \end{array} \tag{3.2}$$

Since u is defined over the interval $[-1, 1]$, it follows that $|f(-u)| = |f(u)|$. Therefore, it is sufficient to consider the radiation pattern $|f(u)|$ only over the interval $[0, 1]$.

The radiation pattern $f(u)$ as given by (3.2), is a complex random process. For the special case where element locations are independent and uniformly distributed over the interval $[-L/2, L/2]$, the expected values of the processes $a(u)$ and $b(u)$ are

$$E\{a(u)\} = \frac{N \sin(\pi u L/\lambda)}{\pi u L/\lambda} = N sinc(uL/\lambda) \tag{3.3}$$

and

$$E\{b(u)\} = 0 \tag{3.4}$$

The process $a(u)$ and $b(u)$, for a given value of u, are sums of N independent, identically distributed random variables. When N is large, the central the central limit theorem justifies approximating $a(u)$ and $b(u)$ as Gaussian random variables. The mean of $a(u)$, as given by (3.3), is approximately zero for u greater than a few beam widths (the nominal beamwidth is λ/L). Furthermore, for imaging problems in which high angular resolution is demanded, $\lambda/L \ll 1$. Thus in most of the sidelobe region, the two orthogonal components of $f(u)$ are approximately zero-mean wide sense stationary Gaussian random processes.

For a given u, the **magnitude** of $f(u)$ is known to be **Rayleigh distributed** [?]. Let us denote the magnitude pattern as $A(u)\Delta |f(u)|$. The probability density function of $A(u)$ will be given by [6]

$$p(A) = \frac{2A}{N} \exp\left(-A^2/N\right) \tag{3.5}$$

It follows that the mean square value $\overline{A^2}$, which is **the average sidelobe power level**, is N (and consequently the **rms amplitude** is $\sqrt{N}$). The **power ratio of the average sidelobe to the main lobe** is $N/N^2 = 1/N$. The average is $\overline{A} = \sqrt{\pi N/2}$. Hence, the variance is $\sigma^2 = \overline{A^2} - \left(\overline{A}\right)^2 = N(1 - \pi/4)$.

The integral [6]

$$\alpha = \int_{A_0}^{\infty} p(A)\, dA = \exp\left(-A^2/N\right) \tag{3.6}$$

is the probability that the magnitude of an arbitrary sample of the radiation pattern, away from the region of the main lobe, exceeds some threshold A_0. Its complement, $1 - \alpha$, is the probability that such a sample is less than A_0. If n independent samples are taken [6]

$$\beta = \left[1 - \exp\left(-A_0^2/N\right)\right]^n \tag{3.7}$$

is the probability that none exceeds A_0. From (3.3), $A_0^2 = -N \ln\left(1-\beta^{1/n}\right)$. It is convenient to normalize this expression to N, the average sidelobe level, and to give the dimensionless power ratio A_0^2/N a new symbol, B. Thus [6]

$$B = -\ln\left(1-\beta^{1/n}\right) \approx \ln(n) - \ln\left(\ln\left(\beta^{-1}\right)\right) \tag{3.8}$$

B may be interpreted as a statistical estimator of the power ratio of the peak-to-average sidelobe of a set of n independent samples. B is a confidence level; it is the probability that none of n independent samples of the sidelobe power pattern exceeds the mean value by the factor B. n is an array parameter, which is a function of all the relevant array properties other than N. It is proportional to the number of sidelobes in the visible region. It maybe calculated in several ways. An interesting method utilizes the Nyquist sampling theorem. The complex radiation pattern of a random array is such a *band-limited* function, the "limit" being due to the finite length of the array. The far-field complex radiation pattern $f(u)$ is related to the radiating element positions according to (3.1). From (3.1) we can define the expression for the **power pattern** of an array of unit radiators

$$f(u)f^{\star}(u) = \sum_{m=0}^{N}\sum_{n=0}^{N} \exp\left(jk\left(x_n - x_m\right)u\right) \tag{3.9}$$

The visible domain is $-1-\sin\theta_0 \le u \le 1-\sin\theta_0$. The length of the non-redundant portion is $1+|\sin\theta_0|$. Consequently, the number of independent samples needed to specify the complex radiation pattern is $2\,(L/\lambda)\,(1+|\sin\theta_0|)$. Half this number may be associated with the amplitude of the array factor and half with its phase. Therefore, the power pattern is uniquely specified by [6]

$$n = \left(\frac{L}{\lambda}\right)(1+|\sin\theta_0|) \tag{3.10}$$

independent samples, the average angular interval between samples being λ/L. n is dominated by the length of the array in units of wavelength and secondarily influenced by the beam steering angle.

Equations (3.8) and (3.10), however, are insufficient to provide an unbiased estimate of the peak sidelobe. **The probability is zero that any finite set of samples of a power pattern falls exactly upon the crest of the largest sidelobe. Hence such estimation is downward biased.** A **correction** to (3.8) may be obtained by calculating the difference between the largest of a set of samples and the height of the lobe from which the sample is taken. The approximate mean increment reduces to $1+2/B$, and the estimator of the normalized peak becomes (for details [6])

$$B_p = B + 1 + \frac{2}{B} \tag{3.11}$$

The power ratio of the peak sidelobe to the main lobe is [6]

$$\frac{peak\ sidelobe}{main\ lobe} = \frac{peak\ sidelobe}{avg} \cdot \frac{avg}{main\ lobe} = B_p \cdot \left(\frac{1}{N}\right) = \frac{B + 1 + 2/B}{N} \tag{3.12}$$

Experimental data indicate that the estimator closely matches the data when $B \gtrsim 3$. The fact that the match is satisfactory for B as small as 3 implies that (3.12) is useful even for small arrays. Using $B = 3$ in 3.8 gives the smallest array for which the estimator is satisfactory.

The minimum number of elements for which the theory is satisfactory ([6]) is the larger of 15 or $2B\,(n, \beta)$, or

$$N_{min} = \max\{15, 2B\} \tag{3.13}$$

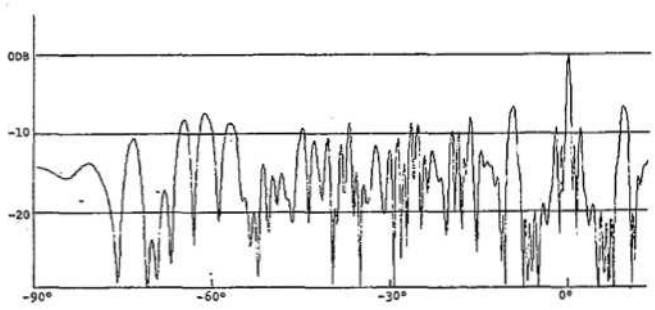


Figure 6. *Random Arrays* - Pattern of 70-wavelength random array of 30 isotropic elements [6].

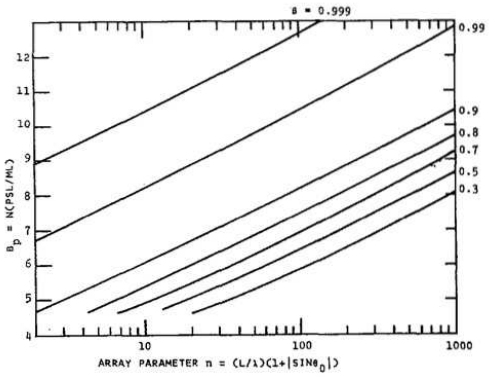
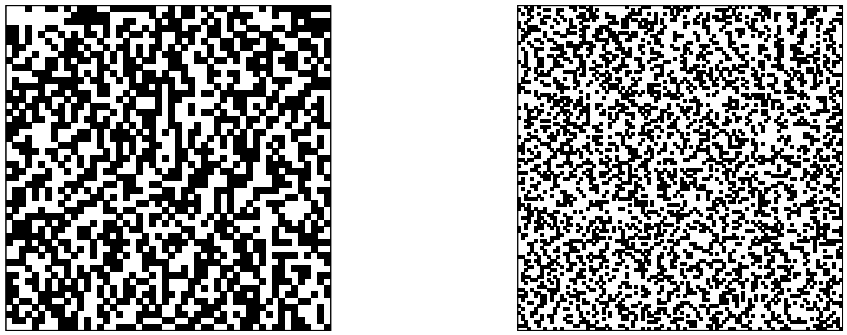


Figure 7. *Random Arrays* - Probabilistic estimator of peak sidelobe of random array. N is the is number of array elements, PSL/ML is power ratio of peak sidelobe to main lobe, β is probability or confidence level that no sidelobe exceeds ordinate, L is array length, λ is wavelength, θ_0 is beam steering angle [6].

3.2.3 Planar Array

Extension of the peak sidelobe theory to two and three dimensional arrays requires only a reevaluation of the array parameter n. Consider as an example a rectangular planar array having sides L_1 and L_2 and uniform pdf of element location. The angular interval for independent sampling of the pattern amplitude in these orthogonal planes is λ/L_1 and λ/L_2. The area in the $u_1 - u_2$ plane associated with each sample point is on the order of $\lambda^2/(L_1 L_2)$. The visible area of the plane, which is a circle of unit radius, is π. Hence the maximum number of independent samples over the hemisphere is approximately $\pi L_1 L_2/\lambda^2$. The same result pertains to a three-dimensional array in which $L_1 L_2$ is the projected area upon a plane perpendicular to the axis of the main lobe of the element factor. Symmetry in the pattern reduces the number of independent samples. With the array steered to the zenith ($\theta_0 = 0$) each lobe in every polar cut has an image lobe in the same plane [6]. Thus the range of variation of n with θ_0 is a factor of two. The logarithmic relation (3.8) between peak sidelobe and the array parameter minimizes the importance of the detailed variation. The dominant feature is the approximate squaring of n when a fixed number of elements N is spread from a linear array to a planar array of the same length and width. The result is (approximately) a doubling, or $3 - dB$ increase, in the peak sidelobe [6].



(a) (b)

Figure 8. *Random Arrays* - Examples of (a) a 50×50 elements square random array and (b) a 100×100 elements square random array [6].

3.2.4 Comparison between the Peak Sidelobe of the Random Array and Algorithmically Designed Aperiodic Arrays [12]

3.2.4.1 Database

In [12], a database of 170 random arrays with various parameters were created by computer, their antenna patterns calculated, and the peak sidelobe of each measured. Approximately half that number algorithmically designed aperiodic arrays were collected from the literature. For each, the peak sidelobe was measured and the pertinent, parameters tabulated.

The aperiodic designs fall into the following categories:

- algorithmic: specific aperiodic designs
- random: element locations chosen at random
- random removal: holes chosen at random
- dynamic programming: quasi-trial-and error

The random arrays were developed for an earlier study of the peak sidelobe of such arrays [6]. The elements were located on a line by random numbers drawn from a population having uniform probability density.

3.2.4.2 Results

Algorithmic design of thinned aperiodic arrays rarely offers enough control of the far radiation pattern to be superior to random location of the array elements. A study of 70 algorithmic arrays and 170 random arrays showed their peak sidelobes, when suitably normalized to permit, comparison, to be indistinguishable in the mean and median [12].

A quasi-trial-and-error procedure called dynamic programming was found to be $3.5dB$ superior in the mean. The distribution of the normalized peak sidelobe of the 170 random arrays found to be log normal with a standard deviation of $1.1dB$. The compactness of the distribution precludes the use of trial-and-error procedures to achieve a peak sidelobe materially below the population mean. The same characteristic almost, guarantees against selections of element locations which produce unexpectedly large sidelobes [12].

3.3 Statistical Removal (Random Removal) [4]

3.3.1 Introduction

This chapter considers the design of "thinned" planar array antennas in which the density of elements located within the aperture is made proportional to the amplitude of the aperture illumination of a conventional "filled" array. (A "thinned" array is one that contains less elements than a "filled" array of equally spaced elements located a half wavelength apart). The selection of the element locations to provide the desired density taper is performed statistically by utilizing the amplitude taper as the probability density functions for specifying the location of the elements. In a "thinned" array all the elements are assumed to radiate equal power if a transmitting array, or equal amplitude weighting if a receiving array. It is further assumed that the element spacings of a "thinned" array are not equal [4].

An unequally spaced, thinned array may be used to

- achieve a narrow main lobe with reduced number of elements
- achieve a wide scan angle or operate over abroad frequency band without the appearance of grating lobes
- achieve desirable radiation patterns without amplitude taper across the aperture.

3.3.2 Analysis of Statistical Density-Tapered Arrays

The usual method for designing directive antennas to achieve low sidelobes is to taper the amplitude of the aperture illumination so that the received (or radiated) energy is greater at the center than at the edges.

A density taper has advantages over an amplitude taper in certain applications. Transmitting arrays, for example, with individual power amplifiers at each element are easier to design and build and more efficient to operate if each amplifier delivers full rated power. The density-tapered array does not suffer any of amplitude taper inconveniences and permits the system designer to employ equal-power amplifiers at each element and still achieve low sidelobes. Receiving antennas can also benefit from density tapering.

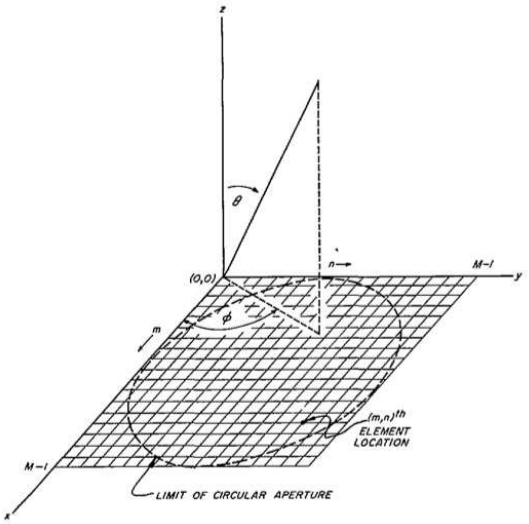
The theory of the design of density-tapered arrays is not on as firm a foundation as that of amplitude tapered arrays. The design techniques of Dolph ([10]) and Taylor ([11]) which are based on the properties of polynomials and which are widely used for amplitude tapered antennas do not seem applicable to unequally spaced arrays.

There are two basic methods for matching a density taper to an amplitude taper. In one technique the density is matched deterministically to the desired amplitude taper by trial and error placement of the elements or by certain approximation techniques applied to the integral of the aperture illumination. The other design technique, and the one which is the subject of this paper, is a statistical method which utilizes the desired amplitude illumination as a probability density function for determining whether or not an element should be located at a particular point within the aperture.

Consider an array antenna with some arbitrary arrangement of N elements. The excitation at each element is assumed to be of equal amplitude. The field intensity pattern (array factor) assuming the elements to be isotropic radiators is [4]

$$E(\theta, \phi) = \sum_{n=1}^{N} \exp(j\psi_n) \tag{3.14}$$

where θ and ϕ are angular coordinates describing the pattern and ψ_n, is the phase of the signal at the $n-th$ element measured with respect to some reference. The phase ψ_n, is a function of θ and ϕ and the location of the $n-th$ element on the aperture. The N elements may be located on any type of aperture.



(a)

Figure 8. *Statistical Arrays* - Geometry of an M by M element array arranged on a square grid. Angular coordinates are also shown [4].

If elements are removed from the array the field intensity pattern may be written [4]

$$E(\theta, \phi) = \sum_{n=1}^{N} F_n \exp(j\psi_n) \tag{3.15}$$

where F_n, is either zero or unity according as the element is removed or left in place. The quantity F_n thus has only the values of 0 and 1. In a statistically designed array, F_n is selected randomly and independently from element to element by a random number generator in such a way that its average value (ensemble average over many selections) is [4]

$$\overline{F_n} = A_n \tag{3.16}$$

where A_n, is the amplitude of the excitation that would normally be applied to the $n-th$ element if it were designed with an amplitude taper across the aperture. The field intensity of the **equivalent amplitude-tapered array** used as the **model** is [4]

$$E_0(\theta, \phi) = \sum_{n=1}^{N} A_n \exp(j\psi_n) \tag{3.17}$$

The radiation pattern of (3.15) is statistical since F_n is statistical. By the Central Limit Theorem of statistics, the distribution of the quantity $E(\theta, \phi)$ for a given θ and ϕ will be approximately Gaussian, if N is large.

The mean of the statistical pattern of (3.15) is found using the fact that the mean of the sum is the sum of the means,

$$\overline{E(\theta, \phi)} = \sum_{n=1}^{N} \overline{F_n} \exp(j\psi_n) = \sum_{n=1}^{N} A_n \exp(j\psi_n) = E_0(\theta, \phi) \tag{3.18}$$

Thus the mean or average pattern is identical with the field-intensity pattern of the amplitude tapered array used as the model. This array factor (3.17) will be referred to as the *model* array factor. The coefficients A_n are selected by standard design procedures [10]-[11] for amplitude-tapered arrays to obtain a desired mean pattern. Since the quantities A_n, are the mean values of a random variable with values 0 and 1, we must always have $0 \leq A_n \leq 1$. This may be obtained by properly scaling the original amplitude taper of the model-array design.

The square of the field-intensity pattern is the power pattern and is written

$$\begin{aligned} |E(\theta, \phi)|^2 &= E(\theta, \phi) \cdot E^{\star}(\theta, \phi) \\ &= \sum_{n=1}^{N} \sum_{m=1}^{N} F_n F_m \exp(j(\psi_m - \psi_n)) \end{aligned} \tag{3.19}$$

where $E^{\star}(\theta, \phi)$ denotes the complex conjugate. There is a theorem which states that the mean of a product of statistically independent random variables is equal to the product of the means of those random variables. The variables F_m and F_n in (3.19) are independent if

and only if $m \neq n$. If $m = n$ they are of course identical. Therefore the double summation is separated into terms with $m = n$ and terms with $m \neq n$, and the average is taken as follows:

$$\overline{|E(\theta,\phi)|^2} = \overline{\sum_n F_n^2} + \overline{\sum_n \sum_m F_n F_m \exp\left(j\left(\psi_m - \psi_n\right)\right)}\Bigg|_{m\neq n} \tag{3.20}$$

Since the values of F_n, are either 0 or 1, $F_n^2 = F_n$, and the first summation of (3.20) becomes

$$\overline{\sum_n F_n^2} = \sum_n \overline{F_n^2} = \sum_n \overline{F_n} = \sum_n A_n \tag{3.21}$$

Using the theorem mentioned above, the second summation of (3.20) involving terms with $m \neq n$ becomes

$$\sum_n \sum_m A_n A_m \exp\left(j\left(\psi_m - \psi_n\right)\right)\Bigg|_{m\neq n} \tag{3.22}$$

This is simply the power pattern corresponding to the model-array pattern $E_0(\theta,\phi)$ of (3.17), except that the terms with $m = n$ are missing. When these terms are restored and subtracted from the result, the following is obtained

$$\begin{aligned} \overline{|E(\theta,\phi)|^2} &= \sum_n A_n + \overline{|E_0(\theta,\phi)|^2} - \sum_n A_n^2 \\ &= \overline{|E_0(\theta,\phi)|^2} + \sum_n A_n\left(1 - A_n\right) \end{aligned} \tag{3.23}$$

where $\overline{|E_0(\theta,\phi)|^2}$ is the power pattern of the model array with "equivalent" amplitude taper A_n, applied to each element.

The fraction of elements removed is controlled by the amplitude taper chosen for the model array. The exact number of elements after the elimination procedure is

$$N_E = \sum_n F_n \tag{3.24}$$

On the average, the number of elements left in the array is [4]

$$\overline{N_E} = \sum_{n=1}^{N} \overline{F_n} = \sum_{n=1}^{N} A_n = NA_n \leq N \tag{3.25}$$

and the variance is

$$\sigma_N^2 = \overline{N_E^2} - \left(\overline{N_E}\right)^2 = \sum_{n=1}^{N} A_n\left(1 - A_n\right) \tag{3.26}$$

If it is assumed that the degree of element removal is such that the omnidirectional component [second term of (3.23)] of the power pattern is larger than the sidelobes of the model amplitude-tapered array pattern, then the average value of the sidelobes is

$$\begin{aligned} \textit{average statistical sidelobes} &= \overline{SL} \\ &= \sum_{n=1}^{N} A_n - \sum_{n=1}^{N} A_n^2 \end{aligned} \tag{3.27}$$

Substituting $\overline{N_E}$ from (3.25)

$$\overline{SL} = \overline{N_E} - \overline{N_E^2}/G_a = \overline{N_E}\left(1 - \frac{\overline{N_E}}{N_{\rho_a}}\right) \tag{3.28}$$

where ρ_a is the aperture efficiency of the model amplitude taper given by A_n ([4]). Since ρ_a is of the order of unity, (3.28) states that the average sidelobe level approaches $\overline{N_E}$, the number of elements left within the array, when the fraction of elements removed $(1 - N_E/N)$ is large. The average sidelobe level relative to the peak value of the main beam after the elimination of elements is

$$\begin{aligned} \textit{average relative sidelobe} &= \\ &= \rho \approx \frac{\sum_n A_n(1-A_n)}{\overline{|E(0,0)|^2}} \end{aligned} \tag{3.29}$$

From (3.23),

$$\begin{aligned} \overline{|E(0,0)|^2} &= \left(\sum_n A_n\right)^2 + \sum_n A_n(1-A_n) \\ &\approx \left(\sum_n A_n\right)^2 \end{aligned} \tag{3.30}$$

Therefore, (3.29) becomes

$$\rho \approx \frac{\sum_n A_n(1-A_n)}{\overline{|E(0,0)|^2}} = \frac{1 - \frac{\sum_n A_n^2}{\sum_n A_n}}{\sum_n A_n} \tag{3.31}$$

and after elaboration

$$\rho \approx \frac{1 - \frac{\overline{N_E}}{N_{\rho_a}}}{\overline{N_E}} \tag{3.32}$$

and

$$\rho \approx \frac{1}{\overline{N_E}} \; for \; \frac{\overline{N_E}}{N} \ll 1 \tag{3.33}$$

where G_a, is the gain of the model amplitude-tapered array, G_s, is the average gain of the statistical designed density-tapered array. If one starts with an N element array and remove elements according to the above statistical procedure, the average number of elements that remain is given by (3.25). The N-element array is said t o be "thinned" and the degree of thinning, or percentage of elements removed, is

$$degree\,of\,thinning = 100\left(1 - \frac{\overline{N_E}}{N}\right)\% \tag{3.34}$$

A given amplitude taper therefore has a certain natural degree of thinning. If it is desired to remove more elements than the natural number, so that the number remaining $N_r = k\overline{N_E}$, where $k < 1$, an examination of (3.25) shows that this may be accomplished by multiplying the amplitudes A_n, by the factor k ([4]). Thus

$$N_r = k\overline{N_E} = \sum_{n=1}^{N} kA_n \tag{3.35}$$

The above analysis can be repeated for $N_r = k\overline{N_E}$ elements. In a statistically designed array F_n, is selected randomly and independently from element to element so that its ensemble average is $\overline{F_n} = kA_n$. When $k = 1$, the array is said to be "naturally" thinned. The average field intensity (ensemble average over many selections) is

$$\overline{|E(\theta,\phi)|} = kE_0(\theta,\phi) \tag{3.36}$$

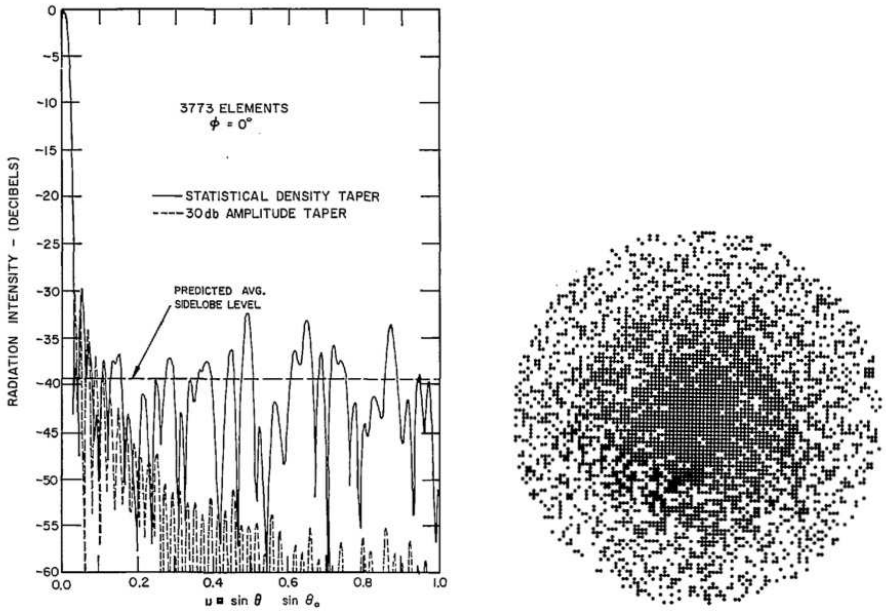
which is similar to that of the model amplitude-tapered array. The average power pattern, or radiation pattern is

$$\overline{|E(\theta,\phi)|^2} = k^2\overline{|E_0(\theta,\phi)|^2} + \sum_{n=1}^{N} kA_n(1 - kA_n) \tag{3.37}$$

The first term of the radiation pattern is proportional to the radiation pattern of the model amplitude-tapered array. When $k = 1$, it is equal to it, corresponding to a naturally thinned array. The second term is independent of angle. Thus the average statistical sidelobes which dominate the pattern outside the vicinity of the main beam (and the near-in sidelobes) may be written [4]

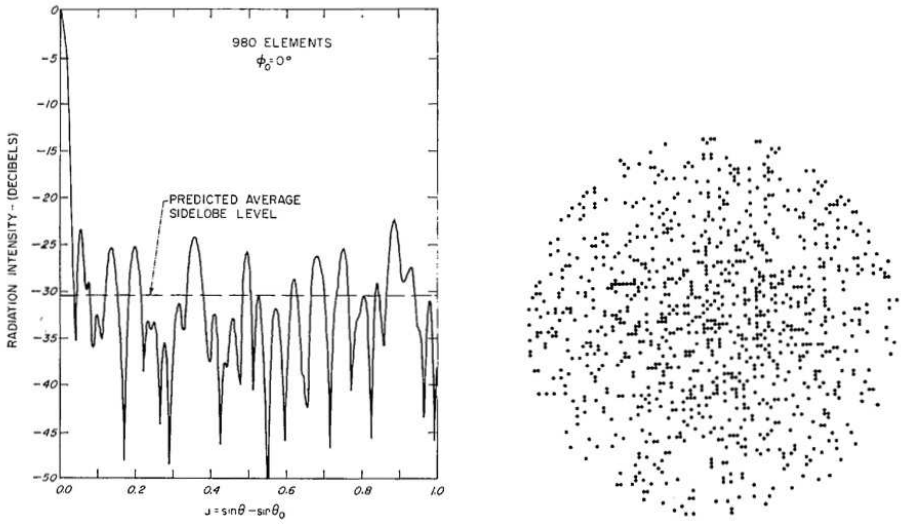
$$\overline{SL} = \sum_{n=1}^{N} kA_n(1 - kA_n) \tag{3.38}$$

The equation (3.38) shows that the statistical sidelobes of a thinned array are determined by the model aperture amplitude distribution A_n, and by k, the factor which determines the number of elements removed. The near in sidelobes are also determined by A_n.



(a) (b)

Figure 9. *Statistical Arrays* - In (a) the solid curve is the computed radiation pattern of a statistically designed array naturally thinned using as a model the $30dB$ Taylor circular aperture distribution whose pattern is shown by the dashed curve. In (b) the locations of the elements for the $30dB$ design with natural thinning [4].



(a) (b)

Figure 10. *Statistical Arrays* - In (a) there is the computed radiation pattern of a statistically designed array using as a model the $25dB$ Taylor design but with approximately 90 per cent of the elements removed. In (b) the corresponding locations of the elements [4].

3.4 Optimization Algorithms Approach

3.4.1 Introduction

Thinning an array means turning off some elements in a uniformly spaced or periodic array to create a desired amplitude density across the aperture. An element connected to the feed network is *on,* and an element connected to a matched or dummy load is *off*. Thinning an array to produce low sidelobes is much simpler than the more general problem of non uniform spacing the elements. Non uniform spacing has an infinite number of possibilities for placement of the elements. Thinning has 2^Q possible combinations, where Q is the number of array elements. If the array is symmetric, then the number of possibilities is substantially smaller. Thinning may also be thought of as a quantized amplitude taper where the amplitude at each element is represented by one bit. Thinning a large array for low sidelobes involves checking a rather large number of possibilities in order to find the best thinned aperture. Exhaustive checking of all possible element combinations is only practical for small arrays. Most optimization methods (including down-hill simplex and conjugate gradient) are not well suited for thinning arrays. They can only optimize a few continuous variable sand get stuck in local minima. Also, these methods were developed for continuous parameters, where as the array thinning problem involves discrete parameters. Dynamic programming can optimize a large parameter set (many elements), but it is vulnerable to local minima.

Simulated annealing ([38])**, genetic algorithms** ([18]), **ant colony** ([39]) and other **stochastic algorithms** ([14][28][27]) are optimization methods that are well suited for thinning arrays. There is no limit to the number of variables that can be optimized. Although quite slow, these algorithms can handle very large arrays. These methods are global in that they have random components that test for solutions outside the current minimum, while the algorithm converges. The global nature of the algorithms and the lack of derivative information causes them to converge very slowly compared to other non-global methods. The purpose of this method is to find a thinned array that produces the *lowest maximum relative sidelobe level* (*rsll*).

3.4.2 Genetic Algorithm [18]

3.4.2.1 GA - Algorithm

A genetic algorithm ([14]) is used to numerically optimize both linear and planar arrays. Genetic algorithms are modeled after genetic recombination and evolution. The

algorithms encode parameters in binary strings called genes and perform the genetic operations of reproduction, crossover, natural selection, and mutation to arrive at the optimum solution. These algorithms arrive at better thinning configurations for arrays than previous optimization attempts or statistical attempts. Other optimization methods cannot be applied to large arrays, while statistical methods cannot find optimum solutions ([14][18]).

The goal of the genetic algorithm is to find a set of parameters that minimizes the output of a function. Genetic algorithms differ from most optimization methods, because they have the following characteristics

1. They work with a coding of the parameters, not the parameters themselves.

2. They search from many points instead of a single point.

3. They don't use derivatives.

4. They use random transition rules, not deterministic rules.

Fig. 11 is a flow chart of a genetic algorithm. Steps are labeled as A through F for easy reference.

Values for all the parameters are represented by a binary code (step A). Each encoded parameter is placed side by side to form a long binary string called a gene. Every gene has an associated output corresponding to the function evaluated at the quantized parameters. Thus, the genetic algorithm has a finite, but possibly very large, number of parameter combinations to check. A gene with N, $B-bit$ parameters has a total of 2^{NB} possible genes. If the parameters are continuous, then the genetic algorithm limits performance due to quantization errors associated with the binary encoding of the parameters. On the positive side, genetic algorithms are ideally suited for optimization of functions with discrete parameters.

A thinned array has discrete parameters. One bit represents the element state as $on = 1$ or $off = 0$. For example, a six element array may be represented by 101101, where elements 2 and 5 are turned off. Assuming the linear array is symmetric about its center allows the $2N$ element array to be represented by a gene with N bits. Our six-element array example can then be represented by the gene 101. The fitness associated with this gene is the maximum relative sidelobe level (rsll) of its associated far-field pattern. The function in this paper is the relative far-field pattern of an array of point sources. Its output to be minimized is the maximum rsll. The parameters affecting the output are whether an antenna element is on or off ([18]).

Genetic algorithms model genetic recombination and evolution in nature. As in nature, the gene is the basic building block. Genetic algorithms start with a random sampling of the output space. Many of the genes from this list have terrible maximum rsll's. Genes that produce a superior output survive, while genes that produce a weak output die off.

Usually initial population and genes are randomly generated (step B). Then genes are ranked from best to worst, (step C) according to their rsll. The most common suitability criterion is to discard (step D) the genes with the worst performance. After this "natural" selection takes place, the genes mate (step E) to produce offspring. Mating takes place by pairing the surviving genes. Once paired, their offspring consist of genetic material from both parents. One last step is to introduce a random mutation in the list of genes (step F). A mutation changes a zero to a one or a one to a zero. The mutation helps the algorithm avoid a local minimum. over again with the parents and the offspring (step C). Mutation usually doesn't improve the solution. It is a very necessary part of genetic algorithms, though. Without it, genetic algorithms are more likely to get stuck in a local minimum. Natural selection, mating, and mutation will take place with these genes. The algorithm continues this process until a suitable stopping point is reached. Eventually, all the genes will be identical except for the single mutated gene ([18]).

3.4.2.2 GA Optimization for the design of Linear Array

For **linear array synthesis problem**, each gene has an associated rsll calculated from

$$FF(u) = \max \left| 2\sum_{n=1}^{N} a_n \frac{\cos(2\pi ndu + \delta_s)}{FF_{max}} elpat(u) \right| \qquad \frac{c_0}{2Nd} \leq u \leq 1 \tag{3.39}$$

where

- $2N$ is the number of elements in the array
- a_n is the amplitude weight of element $n = \begin{cases} 1 & on \\ 0 & off \end{cases}$
- d is the spacing between elements
- $u = \cos(\phi)$
- ϕ is the angle measured from line passing through antenna elements
- $\delta_s = -2\pi du_s$ is the steering phase

- $elpat\,(u)$ is the element pattern
- c_0 is constant
- FF_{max} is the peak of main beam

The region of u for which FF is valid excludes the main beam. The first null for a uniform array occurs at $u = \frac{1}{2Nd}$. Thinning an array causes the null to move, so a constant, c_0, is needed in the formula to adjust for the first null location ([18]).

3.4.2.3 GA Optimization for the design of Planar Array

For **planar $N \times M$ element array synthesis problem**, each gene has an associated rsll calculated from

$$FF\,(\theta, \phi) = \sum_{m=1}^{M}\sum_{n=1}^{N} a_{mn} \cos\left[(2m-1)\,\pi d_y \sin\theta \cos\phi\right] \times \cos\left[(2n-1)\,\pi d_x \sin\theta \sin\phi\right] \quad (3.40)$$

where

- M is the number of elements in the array in the $y-$direction
- N is the number of elements in the array in the $x-$direction
- d_x is the spacing between elements in the $x-$direction
- d_y is the spacing between elements in the $y-$direction

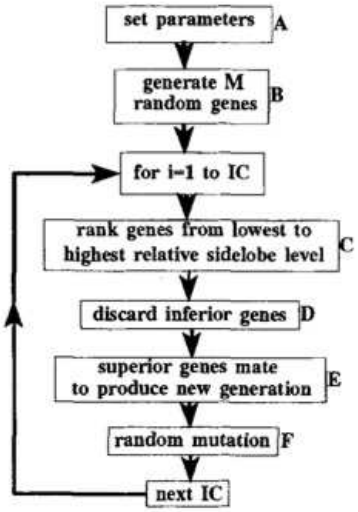


Figure 11. *Thinned Arrays with Genetic Algorithms* - Flow chart of a genetic algorithm [18].

3.4.3 Simulated Annealing [38]

In the past few years, three-dimensional (3-D) acoustic imaging has been one of the main innovations in both underwater and medical applications. To obtain 3-D electronic focusing and beamforming (i.e., 3-D imaging capabilities), a two-dimensional (2-D) aperture should be used to generate and/or receive an acoustic field. When such an aperture is spatially sampled, the adoption of a 2-D-array antenna (also called planar array) is assumed. To prevent grating lobes (i.e., aliasing effects due to spatial under sampling), a half-wavelength ($\lambda/2$) spacing between the elements of the array should not be exceeded. At the same time, in order to obtain a fine lateral resolution, the array should have a wide spatial extension. The $\lambda/2$-condition with the fine resolution requirement often will result in a 2-D array composed of some thousands of elements. As an acquisition channel is associated with each array element, the cost of a 2-D array of this type (i.e., a fully sampled array) is prohibitive ([38]).

One of the most promising approaches to reducing the number of array elements (for both linear and planar arrays) is based on the concept of aperiodic arrays. A fully sampled array is thinned by removing a fraction of the original set of elements, thus obtaining a sparse array. Aliasing effects are avoided because there are no periodicities at the positions of the sparse array elements. The main drawback of the thinning operation is an often unacceptable high level of the side lobes present in the beam pattern (BP) (i.e., the spatial response of the array). Therefore, the thinning should be an optimization operation aimed at reducing the number of elements, while maintaining adequate BP properties for the addressed application ([38]).

In this method, a synthesis method is proposed that is aimed at designing a sparse and aperiodic array to be used as a planar antenna for a narrow-band beamforming processor, mainly for underwater applications. The purpose of the method is to minimize the number of elements able to generate a BP that fulls some a priori fixed constraints by acting on the positions and weights of the elements. The stochastic method proposed in this paper is based on simulating annealing (SA) and is an evolution of the method for the synthesis of linear arrays. The main features, which represent improvements over other methods, are the following:

- very large 2-D arrays can be modeled
- both weights and positions can be optimized
- the number of elements can be minimized

- asymmetric arrays can be synthesized, thus a larger number of degrees of freedom can be exploited
- the overall extent of the 3-D BP can be considered.

3.4.3.1 SA - Algorithm

Initially, simulated annealing (SA) aimed to simulate the behavior of the molecules of a pure substance during the slow cooling that results in the formation of a perfect crystal (minimum energy state). 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 each variable that affects an energy function to be minimized. This function is called the energy function, $f(\mathbf{Y})$, $\mathbf{Y}$ being the vector of state variables. The algorithm is iterative: at each iteration, a small random perturbation is introduced into the current state configuration $\mathbf{Y}_l$ (l being the iteration). If the new configuration, $\mathbf{Y}_n$, causes the value of the energy function to decrease, it is accepted ($\mathbf{Y}_{l+1} = \mathbf{Y}_n$). Instead, if $\mathbf{Y}_n$ causes the value of the energy function to increase, it is accepted with a probability dependent on the system temperature, in accordance with the Boltzmann distribution. The higher the temperature, the higher the probability that the state configuration which caused the energy function to increase will be accepted. In short, the probability that $\mathbf{Y}_n$may be accepted as a new configuration, $Pr\{\mathbf{Y}_{l+1} = \mathbf{Y}_n\}$, can be expressed as:

$$Pr\{\mathbf{Y}_{l+1} = \mathbf{Y}_n\} = \begin{cases} \frac{\exp(f(\mathbf{Y}_l) - f(\mathbf{Y}_n)),}{kT} & if\ f(\mathbf{Y}_l) < f(\mathbf{Y}_n) \\ 1, & otherwise \end{cases} \tag{3.41}$$

where k is the Boltzmann constant and T is the system temperature. As the iterations go on, the temperature T is gradually lowered, following the reciprocal of the logarithm of the number of iterations, until the configuration freezes in a certain final state. This method is very useful to minimize an energy function that has many local minima as, thanks to its probabilistic nature, it represents a notable improvement over classical methods of local descent (despite being computationally demanding). The repetition of the process, using different initial configurations, increases the reliance on the quasi-optimality of results, even though a full optimality cannot be proved ([38]).

3.4.3.2 Optimization Procedure for Linear and Planar Arrays

The conceptual mechanisms and the peculiarities of the SA implementation that have been applied to obtain an efficient minimization of the energy function are presented. Fig. 12 shows a flow-chart of the optimization procedure. One can start the synthesis

by considering a fully sampled array, i.e., a planar array composed of N $\lambda/2$-equispaced elements. Then, according to the process behavior, the elements are divided into two sets: an active set (i.e., having weights different from zero) and an inactive set (i.e., having weights equal to zero). The number of active elements is M and the relation $M \leq N$ is always verified. The initial temperature, T^{start}, is chosen high enough and such that the first configuration perturbations may almost always be accepted, even though they lead to a sharp increase in energy. When one starts the iteration l, one chooses an element i randomly (both active and inactive elements are visited according to a random sequence that does not include any further visit to the same element before all the elements have been visited once). If the chosen element is active, one can move it to an inactive condition by following fixed death probability; whereas, if the chosen element is inactive, one can activate it (with a random weight) by following a resurrection probability. On the basis of the temperature $T(l)$, such state transitions can be accepted or not. If one of these transitions is accepted, the number of active elements M must be updated. If the element i is active and its death does not occur, the weight w_i is perturbed and, on the basis of the temperature $T(l)$, the perturbation can be accepted or not ([38]).

During the optimization procedure, a constraint is imposed to limit to low values the current taper ratio (CTR), which is the ratio between the maximum and minimum weight coefficients. This constraint makes it possible to avoid any consequences of possible unforeseen occurrences regarding the elements with the largest weight coefficients. To limit the CTR value, one should perturb each weight coefficient in a random and continuous way; but one should make sure that the coefficient value is included in an a priori fixed range $[w_{min}; w_{max}]$.

The number of iterations, l, is increased every time all the N elements have been perturbed once. The process terminates when a state of persistent block (freezing) is reached due to the slow temperature reduction. Alternatively, according to previous experiences, one can perform a number of iterations which is large enough to ensure that a block state will be reached ([38]).

Owing to the probabilistic nature of SA, different temperature schedulings and random initial configurations may lead to different final results. However, if a logarithmic scheduling is chosen, almost all the process runs give slightly different results in terms of both energy values and array characteristics. This means that the resulting array configuration is stable and close to the optimal one ([38]).

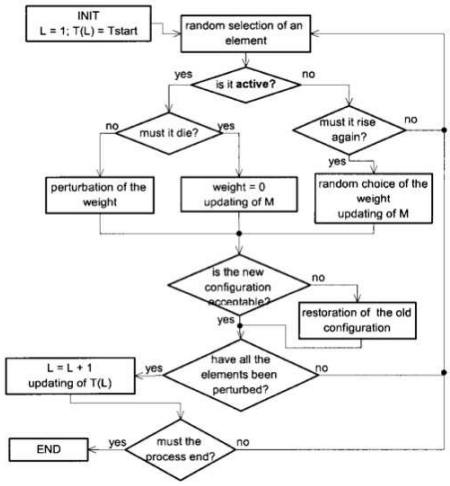


Figure 12. *Thinned Arrays with Simulated Annealing* - Flow-chart of the optimization procedure [38].

3.4.4 Ant Colony [39]

It is known that with aperiodic arrays it is possible to get low sidelobe levels in all directions or only at some regions. The advantage of uniform amplitude excitation is clear from the point of view of the feeding network. However, the synthesis problem is complex and cannot be solved with analytical methods. Therefore, global optimization methods are a good option to afford these problems. Among them, genetic algorithms (GA), particle swarm optimization (PSO), and simulated annealing (SA) have already been used in array synthesis for different requirements.The purpose of using an algorithm based on ant colony optimization (ACO) to synthesize thinned arrays with low SLL without pretending to compare ACO neither with PSO nor with GA or any other optimization technique. The main advantage of ACO algorithms could come from the implicit local search that they perform and also from their simplicity. Of course, it still depends on the problem and in the particular implementation of the algorithm, because all these algorithms have parameters heuristically chosen that can have a strong influence on the algorithm behavior for a particular problem. Moreover, the same algorithm with same parameters applied to the same problem can find different solutions in the same number of iterations ([39]). This is a consequence of their intrinsic randomness.

3.4.4.1 ACO - Algorithm

The ACO is a global search optimization method that is based on the behavior of ant colonies in obtaining food and carrying it back to the nest. It is a “short path” based

algorithm. When the ants walk around in search for food, they give off pheromone on the ground. Ants select paths according to pheromone level. The shorter the trail from the nest to the food source, the higher the pheromone level and thus the probability of ants choosing that path. Furthermore, ants use this to remember the path to the food, and it helps to add new ants to that trail, getting more food from that place to the nest. These pheromone also evaporate slowly with time. This decreases the probability of taking paths toward finished food sources ([39]).

The implementation of an algorithm based on this natural behavior is well suited for discrete problems (although codification using real numbers is also possible). In our case, we have implemented ant colony procedure as follows (having two working modes: forward when the ants search food, and backward when they carry the food to home).

Define pheromone concentration function and desirability function and choose parameters: Number of ants, α, β, ...

```
Initialize I_1,I_2,...,I_n
  For each iteration
   For each ant
    For each adjoining node
      Calculate pheromone function and desirability
    End for
    Chose one node
    If food is found
      Mode 0:  Come back home
    Else-if ant is at home
      Mode 1:  Searching food
    End if
      Update pheromone
   End for
  End for
Solution is I_1,I_2,...,I_n with best result
```

We have N bits, thus corresponding to an N-dimensional space of solutions. In this case, every ant means an array solution, i.e., a vector with N bits. Ants describe paths that are divided into nodes. They move from one node to another through the N-dimensional space of solutions by checking the desirability and the pheromone concentration level of their neighboring nodes before making a probabilistic decision among all of them. A neighboring node is calculated by toggling the state of only one element of the array. This

means that every ant has N neighboring nodes and has to decide which one among them to move toward, in a probabilistic manner. One of the most common and suitable form for combining the two parameters used to calculate the probability of choosing one node in a path is

$$p_{i,j}(t) = \frac{[\tau_j(t)]^{\alpha} \cdot [\eta_j]^{\beta}}{\sum_{l \in \theta_i} [\tau_j(t)]^{\alpha} \cdot [\eta_j]^{\beta}} \tag{3.42}$$

where $p_{i,j}$ is the is the probability of choosing node j at iteration t from node i, $\tau_j(t)$ is the pheromone level of node j at iteration t, η_j is the desirability of node j, α is the parameter controlling the relative importance of pheromone in the decision process while β does the same with the desirability. θ_i is the set of nodes available at decision point i ([39]).

There are different implementations of the function τ_j. This function controls the change in pheromone level in nodes with time. This includes the increase when ants visit that node but also the evaporation with time. We can use, for example,

$$\tau_j(t+1) = \tau_j(t) + \Delta\tau_j(t) - d(t) \tag{3.43}$$

where $\Delta\tau_j(t)$ is is the pheromone addition on node j, and $d_j(t)$ is the pheromone persistence

$$d(t) = \begin{cases} \rho, & if\ mod\left(\frac{t}{\gamma}\right) = 0 \\ 0, & if\ mod\left(\frac{t}{\gamma}\right) \neq 0 \end{cases} \tag{3.44}$$

where γ is the period of pheromone elimination, and is the coefficient of pheromone elimination by period ([39]).

3.4.4.2 Optimization Procedure for Linear and Planar Arrays

There are different methods to synthesize a suitable solution using aperiodic arrays. The most common one entails varying the position of the elements symmetrically. However, when the number of array elements is large, another option is to use the concept of thinned arrays. In this work the positions of the elements will be fixed, but with each element being able to present two states: "on" (when the element is fed) and "off" (when the element is passively terminated in an impedance equal to the source impedance of the fed elements) ([39]).

For a **linear array** where there are $2N$ elements placed symmetrically along the $x-$axis, the far field pattern is

$$AF(\phi) = 2\sum_{n=1}^{N} I_n \cos[\pi(2n-1) \cdot \cos(\phi)] \tag{3.45}$$

where I_n is the excitation amplitude of the $n-$th element. In our case, I_n is 0 if the state of the $n-$th element is "off" and 1 if it is "on". The distance between elements is 0.5λ and all them have identical current phase. In this case, we search the lowest value of SLL with isotropic elements. The desirability $\eta_j$is defined as the absolute value of the normalized SLL (dBs)

$$\eta_j = |SLL\,(dB)| \tag{3.46}$$

For a **planar array structure** of elements, the array factor is given by

$$\begin{aligned} AF\,(\theta,\phi) = \quad & 4\sum_{n=1}^{N}\sum_{m=1}^{M} I_{nm}\cos\left[\pi\,(2n-1)\cdot\sin\,(\theta)\cos\,(\phi)\right]\cdot \\ & \cos\left[\pi\,(2m-1)\cdot\sin\,(\theta)\sin\,(\phi)\right] \end{aligned} \tag{3.47}$$

Therefore, the objective is to find out which array elements should be enabled or disabled ($I_{nm} = 1$ or $I_{nm} = 0$) to get the desired radiation pattern characteristics. In this section, we will deal with the design of a thinned planar array. The SLL level will be checked in the two main planes of the array ([39])

$$\eta_j = \min\left(\left|SLL_{\phi=0^\circ}\,(dB)\right|, \left|SLL_{\phi=90^\circ}\,(dB)\right|\right) \tag{3.48}$$

3.5 Differences Sets [5][19]

3.5.1 Introduction

Tradition **filled phased arrays** have an element placed in every location of a uniform lattice with half wavelength spacing between the lattice points. **Massively thinned** arrays have **fewer than half the elements** of their filled counterparts. Such drastic thinning is normally accompanied by loss of sidelobe control. This chapter describes a class of massively thinned linear and planar arrays that show well behaved sidelobes in spite of massive thinning. Isophoric arrays derive their sidelobe control from a deterministic placement of elements that achieves a uniform weighting of spatial coverage. The term **isophoric** is based on the Greek roots that denote **uniform weight** [5][19].

For a given **aperture size**, massive thinning offers reductions in element count, cost, weight, power consumption, and heat dissipation, albeit with an attendant reduction in antenna gain.

For a given **element count**, thinning offers narrowed beamwidth by making larger apertures possible.

Rather than using a search algorithm, the approach in this chapter attacks the sidelobe control problem directly by applying the properties of difference sets, a topic from combinatorial mathematics, to the placement of antenna elements within a regular lattice. These deterministic placements create an isophoric array with attendant uniformity of spatial coverage. The uniformity consistently produces, with no searching required, a reduction in peak sidelobe level (PSL) when compared to random element placement [5][19].

More specifically, in any linear array of aperture V half wavelengths, the Nyquist sampling theorem shows that the array power pattern can be completely specified from $2V$ uniformly spaced samples of the pattern. In an isophoric array, the even-numbered samples will necessarily be "locked" to a constant value less than $1/K$ times the main-beam peak, where K is the number of elements in the thinned array. While the odd-numbered samples are not so constrained, the net effect is to produce patterns with much lower PSLs than are typical with cut-and-try random placement [5][19].

Isophoric designs apply to linear or planar arrays, whether large or small. While this paper focuses on arrays with 50% thinning, isophoric arrays include arrays thinned to the extent that the number of elements is approximately the square root of the number of elements in their filled counterparts.

Some proposed modern arrays use tens, hundreds, or even thousands of elements

combined with digital beam forming (DBF) to produce multiple simultaneous beams. For these arrays, this paper shows how a filled DBF-based array can be operated as two "interwoven" isophoric arrays, thereby reducing the computational complexity in each. In addition, by simple cyclic shifting of the element assignments overtime, it is possible to produce power patterns for which the entire sidelobe region is approximately a constant value of $\frac{1}{2}K$ relative to the main beam, where K is the number of elements in the original filled array. In other words, the "peaks" in the sidelobe region virtually vanish [5][19].

3.5.2 Notation

This section introduces some definitions and notation needed in later sections.

The **array factor** for a **linear array** of identical isotropic radiators is defined as

$$f(u) = \sum_{m=0}^{V-1} a_m \exp(j2\pi m x_0 u) \tag{3.49}$$

where $a_m = 1$ if an element exists at distance mx_0 wavelengths from the origin and $a_m = 0$, otherwise $u = \sin(\theta)$ is the commonly used direction parameter with θ measured off of a normal to the array, and the lattice has V possible element locations numbered 0 to $V-1$, uniformly spaced at x_0 intervals of wavelengths.

The corresponding **array factor** for a **planar array** on a uniform x, y lattice with x_0, y_0 wavelength spacing is

$$f(u,v) = \sum_{m=0}^{V_x-1} \sum_{n=0}^{V_y-1} a_{m,n} \exp(j2\pi(mx_0 u + ny_0 v)) \tag{3.50}$$

where $a_{m,n} = 1$ if an element exists at location (mx_0, ny_0) wavelengths relative to the origin and $a_{m,n} = 1$, otherwise $u = \sin(\theta)\cos(\phi)$, and $v = \sin(\theta)\sin(\phi)$ are the commonly used direction parameters and the array lattice has $V = V_x V_y$ possible element locations numbered $(0,0)$ to $(V_x - 1, V_y - 1)$. The angle θ is measured off of a normal to the array plane and ϕ is measured off of the x-axis of the array plane.

To simplify both expressions, steering angles have, without loss of generality, been set to zero. As usual, applying an appropriate linear phase variation across the elements will allow the main beam to be steered.

Array power patterns for **linear and planar arrays** are represented as

$$\begin{aligned} ff^*(u) &= f(u) \cdot f^*(u) = |f(u)|^2 \\ ff^*(u,v) &= f(u,v) \cdot f^*(u,v) = |f(u,v)|^2 \end{aligned} \tag{3.51}$$

Since the array factor and power pattern are **periodic** as well as **band limited**, a finite number of samples, taken from a single period, are sufficient to regenerate the entire factor or pattern over all u. The derivations of the sampling theorem for $f(u)$ and $ff^*(u)$ are straightforward. For linear and planar arrays are ([5][19]):

$$f(u) = \sum_{n=0}^{V-1} f\left(\frac{n}{Vx_0}\right) \frac{\sin\left[\pi V x_0 \left(u - \frac{n}{Vx_0}\right)\right]}{V \sin\left[\pi x_0 \left(u - \frac{n}{Vx_0}\right)\right]} \tag{3.52}$$

$$ff^*(u) = \sum_{n=0}^{2V-1} ff^*\left(\frac{n}{2Vx_0}\right) \frac{\sin\left[2\pi V x_0 \left(u - \frac{n}{2Vx_0}\right)\right]}{2V \tan\left[\pi x_0 \left(u - \frac{n}{2Vx_0}\right)\right]} \tag{3.53}$$

The form (3.52) for $f(u)$ is valid for V an odd integer. When V is even, the sine function in the denominator must be replaced by a tangent function. Note that while it takes $2V$ samples to specify the power pattern $ff^*(u)$, it takes only V samples to specify the array factor $f(u)$. The reason is that the samples of $f(u)$ are complex, while those of $ff^*(u)$ are real. Having both a real and imaginary part, each sample of $f(u)$ contains twice the information of $ff^*(u)$ sample. Thus, $f(u)$ both $ff^*(u)$ and are completely specified by $2V$ numbers. The sampling theorem shows that at least $2V$ numbers are required to specify either $f(u)$ or $ff^*(u)$. Conversely, both have, at most, $2V$ **degrees of freedom** in that one can arbitrarily specify only $2V$ sample points in the power pattern. In particular, **control over the power pattern** is equivalent to and limited to control of the $2V$ sample points ([5][19]).

The corresponding forms for planar arrays are

$$f(u,v) = \sum_{m=0}^{V_x-1} \sum_{n=0}^{V_y-1} f\left(\frac{m}{V_x x_0}, \frac{n}{V_y y_0}\right) \frac{\sin\left[\pi V_x x_0 \left(u - \frac{m}{V_x x_0}\right)\right]}{V_x \sin\left[\pi x_0 \left(u - \frac{m}{V_x x_0}\right)\right]} \frac{\sin\left[\pi V_y y_0 \left(v - \frac{n}{V_y y_0}\right)\right]}{V_y \sin\left[\pi y_0 \left(v - \frac{n}{V_y y_0}\right)\right]} \tag{3.54}$$

$$ff^*(u,v) = \sum_{m=0}^{2V_x-1} \sum_{n=0}^{2V_y-1} f\left(\frac{m}{2V_x x_0}, \frac{n}{2V_y y_0}\right) \frac{\sin\left[2\pi V_x x_0 \left(u - \frac{m}{2V_x x_0}\right)\right]}{2V_x \tan\left[\pi x_0 \left(u - \frac{m}{2V_x x_0}\right)\right]} \frac{\sin\left[2\pi V_y y_0 \left(v - \frac{n}{2V_y y_0}\right)\right]}{2V_y \tan\left[\pi y_0 \left(v - \frac{n}{2V_y y_0}\right)\right]} \tag{3.55}$$

3.5.3 Difference Sets

Difference sets and their associated block designs are a branch of combinatorial theory. This section contains a brief introduction to the theory and properties of difference sets [5][19].

By definition, a (V, K, Λ) difference set is a set of K unique integers

$$D = \{d_0, d_1, ..., d_{K-1}\}\,,\ with\ 0 \leq d_i \leq (V-1) \tag{3.56}$$

such that for any integer $1 \leq \alpha \leq (V-1)$

$$d_i - d_j = \alpha mod\,(V)\,,\ i \neq j \tag{3.57}$$

has exactly Λ solution pairs (d_i, d_j) from the set $\{D\}$, where $mod\,V$ means the difference sets is to be taken modulo V.

While three parameters are used to describe a difference set, only two of the parameters are independent. Since there are $K\,(K-1)$ possible differences $(d_i - d_j)$ with $i \neq j$ and since each of the $(V-1)$ possible unique differences is to appear exactly Λ times, it follows that

$$K\,(K-1) = \Lambda\,(V-1) \tag{3.58}$$

As an example, consider the above set $D_2 = \{0, 3, 5, 6\}$ for which $V = 7, K = 4, \Lambda = 2$. As shown in Table I, each of the $V - 1 = 6$ possible unique differences appears exactly $\Lambda = 2$ times and since $K = 4$, (3.58) is also satisfied.

Given a (V, K, Λ) difference set D, the set

$$D' = \{d_0 + s, d_1 + s, d_2 + s, ..., d_{k-1} + s\} = D + s \tag{3.59}$$

where each element is taken modulo V, will also be a (V, K, Λ) difference set. In this case, D' is called a **cyclic shift** of D. If D_p and D_q are two difference sets with the same parameters (V, K, Λ) and $D_p = tD_q + s$ for any integers t and s with t prime to V (that is, t and V have no common factors), then and are called **equivalent difference** sets. If D is a (V, K, Λ) difference set, then its **complement** D^* will be a difference set with parameters $(V, V-K, V-2K+\Lambda)$ [5][19].

For any particular (V, K, Λ) satisfying (3.58) there may be no difference sets, one difference set (disregarding equivalent sets), or several nonequivalent difference sets. Proofs of existence and nonexistence are of great concern to theoreticians. For now, it is sufficient to note that the sets are abundant, that tables of the sets exist, and that construction algorithms can be used to create them. In particular, construction algorithms exist for sets with $K/V \approx \frac{1}{2}, \frac{1}{4}, \frac{1}{8}$, where K/V is defined herein as the **thinning factor**. It is also possible to construct very highly thinned Singer difference sets for which K is approximately the square root of V [5][19].

3.5.4 Difference Sets, Autocorrelations, and Linear Arrays

From a difference set D, we may construct a sequence or "array" of ones and zeros

$$A_V = \{a_j\}\,,\ i = 0, 1, ..., V-1 \tag{3.60}$$

where $a_j = 1$ if j is in D and $a_j = 0$ if j is not in D. For example, set D_3 above gives rise to $A_V = \{1101000001000\}$. If we create an **infinite array** of ones and zeros

$$A_I = \{.., a_{-2}, a_{-1}, a_0, a_1, a_2, ...\}\,,\ i = 0, 1, ..., V-1 \tag{3.61}$$

by periodically repeating A_V, we may define an autocorrelation for A_I given by

$$C_I(\tau) = \sum_{n=0}^{V-1} a_n a_{n+\tau} \tag{3.62}$$

It follows that if and only if A_I is formed from a difference set, then

$$C_I(\tau) = \begin{cases} K, & if\ \tau\,(mod\,V) = 0 \\ \Lambda, & otherwise \end{cases} \tag{3.63}$$

In other words, the autocorrelation function is **two-valued**. Ultimately, it is this property that makes the difference set an effective prescription for the design of thinned arrays. As shown in the next section, by tying the one's and zero's to element locations in a lattice, a periodically repeating element placement sequence dictated by difference sets necessarily has an array power pattern with all sidelobe peaks constrained to be at an identical fixed level that is less than $1/K$ times the main lobe peak. When the infinite sequence is truncated to a single period, these same fixed levels remain, tying down half the sample points of the power pattern. The PSL of the resulting pattern is then determined by the remaining sample points [5][19].

3.5.5 Linear Isophoric Arrays

From any sequence of one's and zero's we can construct a corresponding linear phased array by starting with an empty lattice of element locations spaced $\frac{1}{2}-$wavelength apart, placing an element at each location where the sequence has a "1", and skipping each location where the sequence has a "0". From such a construction we can form an array element location function

$$A_I(x) = \sum_{n=-\infty}^{\infty} a_n \delta(x - n x_0) \tag{3.64}$$

for an infinite length array, where $\delta(x)$ is the usual Dirac delta function, and x_0 is the inter element spacing ([5][19]). Typically, $x_0 = \frac{1}{2}$ wavelength.

While an infinite length array is of no practical interest, a study of its properties will lead to the central result for isophoric arrays. As with any array, the power pattern for this array will be the Fourier transform of the autocorrelation function of the location function. From (3.63), the autocorrelation function of isophoric array is given by

$$\begin{aligned} C_I(\chi) = \quad & (K-\Lambda)\sum_{n=-\infty}^{\infty}\delta(\chi - nVx_0) \\ & +\Lambda\sum_{n=-\infty}^{\infty}\delta(\chi - nx_0) \end{aligned} \tag{3.65}$$

This sum represents an infinite train of impulses at $\chi = 0, \pm x_0, \pm 2x_0, ...$ All the impulses have area Λ except for those at $\chi = 0, \pm Vx_0, \pm 2Vx_0, ...,$ which have area $(K-\Lambda)+\Lambda = K$.

We recall that the Fourier transform of an infinite train of unity-area impulses at $x = 0, \pm x_0, \pm 2x_0, ...$ is itself an infinite train of impulses in u, each with area $1/x_0$ located at $u = 0, \pm 1/x_0, \pm 2/x_0, ...$ From this it follows that the Fourier transform of autocorrelation $C_I(\chi)$ is

$$\begin{aligned} ff_I^*(u) = \quad & (K-\Lambda)\tfrac{1}{Vx_0}\sum_{n=-\infty}^{\infty}\delta\left(u - \tfrac{n}{Vx_0}\right) \\ & +\Lambda\tfrac{1}{x_0}\sum_{n=-\infty}^{\infty}\delta\left(u - \tfrac{n}{x_0}\right) \end{aligned} \tag{3.66}$$

Using (3.58) we can eliminate Λ and create a normalized $\overline{ff_I^*(u)}$ by writing

$$\begin{aligned} \overline{ff_I^*(u)} = \quad & \tfrac{ff_I^*(u)}{K^2} = \rho\left[\tfrac{1}{Vx_0}\sum_{n=-\infty}^{\infty}\delta\left(u - \tfrac{n}{Vx_0}\right)\right] \\ & +(1-\rho)\left[\tfrac{1}{x_0}\sum_{n=-\infty}^{\infty}\delta\left(u - \tfrac{n}{x_0}\right)\right] \end{aligned} \tag{3.67}$$

where

$$\rho = \frac{1}{K}\left[1 - \frac{(K-1)}{(V-1)}\right] \tag{3.68}$$

This normalized power pattern has a "main-lobe" impulse with an area of 1 at $u = 0, \pm 1/x_0, \pm 2/x_0, ...,$ and identical "sidelobe" impulses with area ρ located at $u = \pm 1/(Vx_0), \pm 2/(Vx_0), ...$ ([5][19]).

A finite-length isophoric array will have element location function

$$A_T(x) = \sum_{n=0}^{V-1} a_n \delta(x - nx_0) \tag{3.69}$$

$A_T(x)$ is therefore a single "truncated" cycle of the infinite length array in (3.64). Let $ff_I^*(u)$ and $ff_T^*(u)$ be array power patterns for the infinite and finite arrays, respectively. Then a basic property of the Fourier transform permits us to write

$$ff_I^*(u) = ff_T^*(u)\frac{1}{Vx_0}\sum_{n=-\infty}^{\infty}\delta\left(u-\frac{n}{Vx_0}\right) \tag{3.70}$$

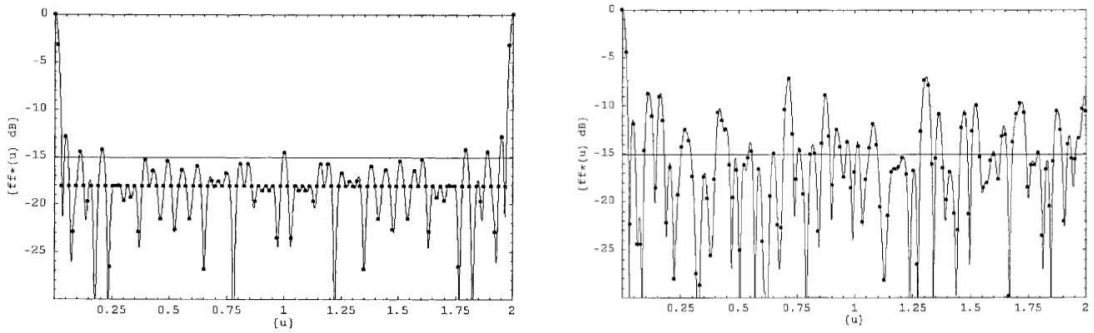
This expression shows that $ff_I^*(u)$ and $ff_T^*(u)$ are "tied together" at $u = 0, \pm 1/x_0, \pm 2/x_0, ...$ It is sometimes said that $ff_T^*(u)$ forms an "envelope" for the $ff_I^*(u)$ impulse train. Therefore, the power pattern $ff_T^*(u)$ for an isophoric array must necessarily pass through the fixed points prescribed by (3.66).

It follows that for an isophoric array

$$\overline{ff_T^*(n/(Vx_0))} = \begin{cases} 1, & for\ n = 0, \pm V, \pm 2V, ... \\ \rho, & for\ all\ other\ n \end{cases} \tag{3.71}$$

Fig. 13-(a) shows the normalized power pattern for a particular isophoric linear array of 32 elements on a 63-slot lattice with uniform $x_0 = \frac{1}{2}$-wavelength spacing. The regularly spaced, dotted points located at $u = 2/63, 4/63, 6/63, ...$ are the sample points referred to in (3.70). At each of these "even numbered" sample points $\overline{ff^*(u)} = 10\log_{10}(\rho) - 18.06\text{dB}$, illustrating the effects predicted by (3.70) and (3.71).

Note that in Fig. 13-(a), the peak at $u = 2$ is simply a repetition of the main beam. From (3.49), it is straightforward to show that any array in which the elements are constrained to be located at the fixed points of a uniform lattice will necessarily have a power pattern that is periodic in u with period $u_0 = 1/x_0$ as well as being symmetric about any integer multiple of $u = 1/(2x_0)$, where x_0 is the spacing between adjacent lattice points measured in wavelengths. For comparison, Fig. 13-(b) shows a power pattern for a random array of 32 elements on the same aperture. Note that: 1) there is no regularity evident in the dotted points and 2) the PSL for this particular array is approximately 6 dB higher than that for the isophoric array. In this chapter, the term **random array** refers to an array in which an element may appear **anywhere** with an aperture with equal likelihood. A lattice array is an array in which elements may only appear at uniformly spaced points in the aperture. A random lattice array is an array in which the elements are located at randomly chosen lattice points [5][19].



(a) (b)

Figure 13. *Isophoric Array* - (a) Isophoric linear array power pattern. Number of elements = 32. Aperture size = 62 half-wavelengths. (b) Random linear array power pattern. Number of elements = 32. Aperture size = 62 half-wavelengths [5].

More generally, the expected PSL of the isophoric array will be lower than that of a corresponding random array by

$$\begin{aligned} &Isophoric\, PSL\, Reduction\, (linear\, array) \\ &\approx 3 + 10\log_{10}(1 - K/V)^{-1}\, dB \end{aligned} \tag{3.72}$$

The 3-dB portion of the PSL reduction comes from constraining the locations to those determined by difference sets. The remainder of the improvement comes from simply constraining the elements locations to the points of a fixed lattice. Note that this latter improvement becomes vanishingly small with increased thinning; that is, as K/V approaches zero. However, the 3-dB improvement remains even for highly thinned arrays [5][19].

The theory of the random array shows that

$$\overline{ff^*(u)} = 10\log_{10}(1/K)\, dB \tag{3.73}$$

is the average power in the sidelobe region of a random array. Both figures show a reference line at this average level for these arrays, namely at 15.05dB.

3.5.6 Expected Power Pattern of a Linear Isophoric Array

Isophoric array PSLs in the preceding section could be reduced still further by trying various cyclic shifts of the difference set that was used to generate the initial array. A **cyclic shift** of a difference set $\{D\}$ simply adds an integer s to each member of $\{D\}$ and then reduces each result modulo V. Clearly, there are V unique such shifts possible for

$s = 0, 1, ..., V-1$. This is a relatively small number to apply in a "cut-and-try" attempt at lowering PSL.

More importantly, as shown in this section, the average power pattern of an isophoric array, taken over all V cyclic shifts of the underlying difference set, is exactly the same as the average power pattern of all the possible arrays that one could create by placing K elements on a lattice with V slots.

The expected (average) power pattern of a linear isophoric array is defined as

$$\mathbf{E}\left[ff^*(u)\right] = ff_E^*(u) = \frac{1}{V}\sum_{s=0}^{V-1} ff_s^*(u) \tag{3.74}$$

where $ff_s^*(u)$ is the power pattern generated by an array whose underlying difference set has undergone a cyclic shift of s units.

As shown below,

$$\overline{ff_E^*(u)} = \rho + (1-\rho)\frac{\sin(\pi u V x_0)}{V^2 \sin^2(\pi u x_0)} \tag{3.75}$$

The derivation of this result is straightforward but lengthy. To conserve space, we simply outline the steps as follows:

1. Note that as with any power pattern, each $ff_s^*(u)$ is the Fourier transform of the autocorrelation of the element location function of the array built from a cyclic shift s of the underlying difference set.

2. By substituting the Fourier transform expression for each $ff_s^*(u)$ in (3.74) and interchanging the order of summation and integration, the average Fourier transform of the power patterns becomes the Fourier transform of the average of the V autocorrelations.

3. Fundamental properties of difference force the average autocorrelation to be

$$C_E(\tau x_0) = \begin{cases} k\delta(0), & \tau = 0 \\ (V-|\tau|)\frac{k(k-1)}{v(v-1)}\delta(x-\tau x_0), & for\ 0 < |\tau| < V \\ 0, & |\tau| \geq V \end{cases} \tag{3.76}$$

4. The (normalized) Fourier transform of $C_E(\tau x_0)$ is $\overline{ff_E^*(u)}$, as given by (3.75).

Note that for a moderately large V, (say, greater than 30), $K/V < \frac{1}{2}$ and u not close to zero (that is, the sidelobe region), the contribution to be made by the second term in (3.75) is quite small. Under these conditions

$$\overline{ff_E^*(u)} \approx \rho = \frac{1}{K}\left[1 - \frac{(K-1)}{(V-1)}\right] \tag{3.77}$$

In the special case $K = V$, the array is filled and the expression reduces to the well-known power pattern of a filled array. The filled array is in fact a special case of an isophoric array [5][19].

(3.75) also represents the grand average power pattern of all possible placements of K elements on a V-slot lattice. One way of viewing the V cyclic shifts of an isophoric array is that they represent a small set of arrays whose average power pattern is the same as the average pattern of the much larger set of all possible of K elements on a V-slot lattice. In the example used thus far, the 63 cyclic shifts of Array 1 have an average power pattern identical to that of the 9.16×10^{17} possible placements of 32 elements on a 63-slot lattice.

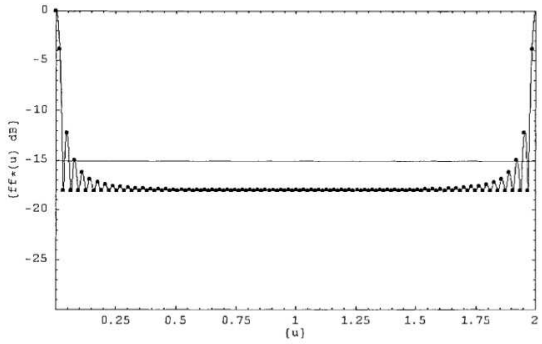


Figure 14. *Isophoric Array* - Expected power pattern of isophoric array with $V = 63$ and $K = 32$ [5].

Note also that while the average sidelobe power of a random array is $1/K$, the average power of a **random lattice** array is $\approx (1/K)(1 - K/V)$. Thus, simply **constraining the element placements** to **lattice positions reduces sidelobe levels** to some extent, although the improvement becomes vanishingly small with increased thinning. As stated previously, further constraining the element placements to be those dictated by a difference set produce another 3 dB of expected PSL reduction. This 3-dB reduction is **independent** of how much the array has been thinned [5][19].

3.5.7 Extension to Planar Arrays

Isophoric arrays, both static and spatially hopped, can be planar as well as linear. The principals are the same. We seek a deterministic placement of K elements in a rectangular lattice such that the element location function has a two-level autocorrelation function in two dimensions [5][19].

The element location function for a planar array is defined by

$$A_T(x, y) = \sum_{m=0}^{V_x-1} \sum_{n=0}^{V_y-1} a_{m,n} \delta(x - mx_0, y - ny_0) \tag{3.78}$$

where the array has dimensions $V_x V_y$, $\delta(x-g, y-h)$ is interpreted as a unit impulse at location $(x, y) = (g, h)$, and the coefficients form a V_x-by-V_y matrix of ones and zeros that designate the presence or absence of an array element at (mx_0, ny_0).

Analogous to (3.62), we form a two-dimensional autocorrelation for an infinitely repeated version $A_I(x, y)$of $A_T(x, y)$ as

$$C_I(p, q) = \sum_{m=0}^{V_x-1} \sum_{n=0}^{V_y-1} a_{m,n} a_{m+p,n+q} \tag{3.79}$$

We let the number of ones in the $a_{m,n}$ coefficients equal K and assume that we can discover a placement of ones and zeros such that

$$C_I(p, q) = \begin{cases} K, & if\ V_x\ divides\ p\ and\ V_y\ divides\ q \\ \Lambda, & otherwise \end{cases} \tag{3.80}$$

That is, $A_I(x, y)$ has a two-level autocorrelation function. If this can be done, then we know that all the $V_x V_y$ sample point sin the sidelobe region of $f(u, v)$ (3.54,3.55) will necessarily have magnitude K. We also know that the even-numbered samples from the sidelobe region of $ff^*(u, v)$ will have magnitude K^2. The odd-numbered samples will be the ones that determine the PSL [5][19].

Results from Monte Carlo simulations show that compared to a random (nonlattice) placement of elements on the same aperture, a static (not spatially hopped) isophoric array will have an expected improvement in PSL of

$$\begin{gathered} Isophoric\ PSL\ Reduction\ (planar\ array) \\ \approx 1.5 + 10 \log_{10}(1 - K/V)^{-1}\ dB \end{gathered} \tag{3.81}$$

where $V = V_x V_y$. This improvement is 1.5dB smaller than it was for linear arrays. As with linear arrays, if we can find a placement algorithm with the property described by (3.80), then we can spatially hop the array element assignments as we did for linear arrays, thereby guaranteeing a fixed low-sidelobe power pattern for $ff^*(u, v)$ as we did for $ff^*(u)$ ([5][19]).

Assume we have a linear sequence of V ones and zeros

$$A_V = \{a_i\},\ i = 0, 1, ..., V-1$$

dictated by a difference set as in (3.57). Then the assignment

$$\begin{gathered} a_{m,n} = a_i \qquad where \qquad m = i\,(mod V_x) \\ n = i\,(mod V_y) \quad i = 0, 1, ..., V-1 \end{gathered} \tag{3.82}$$

will create a rectangular array of ones and zeros

$$\begin{gathered} A_{V_x V_y} = \{a_{m,n}\} \\ m = 0, 1, 2, ..., V_x - 1, \quad n = 0, 1, 2, ..., V_y - 1 \end{gathered} \tag{3.83}$$

that has the desired two-level autocorrelation function.

For example, the (63, 32, 16) difference set would be placed in a 9×7 array as shown in Table II. As shown, a_0 is placed in the "southwest" corner of the array and each succeeding coefficient is placed in the slot to the "northeast", continuing from the other side whenever an edge is reached until the entire $V = V_x V_y = (9)(7) = 63$ coefficients have been placed. The table shows the placement of the first 18 coefficients. An antenna element will be placed in each location where $a_i = 1$ and not placed where $a_i = 0$.

With the approach above, we can create a static isophoric array with expected power pattern

$$\overline{ff_E^*(u,v)} = \rho + (1-\rho)\frac{\sin^2(\pi u V_x x_0)}{V_x^2 \sin^2(\pi u x_0)} \frac{\sin^2(\pi u V_y y_0)}{V_y^2 \sin^2(\pi u y_0)} \tag{3.84}$$

As with linear arrays, once we move into the sidelobe region (that is, u and v not too close to $0, \pm 2, \pm 4, ...$), the expected normalized pattern is approximately the constant ρ, where ρ is given by (3.67). Fig. 15 shows for a $\overline{ff_E^*(u,v)}$-slot lattice, with128 elements.

Note that for the special case $V = K$, ρ becomes zero and $\overline{ff_E^*(u,v)}$ becomes the power pattern of the familiar filled rectangular-lattice array. Note also that the beamwidth implied by (3.84) is independent of the thinning factor $\beta = K/V$. Even a very highly thinned isophoric array will have the same beamwidth as a filled array.

Again, as with linear arrays, if we begin with a filled lattice and operate it as two independent interwoven isophoric arrays with spatially hopped element assignments, we can actually achieve two independent patterns obeying $\overline{ff_E^*(u,v)}$ on a time-averaged basis.

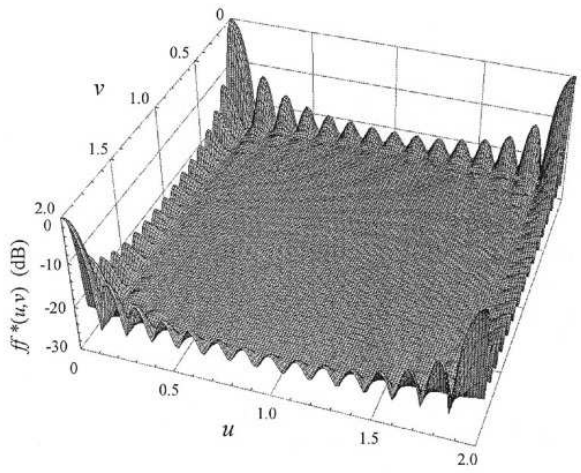


Figure 15. *Isophoric Array* - Expected power pattern of isophoric planar array with $V = V_x V_y = 15 \times 17$ half-waves and $K = 128$ elements. this exact pattern is realizable with "spatial hopping". Note pattern floor at $10 \log_{10} \rho = -24$dB [5].

3.6 Almost Difference Sets [22]

3.6.1 Introduction

Massive thinning of arrays (i.e., the reduction of the number of the array elements below half of its filled counterpart) is of great importance in practical applications because of the reduction of the array costs, weight, power consumption, HW and computational complexity.

However, such advantages usually come at the cost of a loss of sidelobe level (SLL) control and gain compared to the filled arrangement. In order to overcome these drawbacks, several thinning techniques have been proposed. Deterministic thinning has been first studied, but no significant improvements of SLL control compared to a random element placement have been obtained. More recently, dynamic programming and stochastic optimization techniques, such as simulated annealing (SA) and genetic algorithms (GAs) have been successfully applied. Despite the satisfactory results, statistical methodologies have not an easy application to large arrays because of the computational burden and convergence issues. Moreover, due to their stochastic nature, it is often difficult to a-priori estimate the expected performances for a given aperture size and thinning factor.

The synthesis of massively thinned arrays has been faced in a very promising fashion by considering equally-weighted arrays. Such an approach is based on the use of binary sequences derived from **difference sets** (DSs), which are known to possess two-level periodic autocorrelations. In different works it has been shown that, if the element excitations are chosen according to the binary distribution derived from DSs, the peak sidelobe level (PSL) of the synthesized linear array is 3-dB lower than that of the corresponding random distribution. Such a result has been successfully exploited for the design of both linear and planar arrays, although the PSL reduction is about 1.5-dB smaller when planar architectures are dealt with. The application of DSs has also allowed some improvements in thinned-array design procedures based on GA optimization [22].

Recently, the definition of binary sequences of length N with suitable autocorrelation properties, for which DSs are not available, has been carefully investigated in information theory and combinatorial mathematics. It has been found that it is often possible to determine sequences with a three-level autocorrelation function by taking into account the so-called **almost difference sets** (ADSs). ADSs are a research topic of great interest in combinatorial theory with important applications in cryptography and coding theory. Moreover, although ADS generation techniques are still subject of research, large collections of these sets are already available. As regards the array synthesis, a prelimi-

nary application, although limited to a particular subset of ADSs. In such a framework, the whole class of ADSs seem to be a good candidate for enlarging the set of admissible analytic configurations with respect to the DS case, despite a reduction of expected performances. From this viewpoint, it is of interest to carefully detail the ADS features for antenna arrays synthesis [22].

In this chapter, the exploitation of ADSs properties for the design of linear thinned arrays is discussed and analyze in depth through a solid mathematical description. The proposed ADS based technique is aimed at synthesizing arrays with performances close to those with DSs, but enhancing the set of admissible array configurations. It is also worth while to point out that the paper is not aimed at defining an optimal method for the design of thinned arrays, but its purpose is to propose some guidelines to the array designers who, whether by necessity or choice, are synthesizing a thinned array without considering stochastic optimizations or a random placement, but using a deterministic strategy with predictable results [22].

3.6.2 Almost Difference Sets - Definitions and Properties

Let us provide just some basic definitions and main properties of ADSs.

A K-subset $\underline{D} = \{d_k \in [0, N-1], d_h \neq d_l; k, h, l = 0, ..., K-1\}$ of an Abelian group $\underline{\mathbf{G}}$ of order N is called a (N, K, Λ, t)-almost difference set if the multiset $\underline{\mathbf{M}} = \{m_j = (d_h - d_l), d_l \neq d_k; j = 0, ..., K \times (K-1) - 1\}$ contains nonzero elements of $\underline{\mathbf{G}}$ each exactly Λ times, and the remaining $N-1-t$ nonzero elements each exactly $\Lambda+1$ times. As a consequence, DSs are ADSs for which $t = N-1$ or $t = 0$. 1. An Abelian group is a group satisfying the requirement that the product of elements does not depend on their order. In addition to the other axioms of a group, the product operation is associative, $\underline{\mathbf{G}}$ has an identity element, and every element of $\underline{\mathbf{G}}$ has an inverse [22].

If $\underline{\mathbf{G}} \equiv Z$ and $\underline{D}$ is a (N, K, Λ, t)-ADS of $\underline{\mathbf{G}}$, then the cyclic repetition of the binary sequence $\underline{\mathbf{A}} = \{a_n \in [0, 1]; n = 0, ..., N-1\}$of length N, whose nth element is

$$a_n = \begin{cases} 1, & if\ n \in \underline{D} \\ 0, & otherwise \end{cases} \tag{3.85}$$

defines the characteristic sequence $\underline{\mathbf{S}} = \{s_n, n \in \mathbb{Z}\}$ of $\underline{D}$, where

$$s_n = \begin{cases} 1, & if\ mod_N(n) \in \underline{D} \\ 0, & otherwise \end{cases} \tag{3.86}$$

The corresponding autocorrelation function, $C_s(z)$, is a periodic function defined as fol-

lows

$$C_s(z) = \sum_{n=0}^{N-1} s_n s_{n+1} \; z \in \mathbb{Z} \tag{3.87}$$

and equal to

$$C_s^{ADS}(z) = \begin{cases} K, & z = 0 \\ \Lambda + 1, & z \in \underline{\mathbf{L}} \\ \Lambda, & otherwise \end{cases}, \; K \geq \Lambda + 1 \tag{3.88}$$

in the period $z \in [0, N-1]$, $\underline{\mathbf{L}}$ being a set of $N-1-t$ elements (i.e. $\underline{\mathbf{L}} = \{l_p \in \mathbb{Z}; p = 1, ..., N-1-t\}$). For illustrative purposes, let us consider the examples of ADSs reported in Table I together with the corresponding binary sequences and autocorrelation functions. For completeness, the plots of $C_s^{ADS}(z)$ are shown in Fig. 16 [22].

N	13	21	33	45
$\underline{\mathbf{G}}$	$\underline{\mathbf{G}}_1 = \{0, ..., 12\}$	$\underline{\mathbf{G}}_2 = \{0, ..., 20\}$	$\underline{\mathbf{G}}_2 = \{0, ..., 32\}$	$\underline{\mathbf{G}}_2 = \{0, ..., 44\}$
K	3	6	16	22
Λ	0	1	7	10
t	6	10	16	22
$\underline{\mathbf{D}}$	$\underline{\mathbf{D}}_1 = \{5, 6, 9\}$	$\underline{\mathbf{D}}_2 = \{0,1,3,13$ $16,17\}$	$\underline{\mathbf{D}}_3 = \{0,1,2,3,4,5,6,$ $8,13,14,18,20,$ $22,25,28,29\}$	$\underline{\mathbf{D}}_4 = \{0,1,2,3,4,5,6,$ $7,9,11,12,15,$ $16,19,23,24,29,$ $30,32,35,37,39\}$
$\underline{\mathbf{A}}$	$\underline{\mathbf{A}}_1 = \{0000$ 01100 $1000\}$	$\underline{\mathbf{A}}_2 = \{1101000$ 00000010 $0110000\}$	$\underline{\mathbf{A}}_3 = \{11111110100$ 00110001010 $10010011000\}$	$\underline{\mathbf{A}}_4 = \{1111111101$ 01100110010 001100001101 $001010100000\}$
$\underline{\mathbf{S}}$	$\underline{\mathbf{S}}_1 = \{..., \underline{\mathbf{A}}_1, \underline{\mathbf{A}}_1, ...\}$	$\underline{\mathbf{S}}_2 = \{..., \underline{\mathbf{A}}_2, \underline{\mathbf{A}}_2, ...\}$	$\underline{\mathbf{S}}_3 = \{..., \underline{\mathbf{A}}_3, \underline{\mathbf{A}}_3, ...\}$	$\underline{\mathbf{S}}_4 = \{..., \underline{\mathbf{A}}_4, \underline{\mathbf{A}}_4, ...\}$
$C_s^{ADS}(z)$	$3, z = 0$ $0, z = 2,5,6,$ $7,8,11$ $1, z \in \underline{\mathbf{L}} = \{1,3,$ $4,9,$ $10,12\}$	$6, z = 0$ $1, z = 2,6,7,9,$ $10,11,12,$ $14,15,19$ $2, z \in \underline{\mathbf{L}} = \{1,3,4,5,$ $8,13,16,$ $17,18,20\}$	$16, z = 0$ $7, z = 3,6,7,9,10,11,$ $13,15,18,20,22,$ $23,24,26,27,30$ $8, z \in \underline{\mathbf{L}} = \{1,2,4,5,8,12,$ $14,16,17,19,21,$ $25,28,29,31,32\}$	$22, z = 0$ $10, z = 6,9,10,11,12,14,16,18,$ $19,20,21,24,25,26,27,$ $29,31,33,34,35,36,39$ $11, z \in \underline{\mathbf{L}} = \{1,2,3,4,5,7,8,13,15,$ $17,22,23,28,30,37,$ $38,40,41,42,43,44\}$

Table I. *Linear Thinned Arrays based on Almost Difference Sets* - Examples of ADSs and their descriptive functions [22].

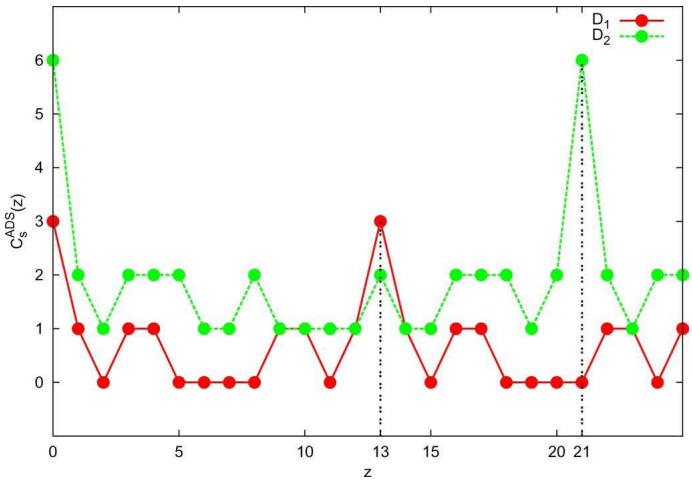


Figure 16. *Linear Thinned Arrays based on Almost Difference Sets* - Autocorrelation function $C_S^{ADS}(z)$ of $\underline{D}_1$ and $\underline{D}_2$ in Table I [22].

It is worth noting that the autocorrelation function $C_s^{ADS}(z)$ of a (N, K, Λ, t)-ADS is close to that of the (if any) corresponding (N, K, Λ)-DS

$$C_s^{DS}(z) = \begin{cases} K, & z = 0 \\ \Lambda, & otherwise \end{cases} \tag{3.89}$$

In fact, the difference is limited to just a unity in $N - 1 - t$ points where $C_s^{ADS}(z) = \Lambda + 1$. Moreover, the ADSs share several other properties with the DSs. In particular, neither DS nor ADS can be defined for every value of N, K, Λ, and t . Indeed, for (N, K, Λ, t)-ADSs in an Abelian group, the following existence condition holds true

$$K(K-1) = t\Lambda + (N - 1 - t)(\Lambda + 1) \tag{3.90}$$

being $K \geq \Lambda + 1$, $0 \leq K \leq N$, and $0 \leq t \leq N - 1$.

On the other hand, if $\underline{D}$ is an ADS, then the set

$$\underline{D} = \begin{Bmatrix} d_k^{(\sigma)} = mod_N(d_k + \sigma), d_k \neq d_l; \\ k, h, l = 0, ..., K - 1 \end{Bmatrix} \tag{3.91}$$

where $\sigma \in \mathbb{Z}$, is still an ADS. Therefore, starting from an (N, K, Λ, t)-ADS, it is possible to build different (N, K, Λ, t)-ADSs by applying a cyclic shift to its elements (i.e., a cyclic shift on the associated binary sequence $\underline{\mathbf{A}}$). Mathematical proofs of existence or non-existence of ADSs for different choices of are currently topic of research in the framework of combinatorial theory and suitable techniques for the generation of new families of ADSs are still in progress. However, several ADSs has been already found and their properties can be profitably exploited for array synthesis [22].

3.6.3 ADS-Based Linear Arrays - Mathematical Formulation

3.6.3.1 ADS-Based Infinite Arrays

An infinite thinned array can be defined from whatever binary sequence $\underline{\mathbf{A}}$ of length N by introducing the **array element location function** $\Psi_\infty(x)$

$$\Psi_\infty(x) = \sum_{n=-\infty}^{\infty} s_n \delta(x - nd) \tag{3.92}$$

where $\delta(.)$ is the Dirac delta function, d and x are the lattice spacing and the spatial coordinate along the linear array, respectively (both expressed in wavelength). In practice, the infinite thinned array is defined by locating the array elements along a uniform lattice with spacing d at those positions where $\Psi_\infty(x) = \infty$ [22].

As with any array, the power pattern of the ADS-based infinite linear array turns out to be the Fourier transform of the autocorrelation function of $\Psi_\infty(x)$, $C_\Psi^{ADS}(z)$, that is

$$PP_\infty(u) = \mathcal{F}\left\{C_\Psi^{ADS}(z)\right\} \tag{3.93}$$

where $\mathcal{F}\{.\}$ denotes the Fourier transform operator, $u = \sin(\theta)$, $u \in [-1, 1]$, and

$$\begin{aligned} C_\Psi^{ADS}(z) &= \Lambda \sum_{n=-\infty}^{\infty} \delta(z - nd) \\ &+ \sum_{p=1}^{N-1-t} \left\{ \sum_{n=-\infty}^{\infty} \delta(z - ndN - l_p d) \right\} \\ &+ (K - \Lambda) \sum_{n=-\infty}^{\infty} \delta(z - ndN) \end{aligned} \tag{3.94}$$

where the index l_p satisfies the condition $C_s(l_p) = \Lambda + 1$ [22].

By substituting (3.94) in (3.93) and recalling the Fourier transformation properties of an infinite train of pulse functions, one can show that

$$PP_\infty(u) = \sum_{n=-\infty}^{\infty} PP_{\infty,n} \delta\left(u - \frac{n}{Nd}\right) \tag{3.95}$$

where, see equation

$$PP_{\infty,n} = \begin{cases} \frac{\Lambda}{d} + \frac{1}{Nd}\left(K - \Lambda + \sum_{p=1}^{N-1-t} \exp(j2\pi l_p n/N)\right), & n = 0, \pm N, \pm 2N, ... \\ \frac{1}{Nd}\left(K - \Lambda + \sum_{p=1}^{N-1-t} \exp(j2\pi l_p n/N)\right), & otherwise \end{cases} \tag{3.96}$$

However, unlike DSs, further simplifications of (3.95) are not trivial since the following term of $PP_{\infty,n}$

$$\left(K-\Lambda+\sum_{p=1}^{N-1-t}\exp\left(k2\pi l_p n/N\right)\right)$$

$$= \left(K-\Lambda+\sum_{p=1}^{N-1-t}\exp\left(k2\pi l_p n/N\right)\right), \quad l_0=0 \tag{3.97}$$

cannot be evaluated in closed form. In fact, the set $\underline{\mathbf{L}}$ depends on the ADS at hand and $PP_{\infty}(u)$ has to be evaluated on a case-by-case basis instead of in a general fashion. However, it is still possible to provide an *a-priori* estimate of the peak sidelobe level of the infinite array, PSL_{∞}, defined as

$$PSL_{\infty}=\max_{n\neq 0}\frac{PP_{\infty,n}}{PP_{\infty,0}} \tag{3.98}$$

Actually, it turns out that PSL_{∞} is limited by the following upper

$$PSL_{\infty}^{MAX}=\frac{K-\Lambda-1+\sqrt{t\left(N-t\right)}}{\left(N-1\right)\Lambda+K-1+N-t} \tag{3.99}$$

and lower

$$PSL_{\infty}^{MIN}=\frac{K-\Lambda-1-\sqrt{\frac{t(N-t)}{(N-1)}}}{\left(N-1\right)\Lambda+K-1+N-t} \tag{3.100}$$

respectively. Moreover, for fixed values of $\eta=t/\left(N-1\right)$ and of the thinning percentage factor ν, $(\nu=K/N)$, the range of variation of PSL_{∞} reduces as N increases until a threshold. Such a behavior is pointed out in a study on the dependence of the confidence range index $\Delta_{\infty}=PSL_{\infty}^{MAX}/PSL_{\infty}^{MIN}$, which by (3.90), (3.99), and (3.100) turns out to be, see the following equation

$$\Delta_{\infty}=\frac{N^2\left(\nu-\nu^2\right)-\eta N+\eta+\left(N-1\right)\sqrt{N^2\left(\eta-\eta^2\right)+N\left(2\eta^2-\eta\right)-\eta^2}}{N^2\left(\nu-\nu^2\right)-\eta N+\eta-\left(N-1\right)\sqrt{N\left(\eta-\eta^2\right)+\eta^2}} \tag{3.101}$$

On N for different values of the ADS-parameters. The asymptotic threshold of Δ_{∞} appears to be equal to

$$\lim_{N\to\infty}\left(\Delta_{\infty}\right)=\frac{\nu-\nu^2+\sqrt{\eta\left(1-\eta\right)}}{\nu-\nu^2} \tag{3.102}$$

As expected, the condition Δ_{∞} is asymptotically verified when $\eta=1$ (i.e., $t=N-1$ and the ADS coincides with a DS), since $PSL_{\infty}=PSL_{\infty}^{DS}$. Such a conclusion identically holds true for $\eta=0$ (i.e., $t=0$), whatever the admissible value of ν [22].

Let us also notice from (3.101) that the following property $\Delta_\infty(\nu) = \Delta_\infty(1-\nu)$ holds true. Moreover, the analysis and the corresponding plots are limited to the range of N values for which an ADS sequence can exist [i.e., (3.90), $K \geq \Lambda + 1$, $0 \leq K \leq N$, and $0 \leq t \leq N-1$]. As it can be observed, the value of the confidence index decreases when $|\nu - 0.5| \to 0$ and it attains its minimum value when $\nu = 0.5$. In such a case, $\Delta_\infty \to \left[1 + 4\sqrt{\eta(1-\eta)}\right]$ asymptotically with a maximum value equal to $\max_n \{\Delta_\infty\rfloor_{\nu=0.5}\} \approx 4.77dB$ for $\eta = 0.5$ [22].

3.6.3.2 ADS-Based Finite Arrays

As regards finite arrays, since the **array element location function** $\Psi(x)$

$$\Psi(x) = \sum_{n=0}^{N-1} s_n \delta(x - nd) \tag{3.103}$$

is now a truncated version of $\Psi_\infty(x)$, then it can be easily shown that $PP_\infty(u)$ and the power pattern of the finite configuration, $PP(u)$, are related by the following relationship

$$PP_\infty(u) = PP(u) \frac{\sum_{n=-\infty}^{\infty} \delta\left(u - \frac{n}{Nd}\right)}{Nd} \tag{3.104}$$

Accordingly, $PP(u)$ necessarily satisfies the sampling condition at each coordinate $u = u_n = n/Nd$, that is

$$PP(u_n) = NdPP_{\infty,n} \qquad n = 0, ..., \left\lfloor \tfrac{N}{2} \right\rfloor \tag{3.105}$$

In order to illustrate such a behavior, Fig. 17 shows the plots of $PP(u)$ and of the coefficients $PP_{\infty,n}$ for the thinned array of $K = 22$ elements on a $N = 45$-locations lattice ($d = 1/2$) defined from the ADS $\underline{\mathbf{D}}_4$. It is worth noting that, since $\Psi(x)$ is real-valued, the beam pattern is symmetric with respect to $u = 0$ and only the range $u \in [0, 1]$ is considered [22].

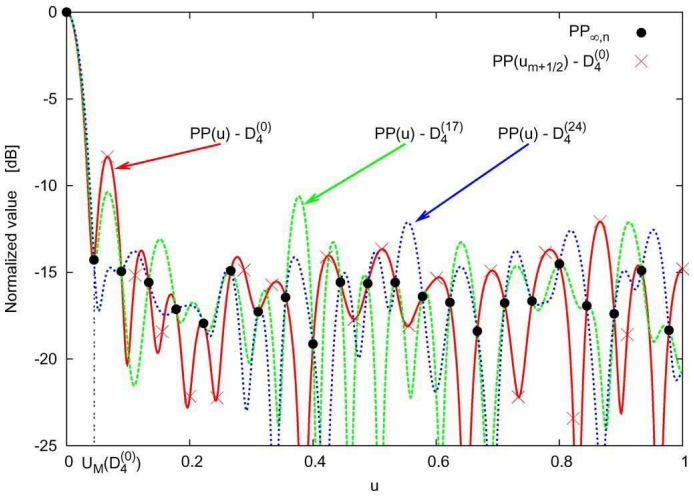


Figure 17. *Linear Thinned Arrays based on Almost Difference Sets* - Normalized $PP(u)$ derived from the ADS derived from the ADS $\underline{D}_4$ ($\underline{D}_4 = \underline{D}_4^{(\sigma)}\Big|_{\sigma=0}$) and its cyclic shifts $\underline{D}_4^{(\sigma)}$ ($\sigma = 17$, $\sigma = 24$). Number of elements: $N = 45$-Aperture size: 22λ [22].

Starting from (3.104), it is then possible to estimate the PSL of a finite array

$$PSL = \frac{\max\limits_{u\in\left[U_M\left(\underline{\mathbf{D}}^{(\sigma)}\right),1\right]}\{PP(u)\}}{PP(0)} \tag{3.106}$$

where U_M is the width of the main lobe region, by using the associated infinite array power pattern $PP_\infty(u)$. It is worth noting that (see Fig. 4) the PSL value is determined by the behavior of the power pattern at $u = u_{m+\frac{1}{2}} = (m+1/2)\,Nd$

$$PSL = \frac{\max\limits_{m}\left\{PP\left(u_{m+1/2}\right)\right\}}{PP(0)},\ m = 1, ..., \left\lfloor \frac{N}{2} \right\rfloor \tag{3.107}$$

being $u_{m+1/2} = (m+1/2)/Nd$.

To evaluate $PP\left(u_{m+1/2}\right)$, let us consider the sampling theorem and (3.104). It follows that

$$PP(u) = \left| \sum_{n=0}^{N-1} \sqrt{NdPP_{\infty,n}} \exp(j\phi_n) \frac{\sin\left[\pi Nd\left(u - \frac{n}{Nd}\right)\right]}{N\sin\left[\pi d\left(u - \frac{n}{Nd}\right)\right]} \right|^2 \tag{3.108}$$

where ϕ_n, $n =, ..., N-1$, are the phase terms of the sampled array factor ($\phi_0 = 0$), which are known quantities only when the ADS at hand is specified. By evaluating (3.108) in $u = 0$ and $u = u_{m+1/2}$ and substituting in (3.107), we obtain

$$PSL = \frac{\max\limits_{m}\left\{\left| \sum_{n=0}^{N-1} \sqrt{PP_{\infty,n}} \exp(j\phi_n) \frac{\sin\left[\pi\left(m-n+1/2\right)\right]}{N\sin\left[\frac{\pi(m-n+1/2)}{N}\right]} \right|^2\right\}}{PP_{\infty,0}} \qquad m = 1, ..., \left\lfloor \frac{N}{2} \right\rfloor \tag{3.109}$$

Consequently, the PSL of an ADS-based **finite** array is fully specified from the knowledge of $PP_{\infty,n}$ and ϕ_n, $n = 0, 1, ..., N-1$. However, since the $PP_{\infty,n}$ coefficients of ADS sequences neither can be expressed in closed-form (as for RDSs) nor have equal expressions (as for DSs), it is not available (although approximated) a threshold value for the PSL as for DSs. Nevertheless, it is possible to yield the following set of inequalities

$$PSL^{MIN} \leq PSL^{DW} \leq PSL^{opt} \leq PLS^{UP} \leq PSL^{MAX} \tag{3.110}$$

where $PSL^{opt} = \min_{\sigma\in[0,N-1]} \left\{PSL\left(\underline{D^{(\sigma)}}\right)\right\}$, $PSL^{MIN} = PSL_{\infty}^{MIN}$, $PSL^{DW} = \max\left\{PSL_{\infty}, PSL^{min}\right\}$, $PSL^{UP} = E\left\{\Phi_N^{min}\right\} PSL_{\infty}$, and $PSL^{MAX} = E\left\{\Phi_N^{min}\right\} PSL_{\infty}^{MAX}$, being $E\left\{\Phi_N^{min}\right\} \approx 0.8488 + 1.128 \log_{10} N$ and $PSL^{min} = E\left\{\Phi_N^{min}\right\} \min_n \left(PP_{\infty,n}\right) / PP_{\infty,0}$. It should be pointed out that PSL^{DW} and PSL^{UP} are determined when the ADS sequence is available since they require the knowledge of the coefficients $PP_{\infty,n}$. On the contrary, PSL^{MIN} and PSL^{MAX} can be always *a-priori* computed from (3.100) and (3.99), respectively [22].

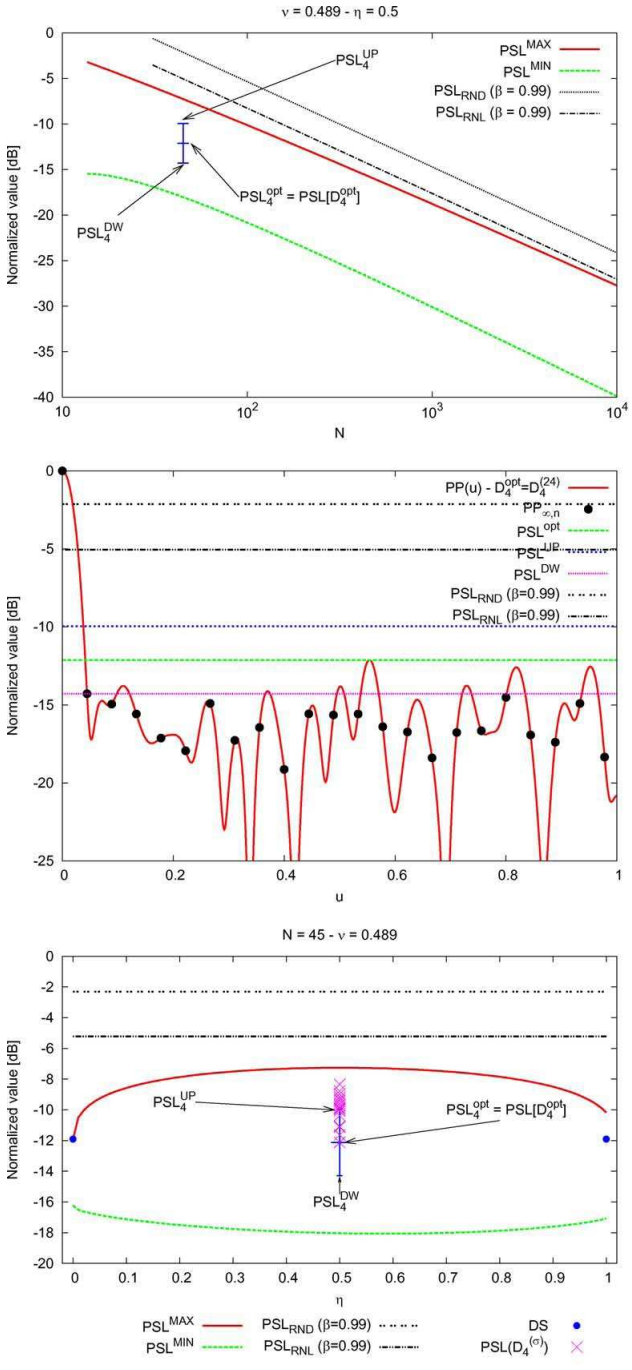


Figure 18. *Linear Thinned Arrays based on Almost Difference Sets - Comparative Assessment* - Plots of the PSL bounds of the ADS-based finite arrays and of the estimator of the PSL of the random arrays (RND - random array, RNL - random lattice array) when $\nu = 0.489$ versus (a) the array dimension, N, and (c) the index η. Normalized generated from $\underline{D}_4^{opt}$ and estimated PSL values of the corresponding random sequences (b) [22].

3.7 Basic Theory of Interferometry for Radio Astronomy [8][9][30][31]

3.7.1 Introduction

The particular interest in correlator antenna arrays for radio astronomy applications can be traced back to 1960s, and it was accompanied by drastic instrumental advances in interferometry techniques. Compared to conventional sum arrays, the enhanced data gathering efficiency of a correlator array is closely related to its spatial-filter-like behavior and the unique signal combination scheme by pair-wisely correlating output signals from all antenna elements. Unlike the well-established synthesis techniques for sum arrays, **determining an appropriate configuration of a correlator array is essentially an optimal sampling problem**. In order to obtain a clear image of a distant radio source, an **ideal correlator array** is desired to have **either the maximum coverage** in the **spatial frequency domain** (the $u-v$ domain) **or the lowest sidelobe level** (SLL) in the **angular domain** (the $l-m$ domain) [31].

3.7.2 Problem Definition

Fig. 19 depicts the measurement of a distant radio source using a correlator antenna array. The source has a brightness distribution $I(l,m)$ in the angular domain and the cosmic signal is collected by the ground-based array with a configuration of $f(x,y)$. The visibility of the source, $V(u,v)$, is defined in a plane perpendicular to the direction of source and this plane is referred as the $u-v$ domain. Here x and y are measured in kilometers; u and v are unitless quantities and $u\lambda$ (or $v\lambda$) has a unit of kilometer, where λ is the freespace wavelength. l and m are directional cosines of a point in the angular domain with respect to the $u-$ and $v-$ axes, respectively. They are measured in radians by applying the small-angle assumption since the desired field of view (FOV) in many practical cases is usually no more than a few degrees. It is worthwhile to mention that the definition of the $u-v$ domain is similar to that of the "$u-v$ domain" in conventional antenna language, which is often used to describe the far field of an antenna. In this thesis, the notion of "$u-v$ domain" follows the radio astronomy nomenclature and represents the spatial frequency domain instead of the angular domain [8][9][31].

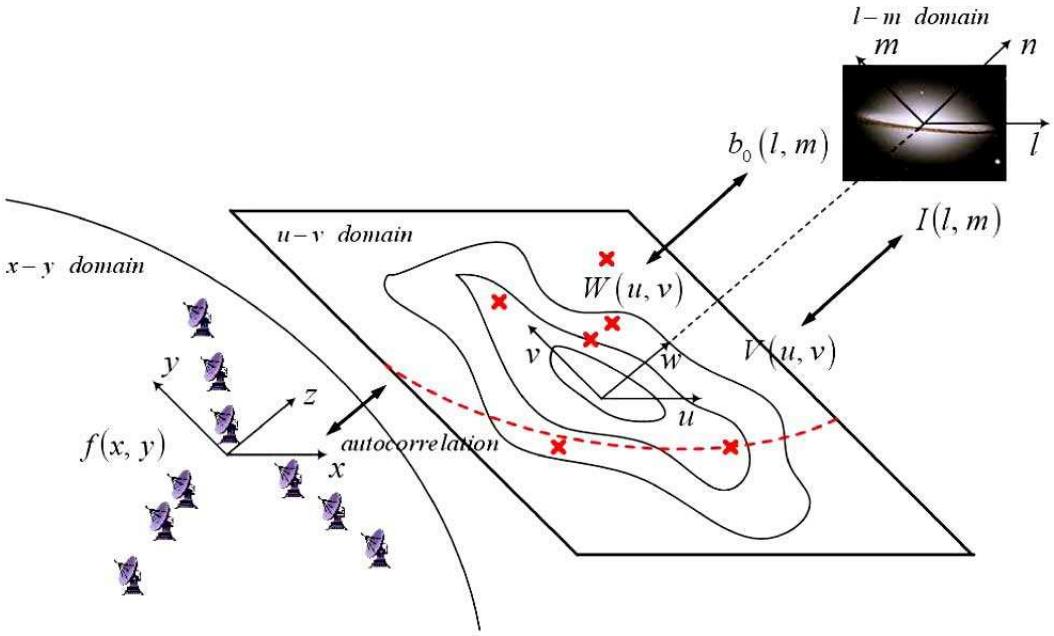


Figure 19. *Radio Astronomy* - Conceptual sketch of a radio astronomical measurement using a correlator antenna array. The brightness distribution $I(l,m)$ in the angular domain is retrieved by the inverse Fourier transform of the samplings of its visibility $V(u,v)$ in the spatial frequency domain. The sampling points are determined by autocorrelating the array configuration $f(x,y)$ in the spatial domain [31].

The particular importance of introducing the concepts of visibility and the $u-v$ domain stems from the Fourier transform relationship between and given in

$$I(l,m) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} V(u,v)\exp\left[j2\pi\left(ul+vm\right)\right]dudv \tag{3.111}$$

that applies to most radio sources with the spatially incoherent feature in their emissions. In other words, the **visibility represents the spatial frequency spectrum** of a **radio source**. The radio astronomical measurement described in Fig. 19, therefore, resembles the microwave holographic imaging in the sense that $I(l,m)$ can be retrieved by the inverse Fourier transform of the sampled components of $V(u,v)$.

Fig. 20 provides a quantitative description of the measurement and summarizes all Fourier transformation pairs between the $u-v$ domain and the $l-m$ domain. Similar to the transient response of a system in signal processing, the point source responses of the array in the $u-v$ domain and the $l-m$ domain are characterized by the $u-v$ coverage $W(u,v)$ and the synthesized beam $b_0(l,m)$, respectively. This spatial-filter-like behavior is only valid when the output signals from antenna elements are pair-wisely processed,

which makes correlator arrays a better option than conventional sum arrays for the sake of signal-to-noise ratio (SNR) and data gathering efficiency [8][9][31].

In general, a uniform $u-v$ coverage is preferable if the array is aimed to observe very bright and complicated sources, and a synthesized beam with a low SLL might function better in extracting images out of noisy data. Although $W(u,v)$ and $b_0(l,m)$ are related as one of the Fourier transformation pairs in Fig. 20, there is not a rigorous proof that the most complete $u-v$ coverage leads to the optimal synthesized beam. The selection of an appropriate array configuration $f(x,y)$ has to be accomplished via the optimizations of $W(u,v)$ and $b_0(l,m)$ in different domains separately [8][9][31].

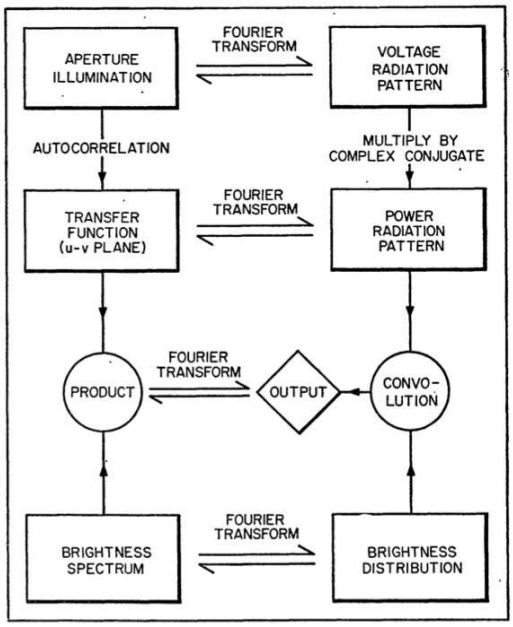


Figure 20. *Radio Astronomy* - Relationship among antenna quantities for an incoherent field.

3.7.3 The U-V Coverage

First let us consider how the array configuration $f(x,y)$ is related to the $u-v$ coverage, $W(u,v)$. Here we assume that two antenna elements are separated by a baseline vector

$$\overrightarrow{B} = \hat{x}u\lambda + \hat{y}v\lambda \tag{3.112}$$

and the antenna dimensions are much smaller than the length of baseline, $\left|\overrightarrow{B}\right|$. It has been shown that for a snapshot observation at zenith, the output signals of the correlator that connects the antenna pair are the sampled visibilities at symmetric spatial frequencies

(u, v) and $(-u, -v)$. Thus for an N-element correlator array with the ith element located at (x_i, y_i) and a configuration of

$$f(x, y) = \sum_{i=1}^{N} \delta\left(\frac{x - x_i}{\lambda}, \frac{y - y_i}{\lambda}\right) \tag{3.113}$$

$W(u, v)$ can be located by searching all baseline vectors via the autocorrelation of the array's configuration

$$\begin{aligned} W(u, v) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y) f(x - u\lambda, y - v\lambda)\, dxdy \\ &= \sum_{i=1}^{N}\sum_{j=1, j\neq i}^{N} \prod\left(u - \frac{x_i - x_j}{\lambda}, v - \frac{y_i - y_j}{\lambda}\right) \end{aligned} \tag{3.114}$$

Here $\prod(u, v)$ is a 2D unit impulse function defined by

$$\prod(u, v) = \begin{cases} 1; & u = v = 0 \\ 0; & elsewhere \end{cases} \tag{3.115}$$

The summation in (3.114) does not include $i = j$ terms since each antenna is **not correlated with itself**. The origin $(u, v) = (0, 0)$ is therefore not included in the $u-v$ coverage, while all other spatial frequencies satisfying $(u, v) = ((x_i - x_j)/(\lambda), (y_i - y_j)/(\lambda))$are sampled. Theoretically an N-element array should have $N(N-1)$ samplings in the $u-v$ domain for a snapshot observation, however, the actual number of samplings is often less than that due to the redundancy in the array configuration. An appealing solution to increase the number of sampling points is to apply a tracking observation in which the Earth rotation effect is incorporated [8][9][31].

3.7.4 The Earth-Rotation Effect

In a tracking observation, each baseline vector tracks an arc of an ellipse in the $u - v$ domain due to the rotation of the Earth. The axial ratio of ellipse and the length of arc are determined by the source declination δ, the elevation $\mathcal{E}$, the latitude $\mathcal{L}$ and the azimuth $\mathcal{A}$ of baseline, along with the observation time duration $2h$ ($h \in [0, 12]$, unit: hours). For each instant hour angle $H \in (-(h\pi)/(12), (h\pi)/(12))$ (unit: radians) during the observation, the particular sampling spatial frequency (u, v) at that instant is specified by a matrix equation

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \sin H & \cos H & 0 \\ -\sin\delta\cos H & \sin\delta\sin H & \cos\delta \end{bmatrix} \begin{bmatrix} X_\lambda \\ Y_\lambda \\ Z_\lambda \end{bmatrix} \tag{3.116}$$

where

$$\begin{bmatrix} X_\lambda \\ Y_\lambda \\ Z_\lambda \end{bmatrix} = \left|\overrightarrow{B}\right| \begin{bmatrix} \cos\mathcal{L}\sin\mathcal{E} - \sin\mathcal{L}\cos\mathcal{E}\cos\mathcal{A} \\ \cos\mathcal{E}\sin\mathcal{A} \\ \sin\mathcal{L}\sin\mathcal{E} + \cos\mathcal{L}\cos\mathcal{E}\cos\mathcal{A} \end{bmatrix} \tag{3.117}$$

Assume the time interval between taking two samplings is Δh, the total number of $u-v$ samplings is increased by a factor of $(h)/(\Delta h)$ compared to a snapshot observation [8][9][31].

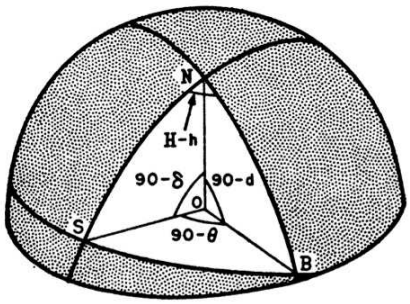


FIG. 5. The geometry of an interferometer.

Figure 21. *Radio Astronomy* - The geometry of an interferometer. The baseline intersects the celestial sphere at B, which has declination d and the local hour angle h. The source is at point S, with coordinates δ and H. The projection of the baseline on the intersection of the plane SOB and a plane tangent to the celestial sphere at S is $D\cos\theta$.

3.7.5 The Synthesized Beam

As shown in Fig. 20, the synthesized beam $b_0(l,m)$ is calculated by the inverse Fourier transform of the $u-v$ coverage $W(u,v)$. This relationship applies to both snapshot and tracking observations in which $W(u,v)$ is obtained using (3.114) and (3.116)-(3.117), respectively. Typically $b_0(l,m)$ is calculated by inverse fast Fourier transformation (IFFT), in which the $u-v$ domain is discretized into $N_u \times N_v$ rectangular grids each with a dimension of $\Delta u \times \Delta v$. Multiple $u-v$ samplings lying in each grid are averaged and relocated at the center of the grid [8][9][31].

The maximum FOV in the $l-m$ domain, L_{max} and M_{max} (in radians), are determined by the grid size Δu and Δv

$$L_{max} = 1/\Delta u, \quad M_{max} = 1/\Delta v \tag{3.118}$$

Under the assumption that the antenna dimension is much smaller than the length of baseline, the maximum FOV is far less than the half-power beamwidth (HPBW) of each

antenna element. The effect of element pattern in the synthesized beam is just multiplying a constant, which can be omitted when investigating a normalized pattern. The resolution of the beam plot is given by:

$$R_l = L_{max}/N_u, \quad R_m = M_{max}/N_v \tag{3.119}$$

By applying zero-padding in the $u-v$ domain, the resolution can be improved to obtain more detailed sidelobe features of $b_0(l,m)$.

Similar to applying amplitude tapering in low-sidelobe aperture antennas, a weighting function $w(u,v)$ is often imposed in the $u-v$ domain to suppress sidelobes of the synthesized beam. Therefore the beam calculation has a general form

$$b_0(l,m) \Leftrightarrow W(u,v)\, w(u,v) \tag{3.120}$$

where $\Leftrightarrow$ represents the Fourier transformation pair [8][9][31].

3.7.6 Image Retrieval

Operating as a spatial filter, the response of a correlator array to an extended source is obtained by a multiplication in the $u-v$ domain

$$V_{mea}(u,v) = W(u,v)\, w(u,v)\, V(u,v) \tag{3.121}$$

or a convolution in the $l-m$ domain

$$I_{mea}(l,m) = I(l,m) \star b_0(l,m) \tag{3.122}$$

as shown in Fig. 20. Since the RHS's of (3.121) and (3.122) are related by the Fourier transformation, the source image $I_{mea}(l,m)$ can be retrieved by the IFFT of sampled visibility $V_{mea}(u,v)$

$$V_{mea}(u,v) \Leftrightarrow I_{mea}(l,m) \tag{3.123}$$

For a better assessment of the array's performance, the image retrieval process can be simulated by specifying a source with a known brightness distribution $I(l,m)$ obtained from an actual astronomical measurement. In this paper, however, due to the lack of measured raw-data, the source is specified in the $u-v$ domain by applying benchmark visibility functions provided. For instance, the visibility of a 2-D Gaussian source is defined by

$$V(u,v) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\left(u^2+v^2\right)/2\sigma^2\right] \tag{3.124}$$

where σ^2 is the variance that modulates the angular width of the source. With this analytical form of $V(u,v)$, the exact value of sampled visibility $V_{mea}(u,v)$ at an arbitrary spatial frequency is calculated by (3.121) [8][9][31].

3.7.7 Basic Two-Elements Interferometer

An interferometer system can be expressed schematically in a fairly general way in Fig. 22. Two antennas, each with its amplifying system, are connected to a correlator (or multiplier), which includes an averaging or integrating circuit with a specified time constant that is much longer than the reciprocal of the frequency bandwidth of the system, so that many voltage impulses are averaged in a simple observation [8][9][31].

The interferometer is assumed to observe an extended source of incoherent and statistically radiation. The antennas are pointed in the same direction. For these conditions the output of the correlator is

$$r(\tau) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{\Gamma}(\xi', \nu) A_1(\nu) A_2^*(\nu) \hat{G}_1(\xi' - \xi, \nu) \hat{G}_2^*(\xi' - \xi, \nu) \exp(j2\pi\nu\tau) d\nu d\xi' \tag{3.125}$$

in which

- $r(\tau)$ is the output of the correlator
- $\hat{\Gamma}$ is the line-integrated brightness distribution of an isolated, finite source
- $\hat{A}$ is the frequency response of the amplifier
- $\hat{G}$ is the antenna voltage gain
- ν is the frequency (Hz)
- $\tau = \tau_g - \tau_i$ is the difference between in transit time from a plane wavefront in space to the correlator via the two possible paths
- τ_g is the geometrical component of τ
- τ_i is the instrumental component of τ
- $\xi = \sin\theta$

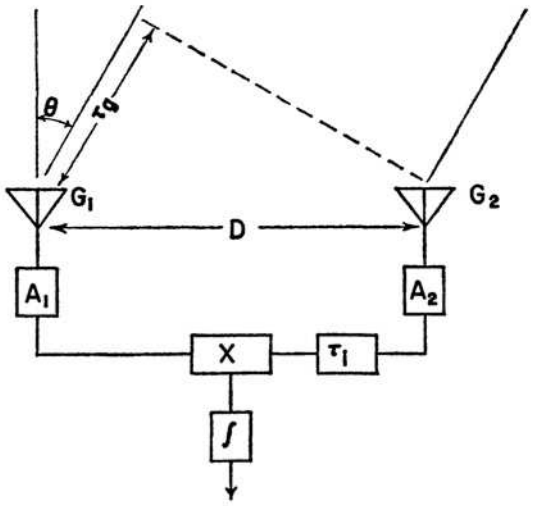


Figure 22. *Radio Astronomy* - Basic correlator interferometer system.

This formula is quite general. In the case of two identical antennas with identical, band-limited amplifiers it reduces to

$$r(\tau) = \int_{-\infty}^{\infty} d\xi' \int_{\nu_0-\Delta\nu/2}^{\nu_0+\Delta\nu/2} d\nu \hat{\Gamma}(\xi', \nu) |A(\nu)|^2 \left|\hat{G}(\xi' - \xi, \nu)\right|^2 \exp(j2\pi\nu\tau) \tag{3.126}$$

The time delay τ is the difference between the geometrical delay τ_g and is the instrumental delay, τ_i. The instrumental delay is adjusted to the value $D\xi_1/c$, so that

$$\tau = \frac{D(\xi' - \xi_1)}{c} \tag{3.127}$$

where D is the separation of the antennas in meters and c is the velocity of the wave in space. If the amplifier passband $\Delta\nu$ is sufficiently small, so that the antenna pattern and the brightness distribution do not vary significantly over the band, Equation 3.126 can be written

$$r(\xi_0, \xi_1, D) = \int_{-\infty}^{\infty} \hat{\Gamma}(\xi', \nu) \hat{P}(\xi_0, \xi', \xi_1) d\xi \tag{3.128}$$

where ξ_0 is the direction in which the antennas are aimed and ξ_1 is the direction for which $\tau = 0$. The function $\hat{P}(\xi_0, \xi', \xi_1)$ is the product of the antenna power pattern $\left|\hat{G}(\xi_0 - \xi', \nu_0)\right|^2$, the bandwidth pattern (or delay pattern)

$$B(\xi_1 - \xi', \Delta\nu, D) = \int_{-\Delta\nu/2}^{\Delta\nu/2} |A(\nu)|^2 \exp[-j2\pi\nu(\xi_1 - \xi')D/c] d\nu \tag{3.129}$$

and the interference pattern

$$F(\xi_1, \xi', D) = \exp[-j2\pi\nu_0(\xi_1 - \xi')D/c] \tag{3.130}$$

The bandwidth pattern has a peak in the direction ξ_1. When the source and the antenna beamwidth are of small angular extent, the integrand in Equation (3.129) is nonzero over only a small range of θ centered at θ_0. The instrumental delay can be adjusted to the value $D\xi_0/c$ so the delay pattern also has a peak at ξ. Now let θ' be defined as $\theta_0 - \theta$; then θ is small and

$$\xi \simeq \sin\theta_0 - \cos\theta_0 \sin\theta = \xi_0 - \xi\cos\theta_0$$

Define u as $(D\cos\theta_0)/\lambda_0$. This is the *spatial frequency* and is the component of the baseline (in wavelengths) in the direction normal to θ_0.

Equation (3.128) can be rewritten

$$r(u) = \int_{-\infty}^{\infty} \hat{\Gamma}(\xi, \nu_0)\, \hat{P}(\xi, u, \Delta\nu)\, d\xi \tag{3.131}$$

Now let us examine the form Equation (3.131) assumes when the bandwidth is narrow enough so that for all baselines the bandwidth pattern is much wider than the antenna pattern, and when source being observed is, in turn, small compared with the antenna pattern. In this case

$$r(u) = \int_{-\infty}^{\infty} \hat{\Gamma}(\xi, \nu_0) \exp(-j2\pi\xi u)\, d\xi = \hat{\gamma}(u, \nu) \tag{3.132}$$

This will be called the "fringe function". It is the Fourier transform of the brightness distribution, and it is apparent, therefore, that the interferometer can be used to make a Fourier analysis of the structure. This is the basis of aperture synthesis. It is seen from (3.132) and the definition of u that the spatial frequency measured with a given baseline is the baseline length, in wavelengths, projected on a plane tangent to the celestial sphere at the location of the source. By using a sufficient number of different baselines, enough Fourier components can be measured to permit the reconstruction of the source by Fourier transformation [8][9][31].

It has been assumed that the source is finite, in fact, that is small compared with the antenna beam. A source of extent $\Delta\xi$ can be completely represented by sampling its spatial frequency spectrum at intervals $u = 1/\Delta\xi$. This follows from the basic properties of the Fourier series representation of a function with a finite base. Furthermore, if the smallest detail to be measured is $\Delta\xi_m$. Thus, the number of baselines needed to perform a complete, one-dimensional analysis on a source is equal to the width of the source divided by the width of the finest detail that is to be resolved. A two-dimensional analysis requires a number of baselines equal to the square of the number for one dimension.

A Fourier series with *discrete, uniform spacing* of the terms *in the frequency domain* is a *periodic function of the spatial coordinate.* If one-dimensional antenna is synthesized

by means of a series of interferometers whose *baselines increase* successively in length by a *uniform interval*, the response to a point source is a *comb-shape series* of *evenly spaced spikes* in the ξ dimension. In an actual observation, an isolated single source can be mapped accurately by this means. If there are other sources present, however, the map of the source under investigation may be seriously distorted by their interactions with the higher-order responses, which are usually termed "grating lobes". The spacing of the responses in the ξ domain is inversely proportional to the increment of the baseline spacing in the u domain; therefore, it is important to plan the observing program according to the nature of the source under investigation. In a two-dimensional synthesis operation, there will be a two-dimensional array of grating lobes, of which examples will be seen [8][9][31].

In the Fourier-series method of aperture synthesis, it is necessary to measure each component of the series only once. If several antennas are available, together with the necessary electronics to permit simultaneous operation of several baselines, the most economical arrangement of the antennas is one which provides the largest number of necessary baselines with the minimum number of duplications. It is possible to arrange four antennas on a straight line in such a way that there are no redundant baselines; but for larger number of elements and for two-dimensional arrays redundancies are inevitable [40].

The aperture illumination is the distribution of the electric field in the plane of the antennas. In a synthesis array consisting of a small number of antennas, for example, the illumination would consist of a number of discrete points in the aperture plane. The autocorrelation function of the illumination is called the transfer function. The Fourier transform of the brightness distribution (in spatial coordinates) is the brightness spectrum (in spatial frequency terms), and the product of the brightness spectrum and the transfer function is the output in terms of spatial frequencies: that is, the observed brightness spectrum, whose Fourier transform is the conventional radio telescope output. Only those spatial frequency components are present in the output which are also present in the transfer function; thus, the performance of the synthetic telescope can be investigated by examinating its transfer function. The transfer function has the same configuration as the diagram of the antennas in the u dimension, or in the $u - v$ plane in the case of a two-dimensional array.

3.7.8 Comparison between Conventional Sum Arrays and Correlator Arrays

In early radio astronomical measurements in 1940s, conventional two-element sum arrays are used as an alternative for 1-D and 2-D image retrieval of radio sources. Different spatial

frequencies are sampled by varying the baseline between two antenna elements. On the other hand, multi element sum arrays are usually used as a probe for directly mapping the source in the angular domain, and the direction of probe is steered by applying phased array techniques. Here we compare both types of sum arrays to correlator arrays, and it will be shown that correlator arrays have unique advantages in both noise reduction and data gathering efficiency [9][31].

Fig. 23(a) shows a schematic diagram of a two-element sum array. The voltage signals from both antennas are summed and squared by a square-law detector, and the output of the detector is low-pass filtered before being recorded. Assume the signal voltage from antenna I is $V\sin(2\pi f_0 t)$. The output of antenna II is therefore delayed by $\tau = \left(\overrightarrow{B}\cdot\overrightarrow{s}\right)$, where $\overrightarrow{B}$ is the baseline vector, $\overrightarrow{s}$ is the unit vector pointed to the source and c is the wave velocity in free space. Noticing that $u = \left(\overrightarrow{B}\cdot\overrightarrow{s}\right)/(\lambda)$. The output of the square-law detector is

$$W'(u) = \left\{V\sin(2\pi f_0 t) + V\sin\left[2\pi f_0\left(t - \frac{u\lambda}{c}\right)\right]\right\}^2 \tag{3.133}$$

By filtering harmonics of $2\pi f_0 t$, which represent radio frequencies, the output of the sum array is

$$W(u) = V^2\left[1 + \cos\left(\frac{2\pi f_0 u\lambda}{c}\right)\right] \tag{3.134}$$

For a certain radio source, the cosine term in (3.134) is a function of u only and represents the spatial frequency to which the array responds. It is not filtered out since varies slowly as the Earth rotates. However, due to the noise power which is typically several orders of magnitude greater than the signal from the source, the large offset represented by the constant term in (3.134) is desired to be removed.

In the two-element correlator array shown in Fig. 23(b), output signals of two antenna elements are multiplied and time-averaged, namely, **correlated**. Using the same expressions as those in (3.133), the output of the multiplier is

$$\begin{aligned} W'(u) &= V^2\sin(2\pi f_0 t)\sin\left[2\pi f_0\left(t - \tfrac{u\lambda}{c}\right)\right] \\ &= \tfrac{V^2}{2}\left[\cos\left(\tfrac{2\pi f_0 u\lambda}{c}\right)\right. \\ &+ \cos(4\pi f_0 t)\cos\left(\tfrac{2\pi f_0 u\lambda}{c}\right) \\ &+ \left.\sin(4\pi f_0 t)\sin\left(\tfrac{2\pi f_0 u\lambda}{c}\right)\right] \end{aligned} \tag{3.135}$$

The second and the third terms in (3.135) vanish after being time averaged. Therefore the output of the correlator is

$$W(u) = \frac{V^2}{2}\cos\left(\frac{2\pi f_0 u\lambda}{c}\right) \tag{3.136}$$

with the cosine term remains only [9][31].

For a sum array with elements, since the output signals of all elements are summed up, it is **not feasible** to identify the $u - v$ domain response of the array. Using such a multi element sum array, a radio source is usually mapped in $l - m$ domain by convolving its power pattern and the brightness distribution of the source, and only one data is obtained at any instant. In this case, what contributes most to the convolution is the brightness distribution within a small angular region determined by the narrow beam formed by the array. In order to achieve a radio map within a reasonably large angular region, the main beam of the array must be phase-steered. On the other hand, a correlator array responds to the entire FOV by sampling multiple spatial frequency components simultaneously. It is therefore more efficient than a sum array in gathering data for mapping purpose [9][31].

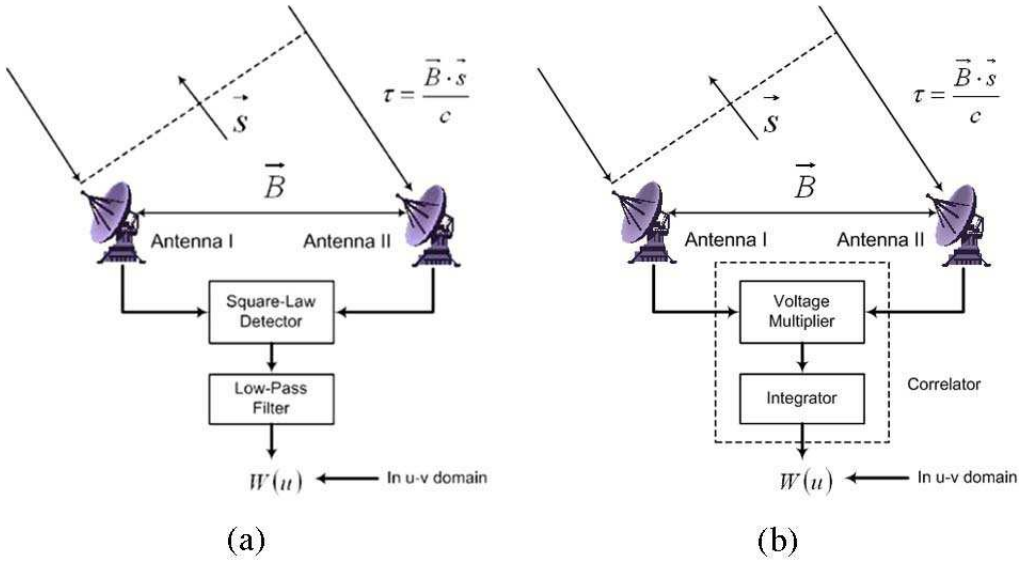


Figure 23. *Radio Astronomy* - Comparison between the signal processing schemes of a 2-element: (a) sum array and (b) correlator array.

3.8 Particle Swarm Optimization for Radio Astronomy [31]

3.8.1 Introduction

Compared to conventional deterministic and pseudodynamic programming techniques discussed in other works, the PSO optimizer provides more flexibilities to optimize the array performance in both the $u-v$ domain and the $l-m$ domain, by performing statistical explorations in high-dimensional, non-linear solution spaces. Benchmark examples are presented to illustrate its effectiveness in designing correlator arrays with typical open-ended and closed configurations such as the "Y" and the Reuleaux triangle, by obtaining optimal arrays that outperform uniform arrays and representative existing designs [31].

3.8.2 A Numerical Example: A Uniform Y-Shaped Array

By utilizing basic formulations (3.113)-(3.123), an analyzer is developed to calculate the $u-v$ coverage and the synthesized beam of a correlator array with an arbitrary configuration $f(x,y)$. Let us take a 27-element array is constructed on a Y-shaped rail track, which is a representative open-ended configuration similar to the Very Large Array (VLA) at Socorro, New Mexico ($\mathcal{L}=34°$, $\mathcal{E}=0°$). The entire array is rotated by 5° from the north-south direction to achieve a better $u-v$ coverage for observations at low declinations. Each arm of the "Y" extends up to 21 km and each antenna element is a 25m-diameter parabolic reflector. The ratio between the maximum baseline ($B_{max}=21\sqrt{3}$km) and the dimension of each individual antenna element is approximately 1400.

A Gaussian source with the visibility specified in (3.124) is used to test the image retrieval capability of the array. The variance of the Gaussian function is selected as $\sigma=(B_{max})/(8\lambda)$and the original source image $I(l,m)$ is plotted in Fig. 24(a) by the inverse Fourier transform of (3.124) at 128×128 FFT grids. The image plot is scaled from -30dB to 0dB [31].

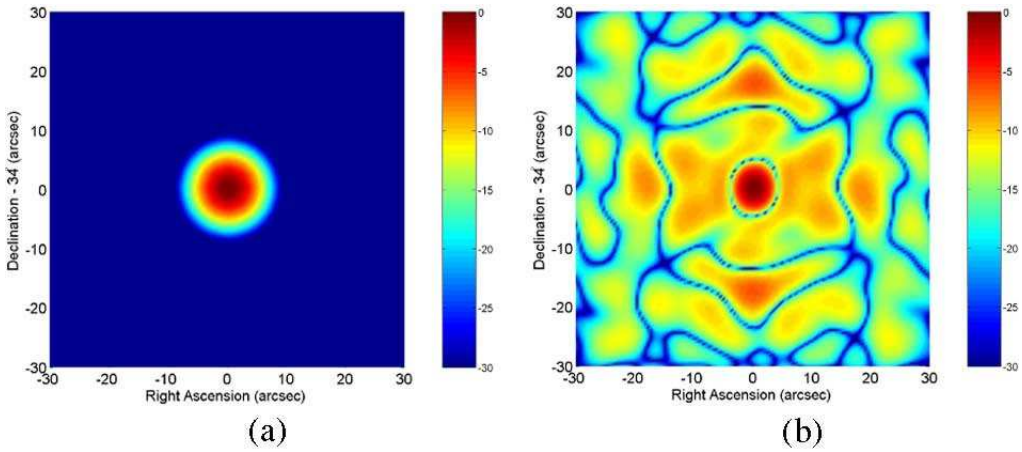


Figure 24. *Radio Astronomy* - (a) Original source image with the visibility specified by the Gaussian function in (3.124). (b) Image retrieved by the uniform Y-shaped array shown in Fig. 4(a).

3.8.3 Optimization of Y-Shaped Arrays

3.8.3.1 The Particle Swarm Optimization Technique

PSO is a recently proposed evolutionary algorithm that addresses both continuous and discrete optimizations by applying the swarming behavior in the nature. The basic principle of PSO is to iteratively explore the solution space using a swarm consists of multiple agents. Each agent represents a candidate design and its performance is quantified by a fitness function representing the goal of optimization. At each iteration, all agents interchange the information of the best design that has ever been found. Each agent is navigated by its own experience and the knowledge from other agents. This procedure repeats until the swarm converges to the global optimum. Being applied to a large variety of practical electromagnetic applications, a robust PSO optimizer has been developed [31].

The PSO algorithm is applied in this section to optimize element positions on each arm of the "Y" in order to reduce the redundancy in the $u-v$ coverage and suppress the sidelobes in the synthesized beam. The number of antenna elements in each optimization is fixed to be 27, and the candidate design has a three-fold symmetry (i.e., the nine elements on each arm have the same distribution) to guarantee a good azimuthal -distribution. To maintain the same B_{max} of $21\sqrt{3}$km, it is also assumed that there is always an element located at the end of each arm. Therefore the array configuration is represented by an eight-dimensional real vector

$$\overrightarrow{x} = [x_1, x_3, .., x_8] \tag{3.137}$$

in which $x_i \in (0, 21)$ (unit: kilometers) represents the radial displacement of the ith

element from the center of the array. The optimization is performed over $\overrightarrow{x}$ and minimizes the fitness functions discussed in following subsections depending on different optimization goals [31].

3.8.3.2 Optimizing the U-V Coverage

The first-order requirement of optimizing the snapshot $u-v$ coverage is to reduce the redundancy while maintaining the uniformity of $u-v$ samplings.

The fitness function can be therefore defined as

$$f = -N_{sampled} \tag{3.138}$$

to maximize the number of sampled grids. The negative sign is used due to the default setting of PSO as a minimizer.

The optimization is executed using a 10-agent swarm for 500 iterations. The optimized array (denoted by Y_1) and its $u-v$ coverage are plotted in Fig. 25(a) and (b), respectively. The radial displacements of nine elements on each arm are tabulated in Table II. The fixed element at the end of each arm is denoted x_9 as and it has a constant radial displacement of 21 km. Compared to the uniform Y-shaped array, the $u-v$ samplings are distributed in 558 grids with 24 more sampled grids obtained. More importantly, there are no more overlapping samplings in the arm directions due to the slight perturbation induced into the uniform element distribution.

In order to verify the robustness of the optimizer, 10 independent optimizations are performed using the fitness function defined in (3.138). All these trials converge to the same optimal design shown in Fig. 25(a) and the $u-v$ coverage with 558 sampled grids is the best result that can be achieved. It is worthwhile to mention that, although the ideal number of 702 sampled grids is used as the target for optimizing element positions, it is not possible to achieve this exact number since there is not such a function $f(x,y)$ whose autocorrelation is completely flat in the $u-v$ domain [31].

A similar criterion is applied to optimize the $u-v$ coverage for an 8-hour tracking observation, with the only difference in selecting the value of N_{grid} defined in (**??**). Under the same observation condition previously mentioned ($h = 8$ hours, $\Delta t = 5$ minutes), the total number of $u-v$ samplings is increased by a factor of $(h)/(\Delta t) = 96$. Ideally N_{grid} should be increased by a factor of $\sqrt{96}$, however, the number of FFT grids (128×128) in the mapping procedure is comparable to the number of $u-v$ samplings in this case. A $N_{grid} = N_u = N_v = 128$ is therefore selected to achieve more sampled FFT grids. The fitness function is defined to be similar to (3.138). The same optimization setup of 10

agents and 500 iterations is applied. By incorporating the analysis of Earth rotation effect in each fitness evaluation, the total optimization time is increased to about 20 minutes [31].

The configuration of optimized array (Y_2) and its tracking $u-v$ coverage are plotted in Fig. 26(a) and (b), respectively. The optimized element locations are also tabulated in Table I. In order to represent the number of sampled grids in a concise manner with such a large N_{grid}, we define the filling ratio of the $u-v$ domain as

$$R = \frac{A_s}{A_c} \tag{3.139}$$

where A_s is the total sampled area and A_c is the area of the big circle of the six-point star. Compared to the uniform array, the filling ratio of array Y_2 is increased from 68.9% to 86.5% by non-uniformly locating antenna elements on each arm [31].

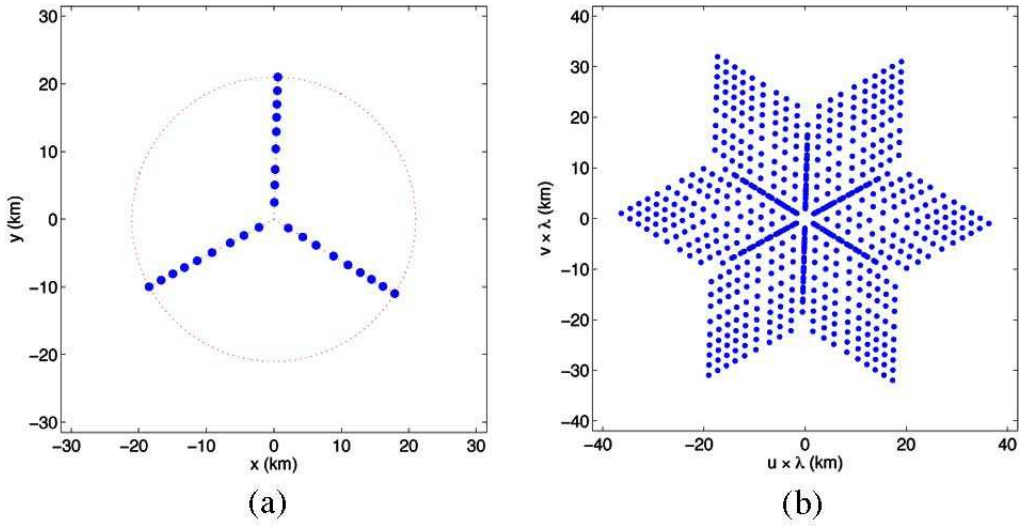


Figure 25. *Radio Astronomy* - (a) Configuration of the optimized 27-element Y-shaped array (Y_1) for the maximum snapshot $u-v$ coverage. (b) Snapshot $u-v$ coverage of Y has 558s sampled grids.

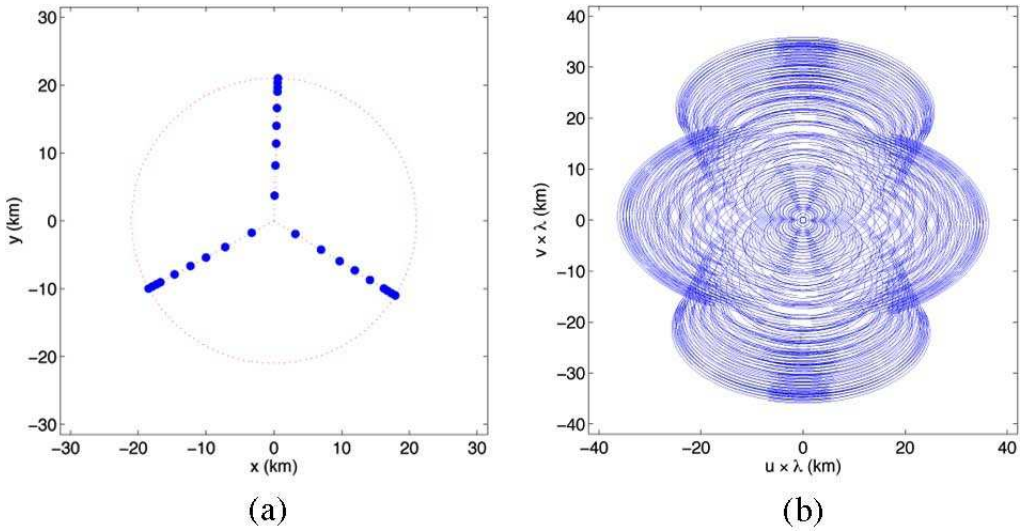


Figure 26. *Radio Astronomy* - (a) Configuration of the optimized 27-element Y-shaped array (Y_2) for the maximum tracking $u-v$ coverage. (b) Tracking $u-v$ coverage of Y_2 has a filling ratio of 86.5%, as defined in (3.139).

3.8.3.3 Optimizing the Synthesized Beam

In order to suppress the sidelobes in the synthesized beam, the peak sidelobe in the 2-D beam plot is identified and a fitness function is defined as:

$$f = \max\left[b_0\left(l,m\right)\right] \ \textit{in sidelobe region} \tag{3.140}$$

The beam is calculated based on the $u-v$ coverage of the 8-hour tracking observation, and a -15dB Gaussian tapering is applied to the $u-v$ samplings as previously mentioned. Fig. 27(a) and (b) plot the optimized array configuration (Y_3) and its synthesized beam using 10 agents for 500 iterations. The element locations of the optimal design are presented in Table II.

Array (Y_3) also has good sidelobe features for other source declinations. As shown in Fig. 28, the optimized array outperforms the uniform Y-shaped array in a wide range of source declinations from $+30°$ to $+80°$ with SLLs around or lower than -18 dB. The deteriorated SLL when tracking a source at $+90°$ is possibly due to the redundancy resulted by the three-fold symmetry. A better SLL at $+90°$ can be achieved by optimizing an array with asymmetrical element distributions on each arm [31].

Design	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
Uniform Array	2.33	4.67	7.00	9.33	11.67	14.00	16.33	18.67	21.00
Array Y_1	2.49	5.05	7.36	10.40	12.92	15.05	17.00	18.97	21.00
Array Y_2	3.72	8.15	11.36	13.99	16.64	19.03	19.67	20.37	21.00
Array Y_3	3.32	3.93	5.49	9.30	11.44	15.48	17.04	20.11	21.00

Table II. *Radio Astronomy* - Radial Element Displacement of Optimized Y-Shaped Arrays (Unit: Kilometers).

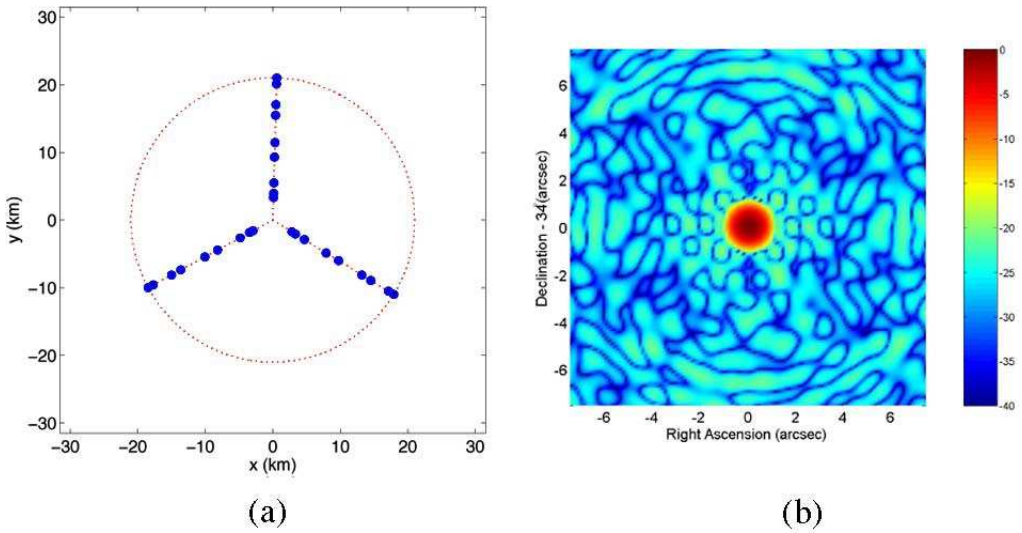


Figure 27. *Radio Astronomy* - (*a*) Configuration of the optimized 27-element Y-shaped array (Y_2) for the lowest SLL. (*b*) Synthesized beam of Y has a peak SLL of -20.3 dB.

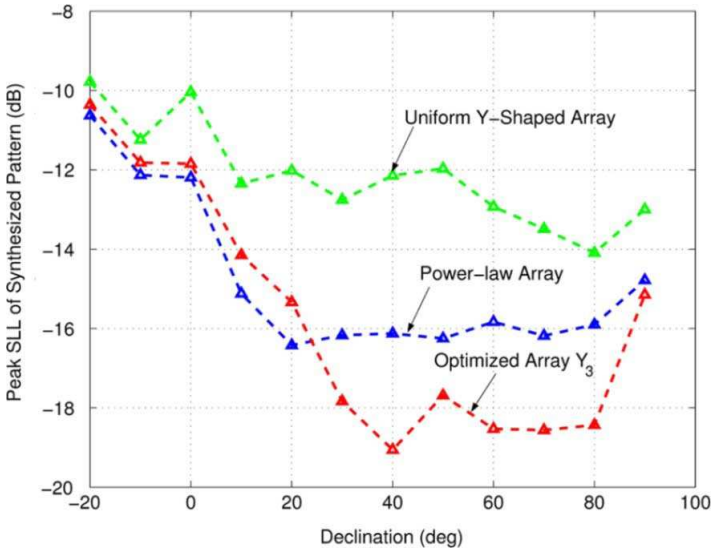


Figure 28. *Radio Astronomy* - Comparison between a uniform array, a power-law array ($\alpha = 1.7$) and the optimized array Y_3 for SLLs in 8-hour tracking observations with different source declinations.

3.8.3.4 Benchmark Comparisons

Table II compares performances of the uniform and three optimized Y-shaped arrays. It is quite obvious that each of (Y_1), (Y_2) and (Y_3) only outperforms other designs in the snapshot $u-v$ coverage, the tracking $u-v$ coverage and the peak SLL of the synthesized beam [the peak SLL referred in Tables II and IV corresponds to the maximum of b_0 in sidelobe region, as defined in (3.140)], respectively. By realizing that these design goals are not directly interrelated to each other, it is appropriate to justify here the advantage of applying PSO to correlator antenna array designs.

First of all, PSO provides a flexible optimization platform to accommodate different circumstances that might be encountered in practical astronomical measurements. Since the only input required by the optimizer is the fitness value, a large variety of design goals can be approached by simply applying different fitness functions without significantly modifying the optimizer itself. On the other hand, in some conventional optimization methods such as the gradient-based method, antenna locations are directly manipulated according to the distribution density function of snapshot $u-v$ samplings, which makes the methodology not as effective for optimizing the synthesized beam.

Secondly, the fitness functions elaborated in (3.138) and (3.140) enable the optimizer to be more effective in obtaining the desired $u-v$ coverage and synthesized beam. For instance, the snapshot $u-v$ coverage of a circular array is optimized by maximizing the summation of $u-v$ separations using simulated annealing (SA). In order to test its applicability in designing Y-shaped arrays, we did four comparative optimizations in PSO using the same fitness function of

$$f = \sum_{j,k;j\neq k}^{M_B} \left| \overrightarrow{B}_j - \overrightarrow{B}_k \right| \tag{3.141}$$

and different element numbers of $N = 9, 12, 18$ and 27. Here $\overrightarrow{B}_j$ and $\overrightarrow{B}_k$ represent the jth and the kth baseline vector, respectively; and $M_B = (N(N-1))/(2)$ is the total number of baselines [31].

Finally, let us consider the actual VLA configuration designed by the power-law, in which the ith element's position is defined by (unit: kilometers)

$$x_i = 21 \times \left(\frac{i}{9}\right)^{\alpha} \tag{3.142}$$

where $\alpha = 1.7$. In PSO-optimized arrays discussed above, it is interesting to notice that Y_2 resembles a "reversed" version of the power-law design, which has more antenna elements concentrated near the center rather than near the edges. In fact, the power-law-based

design is selected largely for reasons of economy. By selecting a proper α, the total number of antenna stations along the rail track is significantly reduced by sharing some stations between multiple array configurations with different scales. However, under the particular observation conditions considered in this chapter, the highly-condensed elements near the edge in the configuration of Y_2 compensate the Earth-rotation effect more efficiently. It is observed in Table II that the filling ratio of the power-law design is only 59.8%, which is even worse than the uniform Y-shaped array. Moreover, its synthesized beam is also outperformed by the optimized low-SLL design Y_3 in a wide range of source declinations, as shown in Fig. 29 [31].

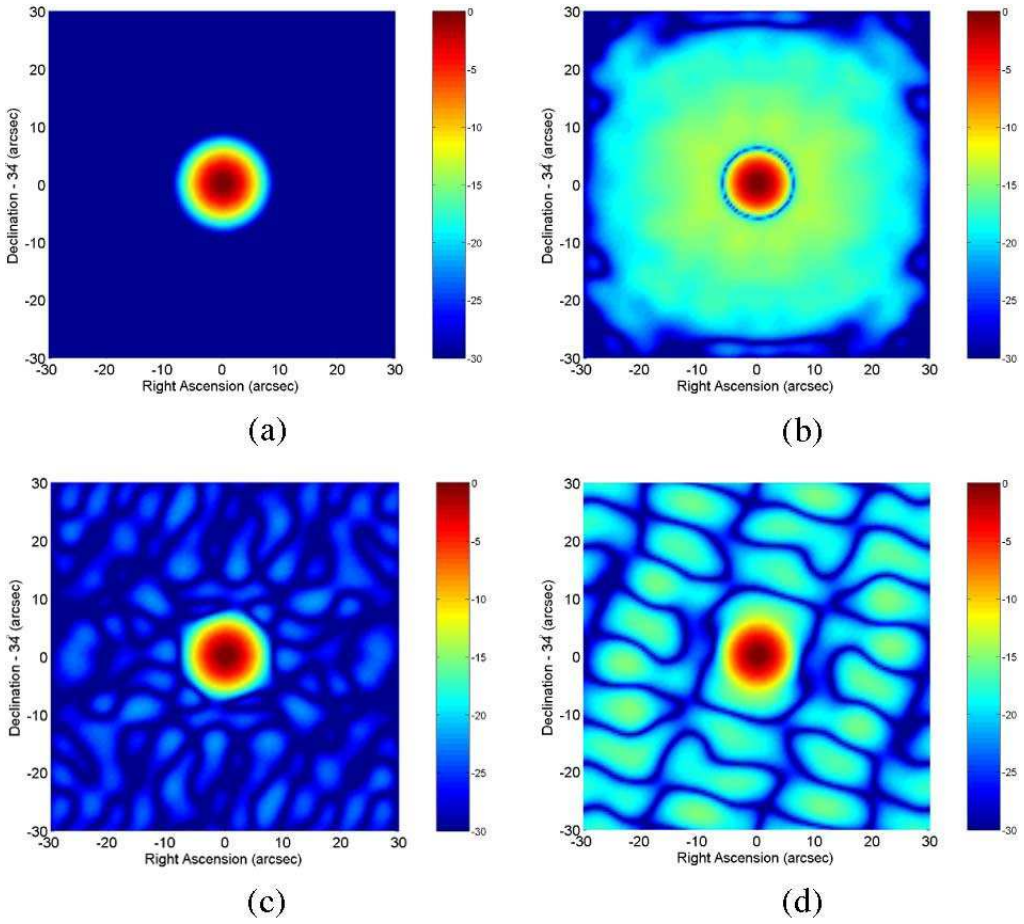


Figure 29. *Radio Astronomy* - (*a*) Original image of a Gaussian source and retrieved images by (*b*) array Y_1, (*c*) array Y_2 and (*d*) array Y_3 . The best image is retrieved by optimized array Y_2.

Chapter 4

Rectangular Thinned Arrays Based on McFarland Difference Sets

4.1 Introduction

ARRAY systems for frequency-modulated continuous-wave (FMCW) radars and SAR applications usually have to exhibit different total main beam widths (TMBWs) in azimuth and elevation and low PSLs [41], [15]. To meet these requirements and provide suitable resolutions, large rectangular layouts are needed [41], [15]. Since large fully-populated rectangular arrangements can yield to unacceptable high costs, weight, power consumption, and feeding network complexity [1], [42], architectural solutions with a reduced number of elements over large apertures with satisfactory PSLs and TMBWs values are often preferred. Towards this end, thinning techniques are generally exploited [1], [42] even though their main drawback is a lower sidelobe control when compared to their filled counterparts [1], [42]. In order to overcome such a limitation, several approaches have been proposed including the random displacement of the array elements [3], [6], the dynamic programming [43], and the stochastic optimization [44]-[45]. In such a framework, analytical techniques seem to be promising tools because of their numerical efficiency and the PSL control [19], [5]. By exploiting the auto-correlation properties of binary sequences, such as difference sets (DSs) [19]-[21] or almost difference sets (ADSs) [22]-[46], a regular and a-priori predictable behaviour of the sidelobes is guaranteed [47]. Unfortunately, only specific geometries and array sizes can be synthesized [5], [25], [48]. Despite the availability of quite large DS-ADS repositories [49]-[50], planar arrays based on DSs and ADSs are usually square [21], [48] or almost square [5], [48], while few examples of DS-based rectangular arrangements with different azimuth and elevation TMBWs are actually used (Following the approach

discussed in [19], [5], a rectangular DS array of size $N_1 \times N_2$ can be generated only if a 1D DS is available with length $N = 2^{r_1 r_2} - 1$ such that $N_1 = 2^{r_1} - 1$ and $N_2 = N/N_1$ are coprime and greater than one. Accordingly, only 6 of such sequences exist for $N_1 < 30$ corresponding to $N = \{15, 63, 255, 511, 1023\}$ [49], and only 3 these exhibit strongly different azimuth and elevation TMBWs [i.e. $(N_1 \times N_2) = \{(3 \times 85), (3 \times 341), (7 \times 73)\}$]). [41], [5]. In this paper, thinned rectangular arrays based on McFarland sequences [51], which are a particular class of DSs, are analyzed for the first time to the best of the authors' knowledge, and a suitable synthesis procedure based on a binary Genetic Algorithm (GA) [44] is proposed (McFarland sequences, likewise two-dimensional DSs [5], exhibit a two-level autocorrelation function). It is worthwhile to point out that the exploitation of such a class of DSs enables the extension of the design approach proposed in [19], [5] to rectangular layouts of size (being a prime number) with different azimuth and elevation TMBWs.

The outline of the chapter is as follows. Section 4.2 introduces McFarland sequences and their application to array thinning. Afterwards, the GA-based synthesis technique for designing McFarland arrays is presented (Section 4.3) and a set of representative numerical results concerned with different apertures and thinning factors is provided (Section 4.4) to show features, potentialities, and limitations of the proposed thinning strategy. An Appendix is present in Section 4.5.

4.2 Mathematical Formulation

Let us consider a two-dimensional regular lattice of $P \times Q$ positions spaced by s_x and s_y wavelengths along x and y, respectively. The array factor of a thinned arrangement defined over such a lattice is equal to [42]

$$F(u,v) = \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} d(p,q) \exp\left[j2\pi \left(p s_x u + q s_y v\right)\right] \tag{4.1}$$

$u = \sin(\theta)\cos(\phi)$ and $v = \sin(\theta)\sin(\phi)$ being the direction cosines. Moreover, $d(p,q)$ is the McFarland binary thinning sequence[48]

$$d(p,q) = \begin{cases} 1 & (p, q\rfloor_P, q\rfloor_{P+2}) \in \mathbf{M} \\ 0 & otherwise \end{cases} \qquad p = 0, ..., P-1, q = 0, ..., Q-1 \tag{4.2}$$

where P is a prime number, $Q = P(P+2)$, $\mathbf{M}$ is a McFarland DS [51] with indexes $N = P^2(P+2)$, $N = P(P+1)$ and $\Lambda = P$. Furthermore, $\cdot\rfloor_P$and $\cdot\rfloor_{P+2}$ stand for the

reminder of division by P and $P+2$, respectively. It is now worth noticing that several McFarland arrays can be generated for each P value. From the McFarland generation technique in the Appendix, it follows that a distinct DS, $\mathbf{D} = \{d(p,q), p = 0, ..., P-1, q = 0, ..., P(P+2)-1\}$, corresponds to ($a$) each value of the integer k in $[0, ..., P+1]$, (b) the set of $P+1$ vectors (a_t, b_t) $(t = 0, ..., P+1, t \neq k)$, and ($c$) the $P+1$ elements $(\hat{w}_1^{(t+1)}, \hat{w}_2^{(t+1)})$ $(t = 0, ..., P+1, t \neq k)$ used for deriving $\mathbf{M}$. As a result, up to $(P+2)! \times P^{2P+2}$ different McFarland sets can be generated for each prime P. In turn, each McFarland set defines up to $P^2(P+2)$ different layouts by performing cyclic shifts of the thinning matrix [5]

$$\mathbf{D}^{(\sigma_x,\sigma_y)}(p,q) = \Big\{ d\left[(p+\sigma_x)\rfloor_P, (q+\sigma_y)\rfloor_{P(P+2)} \right] p = 0, ..., P-1, q = 0, ..., P(P+2)-1 \\ p = 0, ..., P-1, q = 0, ..., P(P+2)-1 \Big\}$$

σ_x and σ_y being the shift indexes along the array axes. In conclusion, the total number of different McFarland arrangements generated for each P turns out to be

$$\Psi(P) = (P+2)^2 \times (P+1)! \times P^{2P+4} \tag{4.3}$$

where $(\cdot)!$ indicates the factorial.

As for the power pattern, a McFarland array defined over a rectangular grid of $P \times P(P+2)$ locations satisfies the following sampling property [5]

$$\left| F\left(\frac{k}{s_x P}, \frac{l}{s_y P(P+2)} \right) \right|^2 = \sum_{m=0}^{P-1} \sum_{n=0}^{P(P+2)-1} \chi(m,n) \times \exp\left[j2\pi \left(\frac{mk}{P} + \frac{ml}{P(P+2)} \right) \right] \tag{4.4}$$

where $\chi(m,n) \triangleq \sum_{m=0}^{P-1} \sum_{n=0}^{P(P+2)-1} d(p,q)\, d\left[(p,+m)\rfloor_P, (q+n)\rfloor_{P(P+2)} \right]$ is the two-valued period autocorrelation function of $\mathbf{D}$ [51] whose values are

$$\chi(m,n) = (K-\Lambda)\,\delta(m,n) + \Lambda$$

$$m = 0, ..., P-1,\ n = 0, ..., P(P+2)-1 \tag{4.5}$$

$\delta(m,n)$ being the delta function [i.e $\delta(m,n) = 1$ if $m = n = 0$ and $\delta(m,n) = 0$ otherwise]. As an example, Fig. 30(a) shows a McFarland array obtained for $P = 3$, while the corresponding autocorrelation reported in Fig. 30(b). From 4.4 and 4.5, it follows that the samples of the power pattern of McFarland arrays are *a-priori* known. Moreover, it has been proved in [5] that they produce patterns with much lower PSLs that are typical with cut-and-try random placement. More in detail, Monte Carlo simulations have shown that compared to a random (nonlattice) placement of elements on the same aperture, a DS array has an expected PSL improvement of $\approx 1.5 + 10\log_{10}(1-(K/N))^{-1}$[dB] [5].

In order to fully exploit the features of McFarland sequences for array thinning, a suitable synthesis procedure is presented in Section 4.3.

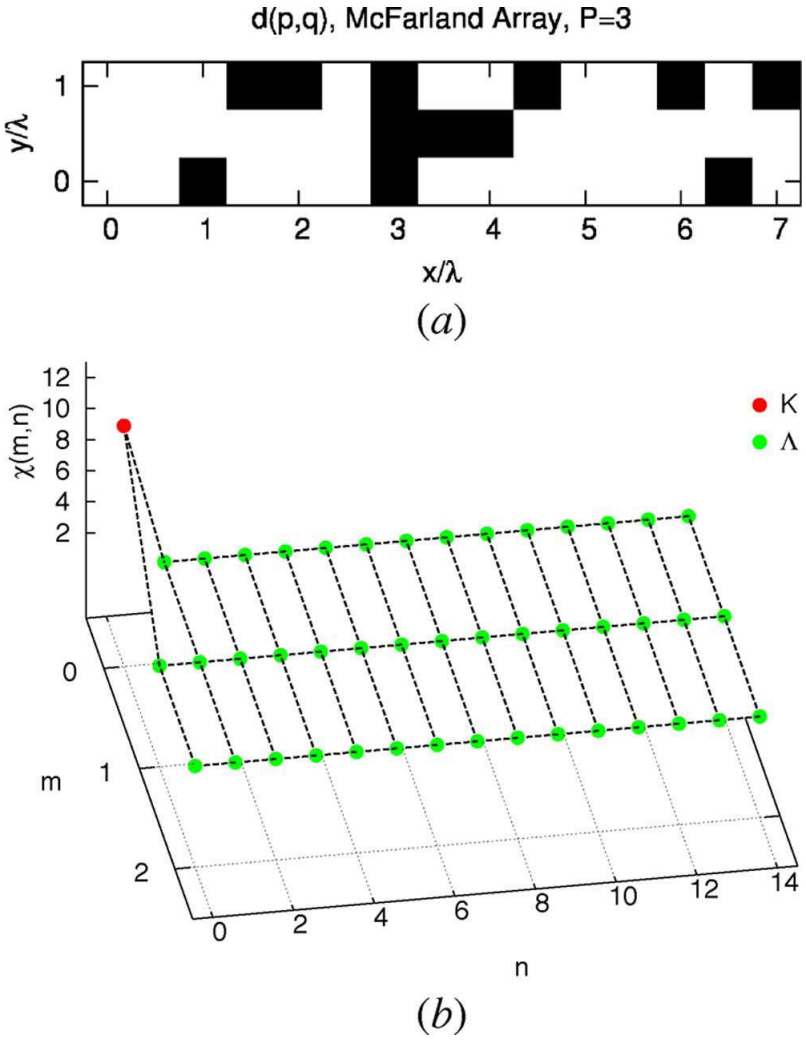


Figure 30. *McFarland Rectangular Arrays* - Example of (a) a McFarland array and (b) the associated (two-level) autocorrelation function ($P = 3$).

4.3 McFarland Array Synthesis Procedure

In order to find the optimal (i.e., with the lowest PSL) McFarland layout for every P value, all $\Psi(P)$ deducible arrays should be, in principle, analyzed. Unlike other 2D DS-based thinned architectures [21], an exhaustive procedure is here computationally unfeasible due to the extremely wide number of layouts even for small P values. As an example, more than $\Psi(P) \approx 2.15 \times 10^{14}$ McFarland arrays can be defined over a lattice of size

$P \times Q = 5 \times 35$ ($P = 5$ - Table I). As a consequence, a different and more efficient selection approach is mandatory to analyze the PSL properties of these arrangements for identifying the optimal layout.Towards this end, the problem of finding the optimal McFarland array among all existing $\Psi(P)$ layouts for a given is recast as an optimization one where the fitness function to be minimized is defined as follows

$$\Phi(\mathbf{D}) \triangleq PSL\{\mathbf{D}\} \tag{4.6}$$

where

$$PSL\{\mathbf{D}\} \triangleq \frac{\max\limits_{(u,v)\in\Omega} |F(u,v)|}{|F(0,0)|^2} \tag{4.7}$$

Ω being the sidelobe region [21].

Because of the discrete nature of the descriptors of the McFarland sets [i.e., σ_x, σ_y, k, (a_t, b_t) and $\left(\hat{w}_1^{(t+1)}, \hat{w}_2^{(t+1)}\right)$ for $t = 0, ..., P+1$, $t \neq k$], a binary GA-based approach [24], [44] is exploited. More specifically, the following procedure is iteratively applied.

1. **Initialization** $(i = 0)$ - A randomly-chosen initial population of C trial solutions (or *individuals*), $\rho_c(i)$, $c = 1, ..., C$ is defined;

2. **Coding** - Each individual $\rho_c(i)$ encodes the values of the McFarland integer descriptors $\sigma_x \in [0, P-1]$, $\sigma_y \in [0, P(P+2)-1]$, $k \in [0, P+1]$, (a_t, b_t) $(a_t \in [0, P-1], b_t \in [0, P-1], t = 0, ..., P+1, t \neq k)$ and $\left(\hat{w}_1^{(t+1)}, \hat{w}_2^{(t+1)}\right) \in \mathbf{V}_{t+1}$ $\left(\hat{w}_1^{(t+1)} \in [0, P-1], \hat{w}_2^{(t+1)} \in [0, P-1], t = 0, ..., P+1, t \neq k)\right)$ into a binary string (or *chromosome*);

3. **GA-Evolution** - At each i-th iteration, the genetic evolution takes places through *selection, crossover, reproduction, mutation* and *elitism* operators [24], [44] taking into account the fitness values $\Phi_c = \Phi_c\{\rho_c(i)\}$, $c = 1, ..., C$ of current trial solutions;

4. **Termination** - The iterative optimization terminates when the optimal fitness value, $\Phi_{POP}(i) = \min_c \Phi_c\{\rho_c(i)\}$, is smaller than an user-defined threshold or when a maximum number of iterations I_{max} has been reached. Then, the fittest trial individual $\bar{\rho} = \arg_\rho \left\{\min_i \left(\min_c \left[\Phi_c\{\rho_c(i)\}\right]\right)\right\}$ is assumed as the *optimal solution* (i.e., the optimal setup for the McFarland descriptors). Otherwise, the iteration index is updated $(i \rightarrow i+1)$ and goto 3.

It is worth to point out that, unlike [18], [24], the objective of the GA procedure is here not to design an optimally thinned array, but the search of the fittest arrangement in terms of PSL among all available McFarland layouts for a given P.

4.4 Numerical Results and Discussion

This section is aimed at *(a)* numerically assessing the features and the potentialities of the McFarland rectangular layouts and *(b)* validating the GA-based synthesis approach for generating optimal PSL arrangements when dealing with both small and large apertures. The GA-based search has been applied with the following setup: cross-over probability equal to 0.7, mutation probability equal to 10^{-2}, maximum number of iterations $I_{max} = 5 \times 10^3$, population size $C = 10$. Moreover, has been assumed $s_x = s_y = 0.5$. It is worth remarking that, although deduced for a broadside steering, the final layouts will be optimal for $s_x = s_y = 0.5$ whatever the steering direction [thanks to 4.4]. Moreover, since in most cases the highest secondary lobes appear near the main lobe in DS planar arrays [19], such layouts are expected to represent the optimal ones also for most other steering directions and inter-element spacings.

The first numerical experiment is concerned with the McFarland sequence with $P = 3$ for which an exhaustive analysis, although computationally cumbersome, can be still performed in a reasonable amount of time. The plot of the PSL values of the whole set of $\Psi(P)\rfloor_{P=3} = 3.54 \times 10^7$ McFarland arrays indicate that several DS layouts exhibit PSLs equal or very close to the optimal one $PSL^{opt} = -9.3$dB [Fig. 31(a)]. This is also confirmed by the index $\Delta(\eta)$ given by

$$\Delta(\eta) \triangleq \frac{\Psi(P)\rfloor_{PSL \leq \eta PSL^{opt}}}{\Psi(P)} \tag{4.8}$$

and defined as the fraction of McFarland layouts that exhibit a PSL equal or below η times the optimal value PSL^{opt} (Fig. 32). As a matter of fact, although the optimal configurations are quite rare [$\Delta(\eta = 1.0) \approx 5.5 \times 10^{-4}$- Fig.32], a non-negligible portion of the randomly-generated layouts exhibits a PSL close to PSL^{opt} [$\Delta(\eta = 0.9) \approx 0.01$]. This suggests that the GA-based search method should quickly find a sub-optimal configuration, while a larger number of iterations may be required to actually reach convergence to the global optimum. Such a behaviour is pointed out by the plot of the evolution of the optimal GA solution within the solution space of McFarland arrays in Fig. 31(b) where the blue crosses identify the elements of the GA solution set at the i-th GA iteration, while the red line is concerned with the overall (ordered) McFarland solution set as a function of the sequence index. Indeed, less than 300 iterations are sufficient to find a McFarland arrangement with $PSL \approx -8.6$dB, while the convergence is reached after $I_{conv} = 1693$ steps. Such an outcome confirms that the GA-based synthesis is able to effectively sample a large solution space finding the optimal McFarland layout characterized by a low PSL value despite only 12 active elements over a lattice of 45 positions [Fig. 30(a)].

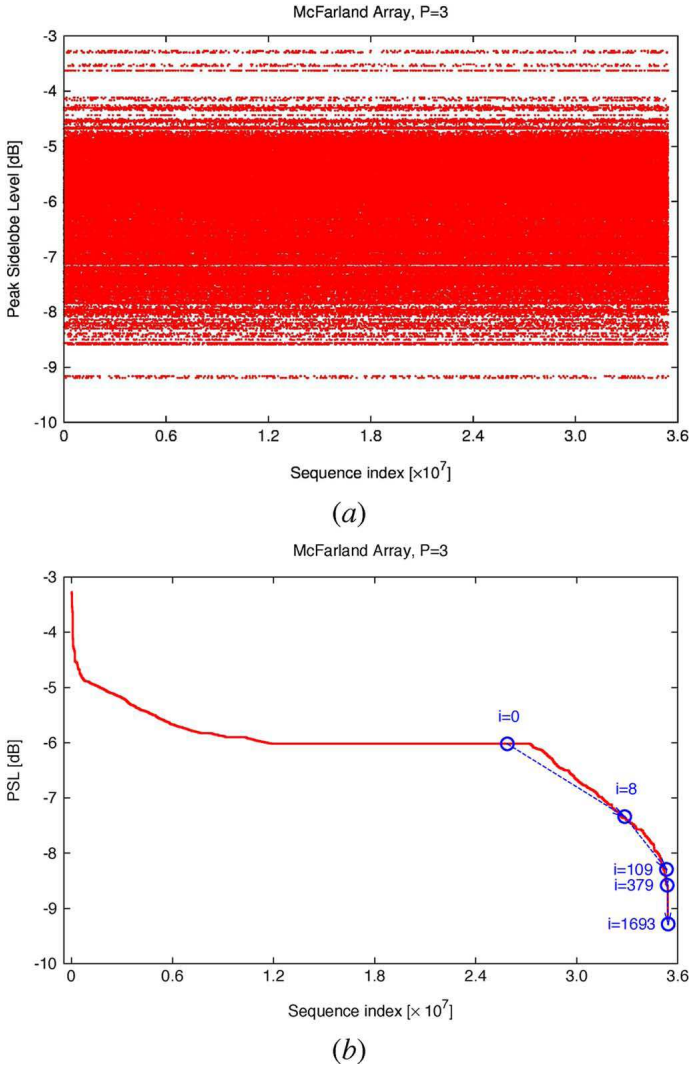
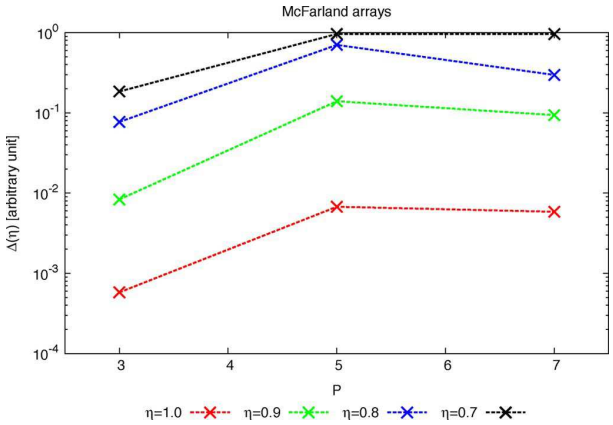


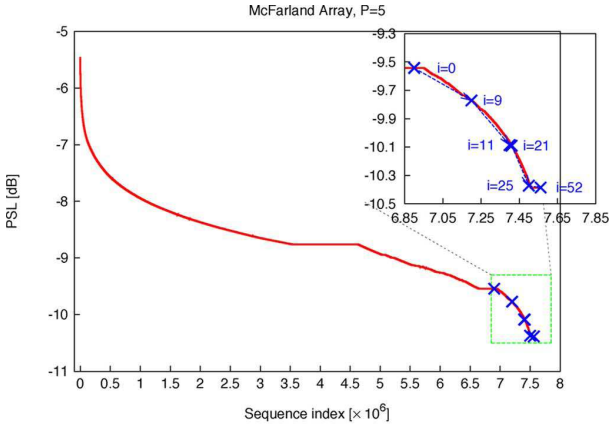
Figure 31. *GA-Based McFarland Synthesis* - Plots of (a) the PSL values of the whole set of McFarland arrays and (b) evolution of the PSL of the GA solution during the iterative (i being the iteration index) sampling of the McFarland solution space.

Similar conclusions can be drawn from the analysis (non exhaustive, but limited to a percentage of the whole set of McFarland configurations) carried out for $P = 5$ and $P = 7$ [Figs. 33(a) and (b)], even though a faster convergence of the GA-search is expected when dealing with larger dimensions as suggested by the values of $\Delta(\eta)$ [e.g., $\Delta(\eta = 0.9) \approx 0.1$ for $P = 5$ vs. $\Delta(\eta = 0.9) \approx 0.01$ for $P = 3$ - Fig. 32]. This is further confirmed by the evolution of the GA solutions in Fig. 33. As a matter of fact, only $I_{conv} = 52$ and $I_{conv} = 47$ iterations are necessary to reach the convergence when $P = 5$ [Fig. 33(a)] and $P = 7$ [Fig. 33(b)], respectively.

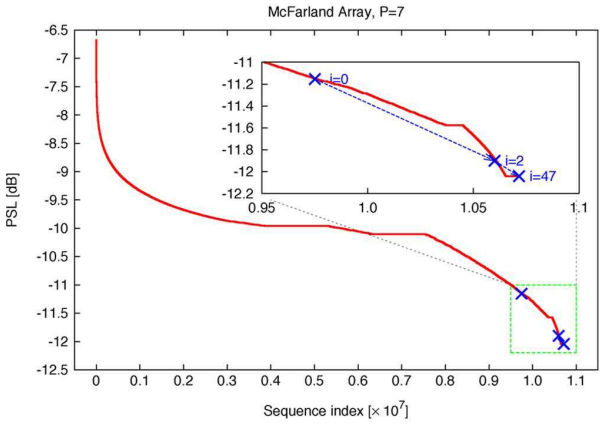


(*a*)

Figure 32. *McFarland Rectangular Arrays* - Behaviour of $\Delta(\eta)$ versus P when $\eta \in \{0.7, 0.8, 0.9, 1.0\}$.



(*a*)



(*b*)

Figure 33. *GA-Based McFarland Synthesis* - Evolution of the PSL of the GA solution during the iterative (i being the iteration index) sampling of the McFarland solution space when (a) $P = 5$ and (b) $P = 7$.

For completeness, Fig. 34 gives the corresponding arrangements and power patterns. As expected from DS theory, the optimal layouts at convergence [$P = 5$ - Fig. 34(a); $P = 7$ - Fig. 34(c)] exhibit controlled and regular sidelobes [$P = 5$ - Fig. 34(b); $P = 7$ - Fig. 34(d)] despite the massive thinning ($\nu \triangleq K/N = (P{+}1)/(P(P{+}2)) \approx 0.17$ for $P = 5$, $\nu \approx 0.13$ for $P = 7$ - Table III). Moreover, thanks to the McFarland distribution, the corresponding architectures give different resolutions in each angular domain as indicated by the locations of the first nulls of the beam pattern (see $z_u$vs. z_v in Table III).

P	(N, K, Λ)	Array Size	$\Psi(P)$	ν	z_u	z_u	PSL^{opt}[dB]
3	$(45, 12, 3)$	3×15	3.54×10^{7}	0.2667	6.66×10^{-1}	1.33×10^{-1}	-9.28
5	$(175, 30, 5)$	5×35	2.15×10^{14}	0.1714	4.10×10^{-1}	5.74×10^{-2}	-10.41
7	$(441, 56, 7)$	7×63	5.31×10^{21}	0.1270	2.85×10^{-1}	3.17×10^{-2}	-12.04
11	$(1573, 132, 11)$	11×143	9.64×10^{37}	0.0839	1.81×10^{-1}	$1..34 \times 10^{-2}$	-15.56
13	$(2535, 182, 13)$	13×195	5.14×10^{46}	0.0718	1.53×10^{-1}	1.02×10^{-2}	-15.54
17	$(5491, 306, 17)$	17×323	1.32×10^{65}	0.0557	1.17×10^{-1}	6.19×10^{-3}	-15.61
19	$(7581, 380, 19)$	19×399	5.47×10^{74}	0.0501	1.05×10^{-1}	5.01×10^{-3}	-15.63
23	$(13225, 552, 23)$	23×575	4.73×10^{94}	0.0417	8.69×10^{-2}	3.47×10^{-3}	-15.50
29	$(26071, 870, 29)$	29×899	1.18×10^{126}	0.0334	6.89×10^{-2}	2.22×10^{-3}	-15.02

Table III. McFarland Rectangular Arrays ($P \leq 29$) - Features and Performance Indexes.

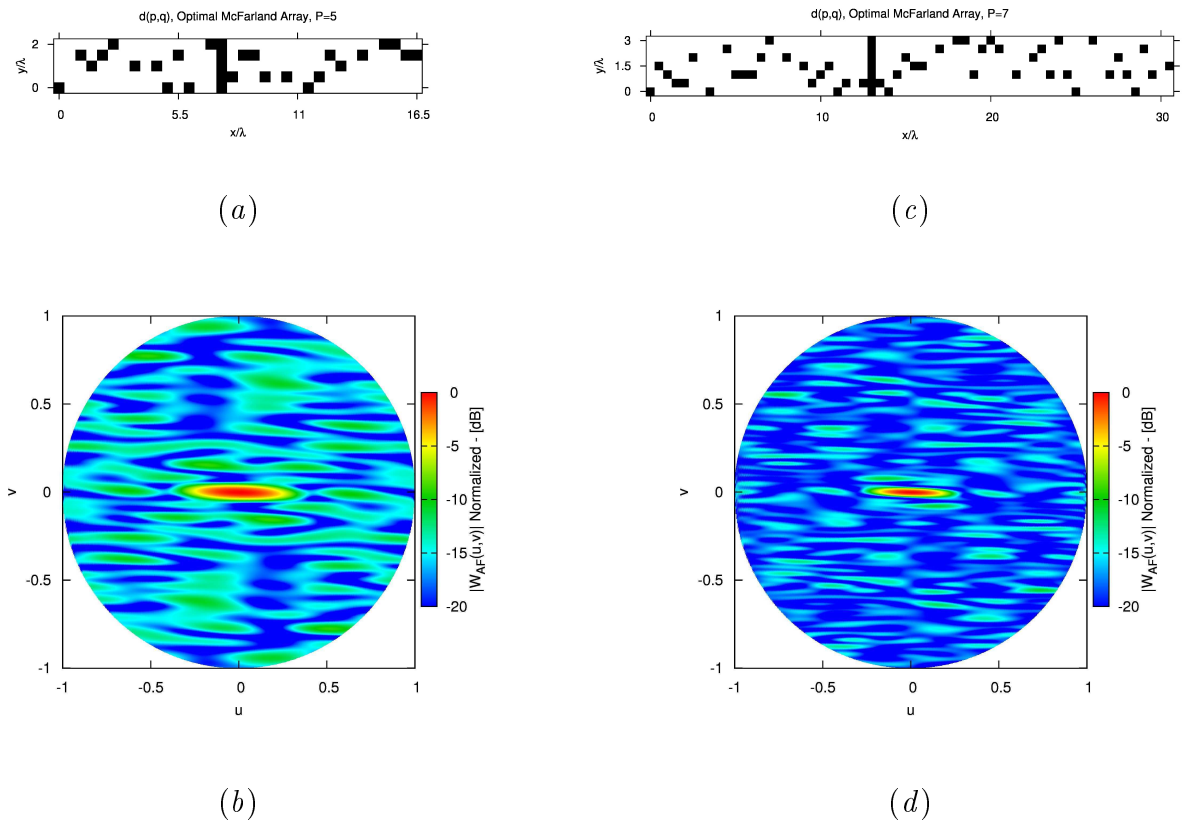
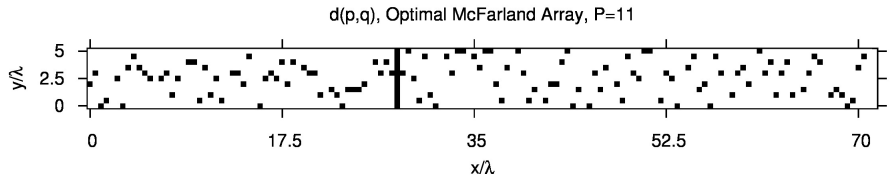
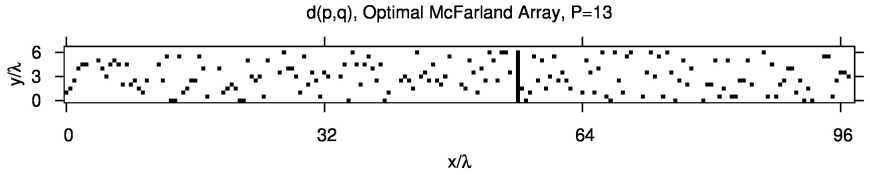


Figure 34. *GA-Based McFarland Synthesis* - Optimal McFarland layouts (a), (c) and the corresponding power patterns (b), (d) when $P = 5$ (a), (b) and $P = 7$ (c), (d).

In order to assess the performances of McFarland thinned arrays also when impractical (for an exhaustive analysis) apertures are at hand, the next experiments are concerned with $11 \leq P \leq 29$. The results of the GA-based synthesis when $P = 11$ and $P = 13$ are provided in Figs. 35 and 36. Despite the decreasing thinning factor ($\nu_{P=11} \approx 8.4 \times 10^{-2}$, $\nu_{P=13} \approx 7.2 \times 10^{-2}$ - Table I), high sidelobe do not appear since $PSL_{P=11} = -15.56$dB and $PSL_{P=13} = -15.54$dB (Table I). Moreover, the power patterns in Fig. 36 [$P = 11$ - Fig. 36(a); $P = 13$ - Fig. 36(b)] show the sidelobe regularity expected from the two-level autocorrelation McFarland layouts notwithstanding the highly-sparse element distribution [$P = 11$ - Fig. 35(a); $P = 13$ - Fig. 35(b)].

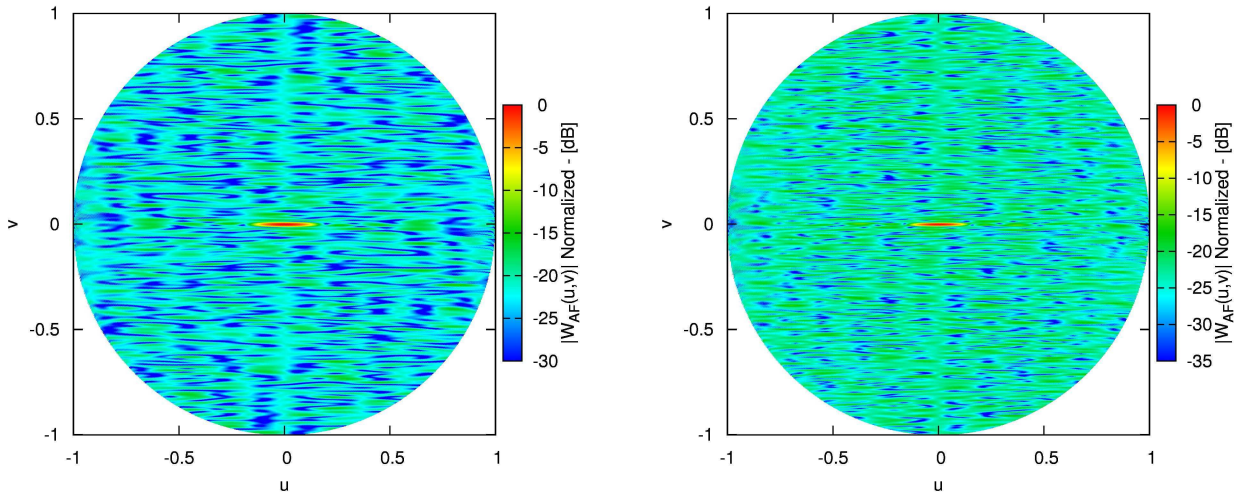


(a)



(b)

Figure 35. *GA-Based McFarland Synthesis* - Optimal McFarland layouts (a) $P = 11$ and (b) $P = 13$.



(a) (b)

Figure 36. *GA-Based McFarland Synthesis* - Power patterns of the optimal McFarland layouts deduced for (a) $P = 11$ and (b) $P = 13$.

Previous conclusions can be also extended to wider McFarland layouts ($P \leq 29$ - Table III). As it can be noticed, low PSL values are obtained whatever the P dimension ($PSL \in [-15.61\text{dB}, -15.0]\text{dB}$ for $P \in [17, 29]$ - Table III), despite the sharp reduction of the thinning factor ($\nu < (1/P)$ - Table III).

As a final numerical validation, a comparison between the performances of the best McFarland array and those of the best sparse array with the same size and thinning factor found by means of a traditional GA-based approach [51], [50] is provided. The GA methodology is applied by assuming standard "binary" descriptors of the geometry [24], [50], rather than the McFarland descriptors introduced above. As a consequence, the obtained design will not be a DS layout. More in detail, a state-of-the-art randomly initialized GA method (see [24], [50] for the implementation details) is employed for designing a thinned rectangular array of size $P \times Q = 7 \times 63$ with $K = 56$ active elements. The stochastic optimization has been carried out by considering a GA population of size 10, a mutation probability equal to 10^{-2} and a crossover probability of 0.7. The maximum number of GA iterations has been set to 5×10^3 [24], [50]. By comparing the performances obtained by the GA-optimized layout [Fig. $37(a)$] with those of the McFarland one [Fig. $34(c)$], it turns out that the stochastically optimized architecture does not to reach a PSL value [Fig. $37(b)$] as low as that of the proposed layout [Fig. $34(d)$] [$PSL_{GA} = -10.76$dB vs. $PSL_{McFarland} = -12.04$dB] even though also non-DS layouts can be synthesized in the former case. Such a result is due to the size of the search space that has to be explored by the standard GA methodology (i.e., 2^{441}), which is extremely larger than that defined by the McFarland descriptors ($\Psi(P = 7) \approx 5.31 \times 10^{21}$ - Table III).

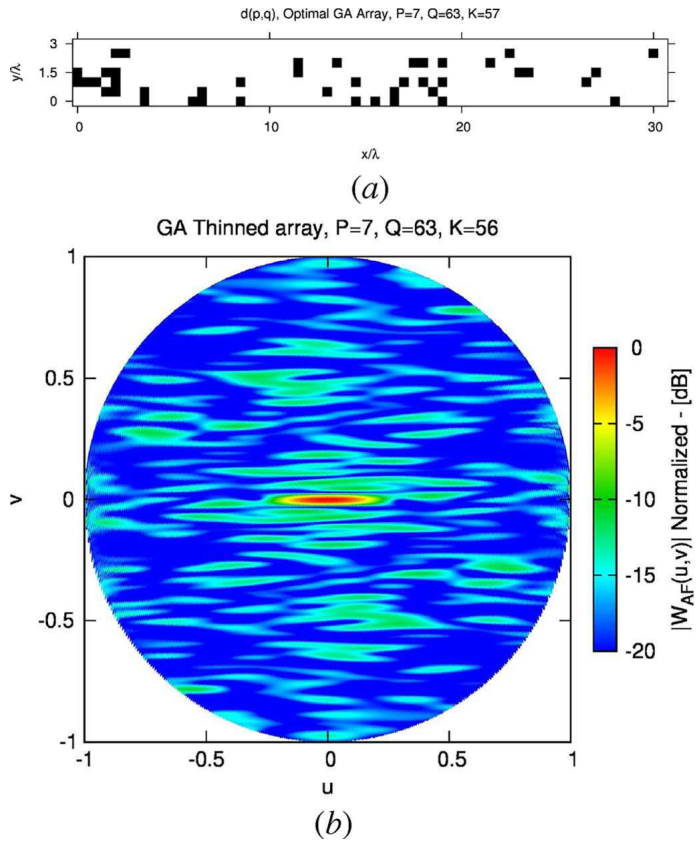


Figure 37. *Comparison with Standard GA-Thinned Rectangular Arrays* - Optimal layout (a) and the corresponding power pattern (b) obtained by GA when $P = 7$, $Q = 63$ and $K = 56$.

4.5 Appendix

In this section, a procedure for the generation of a McFarland Sets is presented.

Let be P a prime number and let us define $\mathbf{V}_0 = \{(w_1, w_2) : 0 \leq w_1 \leq P-1, 0 \leq w_2 \leq P-1, w_2, w_2 \in \mathbb{N}\}$, $\mathbf{H}_0 = \{(0,0)\}$and $\mathbf{M}_0 = \emptyset$. Select an integer $k \in [0, ..., P+1]$ and choose $P+2$ (not necessarily different) vectors $(a_t, b_t) \in \mathbf{V}_0$ with $o \leq t \leq P+1$, $t \neq k$.

For every $t \in [0, ..., P+1]$, let $\mathbf{V}_{t+1} = \mathbf{V}_t \backslash \mathbf{H}_t$ and determine the set $\mathbf{M}_{t+1}$as follows:

$$\mathbf{M}_{t+1} = \mathbf{M}_t,\ \mathbf{H}_{t+1} = \emptyset$$

$$\begin{cases} \mathbf{H}_{t+1} = \left\{ \left[\left(p\hat{w}_1^{(t+1)} \right) \Big\rfloor_P, \left(p\hat{w}_2^{(t+1)} \right) \Big\rfloor_P \right], \ p = 1, ..., P-1 \right\} \\ \\ \mathbf{M}_{t+1} = \mathbf{M}_t \cup \left\{ \left[\left(p\hat{w}_1^{(t+1)} + a_{t+1} \right) \Big\rfloor_P, \left(p\hat{w}_2^{(t+1)} + b_{t+1} \right) \Big\rfloor_P, p = 0, ..., P-1 \right] \right\} \end{cases} if\ t \neq k$$

where $\left(\hat{w}_1^{(t+1)}, \hat{w}_2^{(t+1)} \right)$ is randomly picked element in $\mathbf{V}_{t+1}$.

From [51], it follows that $\mathbf{M}_{P+2}$ is a McFarland DS (i.e. $\mathbf{M} \triangleq \mathbf{M}_{P+2}$) with indexes $N = P^2(P+2)$, $K = P(P+1)$, and $\Lambda = P$.

Chapter 5

Hybrid ADS-Based Techniques for Radio Astronomy Array Design

5.1 Introduction

The design of correlator (also known as interferometric) arrays has been a topic of research since the1960s for applications in radio astronomy [1]-[52]. The efficiency of the data gathering of correlator arrays is related to their spatial filtering properties [31], [8]. Therefore, the design of a correlator array essentially consists in solving an optimal sampling problem [31] where the positions of the array elements are chosen to ensure optimal performances in all possible observation situations (i.e., source positions and durations of the observation), for whatever scientific purpose (e.g., single field imaging, mosaicing, astrometry, detection), and different constraints (i.e., cost, ground composition and practicability, operation of the instrument) [31], [53], [54]. In order to reach these objectives and unlike traditional sum arrays, correlator arrays have to generate either a maximal coverage in the spatial frequency (or) domain or a minimum peak sidelobe level (PSL) in the angular (or) domain [31], [8], [53] as detailed in Section 5.2. Towards this end, many and customized strategies have been proposed including minimum redundancy [55], [40], [33], pseudo randomness [34], power laws [35], difference set arrangements [36], and minimization of the holes in the sampling [37]. As regards optimization-based sum-array design techniques [1], [56]-[59], they also cannot be directly applied since the array spatial coverage evaluation, the Earth rotation effects [60], [29], and the beam calculation must be taken into account in the synthesis procedure as pointed out in [31] and [54]. However, optimization-based design techniques can still represent an important tool for future planned instruments, especially when the underlying architecture is mechanically

reconfigurable (as for the future planned ALMA [57] and SKA [58]).

In such a framework, valuable results have been obtained in [62], [31] thanks to the integration of a tool for the systematic analysis of correlator arrays and an effective particle swarm optimizer (PSO). However, despite the good performances, such a technique does not exploit the available analytical knowledge on interferometric arrays [31]. Usually, introducing a priori information in stochastic optimizers is known to improve their performances in terms of both rate of convergence and final design properties [24], [25]. This is expected to hold true also for the synthesis of correlator arrays. Therefore, this paper is aimed at introducing and numerically validating a set of hybrid techniques that take advantage of the a priori information on suboptimal analytically derived correlator arrangements. The proposed methodologies are based on recently introduced binary sequences with almost ideal autocorrelation properties, named Almost Difference Sets (ADSs) [61]-[63]. Such sequences are exploited in three different ways: (*i*) as a codebook in an exhaustive search approach; (*ii*) as initial trial solutions for a binary optimization process (ADS -hybridized GA); (*iii*) as a-priori information for a real-coded optimization technique (ADS-enhanced PSO). The main motivations of these recipes are:

- ADSs seem to be good candidates for the synthesis of correlator arrays since they exhibit correlation properties very similar to those of DSs [5], [64], whose effectiveness in such a framework has been already shown [36], but they are available in a wider set of admissible configurations [61][65][63];

- GAs are highly efficient tools for discrete optimization problems [44] potentially suitable for the effective design of correlator arrays whose elements lie on a regular lattice;

- PSO [59] has already shown its effectiveness and reliability when dealing with correlator arrays [31];

- the a-priori information can be straightforwardly integrated in stochastic optimization tools and it has proven to be effective in enhancing performances and convergence in array synthesis [24], [25]. Indeed, a good initial population (based on some a priori known sub-optimal solutions) contains good "schemata" [66] which can evolve through genetic operators to improve the GA speed of convergence towards the global minimum (similar considerations apply to PSO, as well).

The outline of the chapter is as follows. After a short review on correlator arrays and a description of the key problems in synthesizing interferometric arrangements (Section 5.2),

the performances of the design methodology (i) are analyzed to point out potentialities and limitations of the analytic ADS-based approach (Section 5.3). Afterwards, the GA-(ii) and PSO-based (iii) hybrid methodologies are presented and numerically validated dealing with benchmark problems (Section5.4).

5.2 Mathematical Formulation and Problem Statement

The interferometer beam, which describe the spatial filtering features of a correlator array, is defined as [8]

$$S(l,m) = IFT\{W(u,v) \times a(u,v)\} \tag{5.1}$$

where $IFT\{\cdot\}$ denotes the Inverse Fourier Transform operator, $a(u,v)$ is a tapering function devoted to suppress the sidelobes in the domain [8], and $W(u,v)$ is the $u-v$ coverage function

$$W(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\, f(x-u\lambda, y-v\lambda)\, dxdy \tag{5.2}$$

where λ is the wavelength and $f(x,y)$ is the element location [8].

As far as tracking observations are concerned, the effects of the Earth rotation must be introduced in the coverage function (5.2), and the interferometer beam in (5.1) turns out modified as [8]

$$S_T(l,m) = IFT\{W_T(u,v) \times a(u,v)\} \tag{5.3}$$

$W_T(u,v)$ being the tracking $u-v$ coverage function [8] which is a function of the source declination $\mathcal{D}$, the elevation $\mathcal{E}$, the latitude $\mathcal{L}$, the azimuth of the baseline $\mathcal{A}$, and the time angle during the observation $\mathcal{T}_k = \frac{kH\pi}{24(K-1)}$. Moreover, H is the total tracking time (in hours) and the number of snapshots collected during the observation.

As for the arising interferometer beam (5.3), the computation of the inverse Fourier transform is usually carried out by means of an IFFT procedure [8]. Towards this end, the $u-v$ domain is partitioned in $N_u \times N_v$ cells of size $\Delta v \times \Delta u$ and the IFFT procedure limits the $l-m$ domain within the range $-\frac{1}{\Delta u} \leq l \leq \frac{1}{\Delta u}$ and $-\frac{1}{\Delta v} \leq m \leq \frac{1}{\Delta v}$, while the beam pattern S_T is sampled at $\left(\frac{1}{\mathcal{Z}N_u\Delta u}, \frac{1}{\mathcal{Z}N_v\Delta v}\right)$, $\mathcal{Z}$ being the IFFT zero-padding factor [31].

For illustrative purposes, the element location function of an Y-shaped array with $N = 27$ elements ($L = 21$ [km] and $\varphi = 5$ [deg]) is shown in Fig. 38(a), while the associated $S_T(u,v)$ pattern is reported in Fig. 38(b) in correspondence with a working

frequency of 3.6 GHz and for the following setup: $\mathcal{D} = 34°$, $\mathcal{E} = 0$, $\mathcal{L} = 34°$, $H = 8$ hours, $K = 97$, $N_u = N_v = 128$, and $\Delta u = \Delta v = 6.82 \times 10^3$. Analogously to [31], the plot in Fig. 38(b) has been generated by applying an all-over Gaussian weighting $a\,(u, v)$ with an edge tapering of -15dB. Moreover, $\mathcal{Z}$ has been set to 8 for visual purposes and only the angular range within ± 01 arc seconds is displayed to highlight the near-in sidelobes. The design of a correlator array requires the optimization of the features of $W\,(u, v)$, $W_T\,(u, v)$, $S\,(l, m)$, and/or $S_T\,(l, m)$ depending on the problem at hand. Standard benchmark synthesis problems are stated in the following subsections and, for comparison purposes, the reference Y-shaped arrangements in [31] similar to the Very Large Array (VLA) at Socorro, NM [8], [67], will be considered unless otherwise stated.

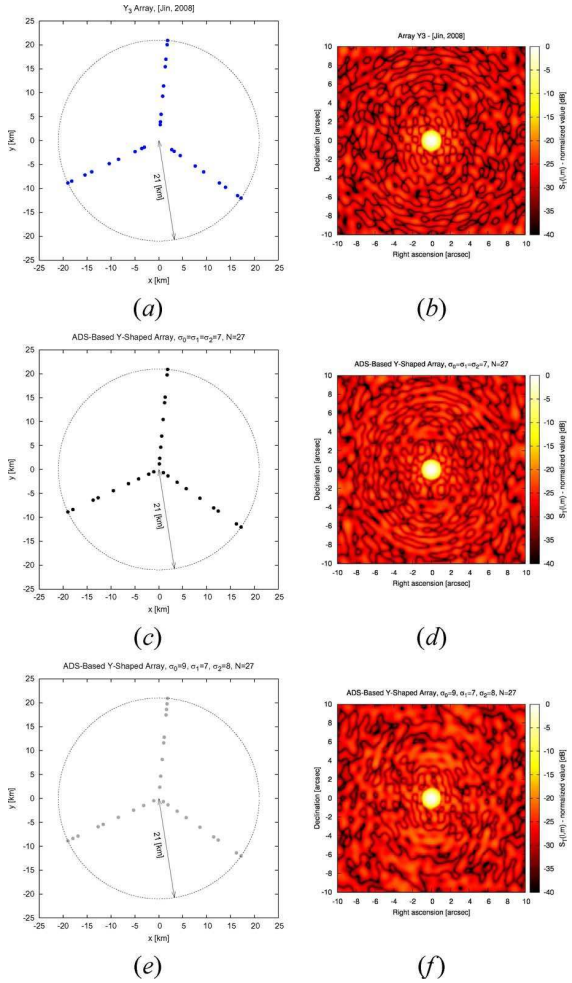


Figure 38. *Y-shaped Arrays* $[P = 18, Q = 9, \Lambda = 4, r = 13,$ *Equal-unequal arms*$]$ - Plots of the arrangement (a) and associated $S_T\,(u, v)$ (b) for the array Y_3 [31]; optimal ADS geometry with equal (c) or unequal (e) arms, and associated synthesized beams (d),(f).

5.2.1 *Problem A* - Optimization of $S_T(u,v)$

The first benchmark problem is concerned with the suppression of the sidelobes of the interferometer beam during tracking observations. Towards this end, the following metric [31].

$$F_A = PSL = \frac{\max\limits_{(l,m)\in R} S_T(l,m)}{S_T(l_0,m_0)} \tag{5.4}$$

has to be minimized, R and (l_0, m_0) being the main lobe region and the main beam steering direction, respectively.

5.2.2 *Problem B* - Optimization of the $u-v$ Coverage in Snapshot Observation

The optimization of the snapshot $u-v$ coverage is the second reference problem. In order to reduce the redundancy of the correlator array , while keeping a uniform sampling, the $u-v$ domain is partitioned in $N_{grid} \times N_{grid}$ square cells of equal size $\Delta_{grid} \times \Delta_{grid}$ and the following cost function, to be minimized, is then defined

$$F_B = \frac{1}{B} \tag{5.5}$$

where B $[B \leq (N_{grid})^2]$ is the number of different cells sampled by the snapshot coverage function $W(u,v)$ given by

$$B = \sum_{i=0}^{N_{grid}-1} \sum_{i=0}^{N_{grid}-1} G(i,j) \tag{5.6}$$

where $G(i,j) = 1$ if $W(u,v) \neq 0$ when $\left(-\frac{N_{grid}}{2}+i\right)\Delta_{grid} < u < \left(-\frac{N_{grid}}{2}+i+1\right)\Delta_{grid}$, $\left(-\frac{N_{grid}}{2}+j\right)\Delta_{grid} < v < \left(-\frac{N_{grid}}{2}+j+1\right)\Delta_{grid}$, and $G(i,j) = 0$ otherwise.

5.2.3 *Problem C* - Optimization of the $u-v$ Coverage in Tracking Observation

In the third problem, the maximization of the tracking $u-v$ coverage is at hand. As for *Problem B* and analogously to [31], the domain is still discretized, but in a finer grid of $N_u \times N_v$ cells of size $\Delta u \times \Delta v$, to define the following cost function to be maximized:

$$F_C = \frac{1}{\nu} \tag{5.7}$$

ν being the filling index defined as the ratio between A_c (e.g., the number of cells belonging to the circle around the "six point star" autocorrelation when dealing with a uniformly spaced array [31]) and A_S [i.e., the number of cells sampled by the tracking coverage function $W_T(u,v)$ given by (5.6) with W_T instead of W].

5.3 ADS-Based Y-Shaped Correlator Arrays

ADSs have been introduced in combinatorial mathematics and code theory to overcome some limitations of DSs while providing similar properties [61][65][63]. Although successfully applied in several fields ranging from cryptography up to antenna array synthesis [22][47][48][69][70], they have never been considered (to the best of the authors' knowledge) in the framework of correlator arrays as proposed in the following exhaustive procedure:

1. **Initialization** - Given a target number of active elements N and an arm length L, select from [68] a reference-ADS $\mathbf{D}$ (for definition and properties, see [61], [65]) with $Q = \frac{N}{E}$, E being the arm number ($E = 3$ for an Y layout). Set $\sigma_e = 0$ (σ_e being the cyclic shift applied to the $e-$th arm of the array) and locate the $i-$th array element of the correlator array at

$$\begin{cases} x_i &= \sin\left(\frac{\pi e}{3} + \varphi\right) \frac{L\left[1+(d_q+\sigma_e)|_{modP}\right]}{P} \\ \\ y_i &= \cos\left(\frac{\pi e}{3} + \varphi\right) \frac{L\left[1+(d_q+\sigma_e)|_{modP}\right]}{P} \end{cases} \tag{5.8}$$

$$i = Qe + q,\ q = 0, ..., Q-1,\ e = 0, ..., E-1$$

2. **Evaluation** - Evaluate the degree of "fitness" to a design problem of the current trial arrangement by computing the cost function in (5.4), or (5.5), or (5.7);

3. **Iteration** - Update σ_e ($\sigma_e \leftarrow \sigma_e + 1$) and use the same shifted ADS-based element distribution on each arm of the correlator ("*equal*" configuration) or a different shift one each arm by iteratively repeating Step 2 and Step 3 ("*unequal*" configuration);

4. **Termination** - Stop when ("*equal*" configuration) or P^E ("*unequal*" configuration) trial designs have been evaluated. Set the "*optimal*" ADS design to the arrangement with the highest "*fitness*".

It is worth to notice that such a procedure is very simple and computationally efficient since just up to P^E evaluations are required for a correlator array with N active elements.

Moreover, the array elements are displaced on a regular lattice of P positions on each arm allowing an easier realization with respect to arbitrary displacements.

In order to evaluate the performance of the ADS-based analytic approach as well as to compare the arising configurations with state-of-the-art arrangements [31], the design of the Y-shaped correlator described in Section 5.2 has been considered as first test case. Because of the design requirements ($Q = 9$), the (18, 9, 4, 13)-ADS $\mathbf{D}_1$ [68] (see Table IV) has been adopted as reference sequence.

As far as *Problem A* is concerned, Fig. 39(a) shows the behavior of the PSL as a function of the shift number σ_e for both the "*equal*" and "*unequal*" arrangements. The figures of merit at the convergence (Table V) indicate that ADS-based designs significantly improve the performance of reference uniform ($PSL_{unf} = -13.1$[dB]) and power-law ($PSL_{pl} = 16.2$[dB]) arrays. Moreover, the arising PSL values turn out to be close to that of stochastically optimized arrays ($PSL_3 = -20.3$[dB]) [31], even though the convergence has been reached after few evaluations of the cost function when the same σ_e is applied to every arm. As expected and because the increased number of degrees-of-freedom (DoFs), a smaller PSL can be yielded by setting different shifts on the arms, but more evaluations are necessary [$\sigma^{un} = 2708$ vs. $\sigma^{eq} = 7$].

On the other hand, it is worthwhile to observe [Fig. 39(b)] that different ADS layouts present performances close to that of the optimal one pointing out an interesting feature of the ADS synthesis to be exploited when "*compromise*" problems with conflicting requirements are at hand.

Concerning *Problems B* and *C*, similar conclusions on the computational efficiency of the analytic ADS-based synthesis arise (Table V). However, the behaviors of B and ν versus σ_e [Fig. 39(c) and (e)] as well as the characteristics of the convergence designs (Table V) indicate that (a) the ADS strategy reaches results with performances comparable or better than those of power-law arrays in *Problem B* ($B^{eq}_{ADS} = 408$ and $B^{un}_{ADS} = 430$ versus $B_{pl} = 428$) and significantly better for the *Problem C* ($\nu^{eq}_{ADS} = 0.828$ and $\nu^{un}_{ADS} = 0.831$ versus $\nu_{pl} = 0.598$); (b) the coverage of ADS-based arrays worsens when compared to *uniform* arrays [Fig. 39(c), $B_{unf} = 534$], while they outperform uniform arrangements in *Problem C* [$\nu_{unf} = 0.689$]; (c) as expected, the PSO-based synthesis gives better performances than the ADS-based strategy in both *Problem B* ($B_1 = 558$) and *Problem C* ($\nu_2 = 0.865$) thanks to the unconstrained displacement of the array elements.

ADS	P	Q	Λ	r	$d_0, ..., d_{Q-1}$
$\mathbf{D}_1$	18	9	4	13	0, 1, 5, 6, 7, 8, 10, 12, 15
$\mathbf{D}_2$	88	44	21	22	3, 4, 5, 7, 8, 9, 10, 11, 1215, 16, 17, 18, 20, 22, 23, 25, 27, 30, 33, 34, 36, 37, 39, 43, 47, 52, 53, 54, 57, 58, 59, 66, 67, 68, 69, 70, 72, 75, 76, 78, 80, 84, 87
$\mathbf{D}_3$	180	90	44	45	0, 4, 8, 9, 10, 11, 12, 14, 15, 18, 19, 22, 23, 24, 25, 28, 30, 33, 34, 40, 42, 45, 48, 52, 53, 54, 55, 57, 61, 63, 65, 68, 71, 73, 76, 77, 78, 79, 80, 82, 84, 89, 93, 95, 96, 97, 98, 99, 100, 102, 104, 105, 111, 112, 113, 121, 126, 128, 129, 131, 132, 133, 137, 138, 139, 140, 141, 143, 145, 146, 149, 150, 151, 152, 153, 156, 158, 159, 162, 163, 166, 167, 168, 170, 172, 173, 175, 176, 177, 179
$\mathbf{D}_4$	42	21	10	31	0, 3, 4, 5, 6, 8, 9, 12, 14, 17, 19, 27, 28, 29, 30, 31, 34, 35, 36, 38, 41
$\mathbf{D}_5$	30	15	7	22	5, 6, 8, 9, 10, 14, 16, 17, 19, 20, 22, 23, 24, 27, 29

Table IV. *ADS* $\mathbf{D}_1$, $\mathbf{D}_2$,$\mathbf{D}_3$, and $\mathbf{D}_4$ and descriptive parameters.

		Equal arms					*Unequal arms*				
Design	*Problem*	PSL [dB]	B	ν	σ	Ξ	PSL [dB]	B	ν	σ	Ξ
Uniform	-	-13.1	534	0.689	-	0.41	-	-	-	-	-
Power-law	-	-16.2	428	0.598	-	0.44	-	-	-	-	-
Y_3 [31]	A	$\mathbf{-20.3}$	412	0.751	-	0.29	-	-	-	-	-
Y_1 [31]	B	-17.3	**558**	0.719	-	0.22	-	-	-	-	-
Y_2 [31]	C	-16.7	366	**0.865**	-	0.39	-	-	-	-	-
	A	$\mathbf{-19.34}$	388	0.758	7	0.33	$\mathbf{-19.98}$	400	0.807	2708	0.29
ADS	B	-15.84	**408**	0.688	1	0.40	-19.00	**430**	0.767	2094	0.26
	C	-17.76	396	**0.828**	9	0.32	-17.65	398	**0.831**	2781	0.32

Table V. *Numerical results - Y_{ADS} Arrays [$P = 18$, $Q = 9$, $\Lambda = 4$, $r = 13$] -* Comparison of ADS-based Y-shaped arrays and some representative designs (bold numbers identify optimized quantities).

Once again, the plots of B and ν versus the cyclic shift [Fig. 39(d), (f)] further confirm that multiple ADS designs with close sub-optimal performances can be synthesized starting from a single ADS with the potential of providing good trade-off solutions in terms of PSL, B, and ν despite negligible computational efforts. To investigate such an issue, Fig. 40 pictorially summarizes the performances of the whole set of trial ADS arrays generated by $\mathbf{D}_1$. For comparisons, the representative points of the solutions in [31] are reported, as well. As expected, good trade-off ADS arrays are available especially in the space [Fig. 40(b)]. They positively compare also with optimized designs and most of them overcome both uniform and power-law architectures [Fig. 40(b)]. In order to quantitatively estimate the effectiveness of the ADS "*compromise*" solutions, let us analyze the *normalized trade-off performance index* Ξ defined as follows:

$$\begin{aligned} \Xi \;=\; & \left\{ \left[\frac{\left(PSL-PSL^{opt}\right)\times\mathcal{H}\left(PSL-PSL^{opt}\right)}{PSL^{opt}} \right]^2 \right. \\ & + \left[\frac{\left(B-B^{opt}\right)\times\mathcal{H}\left(B-B^{opt}\right)}{B^{opt}} \right]^2 \\ & + \left. \left[\frac{\left(\nu-\nu^{opt}\right)\times\mathcal{H}\left(\nu-\nu^{opt}\right)}{\nu^{opt}} \right]^2 \right\}^{1/2} \end{aligned} \tag{5.9}$$

where $\mathcal{H}(\cdot)$ is the Heaviside function and the "*optimal*" values (i.e., $PSL^{opt} = -20.3$[dB], $B^{opt} = 558$ and $\nu^{opt} = 0.865$) have been set to those of the layouts Y_3, Y_1, and Y_2 in [31]. As it can be noticed (Fig. 41), the Ξ indexes of several ADS designs turn out to be better than power law ($\Xi_{pl} = 0.44$), uniform ($\Xi_{unf} = 0.41$), and Y_2 ($\Xi_2 = 0.39$) architectures. Moreover, ADS layouts with different shifts on the array arms also improve the results from Y_3 ($\Xi_3 = 0.29$). On the contrary, no bare ADS design outperforms Y_1 ($\Xi_1 = 0.22$). As a matter of fact, the arising number of ADS baselines turns out to be significantly smaller than that of the optimized design in [31] and of the uniform arrangement [Fig. 39(b)-(c)].

Summarizing, the obtained results suggest that (a) ADS layouts provide ν, PSL and Ξ values which are close to or better than those of state-of-the-art globally optimized architectures when dealing with *Problems A* and *C*; (b) the "*bare*" ADS approach cannot be profitably applied when *Problem B* is of interest and suitable hybridization and/or an increasing of the DoFs (e.g., no position constraints) is mandatory.

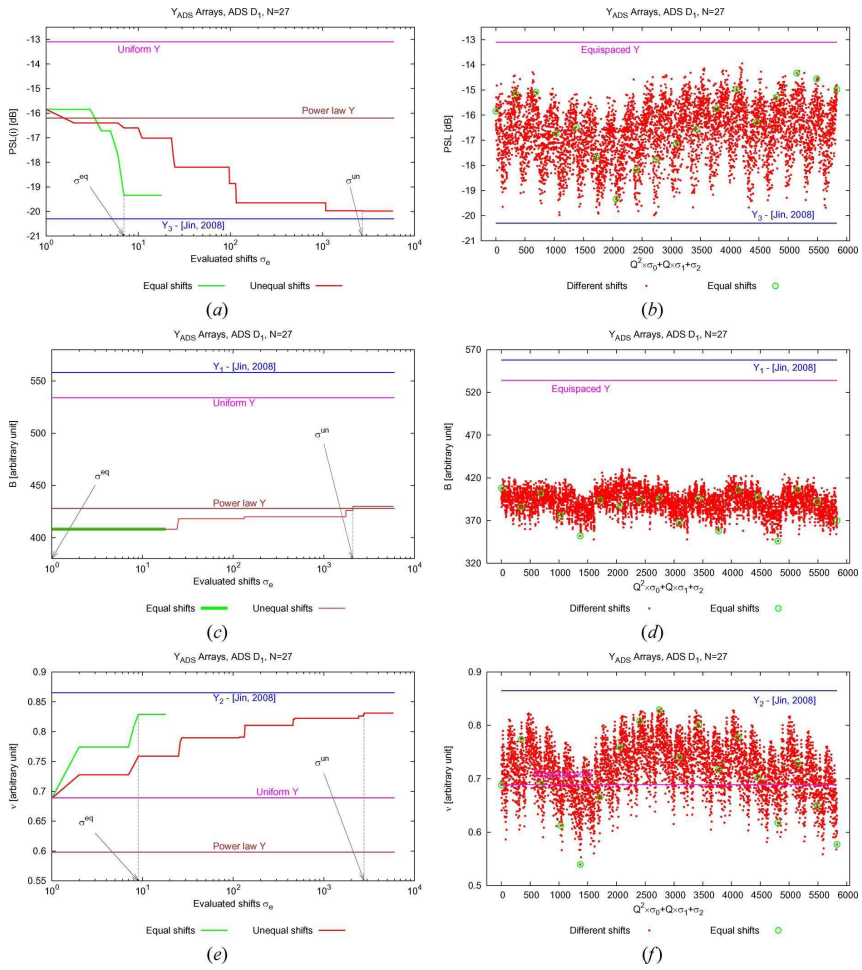


Figure 39. Y_{ADS} *Arrays* $[P = 18, Q = 9, \Lambda = 4, r = 13,$ *Equal-unequal arms*$]$ - Behavior of optimal (a) PSL, (c) , and (e) ν versus evaluated shift for ADS-based Y arrays, and comparison with reference designs from $[31]$. Plots of (b) PSL, (d) B, and (f) versus evaluated shift for ADS-based Y arrays.

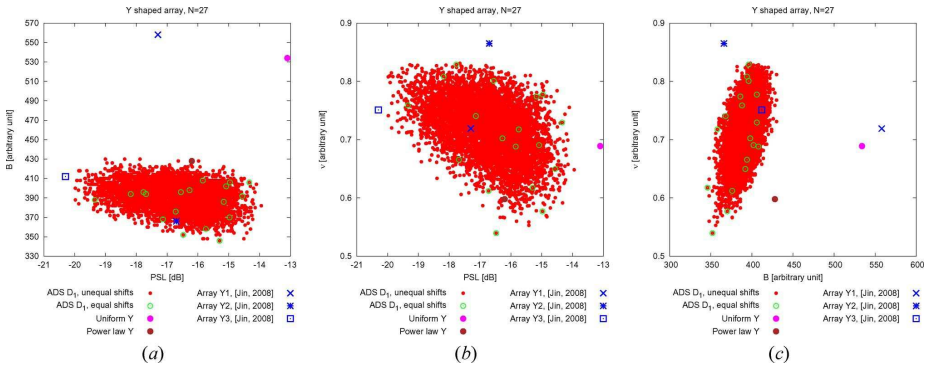


Figure 40. Y_{ADS} *Arrays* $[P = 18, Q = 9, \Lambda = 4, r = 13,$ *Equal-unequal arms*$]$ - Behavior of (a) B versus PSL, (b) ν versus PSL, and (c) ν versus for all Y_{ADS} arrays derived from $\mathbf{D}_1$, and comparison with reference designs from $[31]$.

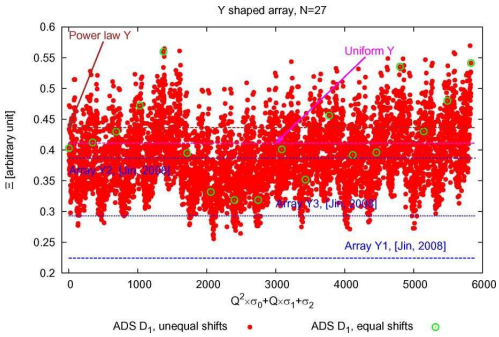


Figure 41. Y_{ADS} *Arrays* [$P = 18$, $Q = 9$, $\Lambda = 4$, $r = 13$, *Equal-unequal arms*] - Behavior of for Ξ all Y_{ADS} arrays derived from $\mathbf{D}_1$, and comparison with reference designs from [31].

5.4 ADS-Based Hybrid Methodologies

A first attempt to improve the performance of ADS-based designs while keeping their favorable properties (i.e., the computational efficiency of the synthesis process and the geometric simplicity of the lattice architecture) is aimed at defining an iterative hybrid GA-ADS (in the following, ADSGA) approach. The iterative approach is based on a standard GA implementation where the positions of Q active elements over a lattice of P positions are encoded in a binary string of length P, thus defining a chromosome of length $E \times P$. To exploit the ADS properties, the initial GA population of dimension is determined by first sorting the shifted versions of the reference ADS arrangement according to their fitness values and selecting the first αV_{GA} highly ranked sequences (α being the hybridization factor,$0 \leq \alpha \leq 1$) as trial array solutions. The remaining of the population is randomly chosen within the range of admissibility of the problem unknowns. As regards the GA operators, both crossover and mutation are applied with crossover probability $\mathcal{P}_C$ and mutation probability $\mathcal{P}_M$ according to standard binary implementations [44], but constraining to Q the number of active elements on each arm of the correlator.

The first numerical assessment is still concerned with the Y-shaped correlator and it deals with *Problem A* (i.e., the PSL minimization) by fixing the following setup: $V_{GA} = 10$, $\alpha = 0.5$, $\mathcal{P}_C = 0.9$, $\mathcal{P}_M = 0.01$, and a maximum number of iterations equal to $I_{MAX} = 400$. "*Equal*" and "*unequal*" arrangements on each arm have been simulated and a standard GA (RNDGA) has been also applied for evaluating in a comparative fashion the impact of the ADS initialization. The results reported in Table VI indicate that the ADSGA array [Fig. 40(b)] outperforms the bare ADS-based geometries both in the "*equal*" ($PSL^{eq}_{ADSGA} = -19.84$[dB] versus $PSL^{eq}_{ADS} = 19.34$[dB]) and "*unequal*" ($PSL^{un}_{ADSGA} = -20.93$[dB] versus $PSL^{un}_{ADS} = -19.98$[dB]) configurations.

		Equal arms					*Unequal arms*				
Design	*Problem*	PSL [dB]	B	ν	I	Ξ	PSL [dB]	B	ν	I	Ξ
GA	A	$\mathbf{-19.57}$	400	0.770	90	0.31	$\mathbf{-20.14}$	460	0.794	331	0.29
	B	-13.55	**534**	0.737	279	0.37	-15.00	**534**	0.748	1719	0.30
	C	-16.40	394	**0.838**	244	0.35	-16.14	412	**0.841**	399	0.33
ADSGA	A	$\mathbf{-19.84}$	424	0.769	175	0.27	$\mathbf{-20.93}$	404	0.773	231	0.30
	B	-13.55	**534**	0.737	203	0.37	-14.75	**534**	0.742	1799	0.31
	C	-16.01	400	**0.839**	283	0.35	-18.11	396	**0.845**	432	0.31
PSO	A	$\mathbf{-20.83}$	457	0.763	407	0.21	$\mathbf{-21.25}$	453	0.781	414	0.22
	B	-16.88	**550**	0.768	186	0.20	-17.97	**552**	0.759	96	0.17
	C	-17.57	407	**0.878**	260	0.30	-17.94	387	**0.881**	464	0.32
ADSPSO	A	$\mathbf{-20.91}$	457	0.800	312	0.20	$\mathbf{-21.35}$	489	0.781	493	0.16
	B	-17.80	**554**	0.747	222	0.18	-18.44	**554**	0.787	269	0.13
	C	-17.48	379	**0.879**	245	0.35	-17.94	415	**0.882**	288	0.28

Table VI. *Numerical results* - Comparison of optimized Y-shaped arrays (bold numbers identify optimized quantities).

On the other hand, the PSL value in correspondence with the "*unequal*" ADS array turns out to be lower than that for GA-based "*equal*" arrangements ($PSL^{eq}_{ADSGA} = -19.84$[dB], $PSL^{eq}_{RNDGA} = -19.57$[dB]). Such a result further confirms that unequally displacing the array elements over the correlator arms can provide non-negligible performance improvements independently of the synthesis technique. However, *"unequal"* layouts usually require a larger number of iterations to reach the convergence due to the additional *DoFs* ($I^{eq}_{ADSGA} = 175$ vs. $I^{un}_{ADSGA} = 231$, $I^{eq}_{RNDGA} = 190$ vs. $I^{un}_{RNDGA}331$).

As a further observation, it is worth noting that the ADSGA array outperforms the corresponding RNDGA solution both in terms of fitness ($PSL^{eq}_{ADSGA} = -19.84$[dB] versus $PSL^{eq}_{RNDGA} = -19.57$[dB], $PSL^{un}_{ADSGA} = -20.93$[dB] versus $PSL^{un}_{RNDGA} = -20.14$[dB]) and convergence rate [Fig. 42(a) and Table VI] assessing the effectiveness of an ADS initialization to improve the GA optimization. Thanks to the properties of "*unequal*" layouts and the effectiveness of an ADS initialization, the hybrid GA-based approach is even able to improve the state-of-the-art PSO solution [31] ($PSL^{un}_{ADSGA} = -20.93$[dB] versus $PSL_3 = -20 - 30$[dB]), despite the wider set of *DoFs* of this latter.

Concerning the reliability of the ADSGA and RNDGA layouts as "*compromise*" solutions, it is note worthy (Table VI) that they exhibit trade-off indexes very close or better than Y_3 (e.g., $\Xi^{eq}_{ADSGA} = 0.27$ versus $\Xi_3 = 0.29$). Such a behavior seems to assess the ability of the approach to intrinsically provide good compromise solutions also without directly optimizing the "*compromise index*" Ξ.

Dealing with the application of ADSGA to the other benchmark problems, even though the positive effect of the ADS integration still holds true, it results that (Table VI) both B and ν indexes, as well as the corresponding *"compromise"* performances, cannot be improved significantly without resorting to non-regular designs (i.e., avoiding regular lattices) as for PSO-based state-of-the-art solutions [31].

Towards this aim, an hybrid real-valued multiple-agent optimization approach based on a standard iterative PSO method [31], [59], [25] is then investigated. Likewise the ADSGA, the initial set of trial solutions is generated by exploiting the ADS sequences. Otherwise, the positions of the N active elements of the array are encoded in a PSO particle by setting each unknown as the distance between two adjacent array elements.

Because of the limitations of the ADS and ADSGA approaches in dealing with *Problem B* and *Problem C*, such benchmark tests will be first considered for validating the ADSPSO. Towards this end, the following setup for the PSO parameters has been adopted: $V_{PSO} = 10$, $c_1 = c_2 = 2$, $w = 0.4$, and $I_{MAX} = 500$. For a complete comparison, a PSO approach with a random initialization (RNDPSO) has been implemented, as well. The numerical simulations related to *Problem B* and whose results are illustrated in Fig. 43

indicate that the hybrid ADSPSO procedure is able to reach higher fitness values than ADS and ADSGA techniques ($B^{un}_{ADSPSO} = 554$ versus $B^{un}_{ADSGA} = 534$ and $B^{un}_{ADS} = 430$) and very close to [31] ($B_1 = 558$), while significantly outperforming uniform and power law layouts ($B_{unf} = 534$, $B_{pl} = 428$) thanks to the additional *DoFs* of the real-valued formulation (i.e., arbitrary element positions over each arm).

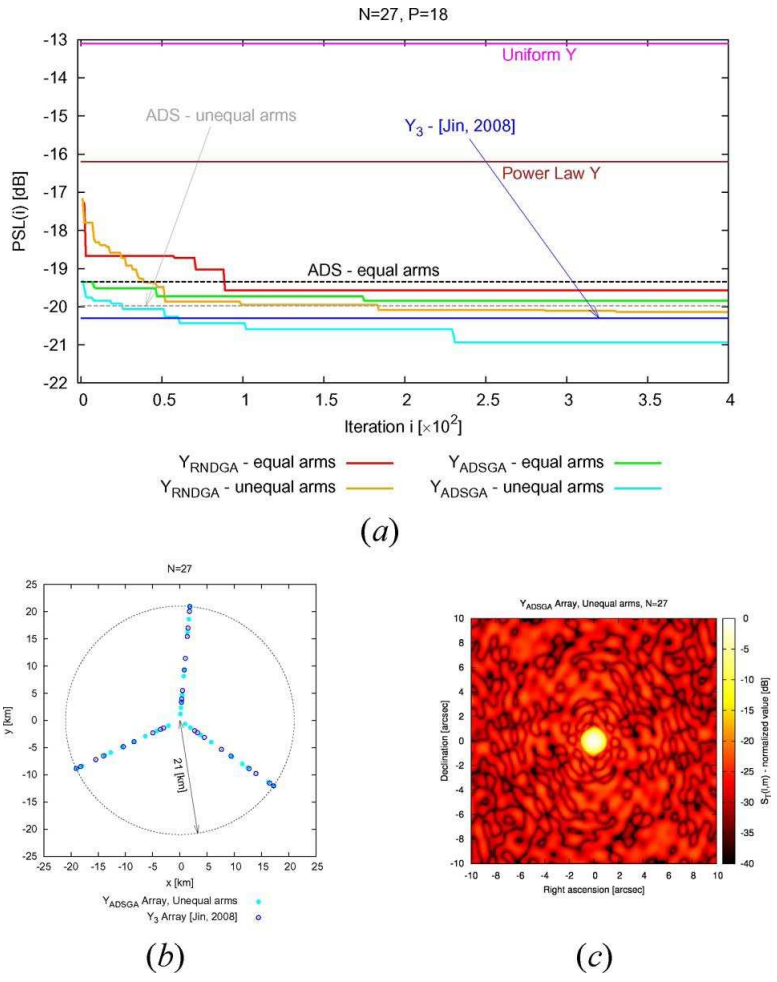


Figure 42. *Problem A* [*Equal-unequal arms*, $N = 27$] - Synthesis results for the GA and ADSGA approaches: (a) behavior of the optimal PSL versus the iteration number i, and comparison with reference designs from [31], (b) optimal Y_{ADSGA} array arrangement, and (c) associated synthesized pattern.

Moreover, as for the GA-based approaches, the hybrid ADS implementation exhibits improved performances ($B^{eq}_{RNDPSO} = 550$ vs. $B^{eq}_{ADSPSO} = 554$, $B^{un}_{RBDPSO} = 552$ vs. $B^{un}_{ADSPSO} = 554$) and convergence properties ($I^{eq}_{RNDPSO} = 286$ vs. $I^{eq}_{ADSPSO} = 222$, $I^{un}_{RNDPSO} = 296$ vs. $I^{un}_{ADSPSO} = 269$) with respect to the randomly initialized PSO also

when real-coded unknown are at hand. Moreover, the PSO-based hybrid technique always guarantees the best "*compromise*" performances (Table VI). As a matter of fact, it turns out that $\Xi^{eq}_{ADSPSO} = 0.18$ and $\Xi^{un}_{ADSPSO} = 0.13$, while $\Xi_1 = 0.22$.

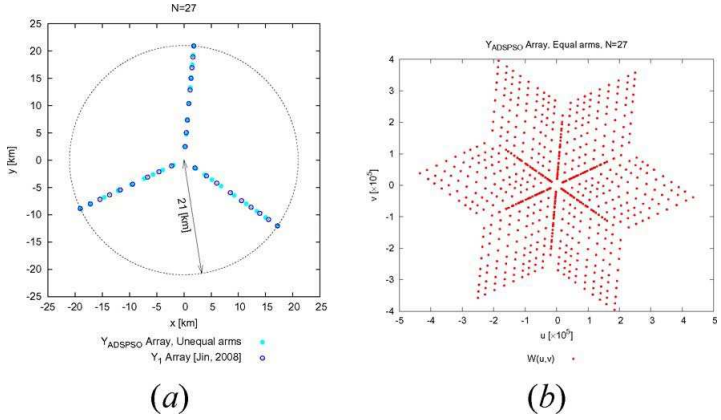


Figure 43. *Problem B* [*Equal-unequal arms*, $N = 27$] - Synthesis results for the RNDPSO and ADSPSO approaches: (a) optimal Y_{ADSPSO} array arrangement and (b) associated $u - v$ coverage function.

The improvements allowed by the ADSPSO are even more evident when addressing *Problem C* (Fig. 44), as confirmed by the indexes in Table VI (e.g., $\nu^{un}_{ADSPSO} = 0.882$ versus $\nu_2 = 0.865$). As far as the trade-off index Ξ is concerned, one could notice that the ADSPSO solution for the *Problem C* still overcomes the corresponding ADSGA design ($\Xi^{un}_{ADSPSO} = 0.28$ versus $\Xi^{un}_{ADSGA} = 0.31$), but it does not reach the optimal value yielded by the ADSPSO when applied to *Problem B* (Table VI). Such results, together with that from the bare ADS ($\Xi^{un}_{ADS} = 0.32$) indicate that, whatever the design technique, the configurations suitable for *Problem C* are not reliable compromise solutions.

For completeness and further verification of the positive effect of the increased number of *DoFs* of the real-valued optimization, the ADSPSO has been applied to *Problem A* as well (Fig. 45), although the ADSGA was already able to improve state-of-the-art performances. The flexibility of the real-valued encoding used in the ADSPSO allows a further reduction of the array PSL with respect to the ADSGA (and obviously Y_3) in both the equal case ($PSL^{eq}_{ADSPSO} = -20.91$[dB] versus $PSL^{eq}_{ADSGA} = -19.84$[dB]) and unequal one ($PSL^{un}_{ADSPSO} = -21.35$[dB] versus $PSL^{un}_{ADSGA} = -20.93$[dB]), but at the expense of the geometric regularity of the GA or bare ADS lattice-based solution [Fig. 45(b) versus Figs. 42(b) and 38(c)].

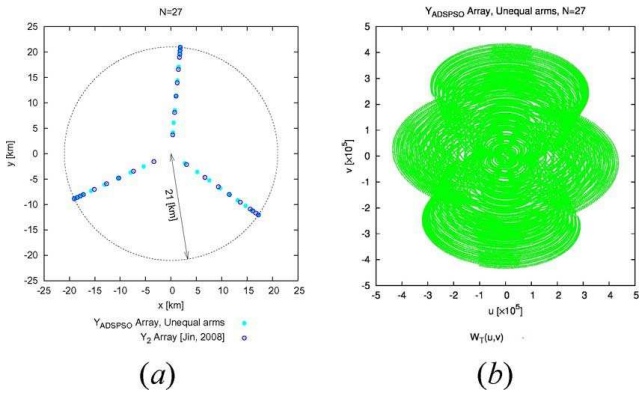


Figure 44. *Problem C [Equal-unequal arms, $N = 27$]* - Synthesis results for the RNDPSO and ADSPSO approaches: (a) optimal array arrangement and (b) associated tracking $u - v$ coverage function.

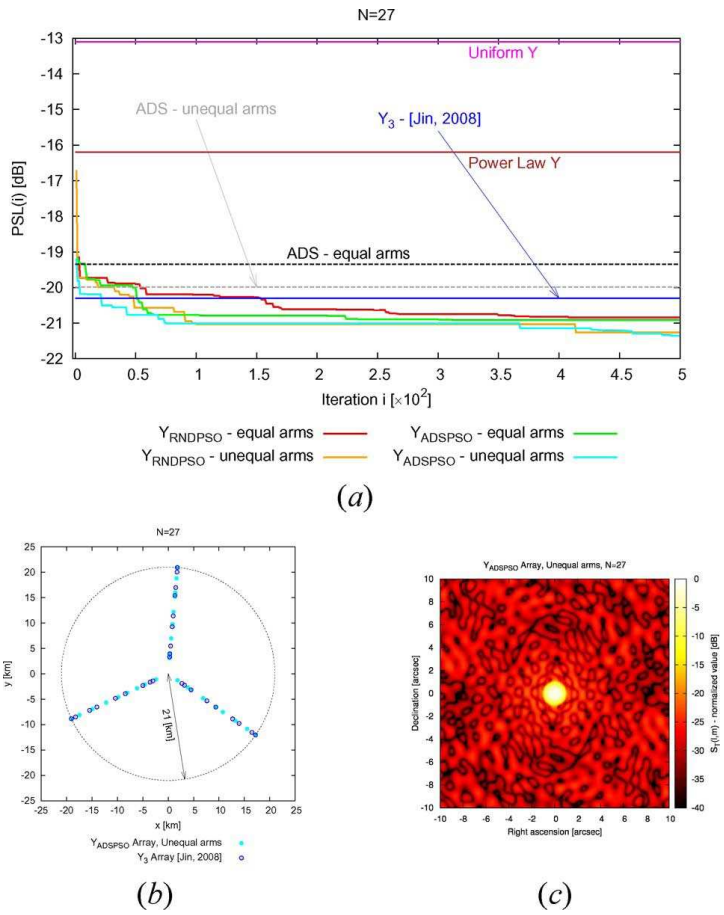


Figure 45. *Problem A [Equal-unequal arms, $N = 27$]* - Synthesis results for the RNDPSO and ADSPSO approaches: (a) Behavior of the optimal PSL versus the iteration number i, and comparison with reference designs from [31], (b) optimal Y_{ADSPSO} array arrangement, and (c) associated synthesized pattern.

As it can be observed, the ADS initialization allows an improvement of the optimization technique performance whatever the problem at hand [Table VI]. Moreover, from previous outcomes, the real-valued ADS hybrid approach seems to always yield better performance than the GA-based technique (Table VI). Such a conclusion could be misleading since it has been drawn for arrays with a small number of active elements ($N = 27$) [31]. In order to evaluate the performance of the ADS-based methods when dealing with larger N, the *Problem A* is still addressed, but considering medium (large) N. More in detail, *Problem A* is formulated by assuming $L = 100$ (210) Km, $\Delta u \times N_u = \Delta v \times N_v = \Delta_{grid} \times N_{grid} = \frac{4.16 \times 10^5}{\lambda} \left(\frac{4.2 \times 10^5 \sqrt{3}}{\lambda} \right)$, and $\mathcal{Z} = 2$. Consequently, the hybrid solvers have been applied with the following setup: $V_{GA} = V_{PSO} = 20$, $I_{MAX} = 400$ and $Q = \frac{N}{E} = 44$ (90). Moreover, the reference ADS sequence is the (88,44, 21, 22)- ADS $\mathbf{D}_2$ [(180, 90, 44, 45)- ADS $\mathbf{D}_3$] [68]. In order to point out the efficiency of binary-valued techniques, Fig. 46 shows the optimization of the PSL during the iterative minimization. As it can be observed, the GA-based approaches outperform the corresponding PSO implementations when dealing with both medium and large arrays ($PSO^{eq}_{ADSGA}\rfloor_{N=132} = -15.86$[dB] vs. $PSL^{eq}_{ADSGA}\rfloor_{N=132} = -17.54$[dB], $PSL^{eq}_{ADSPSO}\rfloor_{N=270} = -18.35$[dB] vs. $PSL^{eq}_{ADSGA}\rfloor_{N=270} = -20.15$[dB]). Such a behavior is mainly related to the greater efficiency of the binary optimizers in sampling very large solution spaces as those when is medium/large. On the other hand, it is worthwhile to point out the more significant effect of the ADS initialization on the arising PSL and the convergence rate of the optimization when applying GA-based approaches (Fig. 46),while the improvement turns out to be less important using real-valued PSO approaches.

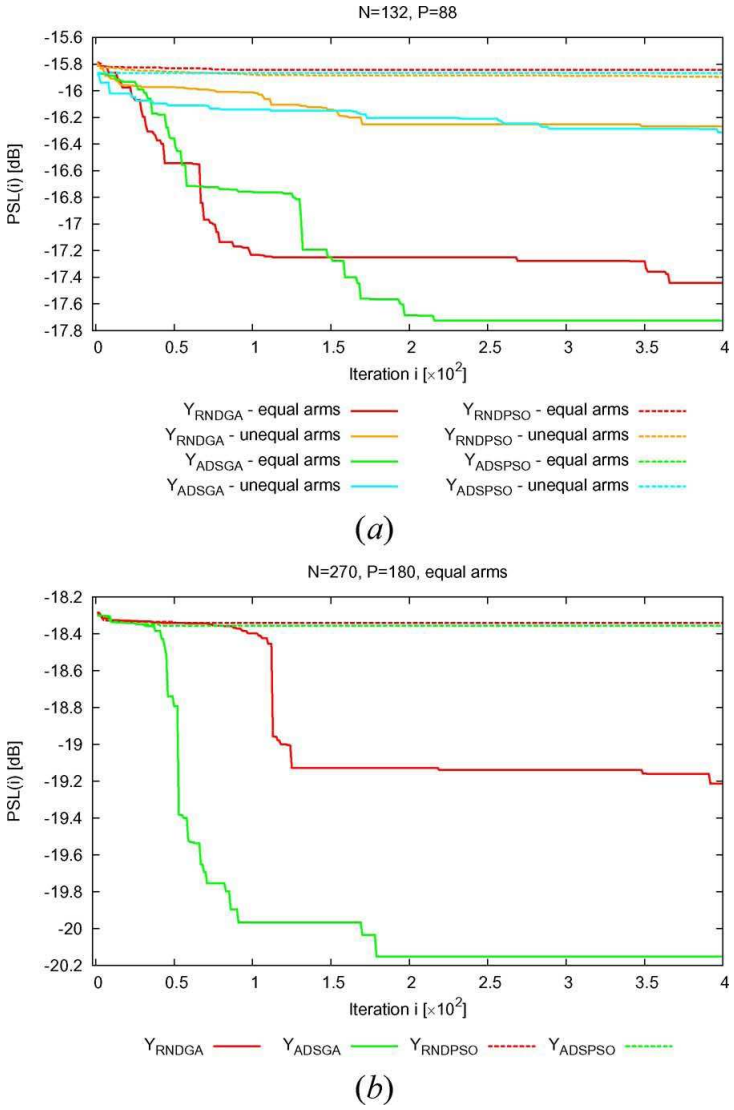


Figure 46. *Problem A* - Behavior of the optimal PSL versus the iteration number i for the RNDGA, ADSGA, RNDPSO, and ADSPSO approaches for (a) $N = 132$ (equal and unequal arms) and (b) $N = 270$ (equal arms).

In order to provide further insights on the expected improvements over existing approaches, the next experiment deals with a design example for the 12-m Atacama Large Millimeter/Sub millimeter Array (ALMA) [57] (*Problem A* has been considered). In this case, a *Y-shaped* ($E = 3$) layout with $L = 9000$[m], $N = 63$, $\phi = \pi/6$, $\mathcal{L} = \mathcal{D} = -23^{\circ}$ and $\mathcal{E} = 0$ [57] has been optimized at 300GHz assuming $N_u = N_v = 256$, $\Delta u \times N_u = \Delta v \times N_v = \Delta_{grid} \times N_{grid} = 3.2 \times 10^7$, and $\mathcal{Z} = 2$. The results obtained starting from the (42,21,10, 31)-ADS $\mathbf{D}_4$ (Table IV) indicate that PSO-based approaches overcome GA-based techniques (Table VII), as expected because of the moderate value of [Fig. 47(a)], by achieving

$PSL^{un}_{ADSPSO} = -18.55$[dB] [Fig. 47($b$)]. Moreover, it is worth noticing that the "*unequal*" layouts always guarantee non-negligible improvements over their equally spaced counterparts (e.g., $PSL^{eq}_{ADSGA} = -17.25$[dB] versus $PSL^{un}_{ADSGA} = -17.56$[dB] - Table VII). Furthermore, the comparisons with uniform and power law analytical designs (Table VII) further assess the effectiveness of the proposed approaches (e.g., $PSL^{un}_{ADS} = -15.57$[dB] versus $PSL_{pl} = -11.01$[dB] - Table VII).

		Equal arms				*Unequal arms*			
Design	*Problem*	PSL [dB]	B	ν	I	PSL [dB]	B	ν	I
Uniform	-	-12.40	2766	0.712	-	-	-	-	-
Power-Law	-	-11.01	2479	0.610	-	-	-	-	-
ADS	A	$\mathbf{-15.48}$	2412	0.731	-	$\mathbf{-15.57}$	2372	0.721	-
GA	*A*	$\mathbf{-16.82}$	998	0.550	207	$\mathbf{-17.02}$	1618	0.716	370
ADSGA	*A*	$\mathbf{-17.25}$	1044	0.511	87	$\mathbf{-17.56}$	1544	0.671	282
PSO	*A*	$\mathbf{-17.58}$	931	0.607	233	$\mathbf{-17.61}$	779	0.501	309
ADSPSO	*A*	$\mathbf{-18.08}$	893	0.562	152	$\mathbf{-18.55}$	877	0.596	266

Table VII. *Numerical results* - Comparison among optimized *ALMA* configuration (bold numbers identify optimized quantities).

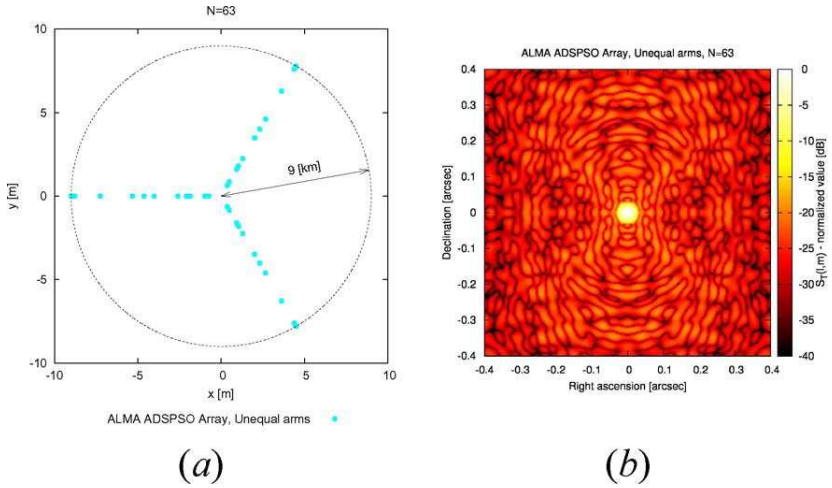


Figure 47. *ALMA - Problem A* [*Equal-unequal arms*, $N = 63$] - Synthesis results for the ADSPSO approach: (a) optimal array arrangement and (b) associated $S_T\,(u, v)$.

Finally, the last example is aimed at analyzing the hybrid approaches when applied to the synthesis of a different geometry and set of parameters. Let us consider a "*Cross*" geometry ($E = 4$) at 1.42GHz characterized by $L = 189$[m], $N = 60$, $\phi = 0$, $N_u = N_v = 256$, $\Delta u \times N_u = \Delta v \times N_v = \Delta_{grid} \times N_{grid} = 4000$, $\mathcal{Z} = 2$, $\mathcal{D} = -33.8°$, $\mathcal{E} = 0$, $\mathcal{L} = -33.8°$ (i.e., similar to the "*Chris-Cross*" array [8][60]). The results from the synthesis process starting from the reference sequence (30,15,7,2)-ADS $\mathbf{D}_5$, indicate that, as expected, PSO-based approaches provide better layouts [Fig. 48(a)] than GA-based techniques because of the relatively small dimension of the solution space (i.e., low number of active elements). Moreover, the improvement caused by "*unequal*" element placement is more significant than for the Y geometry. This is due to the highest redundancy of the Cross geometry that can be more easily broken by an unequal arm displacement [e.g., Fig. 48(b)]. For completeness, a summary of the performance indexes is given in Table VIII. These results further confirm the effectiveness of an ADS initialization to enhance the efficiency of the optimization procedures (e.g., $PSL_{RNDPSO} - PSL_{ADSPSO} \approx 1.4$[dB] - Table VIII).

		Equal arms				*Unequal arms*			
Design	*Problem*	PSL [dB]	B	ν	I	PSL [dB]	B	ν	I
GA	A	$\mathbf{-14.21}$	201	0.572	157	$\mathbf{-14.69}$	283	0.763	397
ADSGA	A	$\mathbf{-14.90}$	261	0.692	159	$\mathbf{-15.16}$	283	0.756	389
PSO	A	$\mathbf{-16.29}$	265	0.905	387	$\mathbf{-17.83}$	339	0.873	328
ADSPSO	A	$\mathbf{-17.69}$	265	0.912	324	$\mathbf{-21.10}$	301	0.847	266

Table VIII. *Numerical results* - Comparison of optimized *Cross* arrays (bold numbers identify optimized quantities).

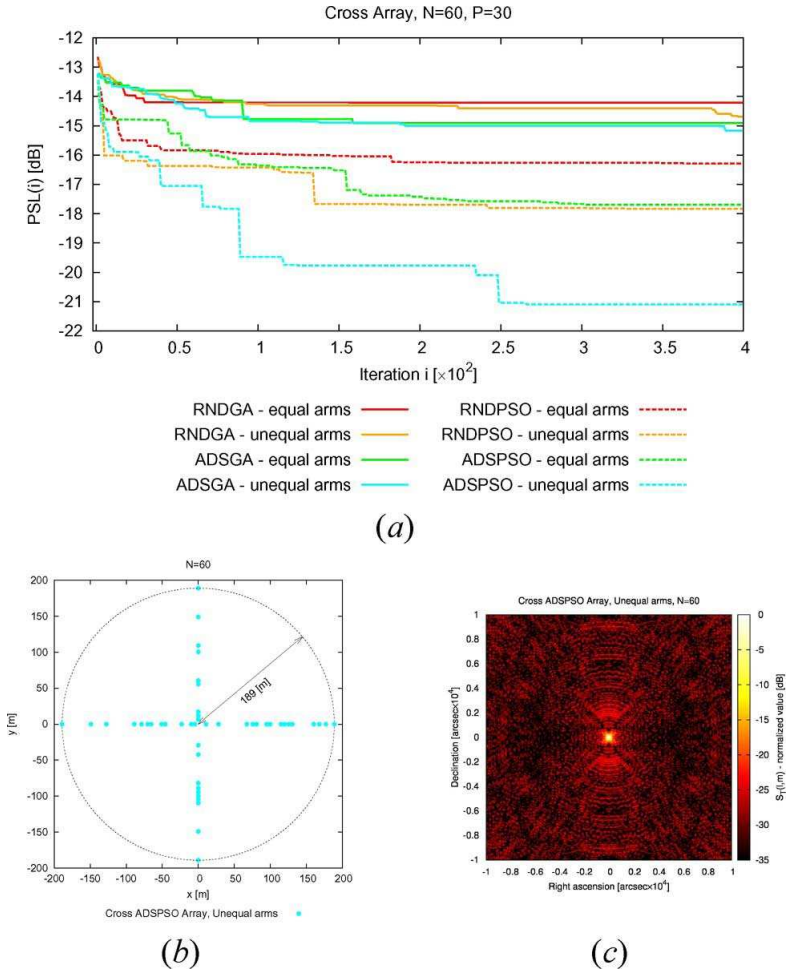


Figure 48. *Cross arrays - Problem A [Equal-unequal arms, $N = 60$]* - Synthesis results for the RNDGA, ADSGA, RNDPSO and ADSPSO approaches: (a) behavior of the optimal PSL versus the iteration number i, (b) optimal ADSPSO array arrangement and (c) associated $S_T\,(u, v)$.

Chapter 6

Hybrid Almost Difference Set (ADS)-based Genetic Algorithm (GA) Method for Planar Array Thinning

6.1 Introduction

In the framework of antenna arrays for communication and space applications, such as radars for remote sensing, arrays for microwave imaging or satellite and ground communications one of the most important requirements is represented by high directivity and low peak sidelobe level (PSL) [1]. To satisfy these requirements an interesting solution is represented by large thinned arrays. Thinned arrays, as put in evidence in [6] are a good solution since thinning offers reduction in element count, cost, weight, power consumption, and heat dissipation, albeit with an attendant reduction of the antenna gain. In scientific literature ([5][6][12]) it is showed that the main drawback associated to thinning is the loss of sidelobe control. Several different techniques have been proposed and developed to overcome such a problem: e.g. random technique [3][12], algorithmic approaches [12], dynamic programming [71], genetic algorithms [18][44], simulated annealing [38], and particle swarm optimisers [25]. One of the more interesting approaches is based on the use of deterministic combinatorial sequences called difference sets (DSs) that have been successfully employed to analytically determine thinned arrangements with well controlled sidelobes [5]. This approach generate arrays that have element locations constrained by an algorithm based on difference sets. These constraints produce arrays with PSLs demonstrably better than those obtainable with simple cut-and-try placement techniques, as well as many previously published algorithmic techniques. Since only a limited number

of DS sequences exists, recently a new analytical approach have been proposed. Such an analytical approach has been extended to a wider class of geometries by exploiting the mathematical properties of almost difference sets (ADSs) [61][65]. ADSs are characterized by a three-valued autocorrelation function that allows to obtain deterministic arrays configuration with a controlled and predictable PSL [50]. Moreover, the reliability of the analytic ADS-based thinning has been analysed also taking into account the mutual coupling effects among array elements [46]. However, despite several interesting features and advantages, the use of ADS sequences for array thinning has some limitations. In more detail

- arrays based on ADS sequences usually provide sub-optimal and not optimal PSL performance;
- although large repositories of ADSs are available [16], the possible array configurations are limited. ADS arrays with arbitrary aperture sizes and thinning factors cannot be designed, since ADS sequences exist only for specific sets of descriptive parameters;
- a general purpose ADS construction techniques do not exist at present. The explicit forms of ADS sequences has to be determined on a case by case basis using suitable construction theorems [61][65] or other approaches.

The aim of this chapter is to introduce a way to improve and enhance the ADS-based design technique and to overcome the above limitations [50]. The main idea is to exploit a GA-based procedure, that is particularly suitable for these applications for the following considerations

1. GAs are able to deal with binary optimisation problems;
2. GAs have been used and applied to thin antennas arrays [18];
3. GA optimization procedure can integrate information and constraints of ADSs [44].

In other words the method that is proposed in this chapter is a GA-enhanced ADS technique, called hereinafter ADSGA. Differently from other works published about exploiting ADS to thin antenna arrays [22][46], as well as other array design problems (such as interleaved arrays [69]), the proposed approach does not rely on a analytic technique but on a hybrid one. Consequently it is not possible to determine a priori performance bounds. The main objectives of this chapter are not only to propose a hybrid technique to design linear thinned arrays, but also to proposed an approach useful when either the ADS-based

array performance do not comply with the radiation requirements of the application at hand or no ADS is available for the geometry (aperture size or thinning factor) under study [50].

The structure of the chapter is as follows. First of all a review of ADS design techniques for planar array thinning is presented. Then a GA-enhanced methodology is proposed to address three different problems concerned with ADS-based planar arrays (Section 6.2). In Section 6.3, the hybrid technique is applied to the three problems and validated by means of several numerical simulations. Representative results concerned with both small and large arrays as well as different thinning factors are discussed to point out its reliability.

6.2 Problem statement and mathematical formulation

Let us consider a planar arrangement defined over a lattice of $P \times Q$ positions ($N = P \times Q$ being the total number of elements) [23]. The array factor of is equal to

$$W(u,v) = \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} s(p,q) \exp\left[2\pi i \left(p d_x u + q d_y v\right)\right] \tag{6.1}$$

where $s(p,q)$ is the array weight of the (p,q)th element, d_x and d_y are the lattice spacings along the x and y directions (in wavelengths), $u = \sin(\theta)\cos(\phi)$, and $v = \sin(\theta)\sin(\phi)$ ($u^2 + v^2 \leq 1$) [23]. Dealing with equally weighted thinned arrays, $s(p,q) = 0$, $p = 0, ..., P-1$, $q = 0, ..., Q-1$, can either assume the value 1 or 0 when an element is present or not at the (p,q)th lattice position. In ADS-based thinning techniques the lattice weights are selected as follows

$$s(p,q) = \begin{cases} 1 & if\ (p,q) \in \mathbf{D} \\ \\ 0 & otherwise \end{cases} \tag{6.2}$$

where $\mathbf{D}$ is a (N, K, Λ, t)-ADS, where N is the array size, K the number of active elements, and Λ and t are parameters which define the autocorrelation properties of the considered ADS [23]. A (N, K, Λ, t)-ADS is defined as a K-subset $\mathbf{D} = \{\mathbf{d}_k \in \mathbf{G}, k = 0, ..., K-1\}$ of the Abelian group $\mathbf{G}$ of order $P \times Q$ ($\mathbf{G} = \mathbb{Z}^P \otimes \mathbb{Z}^Q$, P and Q being chosen according the Kronecker Decomposition Theorem) for which the multiset

$$\mathbf{M} = \{\mathbf{m}_j \in (\mathbf{d}_h - \mathbf{d}_l), \mathbf{d}_h \neq \mathbf{d}_l; j = 0, ..., K(K-1) - 1\}$$

contains t nonzero elements of $\mathbf{G}$ each exactly Λ times and the remaining $PQ-1-t$ nonzero elements each exactly $\Lambda+1$ times [23]. Therefore, an ADS satisfies the following existence condition

$$K(K-1) = t\Lambda + (PQ-1-t)(\Lambda+1) \tag{6.3}$$

where $K \geq \Lambda+1$, $0 \leq K \leq PQ$, and $0 \leq t \leq PQ-1$. Moreover, it is worth noticing that DSs and ADSs for which $t = PQ-1$ or $t = 0$. if $\mathbf{D}$ is a (N, K, Λ, t)-ADS, then it is possible to define the two dimensional binary sequence $\mathbf{S} = \{s(p,q) = 1(0),\ if\ (p,q) \in (\notin)\mathbf{D};\ p = 0, ..., P-1, q = 0, ..., Q-1\}$ [23].

In more detail, by exploiting the properties of the autocorrelation function, $\xi(\tau_x, \tau_y) = \sum_{p=0}^{P-1}\sum_{q=0}^{Q-1} s(p,q)s\left[(p+\tau_x)|_{modP}, (q+\tau_y)|_{modQ}\right]$ (being $P \times Q$ its period), of ADS binary sequences, which is known to be the three-level function

$$\xi(\tau_x, \tau_y) = \begin{cases} K & (\tau_x, \tau_y) = 0 \\ \Lambda & for\ t\ values\ of\ (\tau_x, \tau_y) \\ \Lambda+1 & otherwise \end{cases} \tag{6.4}$$

it turns out that the power pattern $|W(u,v)|^2$ of an ADS-based array satisfies the following constraint

$$\begin{gathered} \left|W\left(\frac{k}{s_x P}, \frac{l}{s_y Q}\right)\right|^2 \quad = \quad \Upsilon(k,l) \\ k = 0, ..., P-1, \quad l = 0, ..., Q-1 \end{gathered} \tag{6.5}$$

i.e., the samples of the power pattern are equal to the inverse discrete Fourier transform (IDFT) of the autocorrelation function $\xi(\tau_x, \tau_y)$,

$$\Upsilon(k,l) = \sum_{p=0}^{P-1}\sum_{q=0}^{Q-1} \xi(\tau_x, \tau_y) \exp\left[2\pi i\left((\tau_x k)/P + (\tau_y l)/Q\right)\right],$$

which, from (6.4), turns out to be equal to

$$\Upsilon(k,l) = K - \Lambda + N\Lambda\delta(k,l) + \Psi(k,l) \tag{6.6}$$

In (6.6), $\delta(k,l)$ is the discrete impulse function [$\delta(k,l) = 1$ if $k = l = 0$, and $\delta(k,l) = 0$] otherwise $\Psi(k,l) = IDFT\{\psi(\tau_x, \tau_y)\}$, where $\psi(\tau_x, \tau_y) = \sum_{r=1}^{N-1-t} \delta\left(\tau_x - \tau_x^r, \tau_y - \tau_y^r\right)$, and $\left(\tau_x^r, \tau_y^r\right)$, $r = 1, ..., N-1-t$, are the indexes at which $\xi\left(\tau_x^r, \tau_y^r\right) = \Lambda+1$ [23]. According to (6.4), the ADS sequence exhibits a three-level autocorrelation function.

Thanks to (6.5), the following a priori bounds have been derived for the one-way PSLs of ADS-based thinned arrays [23]:

$$PSL^{INF} \leq PSL^{MIN} \leq PSL^{OPT} \leq PSL^{MAX} \leq PSL^{SUP} \tag{6.7}$$

where $PSL^{OPT} = \min_{\sigma_x,\sigma_y} \left\{ PSL\left[\mathbf{D}^{(\sigma_x,\sigma_y)}\right] \right\}$, $\sigma_x = 0, ..., P-1$, $\sigma_y = 0, ..., Q-1$,

$$PSL\left[\mathbf{D}^{(\sigma_x,\sigma_y)}\right] = \frac{\max\limits_{(u,v)\notin M} \left| W^{(\sigma_x,\sigma_y)}(u,v) \right|^2}{\left| W^{(\sigma_x,\sigma_y)}(0,0) \right|^2}$$

where (u_0, v_0) is the main lobe steering direction, M is the main lobe region [23], and $\left|W(\sigma_x,\sigma_y)(u,v)\right|^2$ is the power pattern [23] of the layout generated from the cyclical shift sequence of the reference ADS, $\mathbf{D}^{(\sigma_x,\sigma_y)}$,

$$\mathbf{D}^{(\sigma_x,\sigma_y)} = \left\{ \left((p+\sigma_x)_{modP}, (q+\sigma_y)_{modQ} \right);\ (p,q) \in \mathbf{D};\ \sigma_x, \sigma_y \in \mathbb{Z} \right\}.$$

The analytic expressions of the bounds in (6.6) state the peak sidelobe level of ADS-based arrays is constrained by the *a priori* known quantities PSL^{INF}, PSL^{MIN}, PSL^{MAX}, PSL^{SUP} [23]:

- $PSL^{INF} = \frac{K-\Lambda-\sqrt{\frac{(t+1)(N-t-1)}{(N-1)}}}{K^2}$
- $PSL^{MIN} = \frac{\left[\min\limits_{(k,l)\in\mathcal{H}_0} \{\Upsilon(k,l)\}\right]^{[0.5+0.8\log_{10}(N)]}}{K^2}$
- $PSL^{MAX} = \frac{\left[\max\limits_{(k,l)\in\mathcal{H}_0} \{\Upsilon(k,l)\}\right]^{[-0.1+1.5\log_{10}(N)]}}{K^2}$
- $PSL^{SUP} = \frac{\left(K-\Lambda+\sqrt{(t+1)(N-t-1)}\right)^{[-0.1+1.5\log_{10}(N)]}}{K^2}$

These constrains on PSL indicate that ADS-based thinned arrays exhibit a sidelobe level which can be predicted either from the knowledge of the features of the ADS sequence (PSL^{INF} and PSL^{SUP} only depend on N, K, Λ and t) or from the expression of $\Upsilon(k,l)$ (necessary for computing PSL^{MIN} and PSL^{MAX} and returning higher accuracy of estimation) [23].

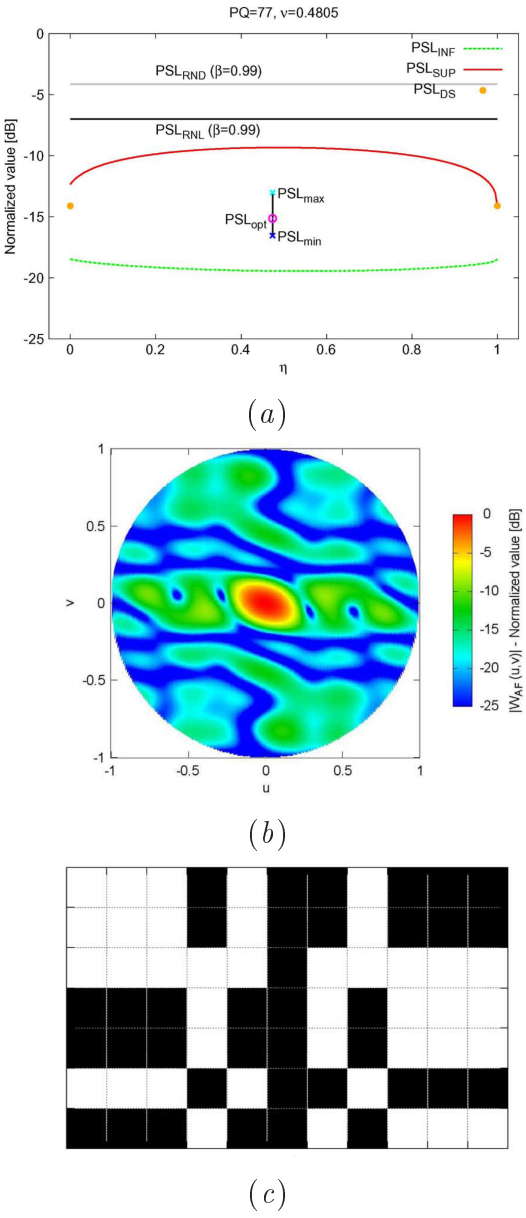


Figure 49. *Example from* [23] *of Planar Array based on* $\mathbf{D}_3^{opt}$- *ADS* . Number of elements: $P \times Q = 7 \times 11$. Plots of the PSL bounds versus $\eta = \frac{t}{PQ-1}$ ($PQ = 77$, $\nu = 0.4805$) (a). Plot of the normalized array factor (b) generated from $\mathbf{D}_3^{opt}$- *ADS* array arrangement (c) (courtesy from [23]).

As put in evidence in the Introduction, the ADS-based approach for array thinning is limited, despite of the advantages of random thinned arrays and computational efficiency. Therefore a methodology able to overcome these limitations while exploiting the ADS analytic features seems to be of some interest in view of communication and space applications [50]. Accordingly, the ADSGA hybrid approach is presented. This methodology tries to exploit the advantages associated to both ADS and GA-based techniques [50].

Concerning the iterative ADSGA optimisation, the standard structure of the GA is modified to exploit the positive key features of the ADSs. The structure of the Genetic Algorithm (GA) considered in this work are briefly described [44][50]:

1. *Initialisation*: The Initial ($i = 0$) population is randomly chosen. A set of M trial solutions, $\rho_m(i), m = 1, ..., M$ is defined. The way to define this set of trial solution allows to define ADSGA method instead of standard GA.

2. *Coding*: Each Individual $\rho_m(i)$ codes the values of an unknown set of parameters into a binary string (called Chromosome);

3. *GA-Evolution*: At each iteration i, the genetic evolution exploit suitable binary operators of evolution and natural selection (selection, crossover, reproduction, mutation and elitism [6, 9]) applied in a probabilistic fashion and taking into account the fitness values $F_m = F\{\rho_m(i)\}, m = 1, ..., M$ of current trial solutions;

4. *Termination*: The iterative procedure ends when one of the following stop criteria is satisfied. The optimal fitness value, $F_{POP} = \min_m \{F_m\}$, is smaller than an user-defined threshold or the maximum number of iterations I_{max} has been reached. The 'final solution' is the fittest trial solution determined throughout the whole iterative process, $\rho_{conv} = \arg\left\{\min_i \left(\min_m [F\{\rho_m(i)\}]\right)\right\}$.

The initial population ($i = 0$, i being the iteration index) is generated as follows for ADSGA method [50]. The $N = P \times Q$ shifted versions of a reference ADS are ranked according to their PSL values. Then, half-trial solutions (M being the dimension of the GA population) are chosen with chromosomes equal to the binary sequences of the first $M/2$ highly ranked shifted ADSs

$$\rho_m(i) = \left\{b_m(n) = w^{(m)}(p,q);\ p = 0, ..., P-1,\ q = 0, ..., Q-1, n = 0, ..., N-1\right\}$$

$$1 \leq m \leq \tfrac{M}{2} \tag{6.8}$$

where $b_m(n)$ is the nth digit of the mth trial solution and $s^{(\sigma_x,\sigma_y)}(p,q) = s^{(m)}(p,q) = 1$ (being $m = \sigma_x + (P \times \sigma_y - 1) = f(\sigma_x, \sigma_y)$) if $(p,q) \in \mathbf{D}^{(\sigma_x,\sigma_y)}$ and $s^{(\sigma_x,\sigma_y)}(p,q) = s^{(m)}(p,q) = 0$, otherwise. Concerning the remaining of the population, the trial solutions are chosen randomly within the range of admissibility of the problem at hand

$$\rho_m(i) = \{b_m(n) = r_m(n); n = 0, ..., N-1\}$$

$$1 \leq m \leq \tfrac{M}{2} \tag{6.9}$$

$r_m(n)$ being a random digit and $N = P \times Q$. Such an initialisation allows the "transfer" into the GA chromosomes of the good ADS-based schemata also providing a sufficient variability within the population to avoid the stagnation [6]. As regards the GA operators, both crossover and mutation are applied following the standard binary implementations [6], but also guaranteeing the updated trial solutions be admissible and comply with the problem constraints (e.g.fixed thinning factor $\nu = K/N$) [50]. Towards this end, the crossover operation is repeated until the new chromosomes satisfy the solution constraints, while a conditioned mutation is applied. More specifically, let ν be the user-defined thinning factor, then the bit-mutation probability is defined as follows [50]

$$P_{BM}(n) = \frac{\left[N \times \nu - \sum_{h=0}^{n-1} b(h)\right]}{N-n} \times [1 - 2b(n)] + b(n) \tag{6.10}$$

The set of parameters of he GA-based procedure are: $P_C = 0.9$ (crossover), $P_M = 0.01$ (mutation rate) and $N = P \times Q$ (population size) if not otherwise stated.

6.2.1 Problem I - PSL minimisation in array synthesis

In order to determine an optimal thinned configuration starting from the (usually) sub-optimal ADS arrangement with a given aperture size $N_{ADS} = P_{ADS} \times Q_{ADS}$ and thinning factor ν_{ADS}, let us formulate the following constrained optimisation problem, similarly to [50]

Problem I: Minimise $F\{\rho\} = \max_{(u,v)\notin R_M} \left\{|W(u,v)|^2\right\} / |W(0,0)|^2$, R_M the main lobe region as

$$R_M = \left\{ (u,v) \in [-1,1] \times [-1,1] : \; u^2 + v^2 \leq 1, \; uv \leq \frac{K}{4PQs_x s_y \max_{(k,l)\in\mathcal{H}_0} \{\Upsilon(k,l)\}} \right\},$$

subject to $K = K_{ADS}$ and $N = N_{ADS}$ (namely $P = P_{ADS}$ and $Q = Q_{ADS}$).

to be solved through ADSGA. In Such a case, the GA fitness function is defined as the PSL of the array while the constraints force the array to kept its descriptive parameters (i.e. original dimension, $N = N_{ADS}$, and thinning, $\nu = \nu_{ADS}$).

6.2.2 Problem II - extension of the range of ADS applicability in array synthesis

The use of an ADS-based technique for array synthesis is sometimes limited to fixed array dimensions and thinning values because of the limited, although quite large, set of available ADS sequences. In order to design a thinned configuration with arbitrary values of $N = P \times Q$ and ν, still exploiting the properties of the existing ADS arrangements, the following problem is at hand (in a similar way to [50])

Problem II: Minimise $F\{\rho\} = \max_{(u,v)\notin R_M} \left\{|W(u,v)|^2\right\} / |W(0,0)|^2$, R_M the main lobe region as

$$R_M = \left\{ (u,v) \in [-1,1] \times [-1,1] :\ u^2 + v^2 \leq 1,\ uv \leq \frac{K}{4PQs_x s_y \max\limits_{(k,l)\in\mathcal{H}_0} \{\Upsilon(k,l)\}} \right\},$$

are subject to $K = \hat{K}$ and $N = \hat{N}$, being $\hat{K} \neq K_{ADS}$ and/or $\hat{N} \neq N_{ADS}$ (namely $\hat{P} \neq P_{ADS}$ and $\hat{Q} \neq Q_{ADS}$).

Such a constrained optimisation problem is quite similar to that in previous Section, but, in this case, no ADS-based array is available in correspondence with the array parameters ($\hat{N}$, $\hat{K}$) [50].

6.2.3 Problem III - definition of a general purpose ADS construction technique for array synthesis

With reference to the potential limitation (III) outlined in the Introduction, the aim is now to find the explicit forms of ADSs sequences (i.e. binary sequences with a three-level autocorrelation function) for arbitrary values of N. Towards this end, let us denote with$L\{\rho\}$ and $R\{\rho\}$ the number of levels of the autocorrelation function $\xi(\tau_x, \tau_y)$ of a trial solution ρ and the number of (τ_x, τ_y) values for which $\xi(\tau_x, \tau_y)$ differ from 6.4. Then, the search for admissible (but not available in ADS repositories) ADS sequences is recast as the solution of the following

Problem III: Minimise $F\{\rho\} = \alpha[L\{\rho\} - 3] + \beta R\{\rho\}$ subject to $N = \hat{N}$, where $\hat{N} \neq N_{ADS}$ (namely $\hat{P} \neq P_{ADS}$ and $\hat{Q} \neq Q_{ADS}$) and α and β are suitable user-defined weight coefficients [47].

In such a case, the optimisation at hand turns out to be different from that in Problem I and Problem II. As a matter of fact, it is defined and performed with the ADSGA within the "autocorrelation space" instead of in the "pattern space", while the constraints are still on the set of parameters defining the ADS as well as the corresponding array arrangement [50].

6.3 Numerical analysis

6.3.1 Application to Problem I

As suggested in [5], the combinatorial and the stochastic methods are combined in in order to take advantage from their good characteristics and to compensate for their drawbacks. The ripples caused by ADS sequences can be corrected by GA capabilities, while the controlled PSL of ADS-based arrays is useful to speedup the convergence of the genetic procedure and get optimal PSL for planar arrays. The inter element distances are assumed $d_x = d_y = \frac{1}{2}$ hereinafter.

In particular, now we consider when the application deals with Problem I: obtain an optimal thinned configuration starting from the ADS arrangement and comparing it with standard GA approach.

As stated in previous section, to determine an optimal thinned configuration starting from the (usually) sub-optimal ADS arrangement with a given aperture size $N_{ADS} = P_{ADS} \times Q_{ADS}$ and thinning factor ν_{ADS}, let us formulate the following constrained optimisation problem, that is to minimise $F\{\rho\} = \max_{(u,v)\notin R_M} \left\{|W(u,v)|^2\right\} / |W(0,0)|^2$, R_M the main lobe region, subject to $K = K_{ADS}$ and $N = N_{ADS}$ (namely $P = P_{ADS}$ and $Q = Q_{ADS}$). The constraints force the array to kept its descriptive parameters (i.e. original dimension, $N = N_{ADS}$, and thinning, $\nu \approx \nu_{ADS}$).

The experiments deal with the 2D ADSs described in the following Table

N	P	Q	K	Λ	t	ν
49	7	7	25	12	24	0.555
121	11	11	61	30	60	0.502
289	17	17	145	72	144	0.501
529	23	23	265	132	264	0.500

Table IX: Properties of the ADS sequences

6.3.1.1 Array arrangement $P \times Q = 7 \times 7$

In this example we have used to initialize the population at step $i = 0$, the $(7 \times 7, 25, 12, 24)$-ADS ($N_{ADS} = 49$, $\nu_{ADS} = 0.555$). Fig 1 shows the behaviour of the optimal fitness value against the iteration number i in correspondence with the ADSGA and the standard GA minimization procedure.

The results can be summarized in the following

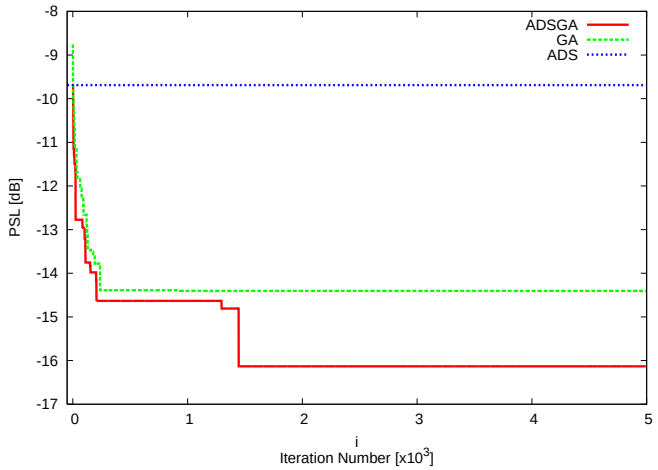
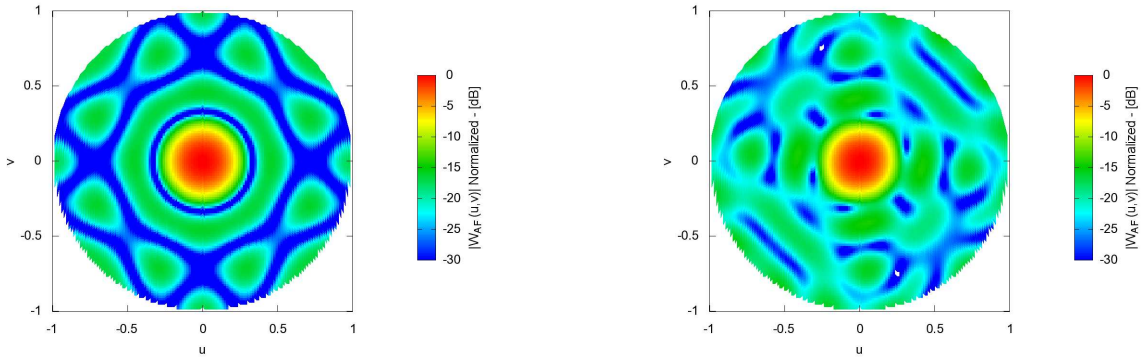
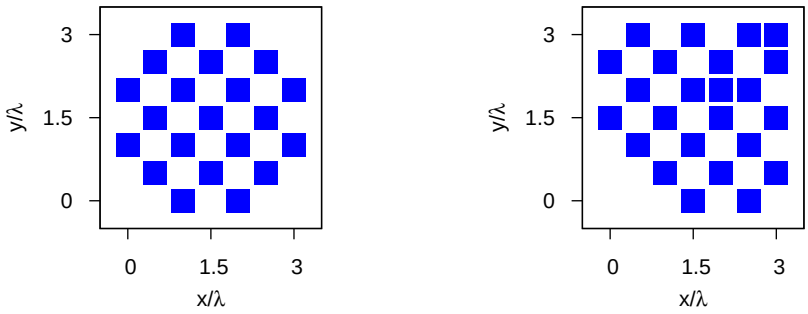


Figure 50: *Problem I- PSL minimisation in array synthesis*: Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.



(a) *(b)*



(c) *(d)*

Figure 51. *Numerical validation - Problem I -PSL minimisation in array synthesis:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.1.2 Array arrangement $P \times Q = 11 \times 11$

In this example we have used to initialize the population at step $i = 0$, the $(11 \times 11, 61, 30, 60)$-ADS ($N_{ADS} = 121$, $\nu_{ADS} = 0.502$). Fig 1 shows the behaviour of the optimal fitness value against the iteration number i in correspondence with the ADSGA and the standard GA minimization procedure.

The results can be summarized in the following Figures.

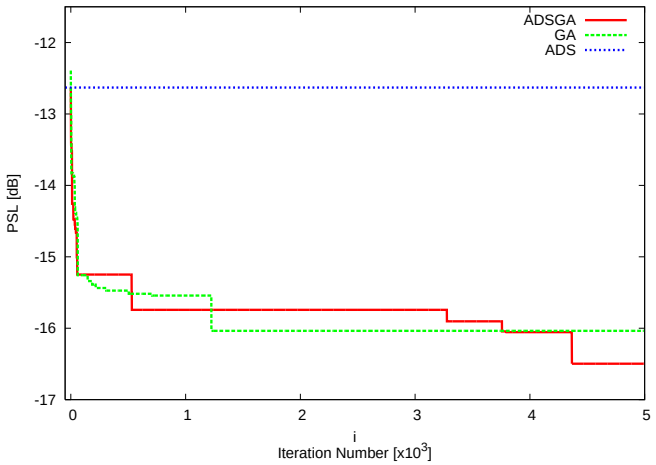
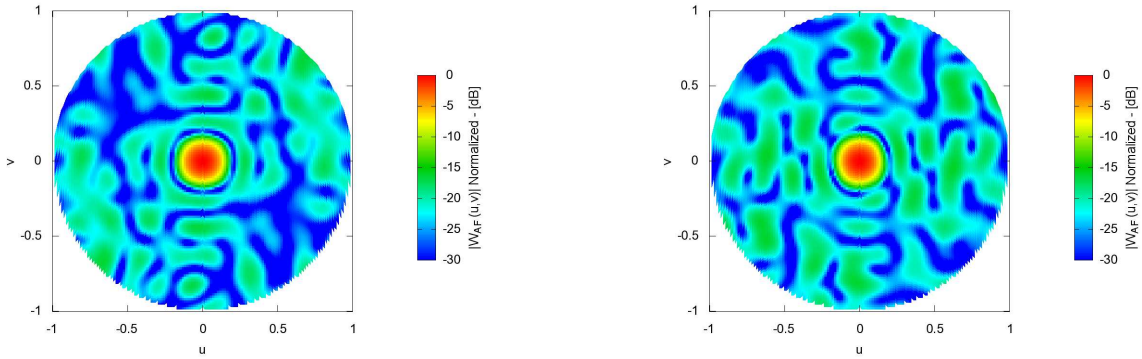
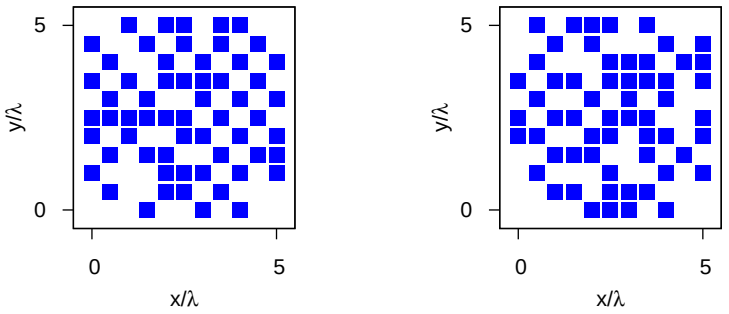


Figure 52. *Numerical validation - Problem I - PSL minimisation in array synthesis*: Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.



(a) *(b)*



(c) *(d)*

Figure 53. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.1.3 Array arrangement $P \times Q = 17 \times 17$

In this example we have used to initialize the population at step $i = 0$, the $(17 \times 17, 145, 72, 144)$-ADS ($N_{ADS} = 289$, $\nu_{ADS} = 0.501$). Fig 1 shows the behaviour of the optimal fitness value against the iteration number i in correspondence with the ADSGA and the standard GA minimization procedure.

The results can be summarized in the following Figures.

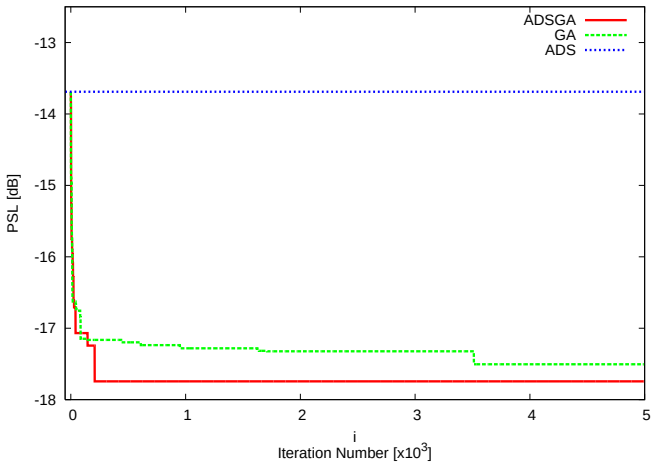
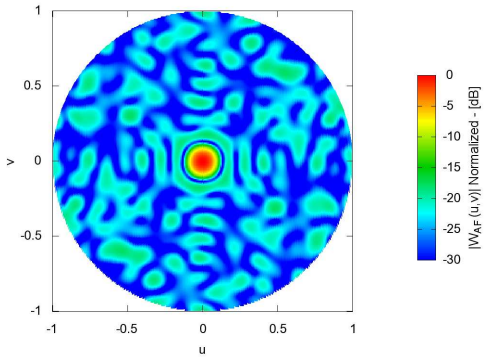
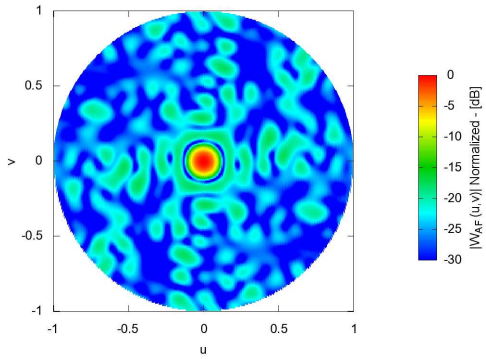


Figure 54. *Numerical validation - Problem I - PSL minimisation in array synthesis*: Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.



(a)



(b)

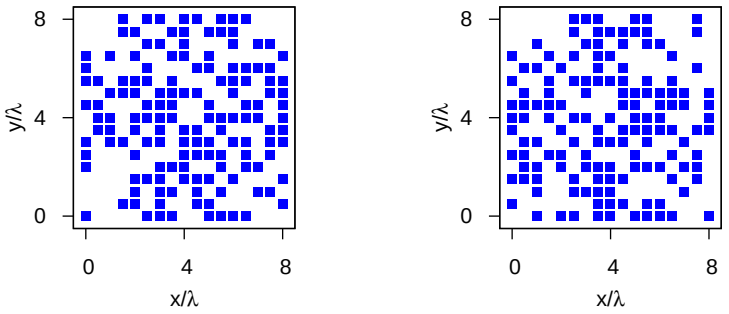


Figure 55. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.1.4 Array arrangement $P \times Q = 23 \times 23$

In this example we have used to initialize the population at step $i = 0$, the $(23 \times 23, 265, 132, 264)$-ADS ($N_{ADS} = 529$, $\nu_{ADS} = 0.500$). Fig 1 shows the behaviour of the optimal fitness value against the iteration number i in correspondence with the ADSGA and the standard GA minimization procedure.

The results can be summarized in the following Figures.

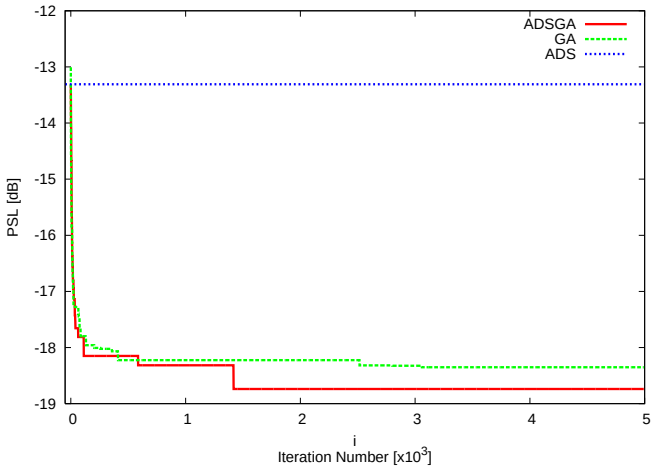
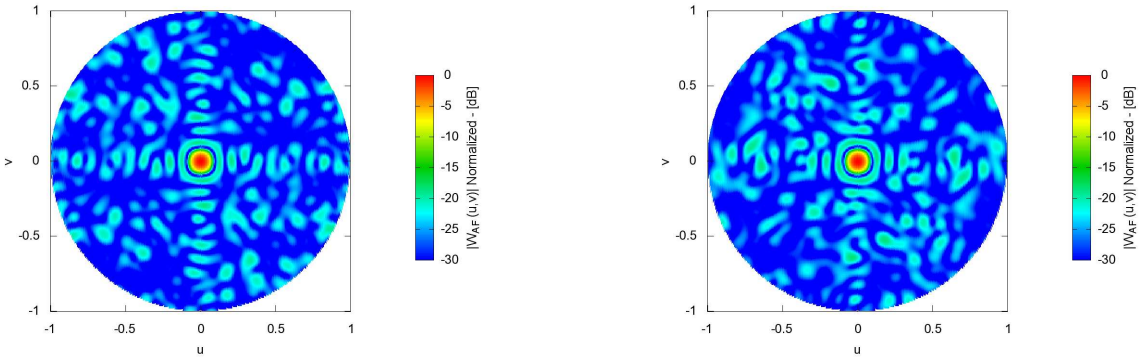


Figure 56. *Numerical validation - Problem I - PSL minimisation in array synthesis*: Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.



(a) *(b)*

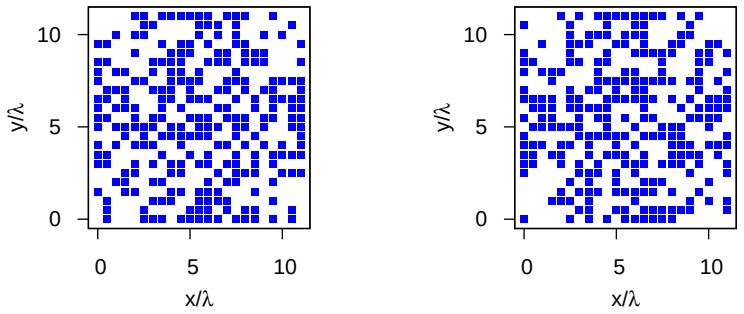


Figure 57. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.1.5 Summary

	I_{conv}		ν			PSL[dB]		
$P \times Q$	$ADSGA$	GA	$ADSGA$	GA	ADS	$ADSGA$	GA	ADS
7×7	1445	920	0.428	0.489	0.555	-16.13	-14.40	-9.69
11×11	4366	1125	0.496	0.487	0.502	-16.50	-16.03	-12.63
17×17	208	3512	0.480	0.494	0.501	-17.74	-17.50	-13.88
23×23	1418	2800	0.484	0.482	0.500	-18.74	-18.35	-13.90

Table X. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Summary of the results obtained. Comparing the results of the new proposed ADSGA technique with the standard GA methodology, we obtain a reduction of PSL that goes from 1.73[dB] to 0.24[dB].

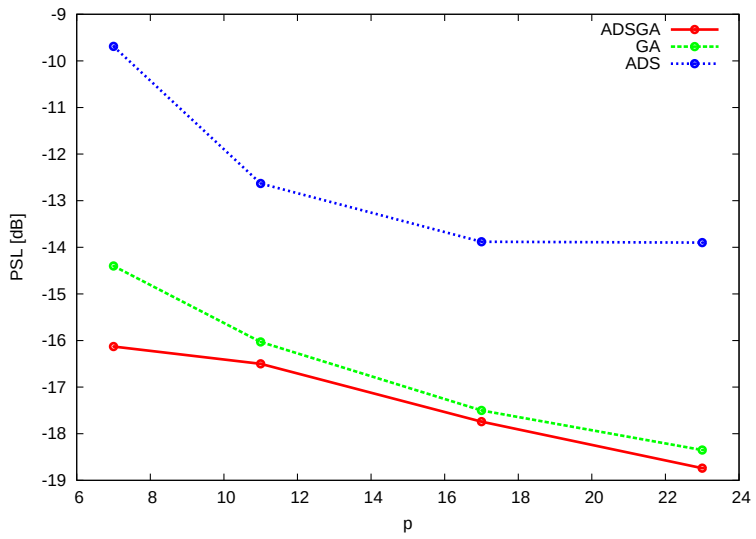


Figure 58. *Numerical validation - Problem I - PSL minimisation in array synthesis:* Graphical comparison of the PSL of different array configurations (the side P on the horizontal axis) for ADSGA an GA methodologies. We can observe that the PSL improvement of the ADSGA method reduces compared with standard GA as the dimension of the array increases.

6.3.2 Application to Problem II

In this section the aim, according to Problem II, is to design antenna arrays with arbitrary values of elements $N = P \times Q$ and thinning ν, still exploiting the existing (and limited) ADS arrangements. In other words, starting from an ADS-based array configuration (with $N_{ADS} = P_{ADS} \times Q_{ADS}$ elements, ν_{ADS}) used as initial guess of the optimization iterative procedure, we want to synthesize a new array configuration with $N \neq N_{ADS}$ and $\nu \neq \nu_{ADS}$.

For the sake of comparison of the performance of the proposed approach, the array configurations are chosen among the state-of-the-art examples, such as [25] and [18].

6.3.2.1 ADSGA method compared with [25]

In order to compare the results of the optimization procedure with **[25]**, we have the define the following problem:

Problem II: Minimise $F\{\rho\} = \max_{(u,v)\notin R_M} \left\{|W(u,v)|^2_{u=0} + |W(u,v)|^2_{v=0}\right\} / |W(0,0)|^2$, R_M the main lobe region as previously defined. The problem is subject to $K = \hat{K}$ and $N = \hat{N}$, being $\hat{K} \neq K_{ADS}$ and/or $\hat{N} \neq N_{ADS}$ (namely $\hat{P} \neq P_{ADS}$ and $\hat{Q} \neq Q_{ADS}$).

6.3.2.2 $P \times Q = 6 \times 6$ Array Configuration

Starting ADS	N_{ADS}	*Array Geometry* [18]	$\hat{N}$
$(5 \times 5, 13, 6, 12)$	25	6×6	36

the

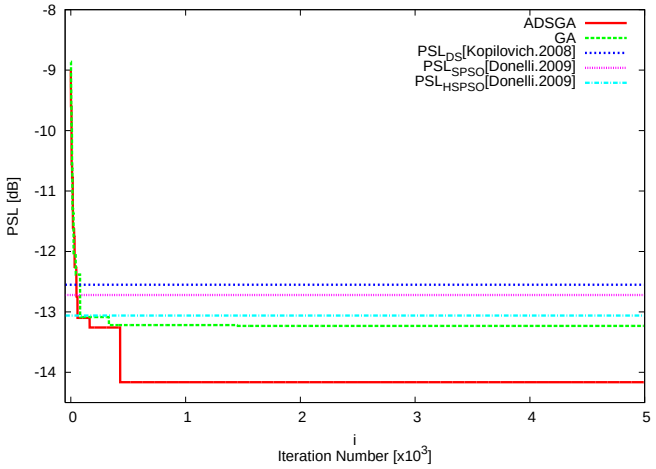
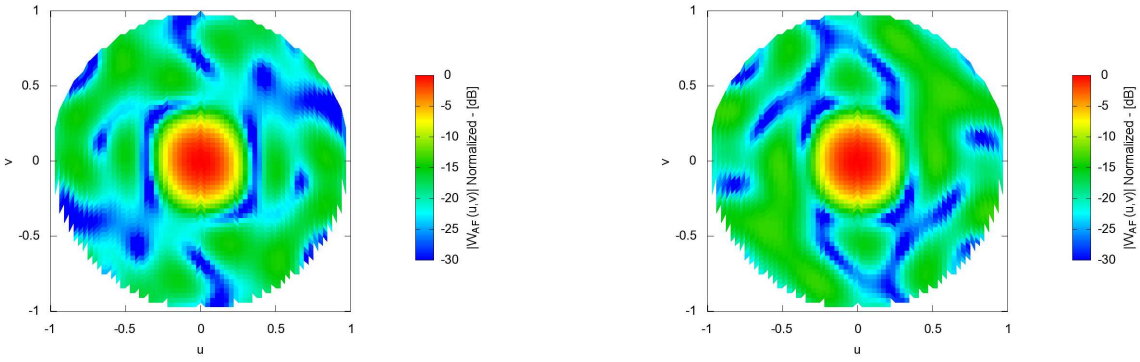
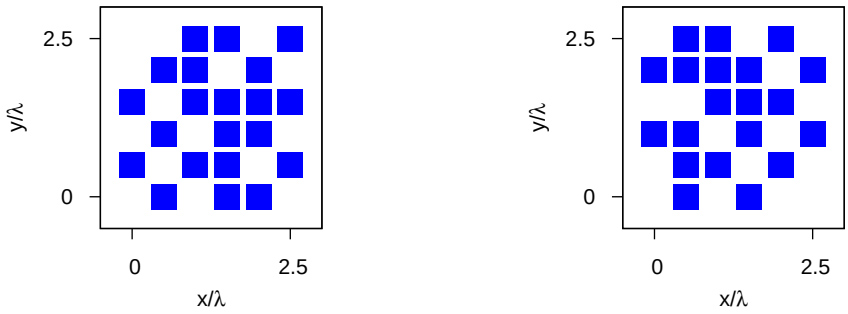


Figure 59: *Problem II- extension of the range of ADS applicability:* Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.



(a) (b)



(c) (d)

Figure 60: *Problem II- extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.2.3 $P \times Q = 8 \times 8$ Array Configuration

Starting ADS	N_{ADS}	*Array Geometry* [18]	$\hat{N}$
$(7 \times 7, 25, 12, 24)$	49	8×8	64

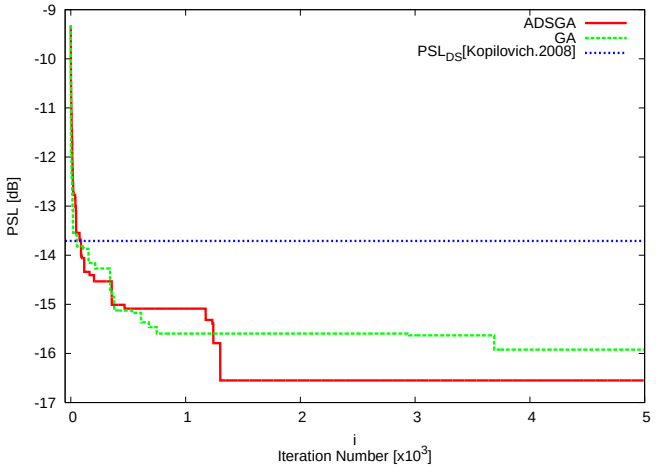
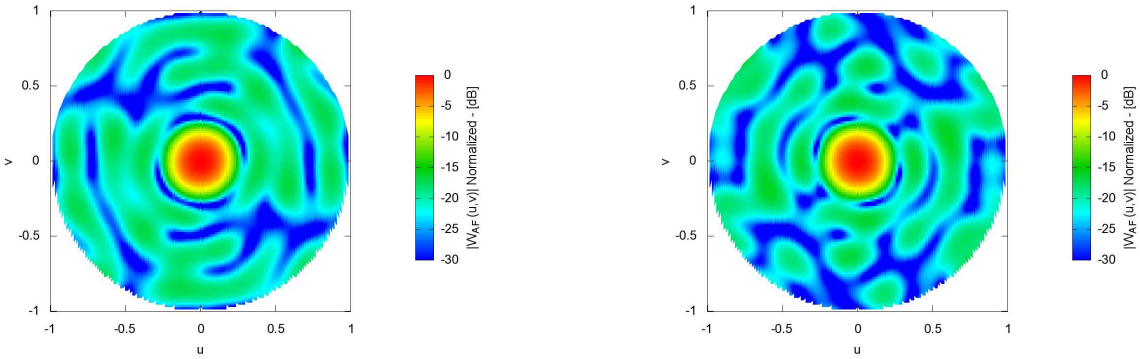
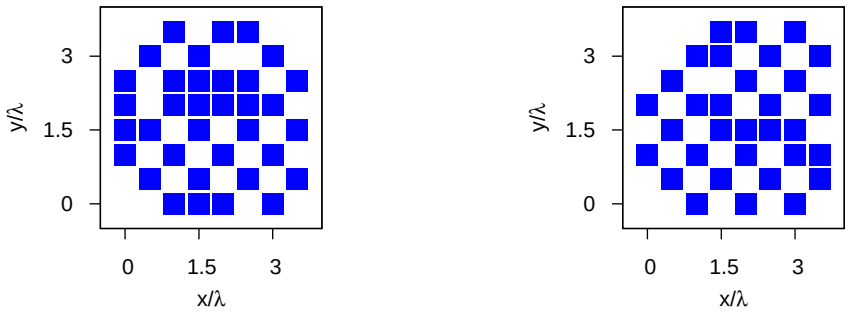


Figure 61: *Problem II- extension of the range of ADS applicability:* Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.



(a) (b)



(c) (d)

Figure 62: *Problem II- extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.2.4 $P \times Q = 12 \times 12$ Array Configuration

Starting ADS	N_{ADS}	*Array Geometry* [18]	$\hat{N}$
$(11 \times 11, 61, 30, 60)$	121	12×12	144

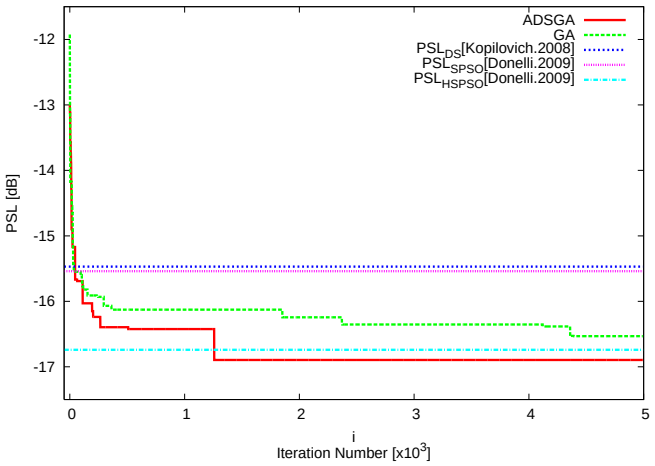
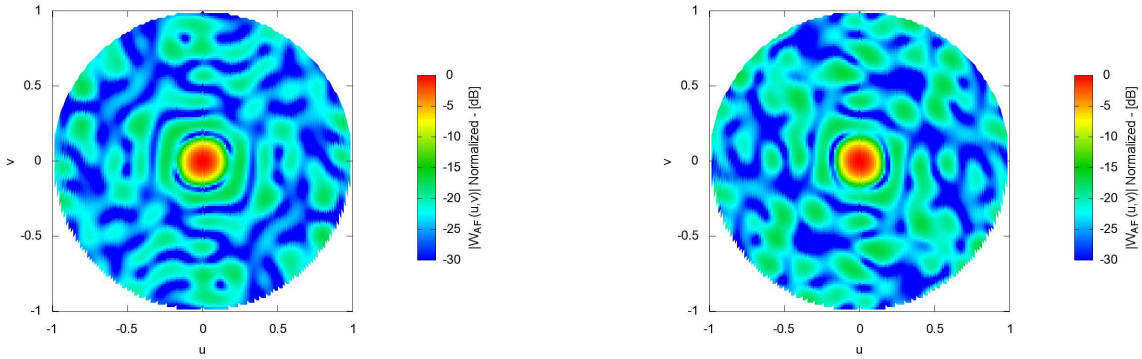
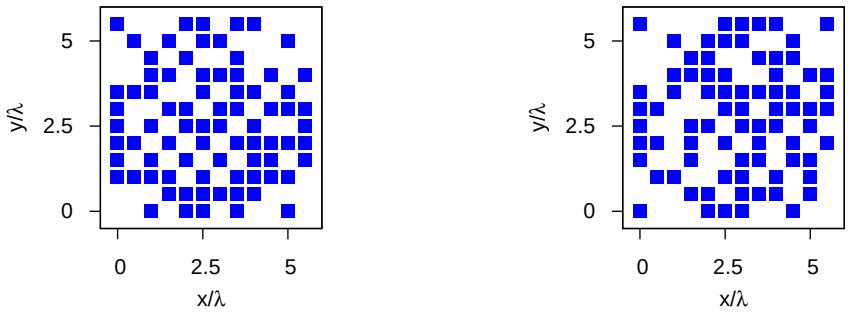


Figure 63: *Problem II- extension of the range of ADS applicability:* Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.



(a) (b)



(c) (d)

Figure 64: *Problem II- extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.2.5 $P \times Q = 16 \times 16$ Array Configuration

Starting ADS	N_{ADS}	*Array Geometry* [18]	$\hat{N}$
$(13 \times 13, 85, 42, 84)$	169	16×16	256

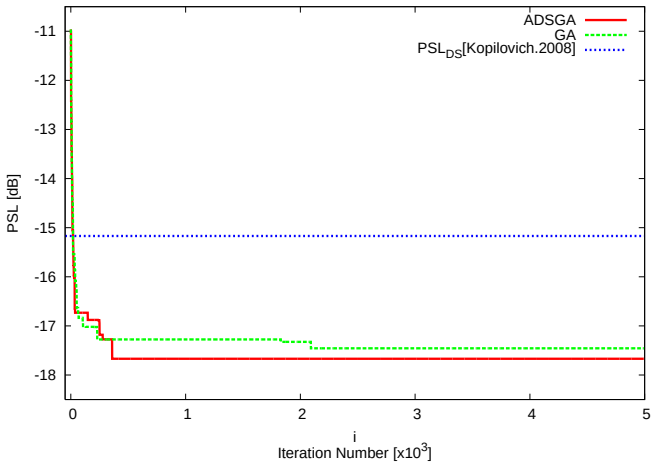
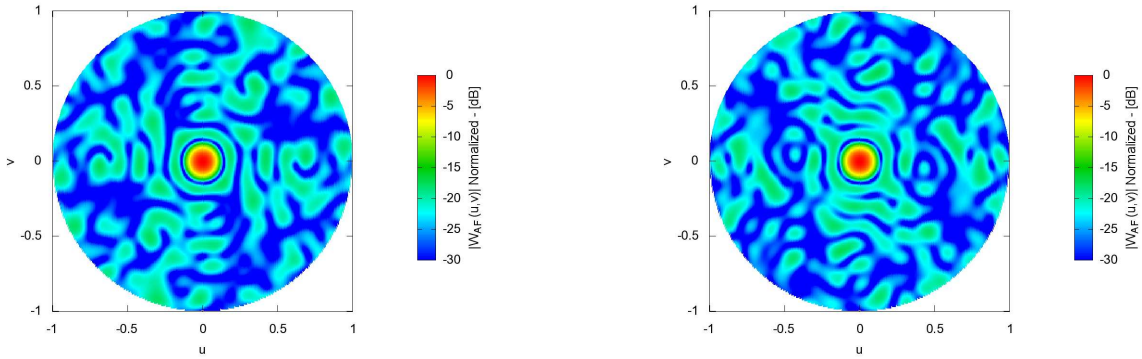


Figure 65: *Problem II- extension of the range of ADS applicability:* Behaviour of the optimal fitness value, $PSL(i)$, against the number of iteration number, i.



(a) (b)

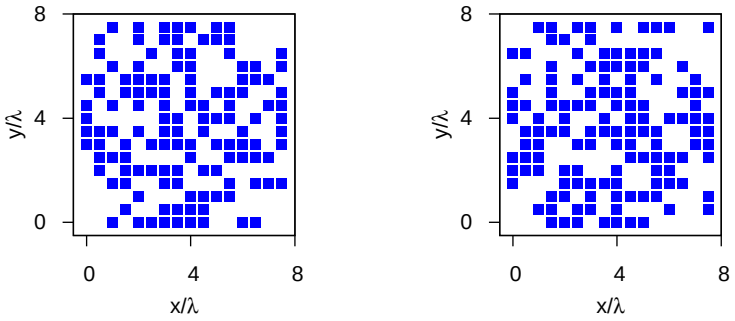


Figure 66: *Problem II- extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.2.6 Summary

$Array - Dimesion$	ν				
$\hat{P} \times \hat{Q}$	$ADSGA$	GA	$SPSO$ [25]	$HSPSO$ [25]	DS [21]
6×6	0.583	0.555	0.50	0.42	0.583
8×8	0.546	0.500	–	–	0.562
12×12	0.541	0.534	0.44	0.48	0.542
16×16	0.500	0.515	–	–	0.531

Table XI: *Problem I- PSL minimisation in array synthesis:* Summary of the results obtained. Comparing the results of the new proposed ADSGA technique with the standard GA methodology, the SPSO, the HSPSO [25] and DS [21], we obtain that ADSGA is able to improve PSL performance also when $\hat{N} \neq N_{ADS}$.

$Array - Dimesion$	PSL[dB]				
$\hat{P} \times \hat{Q}$	$ADSGA$	GA	$SPSO$ [25]	$HSPSO$ [25]	DS [21]
6×6	-14.16	-13.23	-12.72	-13.06	-12.55
8×8	-16.55	-15.92	–	–	-13.71
12×12	-16.90	-16.53	-15.54	-16.74	-15.47
16×16	-17.45	-17.67	–	–	-15.17

Table XII: *Problem I- PSL minimisation in array synthesis:* Summary of the results obtained. Comparing the results of the new proposed ADSGA technique with the standard GA methodology, the SPSO, the HSPSO [25] and DS [21], we obtain that ADSGA is able to improve PSL performance also when $\hat{N} \neq N_{ADS}$.

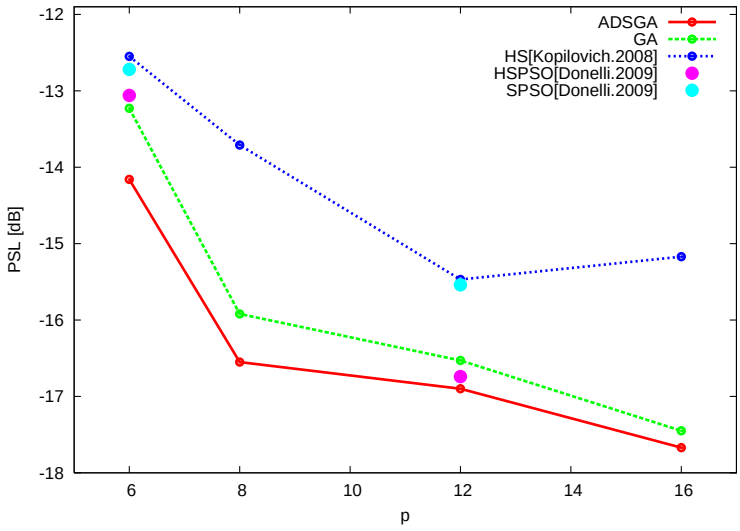


Figure 67: *Problem I- PSL minimisation in array synthesis:* Graphical comparison of the PSL of different array configurations (the side P on the horizontal axis) for ADSGA an GA methodologies. We can observe that the PSL improvement of the ADSGA method reduces compared with standard GA as the dimension of the array increases.

6.3.2.7 ADSGA method compared with [18]

In order to compare the results of the optimization procedure with [18], we have the define the following problem:

Problem II: Minimise $F\{\rho\} = \max_{(u,v)\notin R_M} \left\{|W(u,v)|^2_{u=0} + |W(u,v)|^2_{v=0}\right\} / |W(0,0)|^2$, R_M the main lobe region as defined in [18]. The problem is subject to $K = \hat{K}$ and $N = \hat{N}$, being $\hat{K} \neq K_{ADS}$ and/or $\hat{N} \neq N_{ADS}$ (namely $\hat{P} \neq P_{ADS}$ and $\hat{Q} \neq Q_{ADS}$).

The two examples that are considered are the followings

Starting ADS	N_{ADS}	*Array Geometry* [18]	$\hat{N}$
$(7 \times 11, 37, 17, 36)$	77	10×20	200
$(37 \times 37, 685, 342, 684)$	1369	40×40	1600

6.3.2.8 $P \times Q = 10 \times 20$ Array Configuration

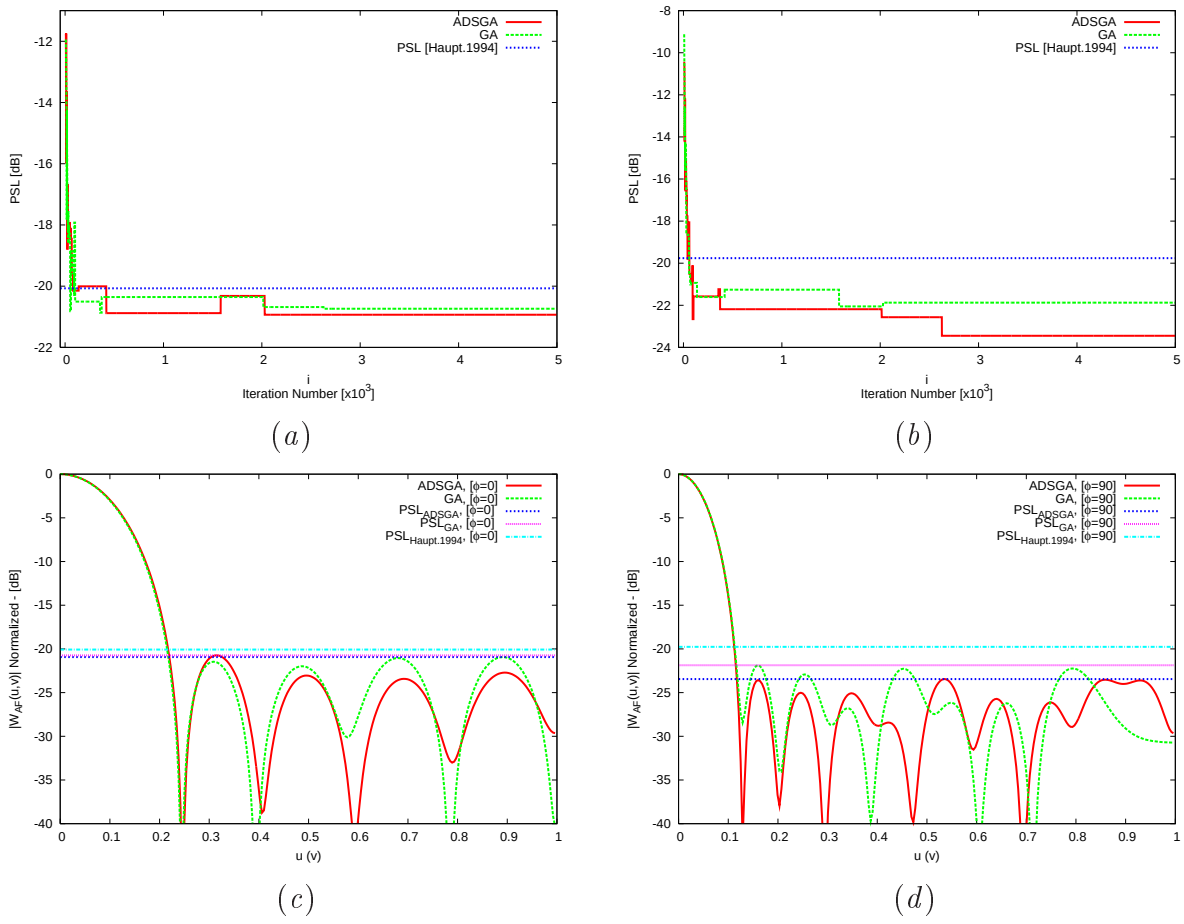


Figure 68: *Problem II- extension of the range of ADS applicability:* Graphical comparison of the PSL against the iteration i of ADSGA, GA and Haupt [18] approaches along the two main directions $\phi = 0^\circ$ (a) and $\phi = 90^\circ$ (b). Slices of the amplitude pattern obtained after optimization procedure along the two main directions $\phi = 0^\circ$ (c) and $\phi = 90^\circ$ (d).

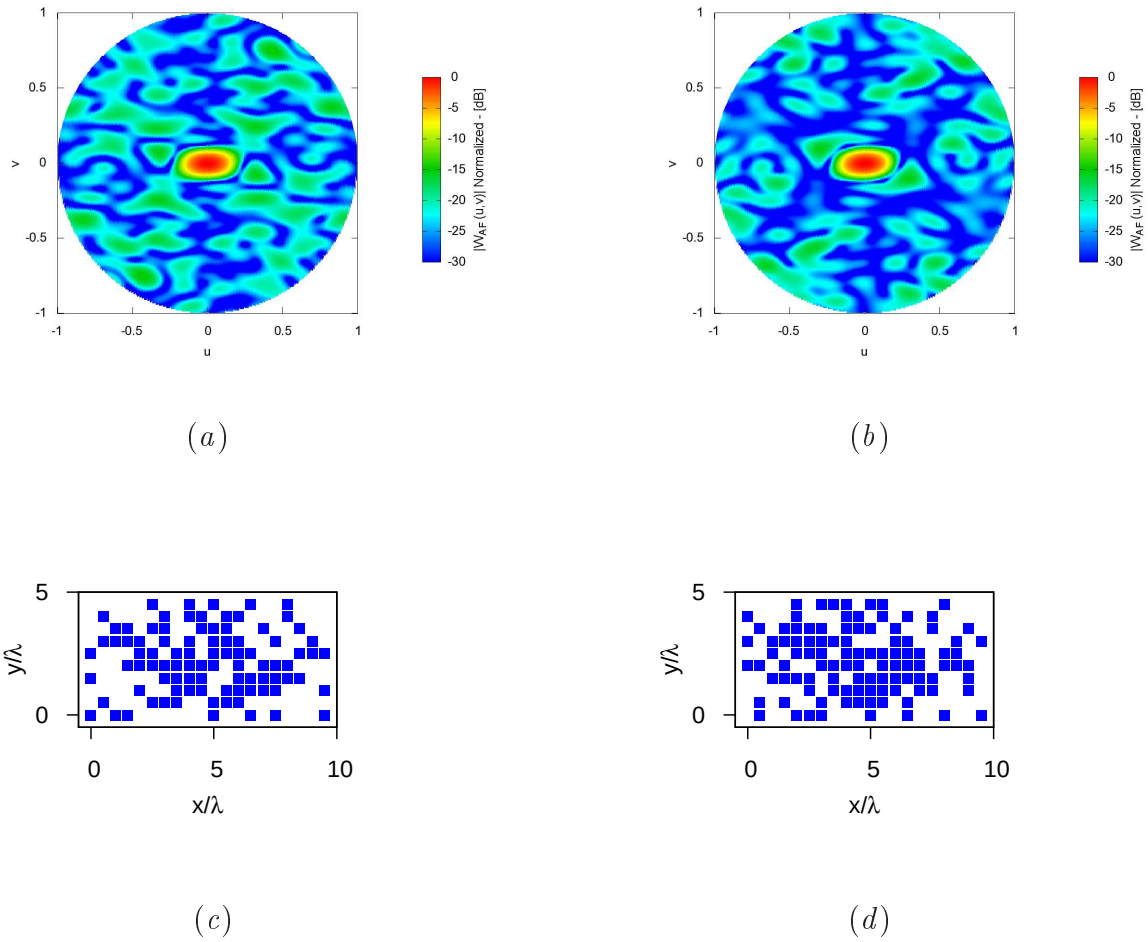


Figure 69: *Problem II- extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.2.9 $P \times Q = 40 \times 40$ Array Configuration

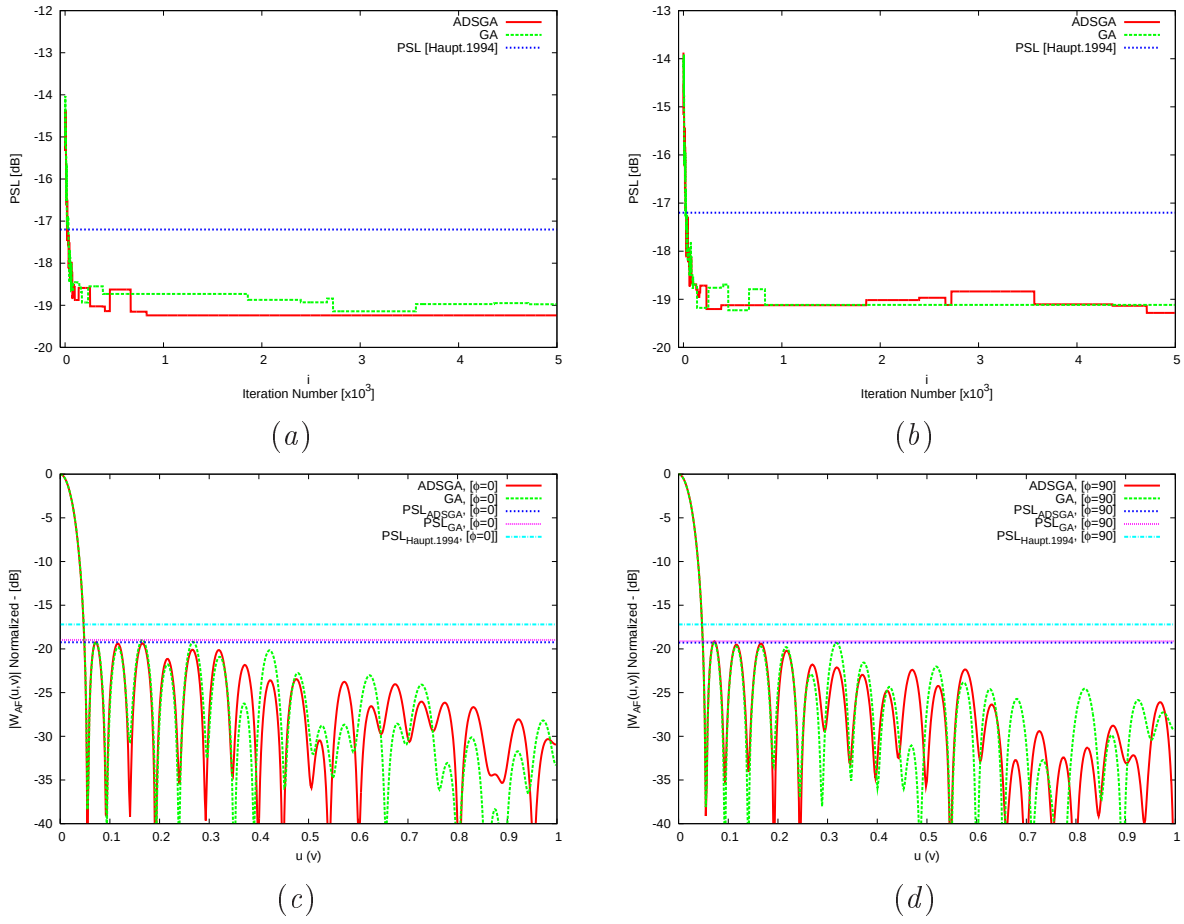


Figure 70: *Problem II- extension of the range of ADS applicability:* Graphical comparison of the PSL against the iteration i of ADSGA, GA and Haupt [18] approaches along the two main directions $\phi = 0^\circ$ (a) and $\phi = 90^\circ$ (b). Slices of the amplitude pattern obtained after optimization procedure along the two main directions $\phi = 0^\circ$ (c) and $\phi = 90^\circ$ (d).

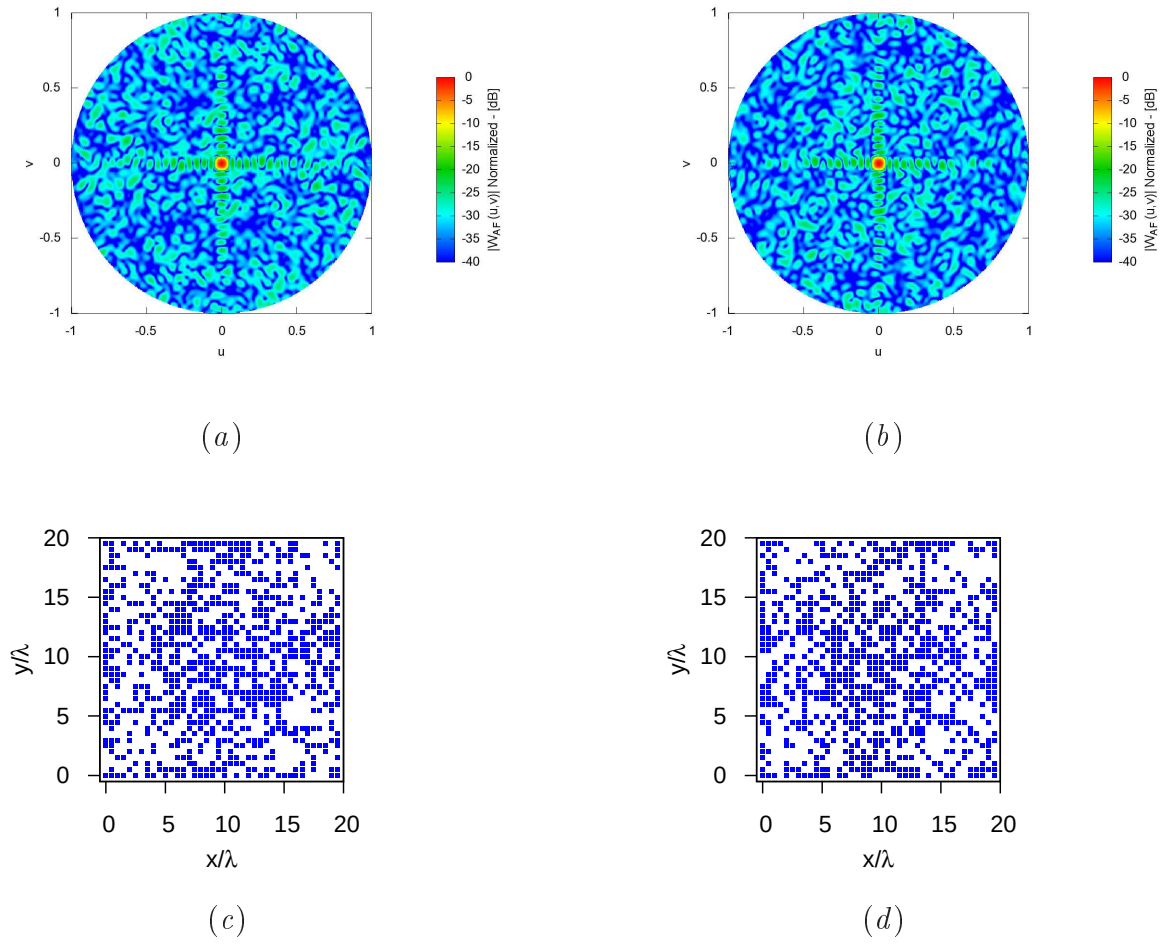


Figure 71: *Problem II- extension of the range of ADS applicability:* Power patterns $|W(u,v)|^2$ for ADSGA (a) and for GA (b) approaches. (c) and (d) show the corresponding array arrangements with ADSGA and GA-based methods, respectively.

6.3.2.10 Summary

		ν		
P	Q	$ADSGA$	GA	$GA-[Haupt]$ [18]
10	20	0.455	0.515	0.54
40	40	0.485	0.491	0.81

Table XIII: *Problem II- extension of the range of ADS applicability:* Summary of the results obtained about thinning factor ν. Comparing the results of the new proposed ADSGA technique with the standard GA methodology and [18].

		$BW_{\phi=0}$			$BW_{\phi=90}$		
P	Q	$ADSGA$	GA	$GA-[Haupt]$ [18]	$ADSGA$	GA	$GA-[Haupt]$ [18]
10	20	0.2412	0.2460	0.2480	0.1289	0.1289	0.1289
40	40	0.0546	0.0546	0.0546	0.0546	0.0546	0.0546

Table XIV: *Problem II- extension of the range of ADS applicability:* Summary of the results obtained about main lobe dimension BW. Comparing the results of the new proposed ADSGA technique with the standard GA methodology and [18].

		$PSL_{\phi=0}[dB]$			$PSL_{\phi=90}[dB]$		
P	Q	$ADSGA$	GA	$GA-[Haupt]$ [18]	$ADSGA$	GA	$GA-[Haupt]$ [18]
10	20	-20.93	-20.74	-20.07	-23.45	-21.87	-19.76
40	40	-19.24	-18.97	-17.20	-19.28	-19.12	-17.20

Table XV: *Problem II- extension of the range of ADS applicability:* Summary of the results obtained. Comparing the results of the new proposed ADSGA technique with the standard GA methodology and [18]. We obtain with ADSGA a reduction of PSL in both examples.

6.3.3 Application to Problem III

As stated in the Introduction and in the description of the Problem III, several construction techniques to obtain ADS sequences have been already developed and even large repositories are now [61][65] available. However, the fact that the ADS sequences of arbitrary length are, at present, not available is a limitation for their use in real-world problems. As a matter of fact, since ADS synthesis techniques are usually based on the cyclotomy property, they generate sequences characterized by specific cyclotomic numbers and not with arbitrary length [47].

As proposed in the description of Problem III, here a new method is proposed for the synthesis of sequences of arbitrary length. The approach reformulates the ADS design in terms of a combinatorial optimization problem where the cost function quantifies the misfit between the autocorrelation of a binary sequence and the three valued function of the DSs. The binary genetic algorithm (GA) is used to minimize such a cost function because of its "hill-climbing" features and its ability to effectively sample the binary solution space [47]. The parameters of the cost function have been set $\alpha = 10^{-2}$ and $\beta = 10^{-4}$. The number of iterations I_{max} depends on how large is the search space.

6.3.3.1 $(36, 32, 28, 23)$-ADS

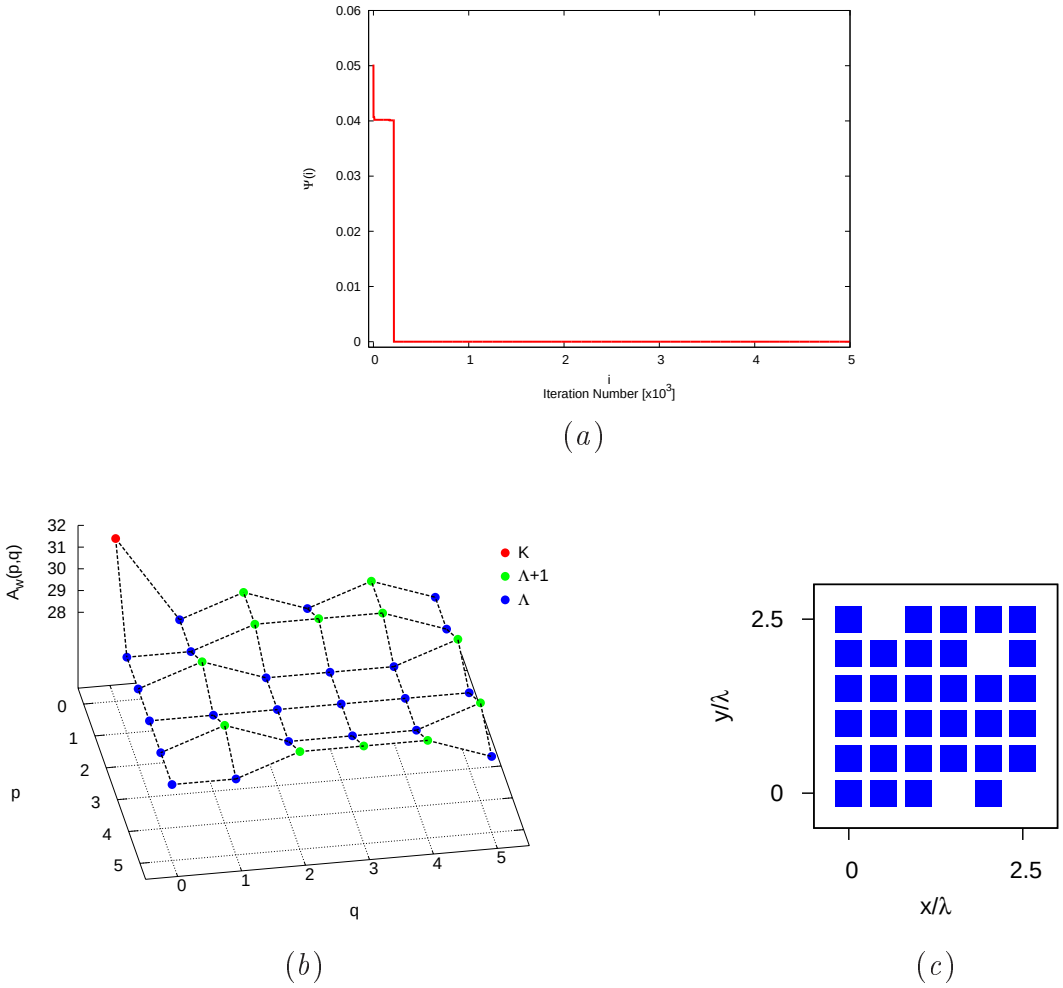


Figure 72. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(36, 32, 28, 23)$-ADS arrangement, (c) Final 2D ADS layout.

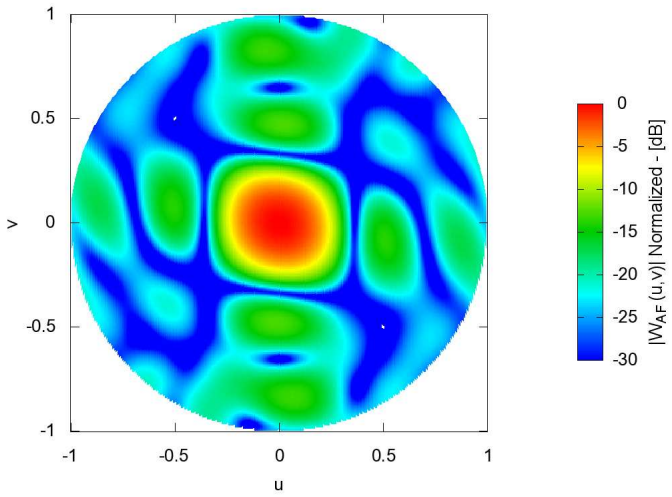


Figure 73. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(36, 32, 28, 23)$-ADS arrangement.

6.3.3.2 $(60, 6, 0, 29)$-ADS

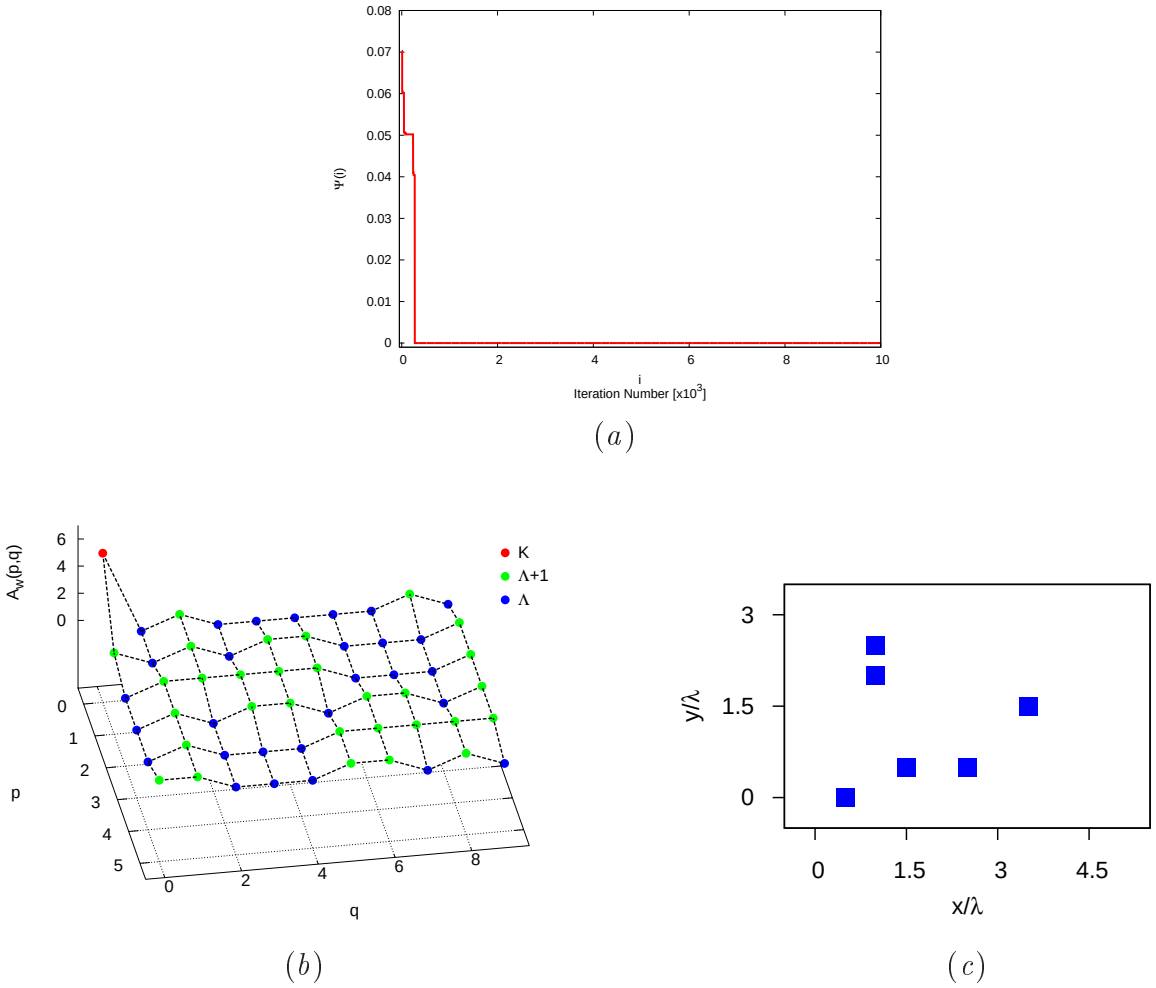


Figure 74. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(60, 6, 0, 29)$-ADS arrangement, (c) Final 2D ADS layout.

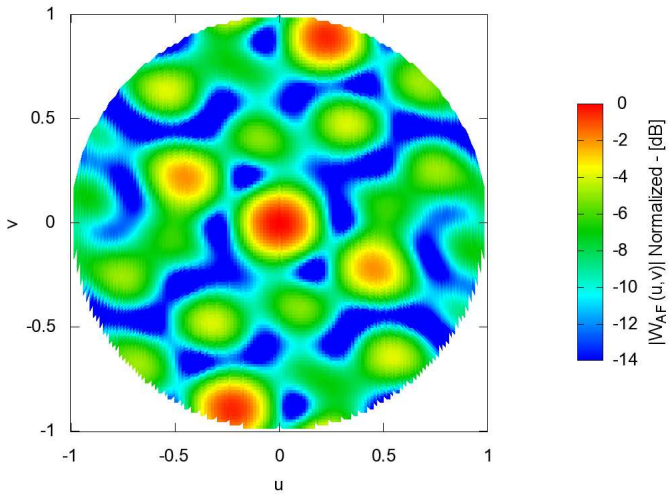


Figure 75. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(60, 6, 0, 29)$-ADS arrangement.

6.3.3.3 $(64, 59, 54, 43)$-ADS

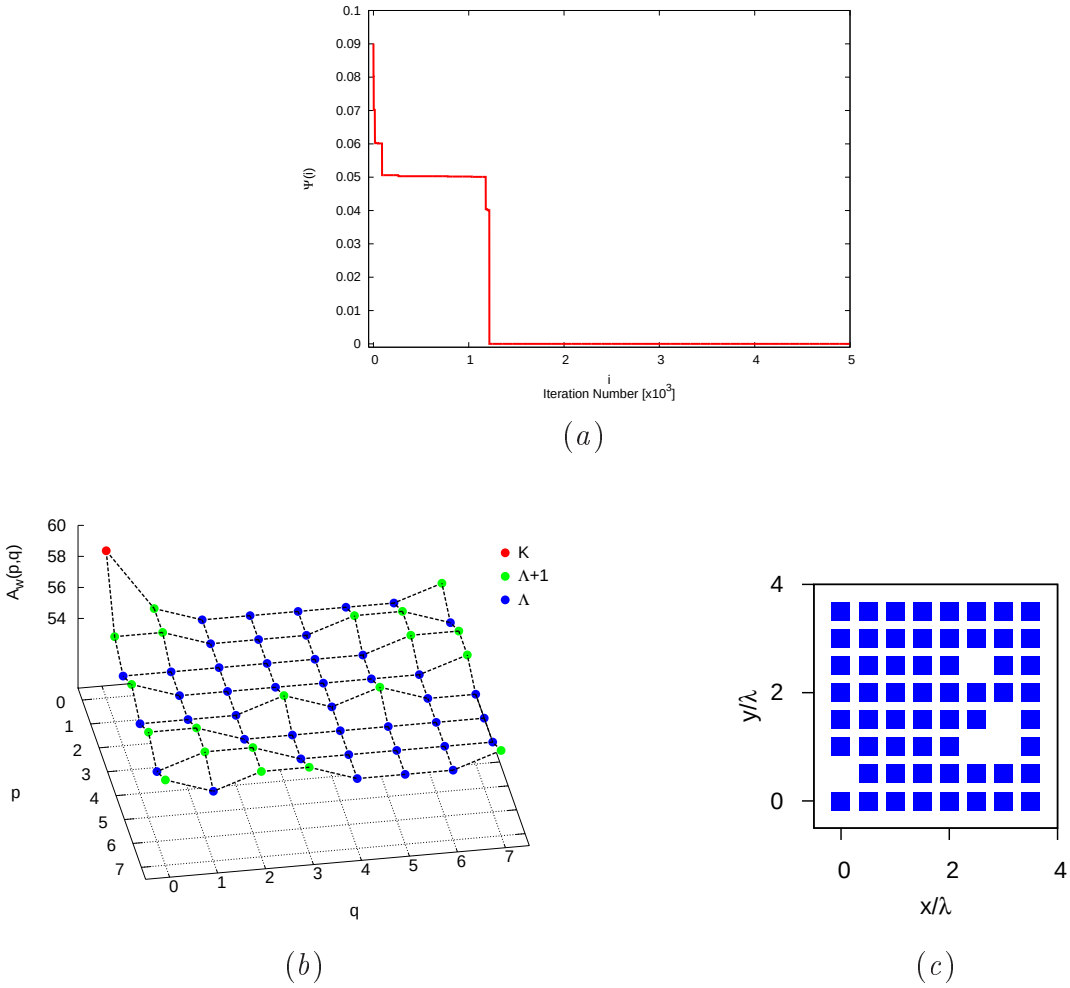


Figure 76. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(64, 59, 54, 43)$-ADS arrangement, (c) Final 2D ADS layout.

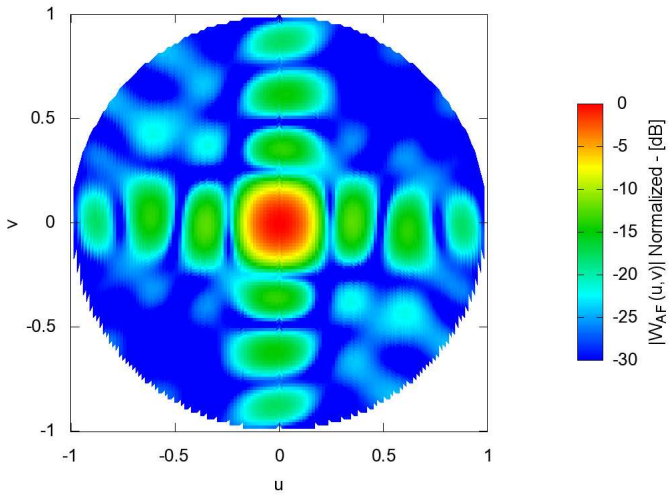


Figure 77. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(64, 59, 54, 43)$-ADS arrangement.

6.3.3.4 $(100, 5, 0, 79)$-ADS

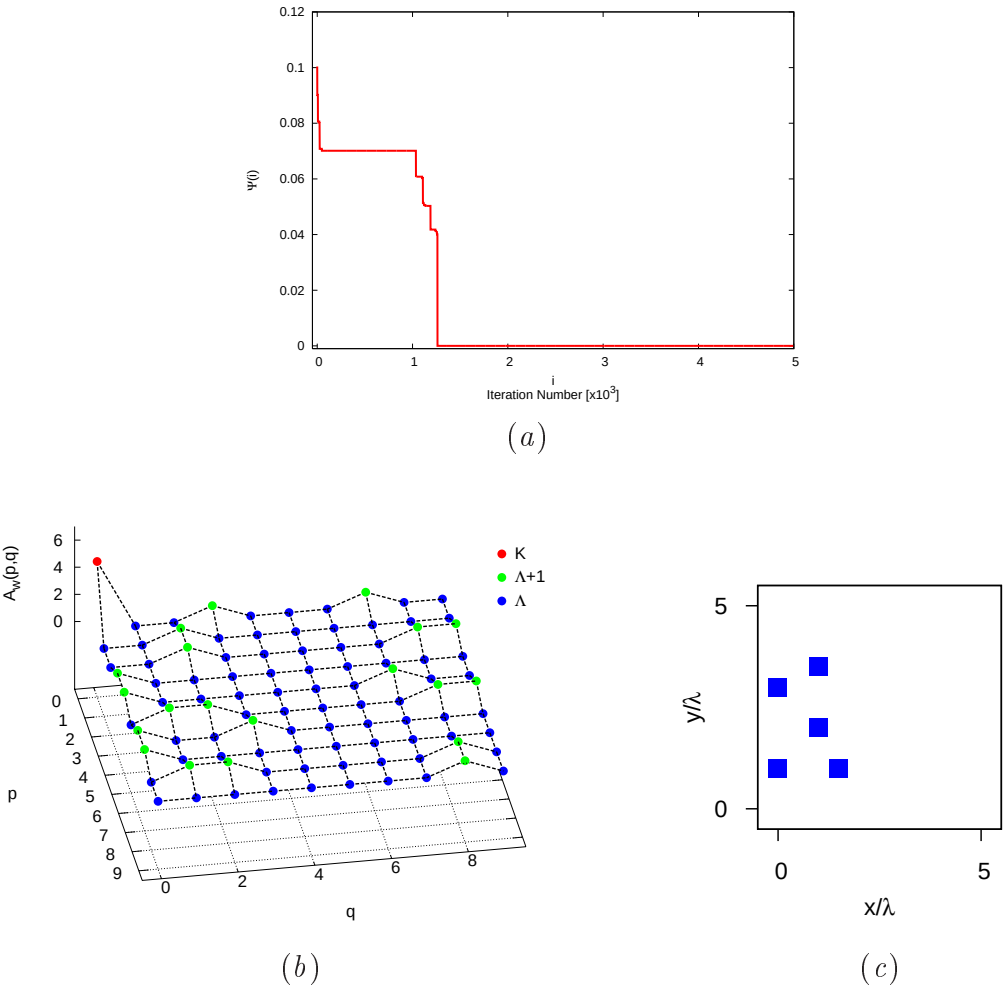


Figure 78. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(100, 5, 0, 79)$-ADS arrangement, (c) Final 2D ADS layout.

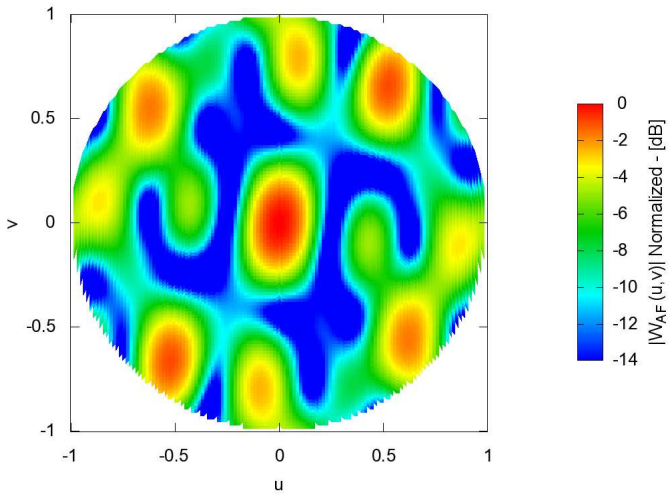


Figure 79. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(100, 5, 0, 79)$-ADS arrangement.

6.3.3.5 $(144, 137, 130, 101)$-ADS

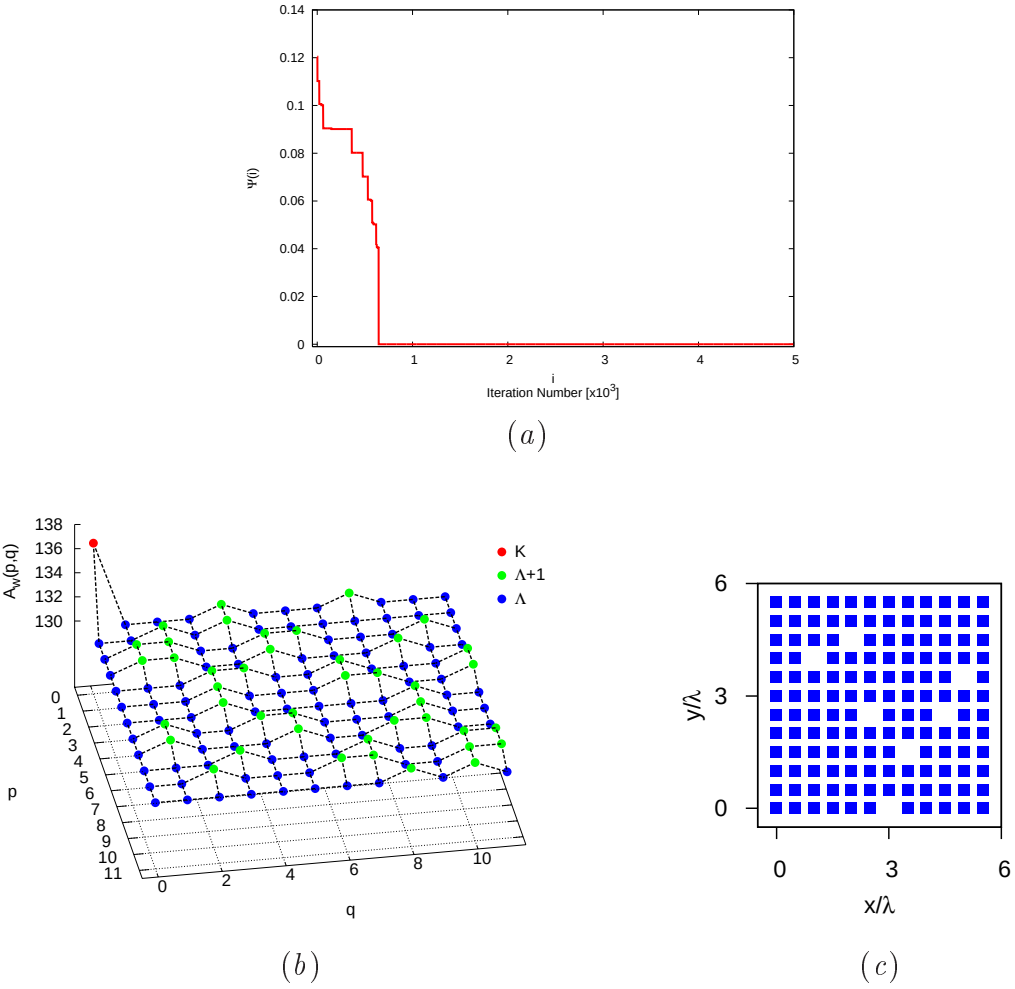


Figure 80. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(144, 137, 130, 101)$-ADS arrangement, (c) Final 2D ADS layout.

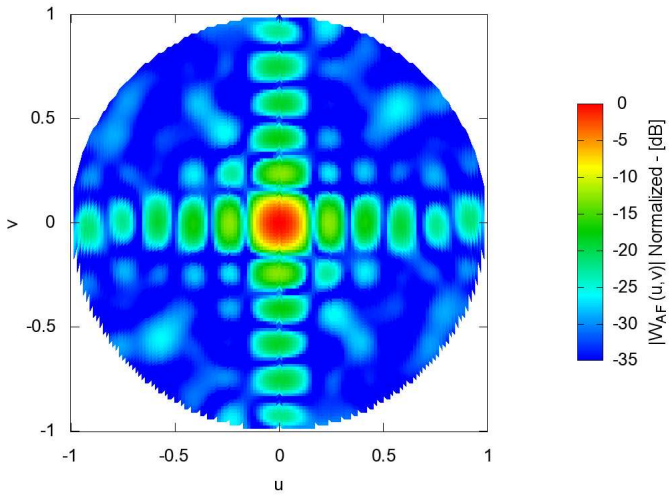


Figure 81. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(144, 137, 130, 101)$-ADS arrangement.

6.3.3.6 $(192, 184, 176, 135)$-ADS

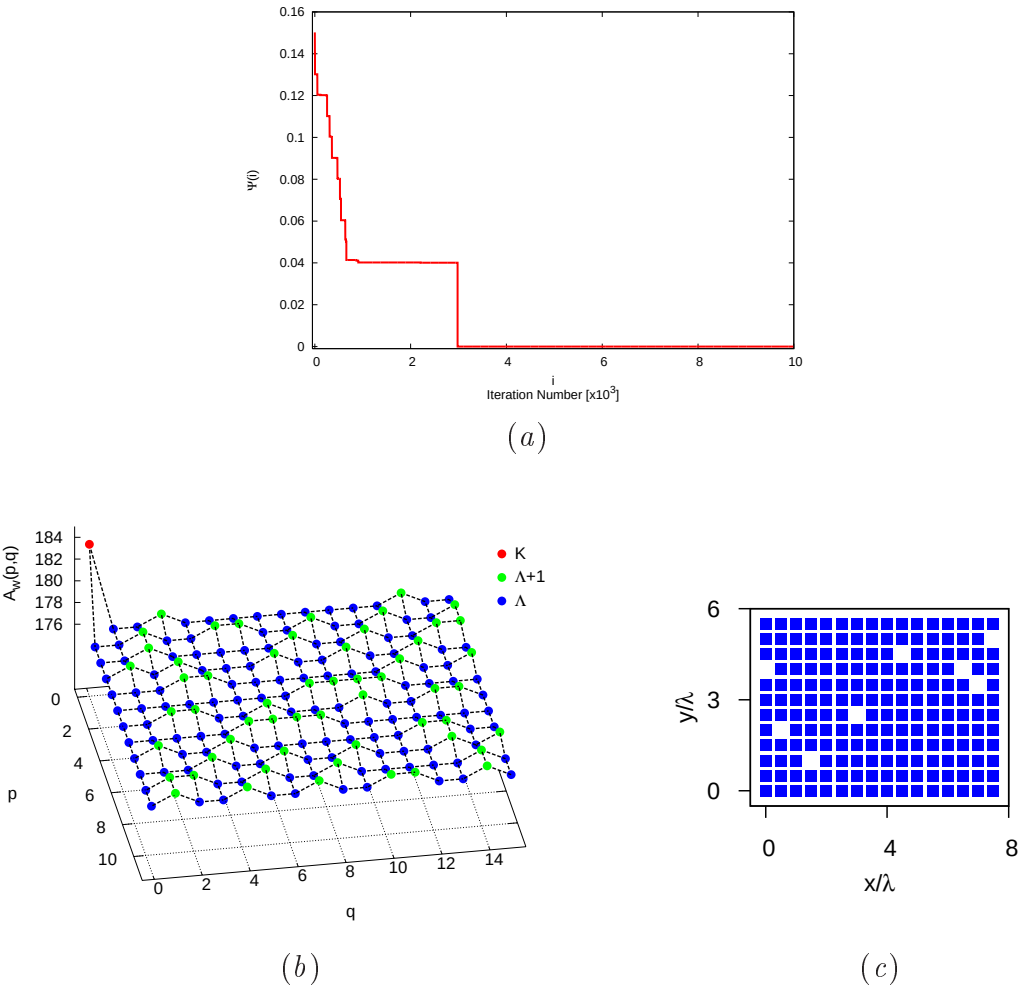


Figure 82. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(192, 184, 176, 135)$-ADS arrangement, (c) Final 2D ADS layout.

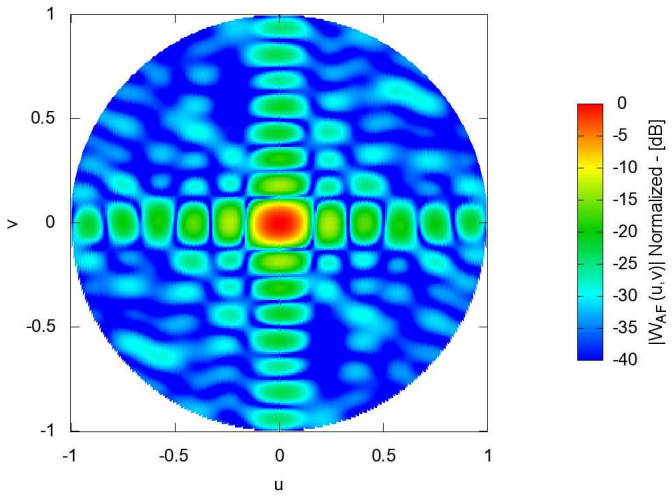
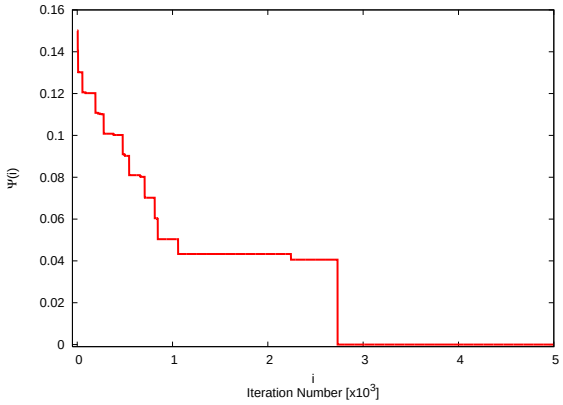
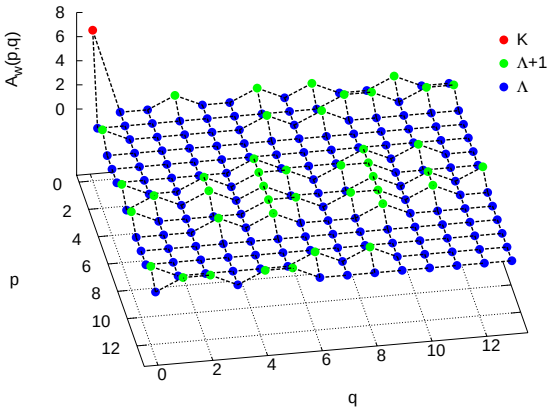


Figure 83. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(192, 184, 176, 135)$-ADS arrangement.

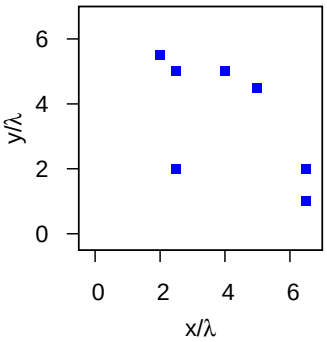
6.3.3.7 $(196, 7, 0, 153)$-ADS



(a)



(b)



(c)

Figure 84. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(196, 7, 0, 153)$-ADS arrangement, (c) Final 2D ADS layout.

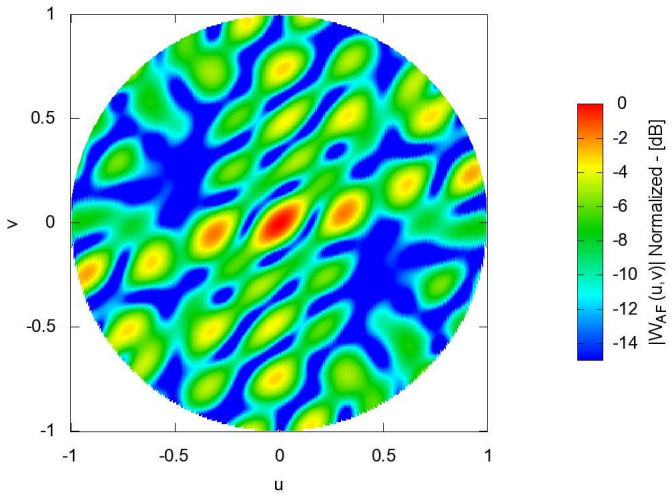
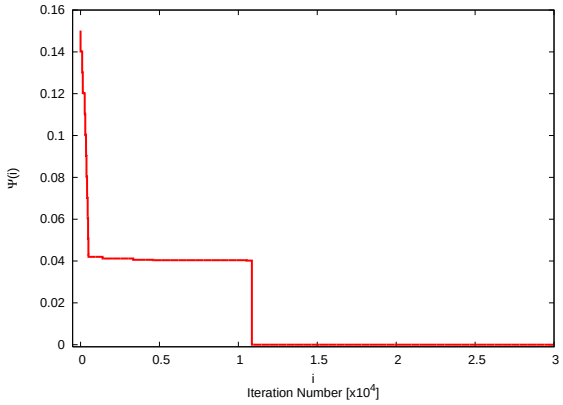
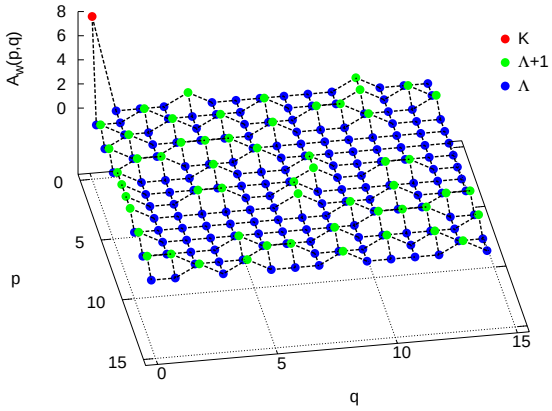


Figure 85. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(196, 7, 0, 153)$-ADS arrangement.

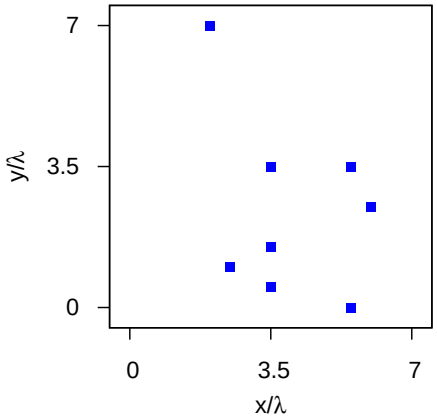
6.3.3.8 $(225, 8, 0, 168)$-ADS



(a)



(b)



(c)

Figure 86. *Numerical validation - Problem III - GA designed ADS construction technique*: (a) Behaviour of the optimal fitness, F_{POP}, against the iteration number i, (b) Three-level autocorrelation function of the convergence $(225, 8, 0, 168)$-ADS arrangement, (c) Final 2D ADS layout.

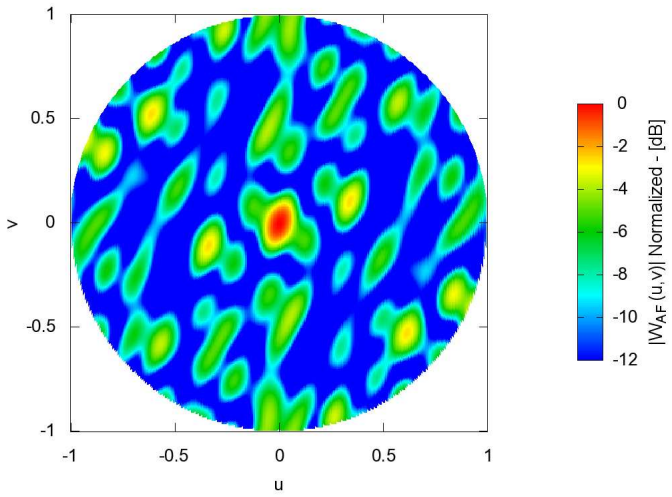


Figure 87. *Numerical validation - Problem III - GA designed ADS construction technique*: Plot of the power pattern associated to the antenna array built with the $(225, 8, 0, 168)$-ADS arrangement.

6.3.3.9 Summary

A GA-based technique has been proposed as a new methodological tool for designing 2D ADS sequences of arbitrary length. As put in evidence in the Introduction Section 6.1, although large repositories of ADSs are available, ADS arrays with arbitrary aperture sizes and thinning factors cannot be designed, since ADS sequences exist only for specific sets of descriptive parameters. Moreover, even for admissible aperture sizes and thinning factors, general purpose ADS construction techniques do not exist at present and the explicit forms of ADS sequences has to be determined on a case by case basis using suitable construction theorems.

To overcome this problem, the original synthesis has been reformulated as a combinatorial optimization. Towards this end, a suitable fitness function exploiting the autocorrelation properties of ADSs has been introduced and minimized by means of a GA-based iterative procedure. In other words, the aim is now to find the explicit forms of ADSs sequences for arbitrary values of N. In such a case, the optimisation at hand turns out to be different from that in Problem I and Problem II. The GA works within the "autocorrelation space", while the constraints are still on the set of parameters defining the ADS as well as the corresponding array arrangement.

In the following Table, the (N, K, Λ, t)-ADS sequences that have been found by means of GA procedure are described.

N	P	Q	K	Λ	t	ν
36	6	6	32	28	23	0.888
60	6	10	6	0	29	0.10
64	8	8	59	54	43	0.921
100	10	10	5	0	79	0.50
144	12	12	137	130	101	0.951
192	12	16	184	176	135	0.958
196	14	14	7	0	153	0.35
225	15	15	8	0	168	0.35

Table XVI: Properties of the ADS sequences that have been designed by the proposed GA-based techniques. Neither of these (N, K, Λ, t)-ADS sequences can found in [61] or [65].

All the sequences in Table XVI are not described by the available theorems and this shows that the proposed ADS-synthesis technique correctly works. It is mandatory to put in evidence that, as expected, the GA-based ADS synthesis technique requires much

more iterations to determine the three-level autocorrelation binary sequence for a given geometry for larger search spaces. Anyway the proposed method assessed its reliability whatever the dimension athand.

As a final observation, it is worthwhile to point out that the new ADSs determined solving different instances of Problem III can be directly used to define new thinned arrays or as starting points for different formulations of Problem I or Problem II. Indeed, the power patterns of different new ADS-based arrays have been plotted.

Chapter 7

Conclusion

As described in the Abstract, this Thesis has presented innovative guidelines for the synthesis of antenna arrays for communication and radioastronomy systems and applications. In more detail in the first part of the Thesis a new family of analytically-designed thinned arrays with different azimuth and elevation TMBWs has been proposed. Thanks to the properties of McFarland DSs, several massively thinned isophoric architectures have been deduced and the PSLs of the arising layouts, defined over grids of size $P \times P(P+2)$ (P being a prime number), have been numerically analyzed. Towards this end, a GA-based search procedure has been exploited due to the extremely large number of admissible McFarland sequences.

The numerical results point out the following issues

- the design of McFarland arrays is highly efficient whatever P, since up to $\Psi(P)$ layouts can be obtained by simply selecting the associated descriptors ,σ_x, σ_y, k, (a_i, b_i) and $\left(\hat{w}_1^{(i+1)}, \hat{w}_2^{(i+1)}\right)$ for $i = 0, ..., P+1$;

- unlike traditional binary encodings used for thinned array designs [44], the GA-based procedure is able to more efficiently identify optimal McFarland layouts thanks to the discrete nature of the McFarland descriptors and also the large number of optimal solutions available within the search space (Fig. 3);

- despite the extremely low number of active elements ($\nu < (1/P)$), McFarland arrays exhibit well-controlled sidelobes especially for large dimensions. This suggests their exploitation for the design of extremely light large arrays as well as of architectures with interleaved functionalities (e.g., multi-function radar arrays in which each function correspond to a highly sparse sub-array [41]).

Further studies will be devoted to analyze the effects of the presence of real array elements and/or mutual coupling. Furthermore, it is still a work in progress the exploitation of McFarland sequences for designing interleaved architectures.

In the second part of this Thesis ADS sequences have been exploited to design correlator arrays for radio astronomy applications in a computationally efficient and reliable fashion. Three strategies have been presented that exhibit different features, computational complexity and flexibility. More specifically: (a) a fully analytic technique based on ADS layouts to provide sub-optimal designs with extremely reduced computational costs; (b) an ADSGA hybrid technique that employs a binary description of the correlator array to obtain optimized configurations with interesting geometric properties and improved PSL performances; and (c) an ADSPSO strategy devoted to enhance the flexibility of the lattice-based approaches and exploiting a real-coded description of the geometry at hand. An extensive numerical validation has been carried out to analyze features and advantages of the proposed approaches, also in comparison with state-of-the-art methodologies, in several working conditions, including design examples for future planned instruments (i.e., the ALMA architecture [57]).

The obtained results have pointed out the following key issues:

- ADS-based analytic layouts outperform equally spaced or power-law state-of-the-art designs in terms of PSL control and snapshot or tracking coverage (e.g., $\nu_{ADS}^{un} = 0.831$ versus $\nu_{pl} = 0.598$ - Table II);

- the analytic ADS technique synthesizes arrays with sub-optimal performances if compared to state-of-the-art stochastically optimized arrangements (e.g., $PSL_3 = -20.3$[dB] versus $PSL_{ADS}^{un} = -19.98$[dB] - Table II), but it is extremely efficient in terms of computational costs and the generation of reliable compromise solutions (versus $\Xi_1 = 0.22$ vs. $\Xi_{ADS}^{un} = 0.26$ - Fig. 4);

- ADS-based hybrid approaches outperform corresponding standard randomly initialized GA and PSO techniques for both convergence rate and array features whatever the synthesis objective and the array geometry (e.g., $PSL_{RNDGA}^{un} = -20.14$[dB] versus $PSL_{ADSGA}^{un} = -20.93$[dB] for *Problem A* - Table II);

- the ADSPSO turns out to be more efficient and effective than the ADSGA when dealing with small arrangements, while the ADSGA outperforms the other hybridizations when medium/large arrays are at hand (Section IV);

- the *"unequal-arms"* geometry usually guarantees fitter solutions than the state-of-the-art *"equal-arms"* displacements, especially when is small (e.g., $PSL^{un}_{ADSPSO} = -21.35$[dB] versus $PSL_3 = -20.3$[dB] - Tables II and III).

Future efforts will be devoted to assess the advantages, potentialities, and limitations of the proposed methodologies when dealing with more realistic scenarios (e.g. directive elements or wideband behavior) and/or considering other geometric architectures such as Reuleaux triangles [31]. Towards this end, the exploitation of linear ADSs in open and closed-ended configuration as well as 2D ADSs [48] will be carefully analyzed. As an additional research topic in future papers, the design parameter spaces for which computational efficiency is a practical limitation will be explored. Indeed, this could allow to discriminate when optimization is impractical even with modern computers and ADS-based techniques are best-in-class or when a full-stochastic approach is more effective.

Finally the third part of this Thesis has been devoted to a hybrid ADSGA-based methodology for planar antenna arrays. This synthesis technique has been presented and developed to improve performance of large thinned arrays. These results can be very useful to design and enhance the features in the far-field and for narrow-band signals of ADS-based binary sequences for planar array thinning. To overcome the main limitations (i.e. flexibility and performance) of ADS-based thinned arrays, while taking advantage of their properties, an innovative methodological approach that, unlike the ADS thinning techniques described in [48], does not rely on purely analytical design method, has been proposed. An extensive numerical analysis has been performed by addressing different kinds of problems, each one concerned with a specific ADS limitation. The obtained results have pointed out the following outcomes:

1. thanks to the ADS initialisation, the ADSGA provides improved performance with respect to a standard GA approach when dealing with linear array thinning, even though the improvements are not always very significant;

2. ADSGA-constrained designs are usually advantageous since they avoid both quasi-dense layouts of limited practical importance as well as large main lobe widths, unlike unconstrained architectures;

3. the knowledge of ADS reference sequences and the a priori information on the performance of the corresponding arrays turn out to be useful even for synthesising antenna arrangements with different (also when ADSs do not exist) thinning factors or sizes;

4. the hybrid approach can be profitably employed to determine the explicit form of new ADS sequences of desired length beyond those already available, thus extending the range of applicability of the ADS-based array thinning.

Bibliography

[1] Balanis, C.A., Antenna Theory, Analysis and Design, Wiley,New York: 1997.

[2] J. D. Kraus, Antennas, 2nd Ed. McGraw-Hill, 2001.

[3] Y. T. Lo, "A Mathematical theory of antenna arrays with randomly spaced elements," *IEEE Trans. Antennas Propag.*, vol.12, pp. 257-268, May 1964.

[4] M. I. Skolnik, J. W. Sherman III, F. C. Ogg Jr. "Statistically Designed Density-Tapered Arrays," *IEEE Trans. Antennas Propag.,* vol. 12, pp. 408-417, Jul. 1964.

[5] D. G. Leeper, "Isophoric Arrays-Massively Thinned Phased Arrays with Well-Controlled Sidelobes," *IEEE Trans. Antennas Propag.*, vol. 47, no. 12, pp. 1825-1835, Dec. 1999.

[6] B. D. Steinberg, "The Peak Sidelobe of the Phased Array having Randomly Located Elements," *IEEE Trans. Antennas Propag.*, vol. 20, no 2, pp. 129-136, Mar. 1972.

[7] D. H. Rogstad, A. Mileant, T. T. Pham, "Antenna Arraying Techniques in the Deep Space Networks," Deep Space Communications and Navigation Series, Jet Propulsory Lab., Caltech, pp. 2-4.

[8] A. Thompson, J. Moran and G. Swenson, "Interferometry and Synthesis in Radio Astronomy," 2nd Ed. New York: Wiley 2001.

[9] G. W. Swenson Jr., N. C. Mathur, "The Interferometer in Radio Astronomy," *Proc. of the IEEE*, vol. 56, no. 12, pp. 2114-2130, Dec. 1968.

[10] C. L. Dolph, "A current distribution for broadside arrays which optimizes the relationship between beamwidth and side lobelevel," *Proc. IRE*, vol. 34, pp. 335-348; Jun., 1946.

[11] T. T. Taylor, "Design of line-source antennas for narrow beamwidth and low sidelobes," *IRE Trans. Antennas Propag.*, vol. AP-3, pp. 16-28; Jan., 1955.

[12] B. D. Steinberg, "Comparison between the Peak Sidelobe of the Random Array and Algorithmically Designed Aperiodic Arrays," *IEEE Trans. Antennas Propag.*, May 1973.

[13] Y. T. Lo and S. W. Lee, "A study of space tapered arrays," *IEEE Trans. Antennas Propag.*, vol. AP-14, pp. 22-30, Jan, 1966.

[14] D. E. Goldberg, "Genetic Algorithm in Search, Optimization and Machine Learning". Reading, MA: Addison-Wesley, 1989.

[15] M. Skolnik. Radar Handbook, 2001.

[16] C. S. Ruf, "Numerical Annealing of Low-Redundancy Linear Arrays," *IEEE Trans. Antennas Propag.*, vol. 41, no. 1, pp. 85-90,1993.

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

[18] R. L. Haupt, "Thinned Arrays Using Genetic Algorithms," *IEEE Trans. Antennas Propag.*, vol. 42, no. 7, Jul. 1994.

[19] D. G. Leeper, "Thinned aperiod antenna arrays with improved peak sidelobe level control," January 31, 1978. U. S.Patent.

[20] L. D. Baumert, Cyclic Difference Sets. New York: Springer-Verlag. 1971.

[21] L. E. Kopilovich, "Square Array Based on Hadamard Difference Sets," *IEEE Trans. Antennas Propag.*, vol. 56, no. 1, pp. 263-266, Jan. 2008.

[22] G. Oliveri, M. Donelli, A. Massa, "Linear Array Thinning exploiting Almost Difference Sets," *IEEE Trans. Antennas Propag.*, vol. 57, no. 12, pp. 3800-3812, Dec. 2009.

[23] G Oliveri, L. Manica, A. Massa, "ADS-Based Guidelines for Thinned Planar Arrays," *IEEE Trans. Antennas Propag.*, vol. 58, no. 6, pp. 1935-1948, Jun. 2010.

[24] S. Caorsi, A. Lommi, A. Massa, M. Pastorino, "Peak Sidelobe Level Reduction With a Hybrid Approach Based on GAs and Difference Sets," *IEEE Trans. Antennas Propag.*, vol. 52, no. 4, pp. 1116-1121, Apr. 2004.

[25] M. Donelli, A. Martini, A. Massa, "A Hybrid Approach Based on PSO and Hadamard Difference Sets for the Synthesis of Square Thinned Arrays," *IEEE Trans. Antennas Propag.*, vol. 57, no. 8, pp. 2491-2495, Aug. 2009.

[26] J. Kennedy, R. Eberhart, "A discrete binary version of the particle swarm algorithm," *Proc. of the IEEE Int. Conf. Systems, Man, and Cybernetics*, vol. 46, pp. 4104-4108, 1997.

[27] N. Jin, Y. Rahmat-Samii, "Advances in particle swarm optimization for antenna designs: real-number, binary, single objective and multiobjective implementations," *IEEE Trans. Antennas Propag.*, vol. 55, no. 3, pp. 556-567, 2007.

[28] J. Kennedy, R. C. Eberhart, and Y. Shi, "Swarm Intelligence". San Francisco: Morgan Kauffman, 2001.

[29] E. B. Fomalont, "Earth-Rotation Aperture Synthesis," *Proc.of the IEEE*, vol. 61, no. 9, pp. 1211-1218, Sep. 1973.

[30] G. W. Swenson Jr., "Synthetic-Aperture Radio Telescopes," *Ann. Rev. Astron. Astrophys.*, vol. 7, pp. 353-374, 1969.

[31] N. Jin, Y. Rahmat-Samii, "Analysis and Particle Swarm Optimization of Correlator Antenna Arrays for Radio Astronomy Applications," *IEEE Trans. Antennas Propag.*, vol. 56, no. 5, pp.1269-1279, May 2008.

[32] R. N. Bracewell, "Radio Interferometry of Discrete Sources," *Proc. of the IRE*, vol. 46, pp. 97-105, Jan. 1958.

[33] M. Ishiguro, "Minimum redundancy linear arrays for a large number of antennas," *Radio Sci.,* vol. 15, pp. 1163-1170, 1980.

[34] N. Mathur, "A pseudodynamic programming techniques for the design of correlator supersynthesis arrays," *Radio Sci.*, vol. 4, no. 3, pp.235-244, Mar. 1969.

[35] Y. Chow, "On designing a supersynthesis antenna array," *IEEE Trans. Antennas Propag.*, vol. 20, no. 1, pp. 30-35, Jan. 1972.

[36] A. Pott et al. (eds), Difference sets, sequences and their correlation properties, Kluver Academic Publisher, The Netherlands, 1999.

[37] R. C. Walker, "VBLI array design," in Indirect Imaging, J.A. Roberts, Ed. Cambridge, England: Cambridge Univ. Press. 1984, pp. 53-65.

[38] A. Trucco, "Thinning and Weighting of Large Planar Arrays by Simulated Annealing," IEEE Trans. on Ultrasonics, Ferroelectrics, and Freq. Control., vol. 46, no. 2, Mar. 1999.

[39] O. Quevedo-Teruel, E. Rajo.Iglesias, "Ant Colony Optimization in Thinned Array Synthesis with Minimum Sidelobe Level," IEEE Trans. on Ant. and Wireless Propag. Letters, vol. 5, 2006.

[40] A. T. Moffet, "Minimum-Redundancy Linear Arrays," *IEEE Trans. Antennas Propag.*, vol. AP-16, no. 2, pp. 172-175, Mar. 1969.

[41] I. E. Lager, C. Trampuz, M. Simeoni, and L. P. Ligthart, "Interleaved array antennas for FMCW radar applications," IEEE Trans. Antennas Propag., vol. 57, no. 8, pp. 2486-2490, Aug. 2009.

[42] R. J. Mailloux, "Phased Array Antenna Handbook," 2nd ed. Norwood,MA: Artech House, 2005.

[43] M. I. Skolnik, G. Nemhauser, and J. W. Sherman, "Dynamic programming applied to unequally-space arrays," *IRE Trans. Antennas Propag.*,vol. 12, pp. 35-43, Jan. 1964.

[44] R. L. Haupt and D. H. Werner, "Genetic Algorithms in Electromagnetics," Hoboken, NJ: Wiley, 2007.

[45] M. Donelli, S. Caorsi, F. De Natale, D. Franceschini, and A. Massa, "A versatile enhanced genetic algorithm for planar array design," *JEMWA*, vol. 18, pp. 1533-1548, 2004.

[46] G. Oliveri, L. Manica, and A. Massa, "On the impact of mutual coupling effects on the PSL performances of ADS thinned arrays," *PIERB*, vol. 17, pp. 293-308, 2009.

[47] G. Oliveri, M. Donelli, and A. Massa, "Genetically-designed arbitrary length almost difference sets," *Electron. Lett.*, vol. 5, no. 23, pp.1182-1183, Nov. 2009.

[48] G. Oliveri, L. Manica, and A. Massa, "ADS-based guidelines forthinned planar arrays," *IEEE Trans. Antennas Propag.*, vol. 58, no. 6, pp. 1935-1948, June 2010.

[49] La Jolla Cyclic "Difference Set Repository" [Online]. Available: http://www.ccrwest.org/diffsets.html

[50] G. Oliveri and A. Massa, "Genetic algorithm (GA)-enhanced almost difference set (ADS)-based approach for array thinning," IET Microw. Antennas Propag., vol. 5, pp. 305-315, Feb. 2011.

[51] R. L. McFarland, "A family of difference sets in non-cyclic groups," J. Combin. Theory, vol. 15, no. 1, pp. 1-10, July 1973.

[52] C. Carilli and S. Rawlings, "Science with the square kilometer array: Motivation, key science projects, standards and assumptions," *New Astron. Rev.*, vol. 48, Dec. 2004.

[53] L. Kogan, "Optimization of an array configuration minimizing sidelobes," MMA Memo., no. 171, May 1997.

[54] F. Boone, "Interferometric array design: Optimizing the locations of the antenna pads," *Astron. Astrophy.*, vol. 377, no. 1, pp. 368-376, Oct. 2001.

[55] J. Arsac, "Nouveau reseau pour l'observation radioastronomique de labrilliance de la soleil a 9350 Mc/s," Acad Sci., vol. 240, pp. 942-945, 1955.

[56] D. W. Boeringer and D. H. Werner, "Particle swarm optimization versus genetic algorithms for phased array synthesis," *IEEE Trans. Antennas Propag.*, vol. 52, no. 3, pp. 771-779, Mar. 2004.

[57] A. Wootten and A. R. Thompson, "The Atacama large millimeter/submillimeterarray," *Proc. IEEE*, vol. 97, no. 8, pp. 1463-1471, Aug. 2009.

[58] P. E. Dewdney, P. J. Hall, R. T. Schilizzi, and T. J. L. W. Lazio, "The square kilometer array," *Proc. IEEE*, vol. 97, no. 8, pp. 1482-1496, Aug. 2009.

[59] J. Robinson and Y. Rahmat-Samii, "Particle swarm optimization in electromagnetics," *IEEE Trans. Antennas Propag.*, vol. 52, no. 2, pp. 397-407, Feb. 2004.

[60] E. Keto, "The shapes of cross-correlation interferometers," Astrophys .J., vol. 475, no. 2, pp. 843-852, Feb. 1997.

[61] C. Ding, T. Helleseth, and K. Y. Lam, "Several classes of binary sequences with three-level autocorrelation," *IEEE Trans. Inf. Theory*, vol. 45, no. 7, pp. 2606-2612, Nov. 1999.

[62] N. Jin and Y. Rahmat-Samii, "Particle swarm optimization for correlator antenna array designs in radio astronomy applications," in Proc. 2nd Eur. Conf. Antennas Propag., Nov. 11-16, 2007, pp. 1-4.

[63] Y. Zhang, J. G. Lei, and S. P. Zhang, "A new family of almost difference sets and some necessary conditions," *IEEE Trans. Inf. Theory*, vol. 52, no. 5, pp. 2052-2061, May 2006.

[64] G. Oliveri, P. Rocca, and A. Massa, "Interleaved linear arrays with difference sets," *Electron. Lett.*, vol. 45, no. 5, pp. 323-324, 2010.

[65] K. T. Arasu, C. Ding, T. Helleseth, P. V. Kumar, and H. M. Martinsen, "Almost difference sets and their sequences with optimal autocorrelation," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2934-2943, Nov. 2001.

[66] P. Rocca, M. Benedetti, M. Donelli, D. Franceschini, and A. Massa, "Evolutionary optimization as applied to inverse scattering problems," *Inv. Probl.*, vol. 25, no. 12, article no. 123003, pp. 1-41, Dec. 2009.

[67] P. Napier, A. Thompson, and R. Ekers, "The very large array: Design and performance of a modern synthesize radio telescope," *Proc. IEEE*, vol. 71, no. 11, pp. 1295-1320, Nov. 1983.

[68] ELEDIA Almost Difference Set Repository [Online]. Available: http://www.ing.unitn.it/~eledia/html/

[69] G. Oliveri and A. Massa, "Fully-interleaved linear arrays with predictable sidelobes based on almost difference sets," *IET Radar, Sonar Nav*, vol. 4, no. 5, pp. 649-661, Oct. 2010.

[70] G. Oliveri and A. Massa, "ADS-based array design for 2D and 3D ultrasound imaging," *IEEE Trans. Ultrason., Ferroelectr., Freq. Control*, vol. 57, no. 7, pp. 1568-1582, Jul. 2010.

[71] Holm, S., Elgetun, B., Dahl, G.: "Properties of the beampattern of weight- and layout-optimized sparse arrays", IEEE Trans. Ultrason. Ferroelectr. Freq. Control, 1997, 44, (5), pp. 983-991.

List of Publications

Publications Related to Thesis

1. Journal Papers

   (a) G. Oliveri, F. Caramanica, and A. Massa, "Hybrid ADS-based techniques for radio astronomy array design," *IEEE Transactions on Antennas and Propagation* - Special Issue on "Antennas for Next Generation Radio Telescopes," v. 59, pp. 1817-1827, Jun. 2011.

   (b) G. Oliveri, F. Caramanica, C. Fontanari, and A. Massa, "Rectangular thinned arrays based on McFarland difference sets," *IEEE Transactions on Antennas and Propagation*, v. 59, pp. 1546-1552, May 2011.

2. Conference Papers

   (a) Oliveri, G.; Caramanica, F.; Rocca, P.; Massa, A.; "ADS-based Y-shaped arrays for interferometry and radio astronomy applications," in Proc. 2010 IEEE International Symposium on Phased Array Systems & Technology, Boston, MA, USA, 12-15 October 2010, p. 1-4.

   (b) G. Oliveri, F. Caramanica, P. Rocca, A. Massa, "Rectangular thinned array design by McFarland difference sets" in Proc. of PIERS2010, Cambridge, USA, 05-08 July 2010: PIER, p. 547-547.

   (c) F. Caramanica, M. Seghi, G. Oliveri, A. Massa , "ADS-based hybrid methods for array thinning" in 26th International Review of Progress in Applied Computational Electromagnetics, ACES 2010, Tampere, Finland, p. 942-946.

Other Publications

1. Journal Papers

   (a) F. Caramanica, G. Oliveri, "An innovative multi-source strategy for enhancing the reconstruction capabilities of inverse scattering techniques" in *ELECTROMAGNETIC WAVES*, v. 101, pp. 349-374, 2010.

(b) R. Azaro, F. Caramanica, G. Oliveri, "Determination of the complex permittivity values of planar dielectric substrates by means of a multifrequency PSO-based technique" in *PROGRESS IN ELECTROMAGNETICS RESEARCH M*, v. 10, pp. 83-91, 2009.

(c) A. Martini, F. Caramanica, M. Franceschetti, A. Massa, "Percolation-based models for ray-optical propagation in stochastic distributions of scatterers with random shape," *IEEE Antennas and Wireless Propagation Letters*, n. 6, pp. 639-642, 2007.

2. Conference Papers

(a) G. Oliveri, F. Caramanica, T. Moriyama, A. Massa, and T. Takenaka; "Potentialities and Effectiveness of the IMSA-FBTS Strategy for the Solution of Inverse Scattering Problems," Progress In Electromagnetics Research Symposium Abstracts, Marrakesh, Morocco, Mar. 20-23, 2011, pp. 381 .

(b) F. Caramanica, M. Donelli, G. Oliveri, P. Rocca, A. Massa , "Evolutionary algorithms for inverse scattering - Advances and state-of-the-art comparisons" in 26th International Review of Progress in Applied Computational Electromagnetics, ACES 2010, Tampere, Finland, p. 783-786.

(c) P. A. Barrière, F. Caramanica, M. Benedetti, A. Massa , "Multi-resolution approaches for inverse scattering problems" in 26th International Review of Progress in Applied Computational Electromagnetics, ACES 2010, Tampere, Finland, p. 102-107.

(d) F. Caramanica, M. Carlin, M. Donelli, D. Franceschini, L. Lizzi, L. Manica, G. Oliveri, L. Poli, P. Rocca, F. Viani, A. Massa, "Antenna synthesis techniques @ ELEDIA Research Group" in Atti XVIII Riunione Nazionale di Elettromagnetismo, Benevento, Italy.

(e) P. A. Barrière, F. Caramanica, M. Carlin, M. Donelli, D. Franceschini, L. Lizzi, L. Manica, G. Oliveri, L. Poli, P. Rocca, F. Viani, A. Massa, , "Methodologies for the solution of inverse scattering problems @ ELEDIA Research Group" in Atti XVIII Riunione Nazionale di Elettromagnetismo, Benevento, Italy.