Chapter 3

Extensions to the Transformation Based Planar Array Synthesis Technique

3.1 Introductory Remarks

The transformation based synthesis technique [95, 94] mentioned earlier in Section 2.7.4 differs from other approaches in several respects. It utilises a transformation that has been used in the design of two-dimensional digital filters [122]. This transformation divides the problem into two decoupled sub-problems. In the antenna array context the one sub-problem involves the determination of certain coefficients of the transformation in order to achieve the required footprint contours. The number of coefficients needed depends on the complexity of the desired contour, but is very small in comparison to the number of planar array elements. The other sub-problem consists of a linear array shaped beam synthesis, for which there already exist powerful methods for determining appropriate element excitations. The size required for this linear array (which will be called the *prototype linear array*) depends on the number of transformation coefficients used and the planar array size. Simple recursion formulas then determine the final planar array excitations from the information forthcoming from the above two sub-problem solutions. As a result the method is computationally efficient, and thus trade-off studies are feasible even for very large arrays.

This technique as first applied by the author [95, 94] was applicable to arrays with quadrantal and centro (about the centre) symmetry only. In this chapter of the present thesis the transformation based technique will be extended to apply to the general, arbitrarily contoured beam synthesis problem.

Planar arrays can be categorised into three groups; firstly arrays with an odd number of elements along both principal planes, secondly arrays with an even number of elements along both principal planes and lastly arrays with an even number of elements along one

principal plane and an odd number of elements along the other pricipal plane. The first two groups will be referred to as the odd case and the even case, respectively. As the formulation differs slightly for each case, Section 3.2 deals with the odd case and Section 3.3 with the even case. The formulation for the last group of planar arrays is simply a combination of the odd and even case formulations and will not be treated. Details of the original transformation based technique, as developed in [95, 94], will be discussed in the first part of each of these sections. The latter part of each of the two sections discusses the extensions to the technique to enable the synthesis of planar arrays with arbitrary footprint contour patterns. A planar array with a rectangular lattice and rectangular boundary is assumed throughout these two sections. In order not to smother the simplicity of the technique under the associated notationally complex algorithmic details, the latter have all been relegated to Appendix A, which is simply referred to at appropriate stages in the text.

The contour approximation, or the problem of determining the transformation coefficients, will be addressed in Section 3.4.

Section 3.5 deals with the application of the method to synthesise planar arrays with non-rectangular boundaries and/or lattices not available in the original formulation [95, 94]. This is achieved by the appropriate selection of the transformation coefficients.

The method is epitomised by a complete, detailed representative example in Section 3.6. The required footprint contour is the shape of the African continent.

In Section 3.7 the transformation based synthesis method is compared to similar existing synthesis methods.

The chapter ends with some conclusions.

3.2 The Transformation Based Synthesis Technique: The Odd Case

Consider a planar array with a rectangular lattice. The inter-element spacings, d_x and d_y, need not be equal. Only planar arrays consisting of $2M+1$ by $2N+1$ elements, that is an odd number of elements in both principal planes (and an odd total number of elements as well) are examined in this section. In addition, the array is assumed to have a rectangular boundary, no elements having been removed from the lattice in order to alter the boundary shape.

3.2.1 Background on Quadrantally Symmetric Contours

In this section we consider quadrantally symmetric contoured footprint patterns, where the planar array factor is symmetrical about both the u- and v-axes. Such contour patterns are generated by planar arrays with quadrantally symmetric excitations. This

section repeats the work reported in [95], which was also published in [94], as background material which facilitates the discussion of the extended transformation method for the odd case in Section 3.2.2

Consider a rectangular planar array in the xy-plane. Using the substitutions

$$\begin{aligned} u &= \mathrm{k}d_x \sin\theta\ \cos\phi \\ v &= \mathrm{k}d_y \sin\theta\ \sin\phi \end{aligned} \tag{3.1}$$

given in (2.14), with θ and ϕ the direction of the far-field observation point, the $2M+1$ by $2N+1$ element quadrantally symmetric planar array space factor is given in (2.16) as

$$F(u,v) = \sum_{m=0}^{M}\sum_{n=0}^{N} \zeta_m \zeta_n a_{mn} \cos(mu)\cos(nv) \tag{3.2}$$

with a_{mn} the normalised excitation of the $mn-$th element and ζ_i as defined in equation (2.17). The constant "4" before the summation may be omitted as the space factor is usually normalised to its maximum.

Next, consider a $2Q+1$ element, uniformly spaced linear array with symmetrical excitation. As indicated in Section 3.1 this will be referred to as the *prototype linear array.* Assume the prototype linear array is positioned along the x-axis, with inter-element spacing d. The observation angle θ_p is measured from the direction broadside to the prototype linear array. The range of θ_p is from -90° to 90° , as mentioned in Section 2.3.6. The path length difference is defined as

$$\psi_p = \mathrm{k}d \sin\theta_p \tag{3.3}$$

From (2.13), the prototype linear array factor in terms of ψ_p is then

$$F_p(\psi_p) = \sum_{q=0}^{Q} \zeta_q\, a_q\, \cos(q\psi_p) \tag{3.4}$$

in which a_q is the q-th normalised element excitation. Use of the relations [123, p. 17]

$$\cos^{2q} x = \frac{1}{2^{2q}} + \frac{1}{2^{2q-1}} \sum_{i=1}^{q} \binom{2q}{q-i} \cos(2ix) \tag{3.5a}$$

$$\cos^{2q-1} x = \frac{1}{2^{2q-2}} \sum_{i=1}^{q} \binom{2q-1}{q-i} \cos\left[(2i-1)x\right] \tag{3.5b}$$

enables us to write the prototype linear array factor (3.4) in a polynomial form

$$F_p(\psi_p) = \sum_{q=0}^{Q} b_q\, \cos^q \psi_p \tag{3.6}$$

The original McClellan transform [122] is a transformation of a one-dimensional FIR filter into a two-dimensional FIR filter by means of the substitution of variables. In the array context the transformation "converts" the prototype linear array factor (a function of one variable) into a planar array factor (a function of two variables) by means of the substitution of variables

$$\cos(\psi_p) = H(u,v) = \sum_{i=0}^{I}\sum_{j=0}^{J} t_{ij}\ \cos(iu)\cos(jv) \tag{3.7}$$

in which t_{ij} are real coefficients. This function, $H(u,v)$, will be called the *transformation function*. We assume here that the transformation is normalised in the sense that $|H(u,v)| \leq 1$. The scaling of the transformation function will be discussed in more detail Section 3.4.3.

Substitution of the transformation function (3.7) into the prototype linear array factor (3.6) converts the prototype linear array factor $F_p(\psi_p)$ into the following function of two variables, (u and v):

$$F_p(u,v) = \sum_{q=0}^{Q} b_q \left[\sum_{i=0}^{I}\sum_{j=0}^{J} t_{ij}\ \cos(iu)\cos(jv)\right]^q \tag{3.8}$$

Application of the relations in (3.5) a second time allows this two-variable function (3.8) to be re-written in the same form as the planar array factor

$$F_p(u,v) = \sum_{m=0}^{M}\sum_{n=0}^{N} c_{mn}\ \cos(mu)\cos(nv) \tag{3.9}$$

with $M = QI$ and $N = QJ$. Comparison of (3.9) with the planar array factor (3.2) shows that the planar array element excitation is

$$a_{mn} = \frac{c_{mn}}{\zeta_m \zeta_n} \tag{3.10}$$

Recursive formulas, given in Section A.1.1 of Appendix A are used to compute b_q in (3.6) from the prototype linear array excitations a_q. Section A.1.2 of Appendix A is devoted to the recursive formulas necessary to compute the coefficients c_{mn} of (3.9). Although writing these formulas in a mathematically elegant fashion is difficult, the recursive formulas are ideally suited for computation.

On those contours where the transformation function is constant $H(u,v) = constant$, the planar array $F(u,v)$ must be constant. These contours are controlled only by the transformation coefficients t_{ij}. The value associated with a particular contour depends only on the excitation coefficients of the prototype linear array and is equal to $F_p(\psi_p)$ with $\psi_p = \arccos[H(u,v)]$. The synthesis problem has thus been reduced to two smaller and more easily solvable problems; the synthesis of a linear array and the selection of a set of

transformation parameters. As the transformation function is quadrantally symmetrical, the planar array factor (as well as the excitation) will also be quadrantally symmetrical.

The desired accuracy for, or amount of detail in the shape of, the desired contours will determine the values of I and J required; that is, the number of contour transformation coefficients required. The use of contour transformations with larger values of I and J will increase the size of the final planar array. The size $(2Q+1)$ of the prototype linear array used is dependent on the width of the main beam and amount of coverage area gain ripple allowable. The size of the final planar array is $2M+1$ by $2N+1$, with $M=QI$ and $N=QJ$. It may be interesting to note that if the prototype linear array has a pure real distribution then the planar array distribution will also be pure real.

3.2.2 Extension to Arbitrarily Shaped Contours

Details of the extension of the transformation based synthesis technique to enable the design of antenna arrays with arbitrarily shaped contoured footprint patterns are discussed in this section. If the desired contour has no symmetry, the resultant excitations will also have no symmetry.

As $\cos u$ and $\cos v$ are even functions in u and v respectively, the transformation function (3.7) is quadrantally symmetrical. Thus the resultant planar array factor will also exhibit quadrantal symmetry. In order to synthesise arbitrarily shaped contours, the transformation function must include both even and odd terms. The transformation for arbitrarily shaped contours is therefore

$$\cos(\psi_p) = H(u,v) = \sum_{i=0}^{I}\sum_{j=0}^{J} t_{ij}^{cc}\cos(iu)\cos(jv) + t_{ij}^{ss}\sin(iu)\sin(jv) + t_{ij}^{cs}\cos(iu)\sin(jv) + t_{ij}^{sc}\sin(iu)\cos(jv) \tag{3.11}$$

in which t_{ij}^{cc}, t_{ij}^{ss}, t_{ij}^{cs} and t_{ij}^{sc} are real coefficients. Inspection of equation (3.11) reveals that the transformation function has the same form as a Fourier series. From the theory of Fourier analysis we know that the transformation function (3.11) can be used to approximate any real function.

Substitution of the general transformation function (3.11) into the prototype linear array factor (3.6) and use of the relations [123, p. 17]

$$\begin{aligned} \cos A\cos B &= \tfrac{1}{2}[\cos(A-B)+\cos(A+B)] \\ \sin A\cos B &= \tfrac{1}{2}[\sin(A-B)+\sin(A+B)] \\ \sin A\sin B &= \tfrac{1}{2}[\sin(A-B)+\sin(A+B)] \end{aligned} \tag{3.12}$$

allow the prototype linear array factor $F_p(\psi_p)$ to be written as

$$F_p(u,v) = \sum_{m=0}^{M}\sum_{n=0}^{N}\zeta_m\zeta_n\,[c_{mn}^{cc}\cos(mu)\cos(nv) + c_{mn}^{ss}\sin(mu)\sin(nv) + c_{mn}^{cs}\cos(mu)\sin(nv) + c_{mn}^{sc}\sin(mu)\cos(nv)] \tag{3.13}$$

with $M=QI$ and $N=QJ$. Let us now consider the array factor of a planar array which has no special symmetry in its excitation. Using the element numbering scheme described in Section 2.2.4 of Chapter 2 for the quadrantally symmetric factor, expression (2.15) can be written as

$$\begin{aligned} F(u,v) = \sum_{m=0}^{M}\sum_{n=0}^{N} \zeta_m\zeta_n \{ & (+a_{mn} + a_{m-n} + a_{-mn} + a_{-m-n})\cos(mu)\cos(nv) + \\ & (-a_{mn} + a_{m-n} + a_{-mn} - a_{-m-n})\sin(mu)\sin(nv) + \\ & \mathrm{j}(+a_{mn} - a_{m-n} + a_{-mn} - a_{-m-n})\cos(mu)\sin(nv) + \\ & \mathrm{j}(+a_{mn} + a_{m-n} - a_{-mn} - a_{-m-n})\sin(mu)\cos(nv)\} \end{aligned} \tag{3.14}$$

Comparison of (3.13) with the planar array factor (3.14) yields the final array excitation

$$\begin{aligned} a_{mn} &= \tfrac{1}{4\zeta_m\zeta_n}[c_{mn}^{cc} - c_{mn}^{ss} - \mathrm{j}(c_{mn}^{cs} + c_{mn}^{sc})] \\ a_{m-n} &= \tfrac{1}{4\zeta_m\zeta_n}[c_{mn}^{cc} + c_{mn}^{ss} + \mathrm{j}(c_{mn}^{cs} - c_{mn}^{sc})] \\ a_{-mn} &= \tfrac{1}{4\zeta_m\zeta_n}[c_{mn}^{cc} + c_{mn}^{ss} - \mathrm{j}(c_{mn}^{cs} - c_{mn}^{sc})] \\ a_{-m-n} &= \tfrac{1}{4\zeta_m\zeta_n}[c_{mn}^{cc} - c_{mn}^{ss} + \mathrm{j}(c_{mn}^{cs} + c_{mn}^{sc})] \end{aligned} \tag{3.15}$$

and the final array size as $M=QI$ and $N=QJ$.

Recursive formulas for computing c_{mn}^{cc}, c_{mn}^{ss}, c_{mn}^{cs} and c_{mn}^{sc} are given in Section A.1.3 of Appendix A.

Again, as stated in Section 3.2.1, on those contours defined by $H(u,v) = \textit{constant}$, the planar array must be constant and equal to $F_p(\psi_p)$, with $\psi_p = \arccos[H(u,v)]$. These contours are controlled only by the transformation parameters t_{ij}^{cc}, t_{ij}^{ss}, t_{ij}^{cs} and t_{ij}^{sc}, and the value associated with the particular contour depends only on the prototype linear array excitation. In this way the synthesis problem is reduced to two smaller and easily solvable problems; the synthesis of a linear array and the selection of a set of transformation parameters.

The selection of the transformation parameters can be viewed as finding the coefficients of a two-dimensional Fourier series representation of the contour shape. Thus, the narrower the beamwidth and/or the more detailed the contour, the more transformation parameters are needed. The increase in the number of transformation coefficients will cause a proportional increase in the final array size. The problem of determining the appropriate values for the transformation coefficients is discussed in Section 3.4

Illustrative Example #1

To illustrate the transformation based synthesis technique for arbitrary contours, let us consider the synthesis of a contoured pattern in the shape of a skewed tear drop. The synthesis process can be divided into a number of steps:

i, j	t_{ij}^{cc}	t_{ij}^{ss}	t_{ij}^{cs}	t_{ij}^{sc}
0, 0	-0.117050	0	0	0
0, 1	0.257781	0	-0.171854	0
1, 0	0.515561	0	0	0.257781
1, 1	0.343707	-0.343707	0.171854	-0.257781

Table 3.1: Illustrative Example #1: Transformation coefficients.

1. Determine the number of transformation coefficients.
2. Determine the number of prototype linear array elements.
3. Prototype linear array synthesis.
4. Determine the transformation coefficients.
5. Compute planar array excitations

These steps are not completely isolated. The contour complexity obtainable is linked to the number of transformation coefficients. The beamwidth, main beam ripple and sidelobe level depend on the number of prototype linear array elements. The number of transformation coefficients (I and J) and the number of prototype linear array elements (Q) in turn determine the size of the planar array ($M=IQ$ and $N=JQ$).

Let us consider an example of the synthesis of a square array with a main beam footprint contour in the shape of a skewed tear drop. The peak-to-peak ripple desired in the main beam is 1dB while the sidelobe level is set at -20dB. The inter-element spacing is chosen as $d_x=d_y=0.662\lambda$. The step by step synthesis of this example is:

1. The desired tear-drop shaped contour can be achieved with 9 transformation coefficients, that is $I=J=1$.
2. The prototype linear array consists of 21 elements, that is $Q=10$. The first two steps determine the total number of element in the planar array.
3. The method of Orchard, Elliot and Stern [54] was used to synthesise the prototype linear array, with two of the roots placed off the unit circle. The peak-to-peak ripple in the main beam region is 1dB and the sidelobe level is set at -20dB. The inter-element spacing used in this step is $d=\frac{1}{2}\lambda$. This step determines the value of θ_p.
4. The inter-element spacing of the planar array is accounted for in this step, $d_x=d_y=0.662\lambda$. The strictly controlled contour is at $H(u,v)=\cos(\pi\sin\theta_p)$. The values of the transformation coefficients are listed in Table 3.1

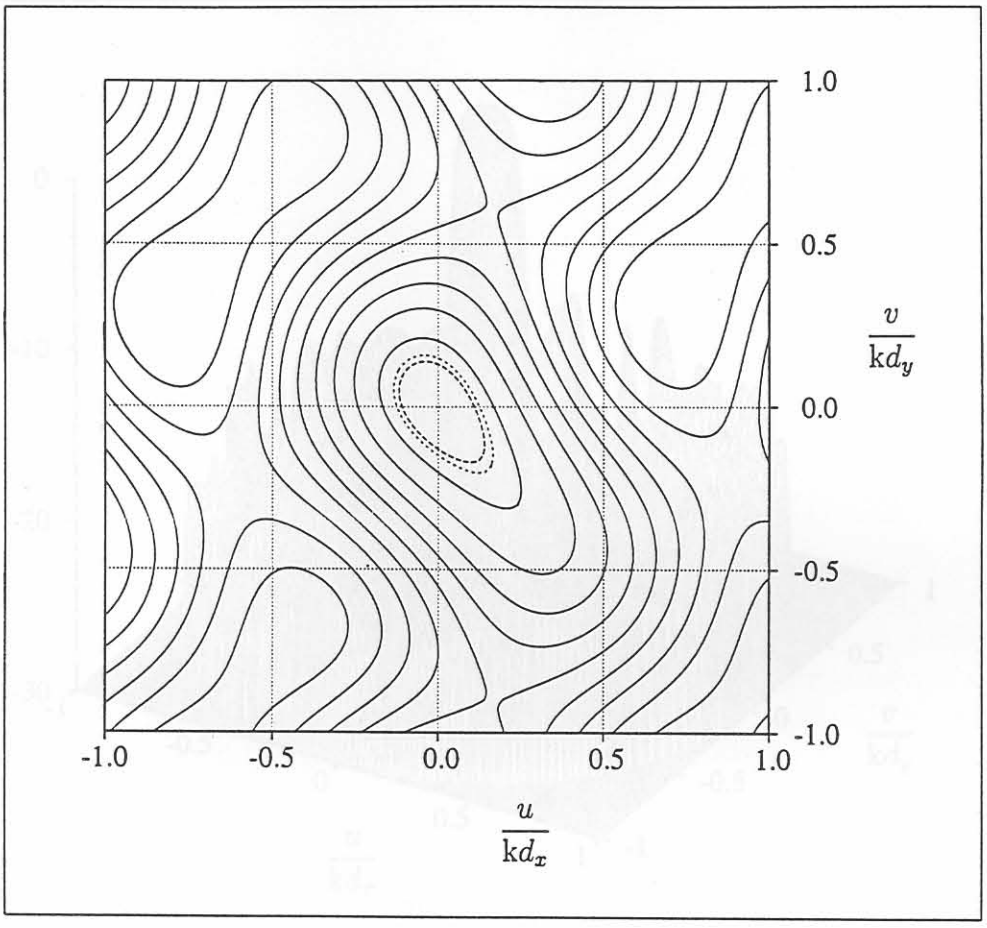

Figure 3.1: Illustrative Example #1: A contour plot of the transformation function. The dashed line represents the -1dB contour and the dotted line the -3dB contour.

5. The planar array elements were computed using the algorithm in Section A.1.3 in Appendix A. This gives the distribution of a planar array with a total of 441 elements, since $M=N=10$ (hence a 21 by 21 square array).

A contour plot of the transformation function is shown in Figure 3.1. The dashed line (in the centre of the graph) represents the -1dB contour at $H(u,v)=0.913$ ($\theta_p=7.7°$) and the dotted line the -3dB contour at $H(u,v)=0.887$ ($\theta_p=8.8°$). The planar array factor of the resulting array is depicted in Figure 3.2.

3.3 The Transformation Based Synthesis Technique: The Even Case

The class of planar arrays regarded in this section consists of an even number of elements in both principal planes, $2M$ and $2N$ elements respectively, and is referred to as the even case. A planar array with a rectangular lattice and boundary is assumed.

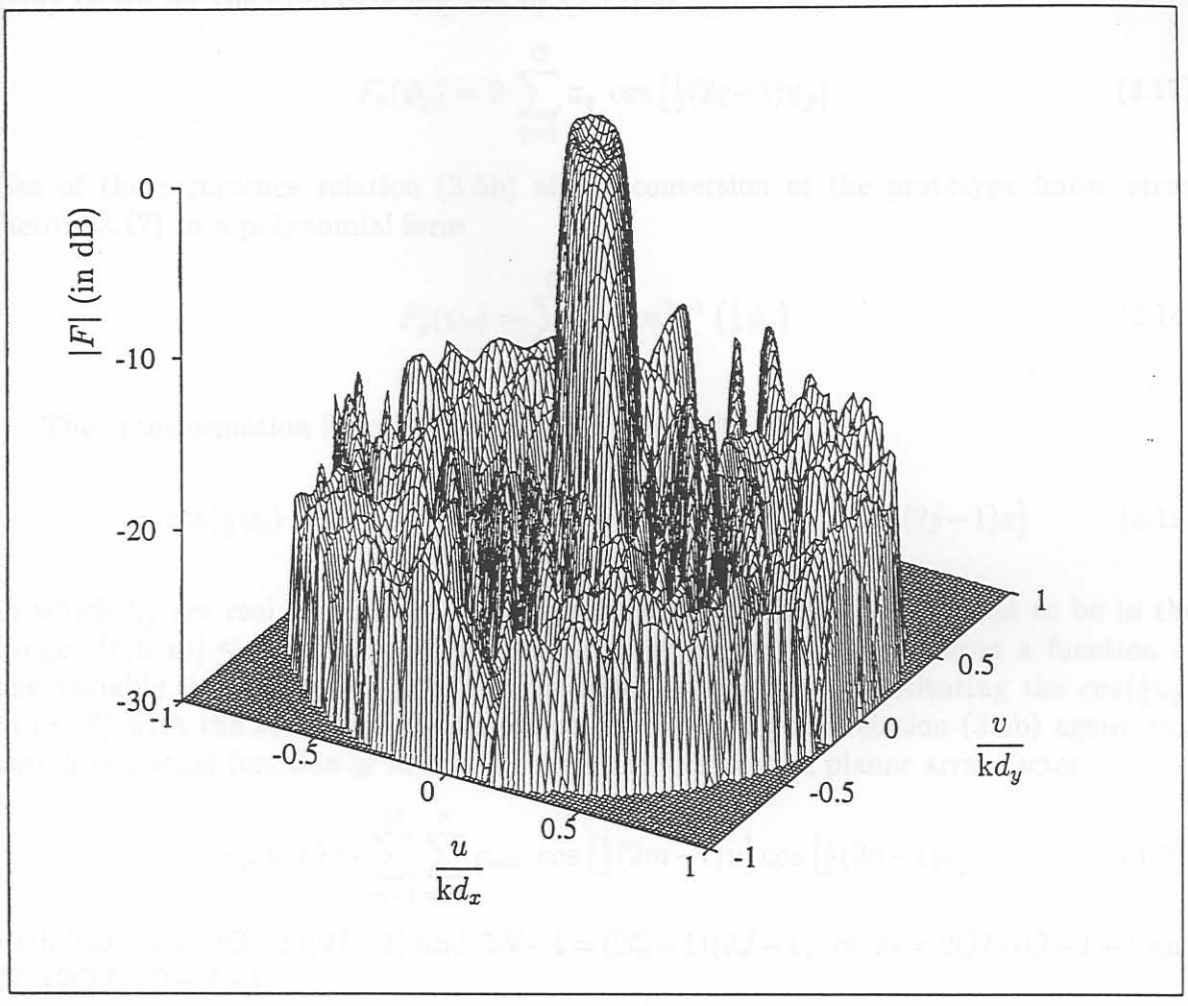

Figure 3.2: Illustrative Example #1: Surface plot of the planar array factor.

3.3.1 Background On Quadrantally Symmetrical Contours

This section deals with quadrantally symmetrical contoured footprint patterns. The information was already published in [95, 94], but is repeated here as background material for the extension described in Section 3.3.2

Consider a rectangular planar array with $2M$ by $2N$ elements excited in phase. The planar space factor is given by (2.18) as

$$F(u,v) = \sum_{m=1}^{M}\sum_{n=1}^{N} a_{mn} \cos\left[\tfrac{1}{2}(2m-1)u\right] \cos\left[\tfrac{1}{2}(2n-1)v\right] \tag{3.16}$$

with a_{mn} the normalised excitation of the mn-th element. The prototype linear array is a $2Q$ element, symmetrically excited, uniformly spaced linear array. The prototype linear

array factor for the even case is given by (2.11) as

$$F_p(\psi_p) = 2\sum_{q=1}^{Q} a_q \cos\left[\tfrac{1}{2}(2q-1)\psi_p\right] \tag{3.17}$$

Use of the recurrence relation (3.5b) allows conversion of the prototype linear array factor (3.17) to a polynomial form

$$F_p(\psi_p) = \sum_{q=1}^{Q} b_q \cos^{2q-1}\left(\tfrac{1}{2}\psi_p\right) \tag{3.18}$$

The transformation function for the even case is [95, 94]

$$\cos(\tfrac{1}{2}\psi_p) = H(u,v) = \sum_{i=1}^{I}\sum_{j=1}^{J} t_{ij} \cos\left[\tfrac{1}{2}(2i-1)u\right]\cos\left[\tfrac{1}{2}(2j-1)v\right] \tag{3.19}$$

in which t_{ij} are real coefficients. The transformation function is assumed to be in the range $|H(u,v)| \leq 1$. The prototype linear array factor is mapped from a function of one variable ψ_p to a function of two variables u and v, by substituting the $cos(\frac{1}{2}\psi_p)$ in (3.18) with the transformation function. By applying the relation (3.5b) again, this two dimensional function is written in the same form as the planar array factor,

$$F_p(u,v) = \sum_{m=1}^{M}\sum_{n=1}^{N} c_{mn} \cos\left[\tfrac{1}{2}(2m-1)u\right]\cos\left[\tfrac{1}{2}(2n-1)v\right] \tag{3.20}$$

with $2M-1=(2Q-1)(2I-1)$ and $2N-1=(2Q-1)(2J-1)$; or $M=2QI-Q-I+1$ and $N=2QJ-Q-J+1$.

The element excitation which is obtained by comparing the planar array factor (3.16) with this function (3.20) is simply

$$a_{mn} = c_{mn} \tag{3.21}$$

The total number of elements in the principal planes of the planar array are $2M = (2Q-1)(2I-1)+1$ and $2N=(2Q-1)(2J-1)+1$. For an $I=J=1$ transformation only one variable is available to control the contour shape, thus only near circular contours can be achieved. As I and J increase, the total number of planar elements, $2M$ and $2N$ increase by $2I-1$ and $2J-1$ respectively. This increase is more rapid than in the case of a planar array with an odd number of elements in both principal planes (as described in Section 3.2.1). The even case suffers another limitation; in order to do a linear scaling of the transformation function a t_{00}-coefficient is needed. Since this coefficient is not available, an additional constraint on the possible values of the transformation function must be included during the determination of the transformation parameters. This will be discussed in Section 3.4.2.

The formula needed to calculate b_q is provided in Section A.2.1 of AppendixA, and the algorithm needed to compute c_{mn} is supplied in Section A.2.2 of Appendix A.

3.3.2 Extension To Arbitrarily Shaped Contours

The extension of the technique for an odd number of elements in both principal planes, to arbitrary contours is very similar to that of Section 3.2.2; thus only the most relevant equations will be given. The transformation function for arbitrarily shaped contours is

$$\begin{aligned} H(u,v) = \sum_{i=1}^{I}\sum_{j=1}^{J} & t_{ij}^{cc}\cos\left[\tfrac{1}{2}(2i-1)u\right]\cos\left[\tfrac{1}{2}(2j-1)v\right] + \\ & t_{ij}^{ss}\sin\left[\tfrac{1}{2}(2i-1)u\right]\sin\left[\tfrac{1}{2}(2j-1)v\right] + \\ & t_{ij}^{cs}\cos\left[\tfrac{1}{2}(2i-1)u\right]\sin\left[\tfrac{1}{2}(2j-1)v\right] + \\ & t_{ij}^{sc}\sin\left[\tfrac{1}{2}(2i-1)u\right]\cos\left[\tfrac{1}{2}(2j-1)v\right] \end{aligned} \tag{3.22}$$

in which t_{ij}^{cc}, t_{ij}^{ss}, t_{ij}^{cs} and t_{ij}^{sc} are real coefficients. The transformation function must be in the range $|H(u,v)| \leq 1$. Normalisation of the transformation function is discussed in Section 3.4.3

Substitution of the general transformation function (3.22) into the prototype linear array factor (3.18), and use of the relations in (3.12), allows the prototype linear array factor $F_p(\psi_p)$ to be written as

$$\begin{aligned} F_p(u,v) = \sum_{m=1}^{M}\sum_{n=1}^{N}\zeta_m\zeta_n & c_{mn}^{cc}\cos\left[\tfrac{1}{2}(2m-1)u\right]\cos\left[\tfrac{1}{2}(2n-1)v\right] + \\ & c_{mn}^{ss}\sin\left[\tfrac{1}{2}(2m-1)u\right]\sin\left[\tfrac{1}{2}(2n-1)v\right] + \\ & c_{mn}^{cs}\cos\left[\tfrac{1}{2}(2m-1)u\right]\sin\left[\tfrac{1}{2}(2n-1)v\right] + \\ & c_{mn}^{sc}\sin\left[\tfrac{1}{2}(2m-1)u\right]\cos\left[\tfrac{1}{2}(2n-1)v\right] \end{aligned} \tag{3.23}$$

with $2M-1=(2Q-1)(2I-1)$ and $2N-1=(2Q-1)(2J-1)$; or $M=2QI-Q-I+1$ and $N=2QJ-Q-J+1$. Now consider the array factor of a planar array which has no special symmetry in its excitation. Using the numbering scheme described in Section 2.2.4 of Chapter 2 for the quadrantally symmetric array factor, expression (2.15) can be written as

$$\begin{aligned} F(u,v) = \sum_{m=1}^{M}\sum_{n=1}^{N} & \{(+a_{mn}+a_{m-n}+a_{-mn}+a_{-m-n})\cos(\tfrac{1}{2}(2m-1)u)\cos(\tfrac{1}{2}(2n-1)v) + \\ & (-a_{mn}+a_{m-n}+a_{-mn}-a_{-m-n})\sin(\tfrac{1}{2}(2m-1)u)\sin(\tfrac{1}{2}(2n-1)v) + \\ & \mathrm{j}(+a_{mn}-a_{m-n}+a_{-mn}-a_{-m-n})\cos(\tfrac{1}{2}(2m-1)u)\sin(\tfrac{1}{2}(2n-1)v) + \\ & \mathrm{j}(+a_{mn}+a_{m-n}-a_{-mn}-a_{-m-n})\sin(\tfrac{1}{2}(2m-1)u)\cos(\tfrac{1}{2}(2n-1)v)\} \end{aligned} \tag{3.24}$$

By equating (3.23) and (3.24) planar array excitations are

$$\begin{aligned} a_{mn} &= \tfrac{1}{4}\left[c_{mn}^{cc} - c_{mn}^{ss} - \mathrm{j}(c_{mn}^{cs} + c_{mn}^{sc})\right] \\ a_{m-n} &= \tfrac{1}{4}\left[c_{mn}^{cc} + c_{mn}^{ss} + \mathrm{j}(c_{mn}^{cs} - c_{mn}^{sc})\right] \\ a_{-mn} &= \tfrac{1}{4}\left[c_{mn}^{cc} + c_{mn}^{ss} - \mathrm{j}(c_{mn}^{cs} - c_{mn}^{sc})\right] \\ a_{-m-n} &= \tfrac{1}{4}\left[c_{mn}^{cc} - c_{mn}^{ss} + \mathrm{j}(c_{mn}^{cs} + c_{mn}^{sc})\right] \end{aligned} \tag{3.25}$$

The final planar array size is $2M-1=(2Q-1)(2I-1)$ by $2N-1=(2Q-1)(2J-1)$. The extension of the transformation based method to arbitrarily shaped contours suffers the same limitations as the quadrantally symmetrical contours; a rapid increase in the number of elements as the number of transformation coefficients increase and linear scaling is not possible. Recursive formulas for computing c_{mn}^{cc}, c_{mn}^{ss}, c_{mn}^{cs} and c_{mn}^{sc} are given in Section A.2.3 of Appendix A.

Illustrative Example #2

Let us consider an example of the synthesis of a rectangular array with a half circle shaped main beam footprint contour. The footprint contour must be symmetrical about the x-axes. The peak-to-peak ripple desired in the main beam is 1dB while the sidelobe level is set at -20dB. The inter-element spacing is chosen as $d_x = d_y = 0.662\lambda$. The step by step synthesis of this example is:

1. Since the desired contour must be symmetrical about the x-axis, $t_{ij}^{ss} = t_{ij}^{cs} = 0$. In order to obtain a good excitation efficiency the ratio of the number of elements along the major axes of the array must be the inverse of the ratio of beamwidths along the principal cuts. From the desired contour shape (and assuming the centre of the half circle will be in the $\theta = 0°$ direction) the ratio of beamwidths along the principal cuts is about 1:1.73. The contour shape is obtainable with $I=3$ and $J=2$; for these values the number of elements along the x-axis would be approximately 5/3 times the number of elements along the y-axis.

2. A prototype linear array of 12 elements was used, that is $Q=6$. The number of transformation coefficients (from the previous step) and the number of prototype linear set the number of planar array elements, $M=28$ and $N=17$

3. The method of Orchard, Elliot and Stern [54] was used to synthesise the prototype linear array, with four roots placed off the unit circle. The peak-to-peak ripple in the main beam region is 1dB and the sidelobe level is set at -20dB. An inter-element spacing of $d=\frac{1}{2}\lambda$ was used in this step.

4. The strictly controlled contour is at $H(u,v) = \cos(\pi \sin\theta_p) = 0.786$. The inter-element spacing is $d_x = d_y = 0.662\lambda$. The values of the transformation coefficients shown in Table 3.2.

i	t_{i1}^{cc}	t_{i1}^{ss}	t_{i1}^{cs}	t_{i1}^{sc}	t_{i2}^{cc}	t_{i2}^{ss}	t_{i2}^{cs}	t_{i2}^{sc}
1	0.430000	0	0	-0.100000	-0.100000	0	0	-0.100000
2	0.250000	0	0	0.100000	0.233891	0	0	-0.150000
3	0.080000	0	0	0.210000	0.080000	0	0	0.000000

Table 3.2: Illustrative Example #2: Transformation coefficients.

5. The final planar array excitations were computed using the algorithm in Section A.2.3 in Appendix A. This gives a total of 1904 elements (a 56 by 34 rectangular array) in the planar array.

Figure 3.3 displays a contour plot of the transformation function. The dashed line (in the centre of the graph) represents the -1dB contour and the dotted line the -3dB contour. In Figure 3.4 a surface plot of the planar array factor is shown.

3.4 The Transformation Function Sub-Problem

This section is devoted to the problem of choosing the transformation parameters to produce iso-contours of the transformation $H(u,v) = constant$ in the uv-plane that have some desired shape. The contours of transformation $H(u,v)$ are controlled by the transformation coefficients t_{ij}^{cc}, t_{ij}^{ss}, t_{ij}^{cs} and t_{ij}^{sc} only. In contoured beam synthesis it will usually be sufficient to firmly control (by the appropriate selection of the transformation coefficients) the shape of a single contour, typically the half-power (-3dB) contour or the first null contour.

Section 3.4.1 deals with transformation grating lobes, a phenomenon unique to the transformation based synthesis technique. The problem of determining the transformation coefficients, or the contour approximation problem, is formulated in Section 3.4.2. To ensure that the transformation function is within certain bounds, it may need to be scaled. A linear scaling technique, only applicable to the odd case, is discussed in Section 3.4.3. If the number of transformation coefficients is small the coefficients of the transformation may be obtained by using some form of approximation. The approximation approach is addressed in Section 3.4.4. The transformation coefficients can also be obtained by using a constrained optimisation technique as described in Section 3.4.5. Alternatively, Section 3.4.6 demonstrates how the transformation function can be viewed as a small planar array, with a pure real array factor and excitation. The problem of selecting the correct values is then a planar array field synthesis problem.

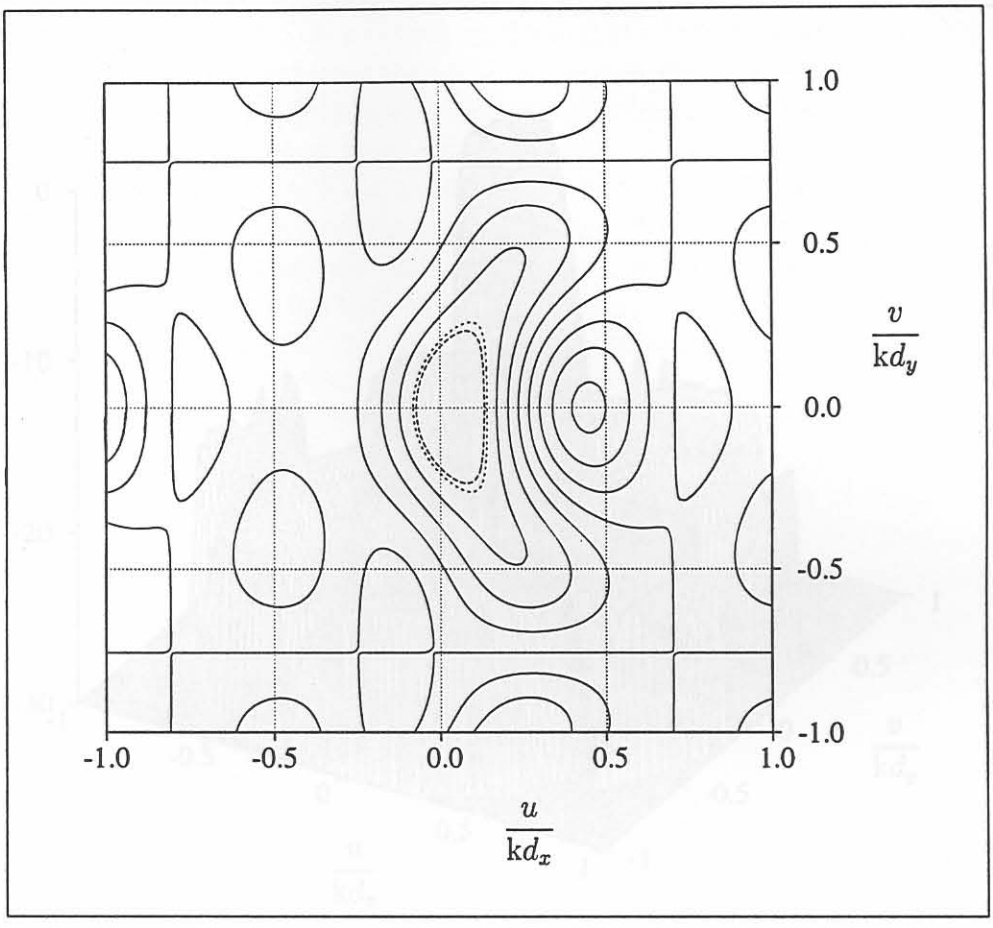

Figure 3.3: Illustrative Example #2: A contour plot of the transformation function. The dashed line represents the -1dB contour and the dotted line the -3dB contour

3.4.1 Transformation Grating Lobes

At this stage it is worthwhile to discuss the relationship between angle θ_p of the prototype linear array factor, and θ for any constant ϕ-cut of the planar array factor. These are of course related via the transformation function $H(u, v)$. In what follows we only consider the odd case; all the comments and conclusions hold for both the odd and even case. Figure 3.5 shows a plot of this relationship in, for example, the $\phi = 0°$ plane for four different transformations. Suffice it to say that two of these are $I = 1$ transformations and the other two $I = 2$ transformations, all having different values for their coefficients t_{ij} (note that the value of J is irrelevant as only the $\phi = 0°$ cut is of interest). Details of the t_{ij} values themselves are unimportant for the purposes of the point being made here. Note then that for both the $I = 1$ transformations, as θ (the "physical" angle) of the planar array moves from 0° out to 90° , we monotonically sweep through values of θ_p, and we will not move through the main beam of the prototype linear array (located around $\theta_p = 0°$) more than once. However, for the $I = 2$ transformations, we see that once the "physical" angle θ moves beyond a certain value the value of θ_p begins to decrease once more, and

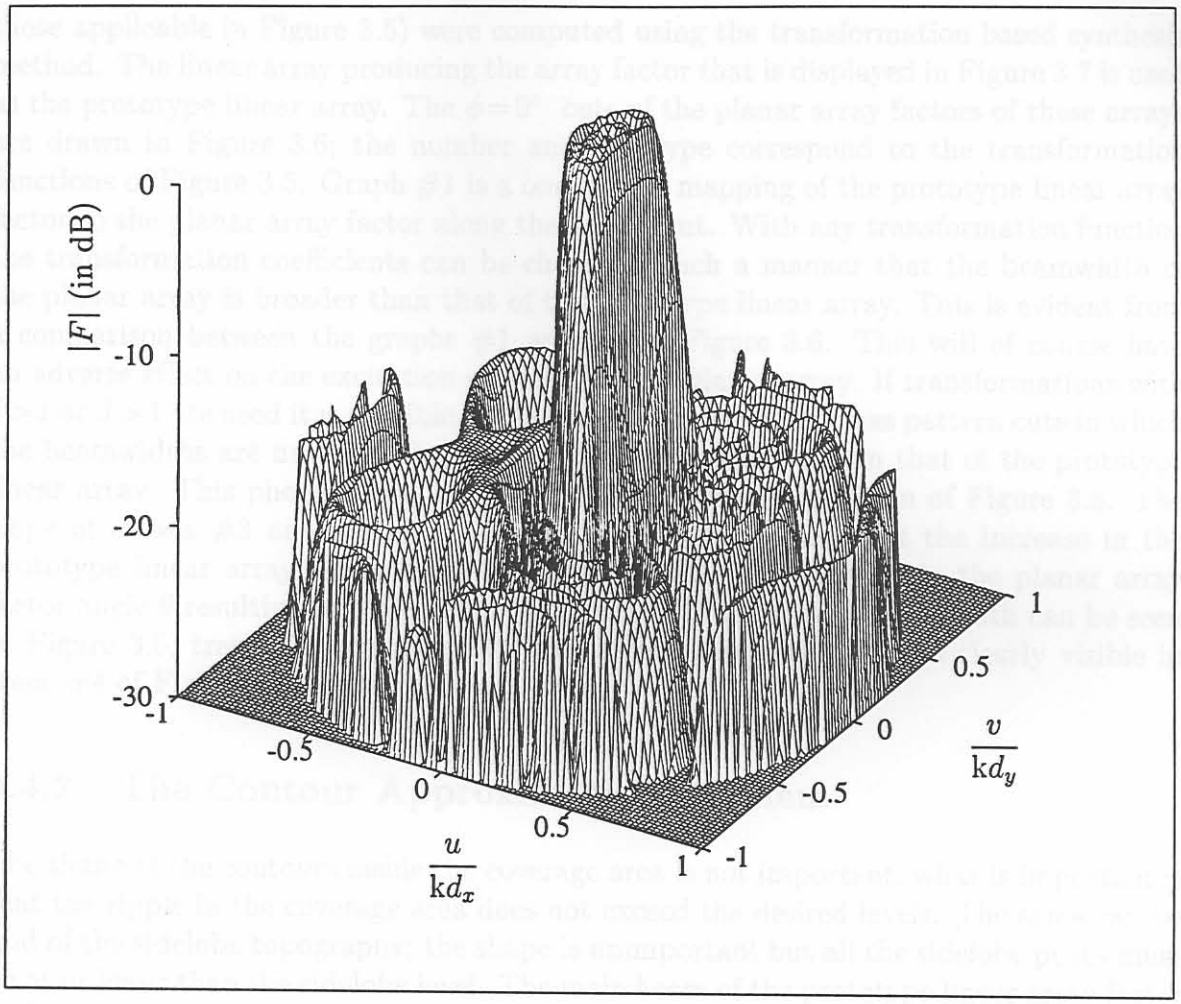

Figure 3.4: Illustrative Example #2: Surface plot of the planar array factor.

may move back into the main beam region of the prototype linear array, thus causing some of the sidelobes of the planar array outside the coverage area to be higher than the sidelobes designed in the prototype linear array. These might be called *transformation grating lobes.* The situation might be such that the $I=2$ transformation marked #3 in Figure 3.5 might be free of transformation grating lobes while #4 might not be (since at $\theta=90°$ the θ_p values of the two transformations differ by 18° , which may determine whether the prototype linear array main beam has been "entered" a second time or not). With contour transformations in which $I>1$ or $J>1$ one must thus ensure that such transformation grating lobes do not arise. These grating lobes are easily located, and can be avoided by imposing additional constraints, equations (3.27) and (3.28), on the transformation function. If the transformation grating lobe persists, it implies that a too narrow beamwidth is being attempted. More transformation coefficients or a prototype linear array with a narrower beamwidth must then be used to get rid of the transformation grating lobe.

Planar array excitations for the above-mentioned transformation coefficients (that is,

those applicable in Figure 3.5) were computed using the transformation based synthesis method. The linear array producing the array factor that is displayed in Figure 3.7 is used as the prototype linear array. The $\phi=0°$ cuts of the planar array factors of these arrays are drawn in Figure 3.6; the number and line type correspond to the transformation functions of Figure 3.5. Graph #1 is a one-on-one mapping of the prototype linear array factor to the planar array factor along the $\phi=0°$ cut. With any transformation function the transformation coefficients can be chosen in such a manner that the beamwidth of the planar array is broader than that of the prototype linear array. This is evident from a comparison between the graphs #1 and #2 in Figure 3.6. This will of course have an adverse effect on the excitation efficiency of the planar array. If transformations with $I>1$ or $J>1$ are used it is possible to obtain a planar array that has pattern cuts in which the beamwidths are narrower (in terms of the view-angle θ) from that of the prototype linear array. This phenomenon can be observed from consideration of Figure 3.5. The slope of curves #3 and #4 are greater than 45° ; indicating that the increase in the prototype linear array factor angle θ_p is greater than the increase in the planar array factor angle θ resulting in a narrower beamwidth. The narrower beamwidth can be seen in Figure 3.6, trace #3 and #4. The transformation grating lobe is clearly visible in trace #4 of Figure 3.6.

3.4.2 The Contour Approximation Problem

The shape of the contours inside the coverage area is not important, what is important is that the ripple in the coverage area does not exceed the desired levels. The same can be said of the sidelobe topography; the shape is unimportant but all the sidelobe peaks must be at or lower than the sidelobe level. The main beam of the prototype linear array factor will be mapped into the coverage area planar array factor. Thus, the prototype linear array factor must, in general, have a main beam with a specified maximum ripple and beamwidth, and sidelobes lower than the required sidelobe level. The transition region is the roll-off region between the main beam region and the sidelobe region. Figure 3.7 displays a typical prototype linear array factor. The end of the main beam region is denoted by α; this is the pattern angle where the prototype linear array factor main beam falls below the ripple specification. The quantity β is the pattern angle where the main beam has fallen to the sidelobe level.

Assume that we want to firmly control the contour $u=f_u(\theta,\phi)$ and $v=f_v(\theta,\phi)$, and that this firmly controlled contour must map to the point $\psi=\psi_0$ (usually the -3dB point).

$$\begin{aligned} \text{Odd case:} \quad & \cos(\psi_0) = H[f_u(\theta,\phi), f_v(\theta,\phi)] \\ \text{Even case:} \quad & \cos(\tfrac{1}{2}\psi_0) = H[f_u(\theta,\phi), f_v(\theta,\phi)] \end{aligned} \tag{3.26}$$

In some methods of determining the transformation function coefficients this will be the only consideration, no additional constraints are enforced; scaling may be necessary for the coefficients obtained with such methods. However, additional constraints may be imposed, which means some optimisation method must be used to calculate the transfor-

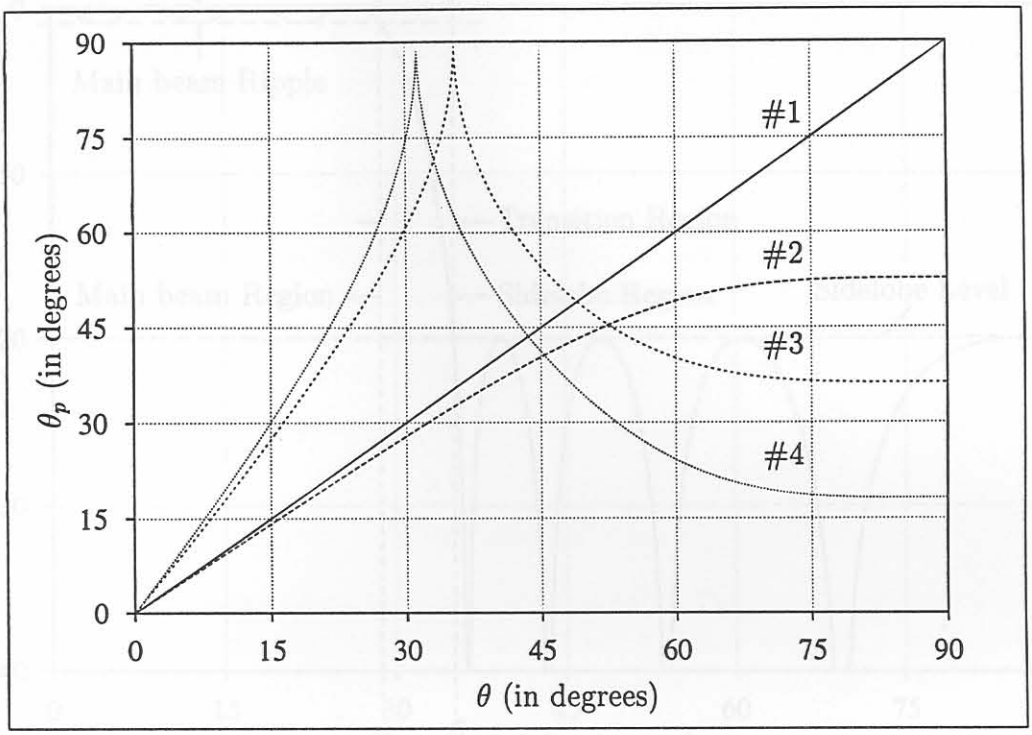

Figure 3.5: Transformation grating lobes: θ_p versus θ.

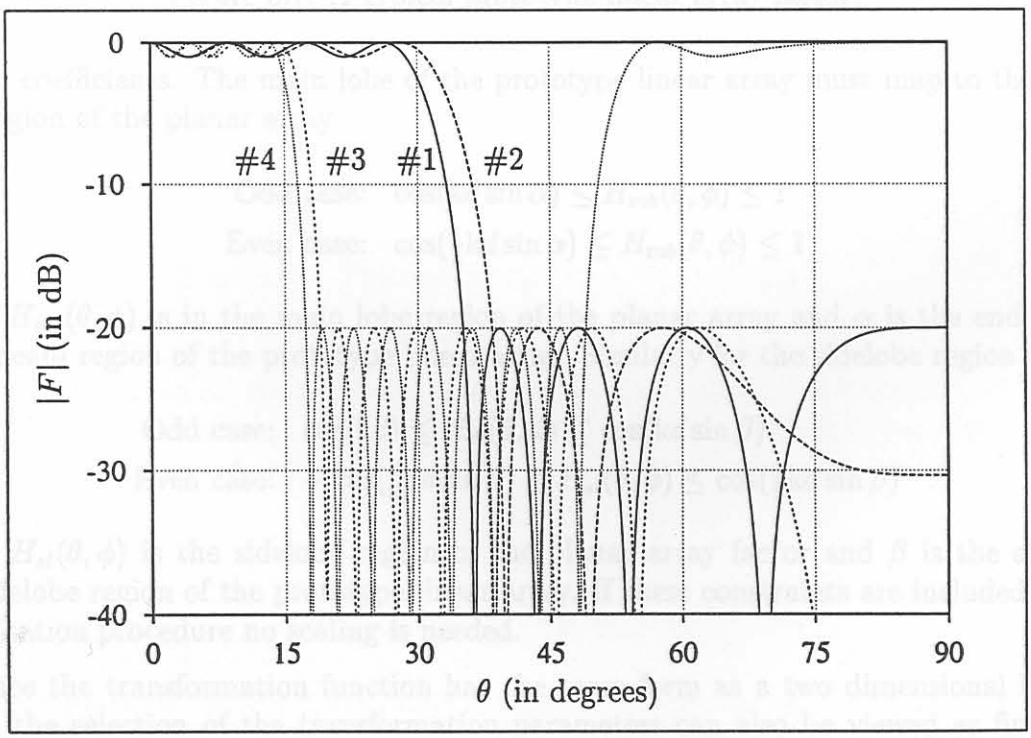

Figure 3.6: Transformation grating lobes: Planar array factors, ϕ=0° cuts only.

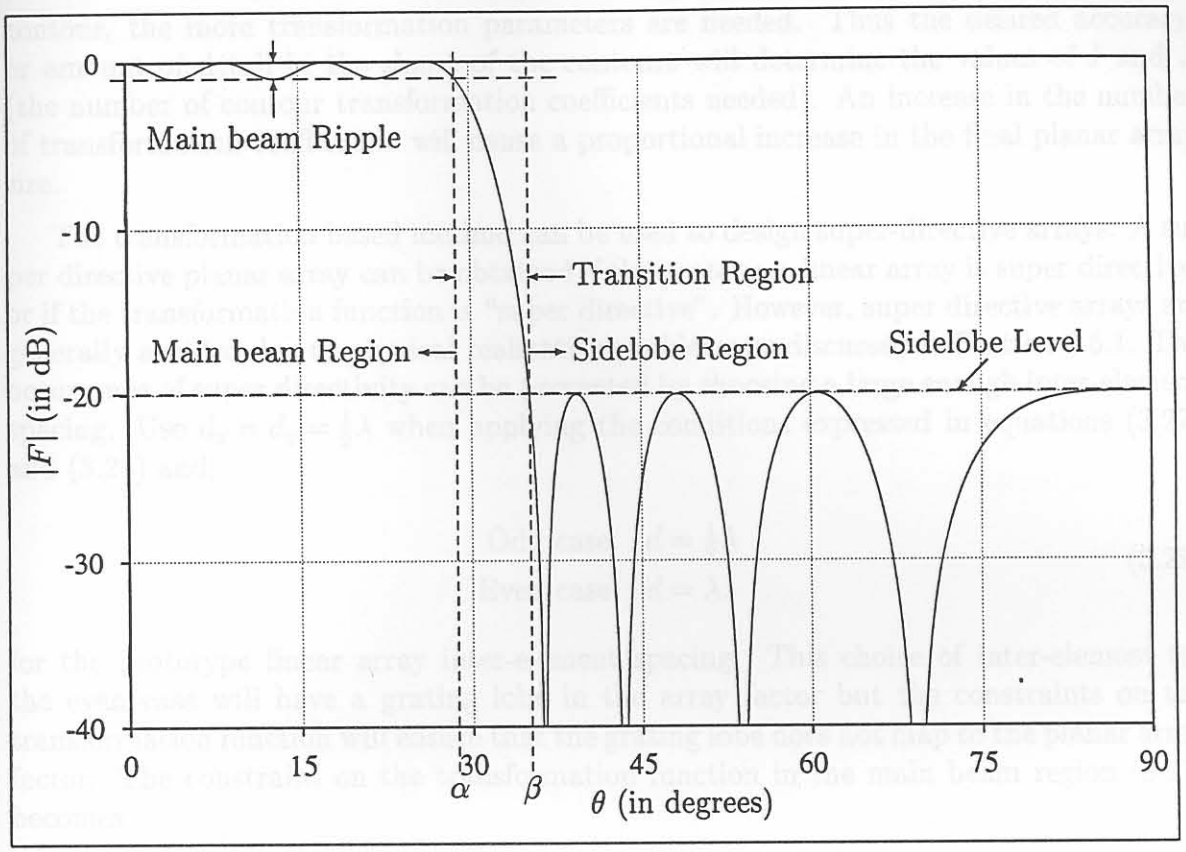

Figure 3.7: A typical prototype linear array factor.

mation coefficients. The main lobe of the prototype linear array must map to the main lobe region of the planar array

$$\begin{aligned} \text{Odd case:} \quad & \cos(\mathrm{k}d\sin\alpha) \leq H_{mb}(\theta,\phi) \leq 1 \\ \text{Even case:} \quad & \cos(\tfrac{1}{2}\mathrm{k}d\sin\alpha) \leq H_{mb}(\theta,\phi) \leq 1 \end{aligned} \tag{3.27}$$

where $H_{mb}(\theta,\phi)$ is in the main lobe region of the planar array and α is the end of the main beam region of the prototype linear array. Similarly for the sidelobe region

$$\begin{aligned} \text{Odd case:} \quad & \cos(\mathrm{k}d) \leq H_{sl}(\theta,\phi) \leq \cos(\mathrm{k}d\sin\beta) \\ \text{Even case:} \quad & -\cos(\tfrac{1}{2}\mathrm{k}d\sin\beta) \leq H_{sl}(\theta,\phi) \leq \cos(\tfrac{1}{2}\mathrm{k}d\sin\beta) \end{aligned} \tag{3.28}$$

where $H_{sl}(\theta,\phi)$ is the sidelobe region of the planar array factor and β is the start of the sidelobe region of the prototype linear array. If these constraints are included in the optimisation procedure no scaling is needed.

Since the transformation function has the same form as a two dimensional Fourier series, the selection of the transformation parameters can also be viewed as finding a two-dimensional Fourier transformation of the strictly specified contour. From Fourier analysis it then follows that the narrower the beamwidth, and the more detailed the

contour, the more transformation parameters are needed. Thus the desired accuracy, or amount of detail in the shape of the contours will determine the values of I and J (the number of contour transformation coefficients needed). An increase in the number of transformation coefficients will cause a proportional increase in the final planar array size.

The transformation based method can be used to design super-directive arrays. A super directive planar array can be obtained if the prototype linear array is super directive, or if the transformation function is "super directive". However, super directive arrays are generally avoided due to physical realization problems as discussed in Section 2.5.1. The occurrence of super directivity can be prevented by choosing a large enough inter-element spacing. Use $d_x = d_y = \frac{1}{2}\lambda$ when applying the conditions expressed in equations (3.27) and (3.28) and;

$$\begin{aligned} \text{Odd case:} \quad & d = \tfrac{1}{2}\lambda \\ \text{Even case:} \quad & d = \lambda. \end{aligned} \tag{3.29}$$

for the prototype linear array inter-element spacing. This choice of inter-element for the even case will have a grating lobe in the array factor but the constraints on the transformation function will ensure that the grating lobe does not map to the planar array factor. The constraint on the transformation function in the main beam region (3.27) becomes

$$\cos(\pi \sin\alpha) \leq H_{mb}(\theta, \phi) \leq 1 \tag{3.30}$$

for both the odd and even case; and the constraint in the sidelobe region (3.28) becomes

$$\begin{aligned} \text{Odd case:} \quad & -1 \leq H_{sl}(\theta, \phi) \leq \cos(\pi \sin\beta) \\ \text{Even case:} \quad & -\cos(\pi \sin\beta) \leq H_{sl}(\theta, \phi) \leq \cos(\pi \sin\beta). \end{aligned} \tag{3.31}$$

3.4.3 Scaling of the Transformation Function

Linear scaling of the transformation function is only applicable to the odd case. Visible space is the region where $0 \leq \theta \leq \frac{1}{2}\pi$ and $0 \leq \phi \leq 2\pi$. From the definition of u and v in (2.14) an elliptical disc, with major axis $\mathrm{k}d_x$ and minor axis $\mathrm{k}d_y$, constitutes visible space in the uv-plane [18]. Since $|\cos x| \leq 1$, the following constraint must be imposed on $H(u, v)$:

$$|H(u,v)| \leq 1 \quad \text{for} \quad \sqrt{\left(\frac{u}{\mathrm{k}d_x}\right)^2 + \left(\frac{v}{\mathrm{k}d_y}\right)^2} \leq 1 \tag{3.32}$$

To avoid super directivity $d_x = d_y = \frac{1}{2}\lambda$ is used when applying this condition.

The transformation parameters are found using an optimisation algorithm, approximation, or some other technique, to fit the desired shape of the target contour. The

solution, however, usually does not satisfy the condition in (3.32) and scaling is therefore necessary. The following linear scaling scheme can be employed [124]

$$H'(u,v) = C_1\, H(u,v) - C_2 \tag{3.33}$$

where

$$C_1 = \frac{2}{H_{max} - H_{min}} \tag{3.34a}$$

$$C_2 = C_1\, H_{max} - 1 \tag{3.34b}$$

and H_{max} and H_{min} are the maximum and minimum values of $H(u,v)$ which usually have to be found numerically. Obviously this scaling does not change the shape of the contours defined by $H(u,v) = constant$;, only the original ψ_p value associated with a specific contour changes to

$$\psi'_p = \arccos(C_1 \cos\psi_p - C_2) \tag{3.35}$$

Since the transformation function for the even case does not have a t_{00} coefficient, the linear scaling technique cannot be applied. For the even case the transformation function must be normalised or the constraint (3.32) must be included and enforced in the determination of the transformation coefficients. Normalisation can be done by

$$H'(u,v) = C\, H(u,v) \tag{3.36}$$

where $C = |H(u,v)|_{max}$.

3.4.4 Transformation Function Synthesis by Approximation

If the desired contour can be written in an analytical form, for example an elliptical contour,

$$\frac{u^2}{a^2} + \frac{v^2}{b^2} = 1 \tag{3.37}$$

where a and b are the major and minor axes of the elliptical contour, then approximation can be used to determine the values of the transformation coefficients. The Taylor series for the trigonometric functions, sin and cos, are

$$\begin{aligned} \cos x &= 1 - \tfrac{x^2}{2!} + \tfrac{x^4}{4!} \cdots \\ \sin x &= x - \tfrac{x^3}{3!} + \tfrac{x^5}{5!} \cdots \end{aligned} \tag{3.38}$$

With small values of x the series can be truncated. These truncated or approximated series are then substituted into the transformation function, and the powers of u and v are set to the values of the corresponding powers in the function describing the contour.

An alternative approach is to select a number of cuts and ensure that the desired contour is at the correct position in these cuts. If the total number of adjustable transformation coefficients is K, one can choose $K-1$ ϕ-cuts and set up $K-1$ equations,

$$\begin{aligned} \text{Odd case:} \quad & H(\theta_k, \phi_k) = \cos(\psi_0) \\ \text{Even case:} \quad & H(\theta_k, \phi_k) = \cos(\tfrac{1}{2}\psi_0) \end{aligned} \tag{3.39}$$

using

$$H(0,0) = 1 \tag{3.40}$$

as the Kth equation, and then simple matrix inversion can be used to determine the transformation coefficients.

The approximation methods work well if only a few transformation function coefficients are used and the desired contours are relatively simple, smooth functions such as circles or ellipses. Thus they are of little use in synthesising planar arrays with arbitrarily contoured array factors. References [95, 92] contain more information on these methods.

3.4.5 Transformation Function Synthesis by Constrained Optimisation

The problem can be formulated as a linear approximation problem. The error function can be defined as

$$\begin{aligned} \text{Odd case:} \quad & \varepsilon = \cos(\psi_0) - H[u, f(u)] \\ \text{Even case:} \quad & \varepsilon = \cos(\tfrac{1}{2}\psi_0) - H[u, f(u)] \end{aligned} \tag{3.41}$$

with $v = f(u)$. Applying the constraint presented in equation (3.32), one variable can be written as a function of the others, leaving one less parameter to be optimised. Standard linear optimisation routines can be used for the optimisation. The transformation parameters obtained with these methods will usually require scaling.

Another procedure for the design of contours of any shape will next be described. The algorithm can design one or more contours simultaneously and it can be used with transformations of any I and J values. We explore the procedure for a one contour mapping. Assume the requirement is the mapping of the point $\psi = \psi_0$ to a contour $u = f_u(\theta, \phi)$ and $v = f_v(\theta, \phi)$. A set of error equations can then be formulated on some set of discrete samples, as

$$\begin{aligned} \text{Odd case:} \quad & \varepsilon(\theta_k, \phi_k) = \cos(\psi_0) - H[f_u(\theta_k, \phi_k), f_v(\theta_k, \phi_k)] \\ \text{Even case:} \quad & \varepsilon(\theta_k, \phi_k) = \cos(\tfrac{1}{2}\psi_0) - H[f_u(\theta_k, \phi_k), f_v(\theta_k, \phi_k)] \end{aligned} \tag{3.42}$$

We must then minimise

$$E_1 = \max |\varepsilon(\theta_k, \phi_k)| \tag{3.43}$$

or

$$E_2 = \sum_k \varepsilon^2(\theta_k, \phi_k) \tag{3.44}$$

subject to equality constraints of the form

$$H(\theta_k, \phi_k) = \alpha \tag{3.45}$$

and inequality constraints of the form

$$\begin{aligned} \text{Odd case:} \quad & -1 \leq H(\theta_k, \phi_k) \leq \beta \\ \text{Even case:} \quad & -\beta \leq H(\theta_k, \phi_k) \leq \beta \end{aligned} \tag{3.46}$$

The quantity E_1 can be minimised by linear programming, or if the inequality constraints (3.46) are not present E_2 can be optimised using least squares techniques. The advantage of this approach is that no analytical function for the contour shape is needed.

The transformation design problem can also be stated in the same form as the design problem for two-dimensional equi-ripple digital filters. The transformation coefficients can be obtained from the impulse response coefficients of a digital filter. Fast and efficient algorithms exist [125, 126] to determine the impulse response of such small digital filters. A more complete discussion of these methods is provided in reference [92].

3.4.6 The Transformation Function as a Planar Array

The transformation function has the same form as a planar array factor. Since the transformation function is a real function the planar array factor must also be pure real, thus the determination of the transformation coefficients is equivalent to a "field synthesis" problem. Although it may seem that we end up where we started, it must be noted that the number of transformation coefficients is only a fraction of the number of final planar array elements. Also, field synthesis is generally simpler than power synthesis [111]. The excitations of the planar array $F_H(u, v)$ formed by the transformation function are

$$\begin{aligned} \mathbf{a}_{ij} &= \tfrac{1}{4}\left[t_{ij}^{cc} - t_{ij}^{ss} - \mathrm{j}(t_{ij}^{cs} + t_{ij}^{sc})\right] \\ \mathbf{a}_{i-j} &= \tfrac{1}{4}\left[t_{ij}^{cc} + t_{ij}^{ss} + \mathrm{j}(t_{ij}^{cs} - t_{ij}^{sc})\right] \\ \mathbf{a}_{-ij} &= \tfrac{1}{4}\left[t_{ij}^{cc} + t_{ij}^{ss} - \mathrm{j}(t_{ij}^{cs} - t_{ij}^{sc})\right] \\ \mathbf{a}_{-i-j} &= \tfrac{1}{4}\left[t_{ij}^{cc} - t_{ij}^{ss} + \mathrm{j}(t_{ij}^{cs} + t_{ij}^{sc})\right] \end{aligned} \tag{3.47}$$

From expression (3.47) it is clear that $\mathbf{a}_{ij} = \mathbf{a}^*_{-i-j}$ and $\mathbf{a}_{i-j} = \mathbf{a}^*_{-ij}$. This will always result in a pure real array factor. Any of the planar array synthesis techniques discussed in Chapter 2 can be used to determine the transformation coefficients, but field synthesis

methods will work best, as the phase of the radiation pattern is known. Since the "array factor" of the transformation function "array" is pure real the generalised projection method (Section 2.8.4) is ideally suited to determine the transformation coefficients; as the sets involved in field synthesis are convex, convergence is guaranteed. In general scaling will be necessary for transformation function parameters computed in this manner.

3.5 Planar Arrays with Non-Rectangular Boundaries and/or Lattices

The transformation based synthesis technique described in Sections3.2 and 3.3 can also be used to synthesise planar arrays with non-rectangular boundaries and/or lattices if we purposefully set specific transformation coefficients equal to zero. We demonstrate this in the present section.

Firstly, the synthesis of planar arrays with rectangular grids but non-rectangular boundaries (say octagonal or near-circular boundaries) will be discussed in Section 3.5.1. Secondly, the synthesis of planar arrays with non-rectangular lattices (for instance hexagonal arrays) is addressed in Section 3.5.2.

3.5.1 Arrays with Rectangular Lattices but Non-Rectangular Boundaries

The transformation function has the same form as a small planar array factor, thus the transformation coefficients can be graphically represented in a fashion similar to planar array excitations. The transformation function must have the same "shape" as the final planar array; for an array with an elliptical boundary the transformation "array" must also have an elliptical "boundary". Two examples are shown in Figure 3.8; the circles represent the transformation coefficients that must be set to zero and the dots represent non-zero, adjustable coefficients. The first example (Figure 3.8(a)) is the transformation needed for an octagonally shaped planar array, while the second (Figure 3.8(b)) is for a planar array with an elliptical boundary.

The synthesis method follows the steps outlined in the illustrative example #1 in Section 3.2.2, except for the first step, which is expanded to take the boundary of the array into account. Not only must the number of transformation coefficients (I and J) be determined, but also which of the transformation coefficients must be set to zero to obtain the correct boundary shape.

Illustrative Example #3

Let us consider the design requirement to be a planar array with octagonal boundary. A circular footprint contour at -3dB is desired.

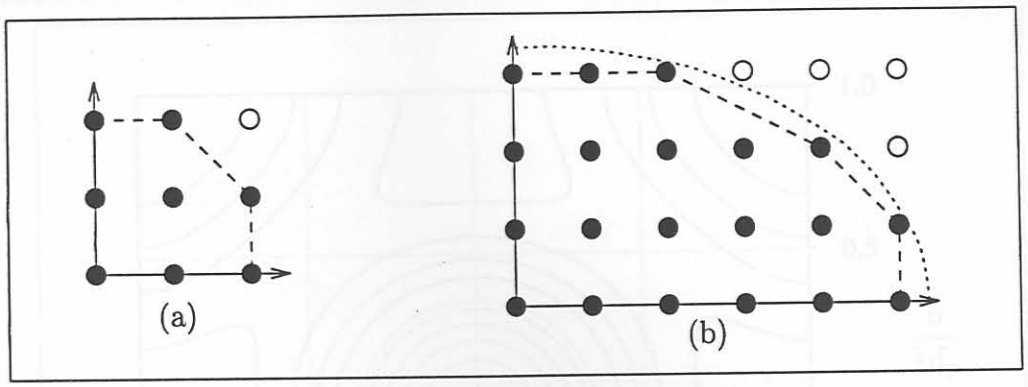

Figure 3.8: Graphical representation of the transformation coefficients for arrays with non-rectangular boundaries, the circles representing zero coefficients. (a) Rectangular grid with an octagonal boundary. (b) Rectangular grid and a near-elliptical boundary. The dashed line indicates the boundary of the final planar array.

Coefficient	Value	Coefficient	Value
t_{00}	-.549205	t_{11}	.739280
$t_{01}=t_{10}$	.129086	$t_{12}=t_{21}$	.129086
$t_{02}=t_{20}$	.146790	t_{22}	0

Table 3.3: Illustrative Example #3: Transformation coefficients.

1. An array with a rectangular grid and an octagonal boundary can be achieved by a transformation with $I = J = 2$, but $t_{22}^{xx} = 0$ in order to obtain the octagonal boundary. The transformation function is shown graphically in Figure 3.8(a), the solid line shows the array boundary shape (only one quadrant is depicted).

2. The number of elements used for the prototype linear array is 15; or $Q=7$.

3. The prototype linear array was synthesised using the method of Orchard, Elliot and Stern [54]. Six roots were placed off the unit circle. The main beam peak-to-peak ripple is 1dB and the sidelobe level is -20dB. The inter-element spacing used in this step is $d=\frac{1}{2}\lambda$. A -3dB beamwidth of 71° was obtained.

4. Due to the octagonal symmetry of the required contour shape only five unique transformation coefficients are available to control the contour shape. The transformation coefficients were computed using the approximation-by-cuts approach discussed in the latter half of Section 3.4.4. The coefficient values are listed in Table 3.3.

5. The algorithm in Section A.1.3 in Appendix A was used to compute the planar array excitation. Although $M=N=14$, suggesting an array of 729 elements, only 617 elements are excited.

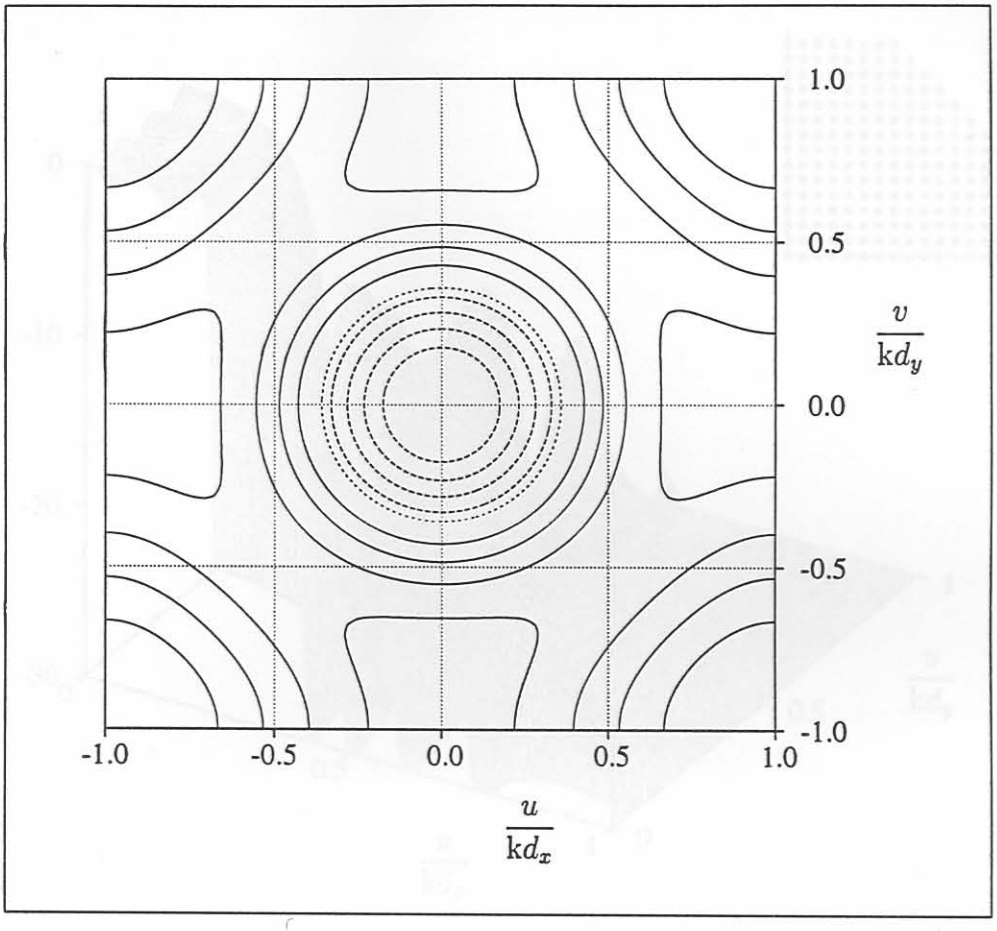

Figure 3.9: Illustrative Example #3: A contour plot of the transformation function. The dotted line represents the -3dB contour, the dashed lines contours in the main beam region and solid lines contours in the sidelobe region.

A contour plot of the transformation function is displayed in Figure 3.9. The -3dB beamwidth contour of the planar array factor, represented by the dotted line in Figure 3.9, is at $\theta = 21°$. A surface plot of the planar array factor is depicted in Figure 3.10. The array geometry is shown in the top right hand corner of Figure 3.10.

3.5.2 Arrays with Triangular Lattices and Boundaries

In general, one can view a triangular grid array as a rectangular grid array with only those elements coinciding on the triangular grid excited. If one chooses the x-axis on one of the triangular lattice axes one can construct an equivalent rectangular grid array, as shown in Figure 3.11. The intersections of the dotted lines with filled circles indicate the positions of excited elements, while the unfilled circles indicate the positions of zero elements. The filled squares represent the collapsed distributions along the principal axes.

First we consider the rows of the array. The inter-element spacing of the triangular

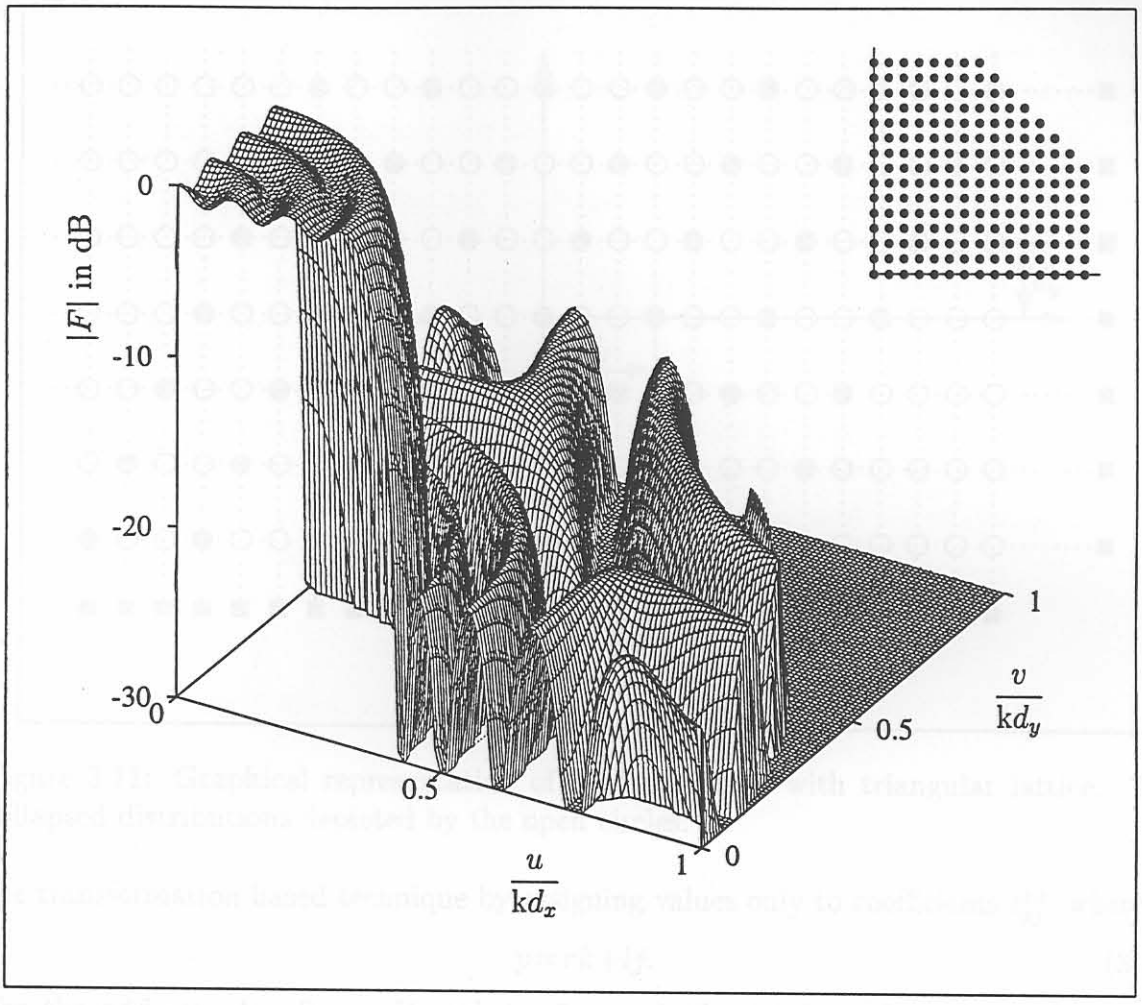

Figure 3.10: Illustrative Example #3: Surface plot of the planar array factor, with the array geometry depicted in the upper right hand corner.

grid array in the $\hat{x}$ direction is denoted by d_t, and the inter-element spacing of the collapsed distribution in the $\hat{x}$ direction by d_x. The inter-element spacings are indicated in Figure 3.11. If the ratio between d_t and d_x is an integer the triangular grid array can be synthesised with the transformation based synthesis method. Let us assume this to be the case and denote the ratio $r = d_t/d_x$. Only the r-th element in each row of the equivalent rectangular grid array will be excited. In addition, the selected elements have a horizontal shift from one row to the next. This shift will be an integer of units d_x, and denoted by k. In the case shown in Figure 3.11 every third element in each row of the rectangular grid is excited, thus $r=3$; and the horizontal shift is $l=1$. The other principal plane is less complicated; the inter-element spacing d_y of the collapsed distribution in the $\hat{y}$ direction is simply the distance between the rows of the triangular grid.

The transformation function coefficients must have the same "shape" as the unit cell of the planar array. Since only the elements at positions $((rm+ln)d_x, nd_y)$ of the equivalent rectangular lattice are excited, the triangular array can be synthesised with

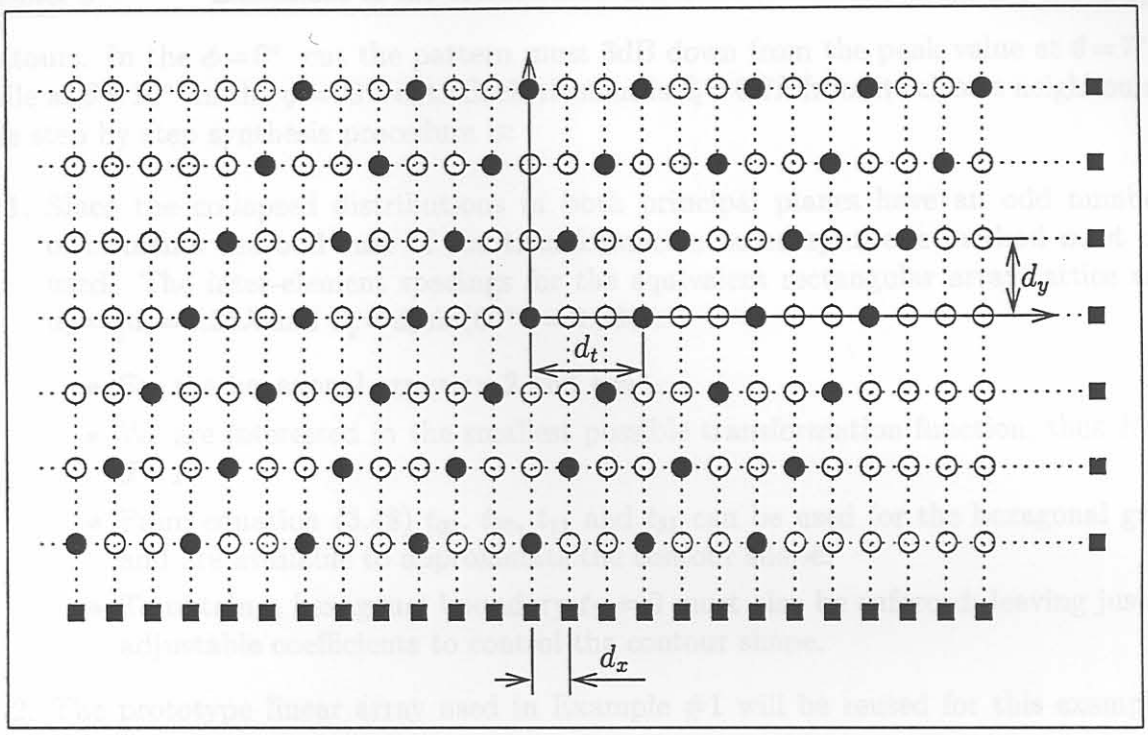

Figure 3.11: Graphical representation of a planar array with triangular lattice. The collapsed distributions depicted by the open circles.

the transformation based technique by assigning values only to coefficients t_{pj}^{xx}, where

$$p=rk+lj. \tag{3.48}$$

For the odd case $k=0,\ldots,K$ and $j=0,\ldots,J$; the total number of transformation coefficients is $(I+1)\times(J+1)$ of which $(K+1)\times(J+1)$ will be non-zero. In the even case $k=1,\ldots,K$ and $j=1,\ldots,J$; from a total of $I\times J$ transformation coefficients $K\times J$ are non-zero. For both the odd and the even case

$$I=rK+lJ \tag{3.49}$$

In order to synthesise triangular arrays yet another extension to the first step in the synthesis procedure must be made. First, r and l must be determined from the geometry of the array considered. The number of coefficients K and J are then selected. From these values the total number of transformation coefficients can be computed, as well as which coefficients are non-zero. Lastly, in order to obtain the correct array boundary some of the non-zero transformation coefficients must also be set to zero. The synthesis method is best illustrated by an example.

Illustrative Example #4

A typical example of an array with triangular lattice is a hexagonal array. Consider the design of a 10 ring hexagonal planar array with a requirement of elliptical footprint

contours. In the $\phi=0°$ cut the pattern must 3dB down from the peak value at $\theta=7°$, while at $\theta=10°$ in the $\phi=90°$ cut. Each element is $d_t=0.7\lambda$ from its closets neighbours. The step by step synthesis procedure is:

1. Since the collapsed distributions in both principal planes have an odd number of elements the odd case of the transformation based synthesis method must be used. The inter-element spacings for the equivalent rectangular array lattice are $d_x=\frac{1}{2}d_t=0.35\lambda$ and $d_y=d_t\sin(60°)=0.606\lambda$.
   - For the hexagonal array $r=2$ and $k=1$.
   - We are interested in the smallest possible transformation function, thus $K=J=1$.
   - From equation (3.48) t_{00}, t_{20}, t_{11} and t_{31} can be used for the hexagonal grid and are available to approximate the contour shape.
   - To obtain a hexagonal boundary $t_{31}=0$ must also be enforced; leaving just 3 adjustable coefficients to control the contour shape.
2. The prototype linear array used in Example #1 will be reused for this example, $Q=10$.
3. The same prototype linear array excitation as in Example #1 is used.
4. The approximation-by-cuts method described in Section 3.4.4 can be used to obtain the contour transformation coefficients. Only two cuts, $\phi=0°$ and $\phi=90°$, are needed. The coefficient values determined in this manner are $t_{00}=-0.208559$, $t_{11}=0.537601$ and $t_{20}=0.670958$. No scaling is needed, even if $d_x=\frac{1}{2}\lambda$.
5. The planar array distribution can now be obtained using the algorithms in Section A.1.2 of Appendix A

Figure 3.12 displays a contour plot of the transformation function. The transformation function coefficients for the hexagonal array are graphically presented in the upper right hand corner of the figure. The filled circles represent adjustable coefficients and the non-filled circles the "off-grid" zero coefficients. The crossed circle represents the "on-grid" transformation parameter that must be set to zero to obtain a hexagonal boundary. The solid line shows the final array boundary shape. The array factor is shown in Figure 3.13.

3.6 Representative Example: An Africa-Shaped Contour Footprint Pattern

Let the requirement for this example be a strictly controlled contour in the shape of Africa as seen from a geo-stationary position above the equator. The geometry of the array is a square lattice ($d_x=d_y=0.72\lambda$) with the corner elements lopped off to fit in a circular boundary.

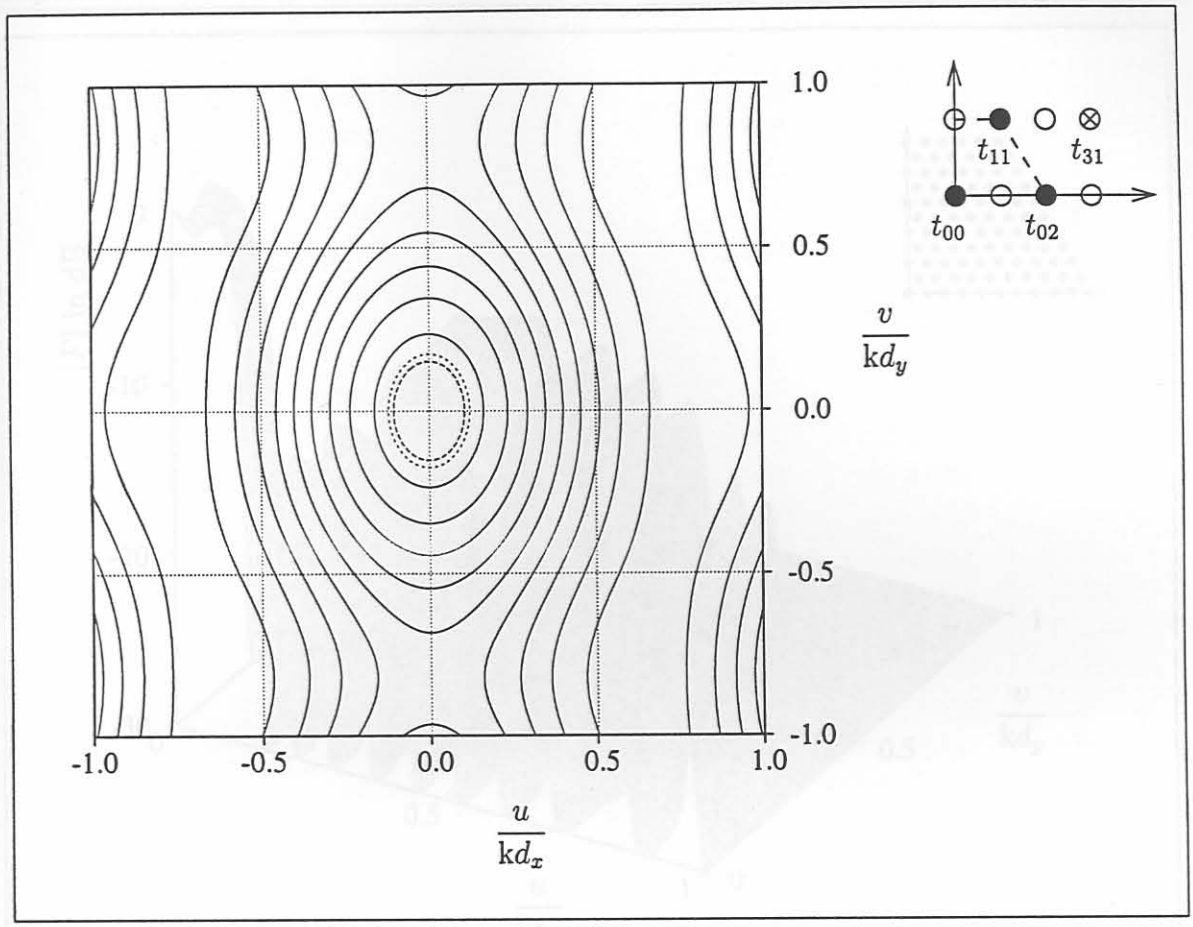

Figure 3.12: Illustrative Example #4: A contour plot of the transformation function, with the transformation function depicted on the right. The dashed line represents the -1dB contour and the dotted line the -3dB contour.

1. A transformation with $I=J=5$ gives enough flexibility to approximate the contour of the African continent adequately. To get a near circular boundary $t_{45}^{xx}=t_{54}^{xx}=t_{55}^{xx}$.

2. The prototype linear array consists of 21 ($Q=10$) elements.

3. The prototype linear array is synthesised using the method of Orchard, Elliott and Stern [53]. The sidelobe level was set at -30dB, and the main beam ripple at 1.0dB. To obtain the narrow beamwidth two zeros had to be moved from the Schelkunoff unit circle.

4. A total of 109 transformation coefficients are available to control the contour shape. The values of the transformation coefficients were obtained using the general projection planar array synthesis method with weighted least squares as the back projector (this method will be discussed in Section 5.2.3). The projection method is ideally suited for determination of the coefficients as the transformation function is pure real, and the sets involved are convex. The transformation parameters are listed in Table 3.4. A contour plot of the transformation function is shown in Figure 3.14.

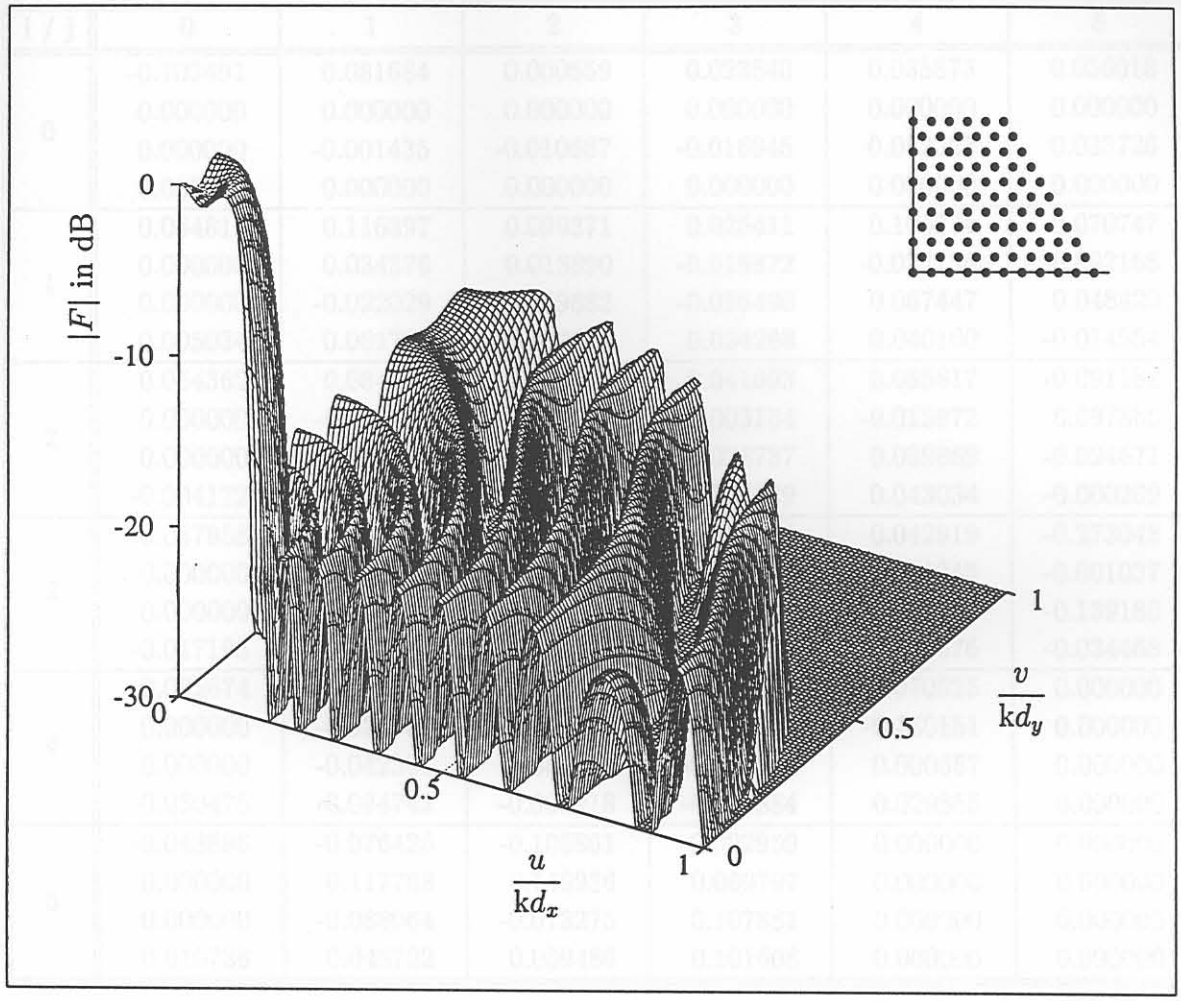

Figure 3.13: Illustrative Example #4: Surface plot of the planar array factor (with the array geometry depicted in the upper right corner).

5. The planar array excitations were then computed using the information forthcoming from the solutions of the two sub-problems.

The planar array distribution is graphically shown in Figure 3.15; one can see that the excitations decrease from the centre outward; this is inherent to the transformation based method. The excitations near the edge of the array are dependent on a few terms of $H^q(u,v)$ with q large. Since the transformation coefficients are generally smaller than one, the excitations at these far-out positions will be small. These small elements far from the origin of the array, with a magnitude of less than 1% the magnitude of the largest excitation, may be removed without seriously degrading the radiation pattern. This will result in an array with a circular boundary of diameter of 64λ and 6293 elements. With all the elements of magnitude less than 1% (or -40dB) of the largest excitation removed, only 0.13% (or -29dB) of the total power of the synthesised array will be lost. This array has 3822 non-zero elements in a circular boundary. The main beam contours are not

i / j	0	1	2	3	4	5
0	-0.103491	0.081684	0.000659	0.023540	0.035873	0.056018
	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
	0.000000	-0.001435	-0.010687	-0.016845	0.062322	0.033726
	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	0.084819	0.116397	0.099371	0.025411	0.106774	0.070747
	0.000000	0.034376	0.013890	-0.019872	-0.024756	0.092168
	0.000000	-0.022029	-0.009882	-0.036490	0.067447	0.048420
	0.005034	0.001793	0.002313	0.034268	0.040160	-0.014554
2	0.054362	0.084796	0.095682	0.041693	0.085817	-0.091182
	0.000000	-0.016720	-0.034881	0.003184	-0.015972	0.097866
	0.000000	-0.081815	-0.043515	0.028737	0.039869	-0.024671
	-0.004172	-0.021291	0.008764	-0.006829	0.043034	-0.000269
3	0.037958	0.038334	0.105628	0.102171	0.042919	-0.273048
	0.000000	-0.030287	-0.027655	-0.019976	-0.025345	-0.001037
	0.000000	-0.037993	-0.040805	0.014134	0.072372	-0.139180
	-0.017198	-0.034351	0.007507	0.016134	-0.008676	-0.034468
4	0.092674	0.190139	0.107466	0.005586	-0.070625	0.000000
	0.000000	-0.024303	-0.079479	-0.098116	-0.140151	0.000000
	0.000000	-0.042361	-0.060640	-0.036315	0.000667	0.000000
	-0.030475	-0.094742	-0.080918	-0.062884	0.029855	0.000000
5	-0.043896	-0.076426	-0.105861	-0.022959	0.000000	0.000000
	0.000000	0.117768	0.149936	0.069797	0.000000	0.000000
	0.000000	-0.088064	-0.073275	0.107881	0.000000	0.000000
	0.010736	0.048722	0.059486	0.101608	0.000000	0.000000

Table 3.4: Representative Example: The transformation function coefficients. The four elements in each cell of the table are t_{ij}^{cc} t_{ij}^{ss} t_{ij}^{cs} and t_{ij}^{sc} , in that order

noticeably changed and all the sidelobes are below a level of -24.5dB (all but one sidelobe is below -27.5dB). The planar array factor is depicted in Figures 3.16 and 3.17 (a surface and a contour plot of only the inner part of the array factor). The highest sidelobe peak is at $(u, v) = (0.193, 0.167)$ and can be seen in these two figures.

3.7 Comparison with Similar Methods

It is instructive to establish the relationship between transformation function and the different transformations that have been proposed to synthesise planar arrays. These transformations were briefly discussed in Section 2.7.2. It will be shown in Sections 3.7.1 to 3.7.4 that all these transformations are special cases of the transformation based synthesis method proposed in this chapter. In addition, some of the limitations suffered by these transformations can be overcome by the transformation based synthesis technique.

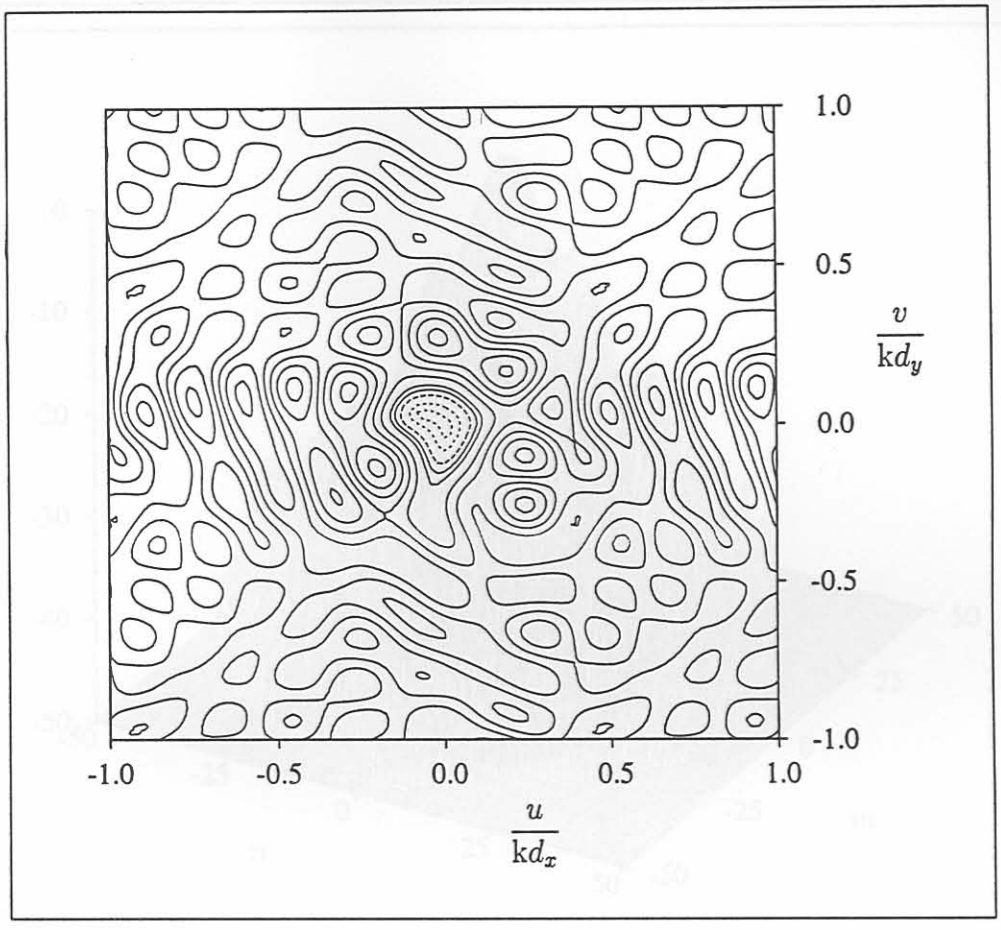

Figure 3.14: Representative Example: A contour of the transformation function. The dashed line is the strictly controled contour, solid lines are contours in the sidelobe region and the dotted lines are contours in the main beam region.

The similarities between the convolution synthesis method (which were mentioned in Section 2.7.3) and the transformation based synthesis technique will be discussed in Section 3.7.5.

3.7.1 The Baklanov Transformation

The use of a transformation to map a linear array excitation to a planar array excitation was first proposed by Baklanov [82] in 1966. Tseng and Cheng [81] substituted the Baklanov transformation into a Chebyshev polynomial to synthesise scanable square planar arrays with Chebychev patterns in every ϕ-cut. They considered arrays with an odd as well as an even number of elements.

The Baklanov transformation, here written in terms of the u and v as defined in this

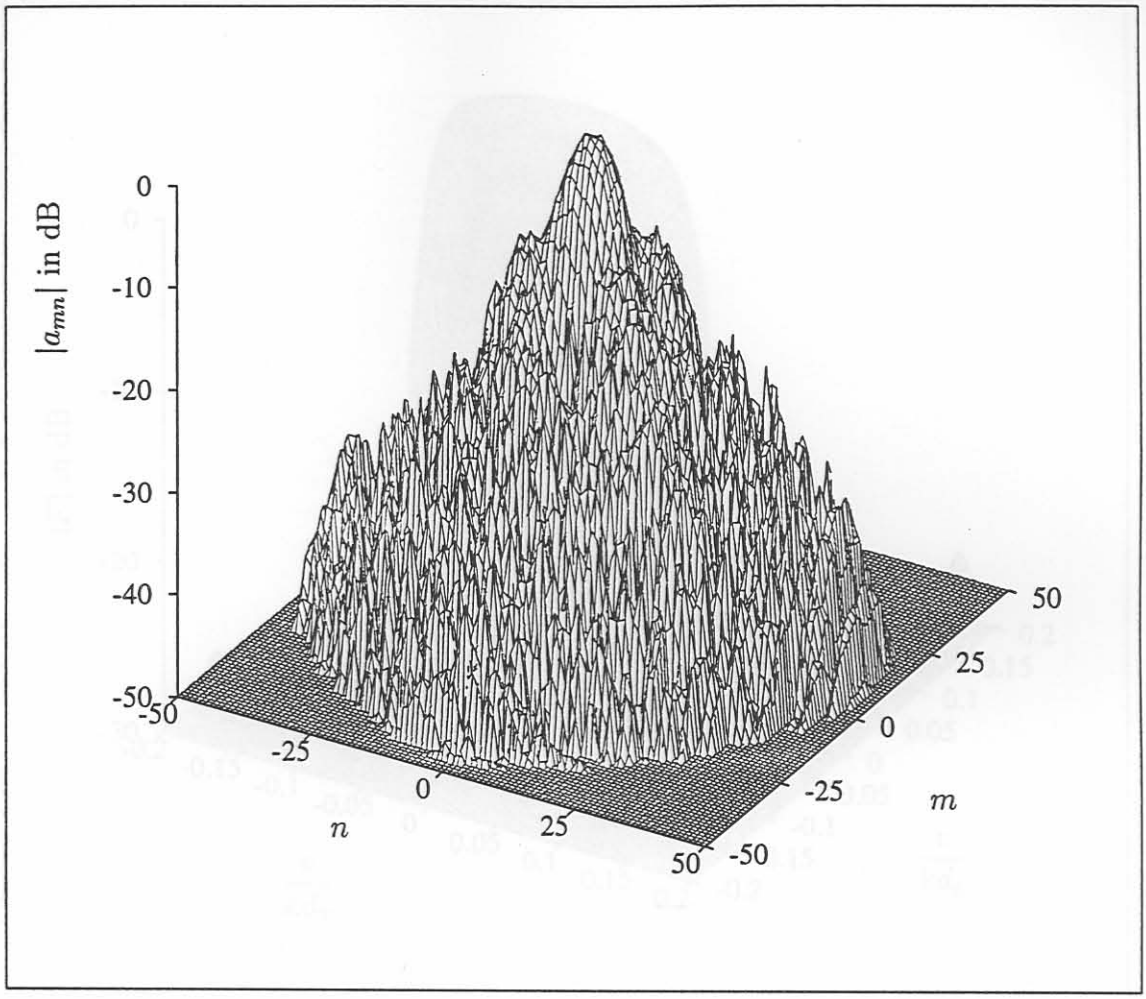

Figure 3.15: Representative Example: A plot of the planar array excitations normalised to the largest excitation.

chapter, is

$$w = w_0 \cos(\tfrac{1}{2}u)\cos(\tfrac{1}{2}v) \tag{3.50}$$

where w is the Chebychev polynomial variable. The Chebychev polynomial is related to the prototype linear array factor by

$$w = w_0 \cos(\tfrac{1}{2}\psi) \tag{3.51}$$

From the definition of the odd case transformation function in equation (3.19) the Baklanov transformation is a special case of the transformation function with $I = J = 1$ and $t_{11} = 1$. For an odd number of array elements the Baklanov transformation reduces to

$$\cos(\tfrac{1}{2}\psi) = \cos(\tfrac{1}{2}u)\cos(\tfrac{1}{2}v) \tag{3.52}$$

If both sides of equation (3.52) are squared and after some manipulation, equation (3.52) can be written as

$$\cos(\psi) = -\tfrac{1}{2} + \tfrac{1}{2}\cos(u) + \tfrac{1}{2}\cos(v) + \tfrac{1}{2}\cos(u)\cos(v) \tag{3.53}$$

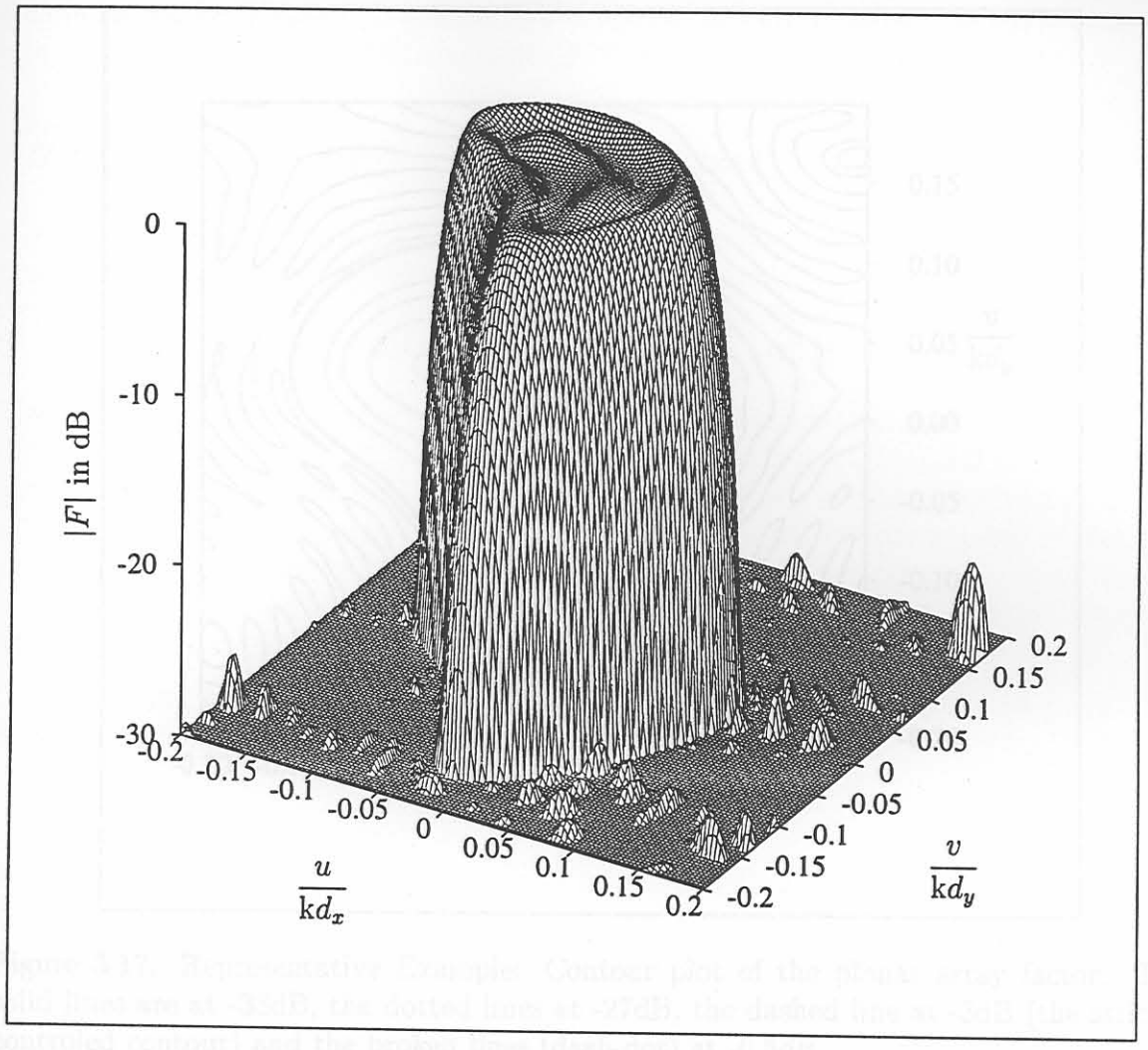

Figure 3.16: Representative Example: A surface plot of the planar array factor, only the centre part is shown.

This is equal to odd case transformation function (defined in equation (3.7)) if $I=J=1$ and $-t_{00}=t_{01}=t_1=t_{11}=1$. The Chebyshev polynomial is related to a prototype linear array with a Chebyshev pattern. Thus the Baklanov transformation synthesis method proposed by Tseng and Cheng is a special case of the transformation based synthesis technique. Tseng and Cheng mentioned that other combinations of $\cos u$ and $\cos(v)$ are possible [81], but did not pursue the possibility.

3.7.2 Extension of Baklanov Transformation by Goto

Goto [83] showed that any symmetrical linear array distribution can be written in an appropriate polynomial form to allow the use of the Baklanov transformation. This allows one to map the symmetrical linear array factor to a planar array factor. The linear array is what we call the prototype linear array.

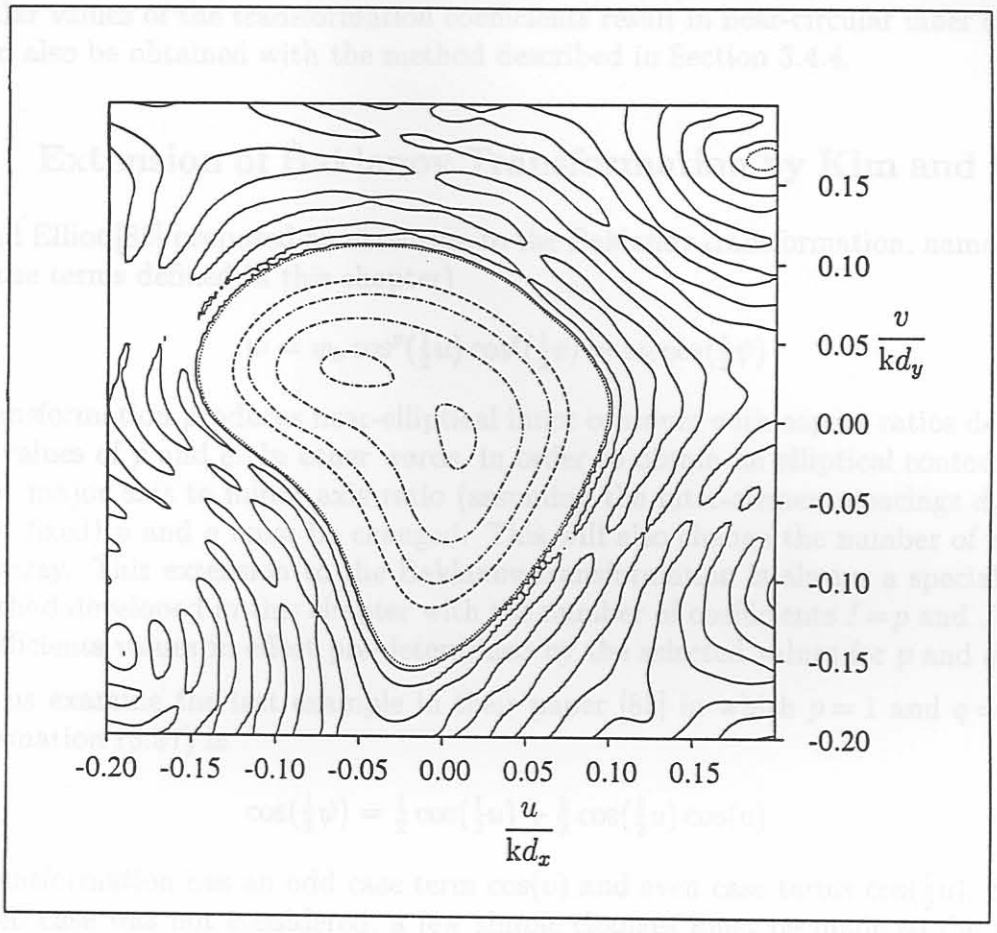

Figure 3.17: Representative Example: Contour plot of the planar array factor. The solid lines are at -33dB, the dotted lines at -27dB, the dashed line at -3dB (the strictly controled contour) and the broken lines (dash-dot) at -0.5dB

In the same paper he used the transformation (written in the terms defined in this chapter)

$$w = \cos(u)\cos(v) \tag{3.54}$$

to obtain a hexagonal lattice array. He later [84] proposed the transformation (again written in the terms defined in this chapter)

$$w = \alpha + \beta\cos(\psi) = \tfrac{1}{2}\cos(u)\left[\cos(u) + \cos(v)\right]. \tag{3.55}$$

where $\alpha = \frac{7}{16}$ and $\beta = \frac{9}{16}$ for a hexagonal array with $d_y = \sqrt{3}d_x$. After some manipulation (3.55) becomes

$$\cos(\psi) = \frac{1}{\beta}\left(\frac{1}{4} - \alpha\right) + \frac{1}{2\beta}\cos(u)\cos(v) + \frac{1}{4\beta}\cos(2u) \tag{3.56}$$

which is equal to the odd case transformation function of size $I=1, J=2$ and coefficients $t_{00} = -\frac{1}{3}, t_{11} = \frac{8}{9}, t_{20} = \frac{4}{9}1$ and $t_{10} = t_{01} = t_{02} = 0$. It is interesting to note that these

particular values of the transformation coefficients result in near-circular inner contours, and can also be obtained with the method described in Section 3.4.4.

3.7.3 Extension of Baklanov Transformation by Kim and Elliott

Kim and Elliot [85] proposed an extension to the Baklanov transformation, namely (written in the terms defined in this chapter)

$$w = w_0 \cos^p(\tfrac{1}{2}u)\cos^q(\tfrac{1}{2}v) = w_0 \cos(\tfrac{1}{2}\psi) \tag{3.57}$$

The transformation produces near-elliptical inner contours with aspect ratios dependent on the values of p and q. In other words, in order to obtain an elliptical contour with a different major axis to minor axis ratio (assuming the inter-element spacings d_x and d_y are kept fixed) p and q must be changed. This will also change the number of elements of the array. This extension to the Baklanov transformation is always a special case of the method developed in this chapter with the number of coefficients $I=p$ and $J=q$ and the coefficients values in effect pre-determined by the selected values for p and q.

Let us examine the last example in their paper [85] in which $p=1$ and $q=2$. The transformation (3.57) is

$$\cos(\tfrac{1}{2}\psi) = \tfrac{1}{2}\cos(\tfrac{1}{2}u) + \tfrac{1}{2}\cos(\tfrac{1}{2}u)\cos(v) \tag{3.58}$$

This transformation has an odd case term $\cos(v)$ and even case terms $\cos(\frac{1}{2}u)$. Since an odd/even case was not considered, a few simple changes must be made to the problem to convert it into an odd case problem. To convert the 10 element prototype linear array to an odd number of elements, a zero element is placed between all the original elements (resulting in a total of 19 elements), thus the inter-element spacing is halved but the radiation pattern stays the same. In addition the planar array inter-element spacings are halved as well. The converted transformation is

$$\cos(\psi') = \tfrac{1}{2}\cos(u') + \tfrac{1}{2}\cos(u')\cos(2v') \tag{3.59}$$

where the variables are primed to indicate the conversion. This is an odd case transformation function with $I'=1$ and $J'=2$. The transformation coefficients are $t'_{10}=\frac{1}{2}, t'_{12}=\frac{1}{2}$ and $t'_{00}=t'_{10}=t'_{11}=t'_{02}=0$. With this arrangement some of the planar array elements will be zero: $a'_{mn}=0$ for $m=$even and $n=$odd. In order to convert the planar array obtained by the synthesis procedure back, the non-zero elements must be mapped to the original array elements

$$a_{m\,n} = a'_{2m-1\,2n}. \tag{3.60}$$

Kim and Elliott remark on the fact that although the angular extent of the flat-top is less in the $\phi=90°$ cut (trace #2 in Figure 3.18) than in the $\phi=0°$ cut (trace #1

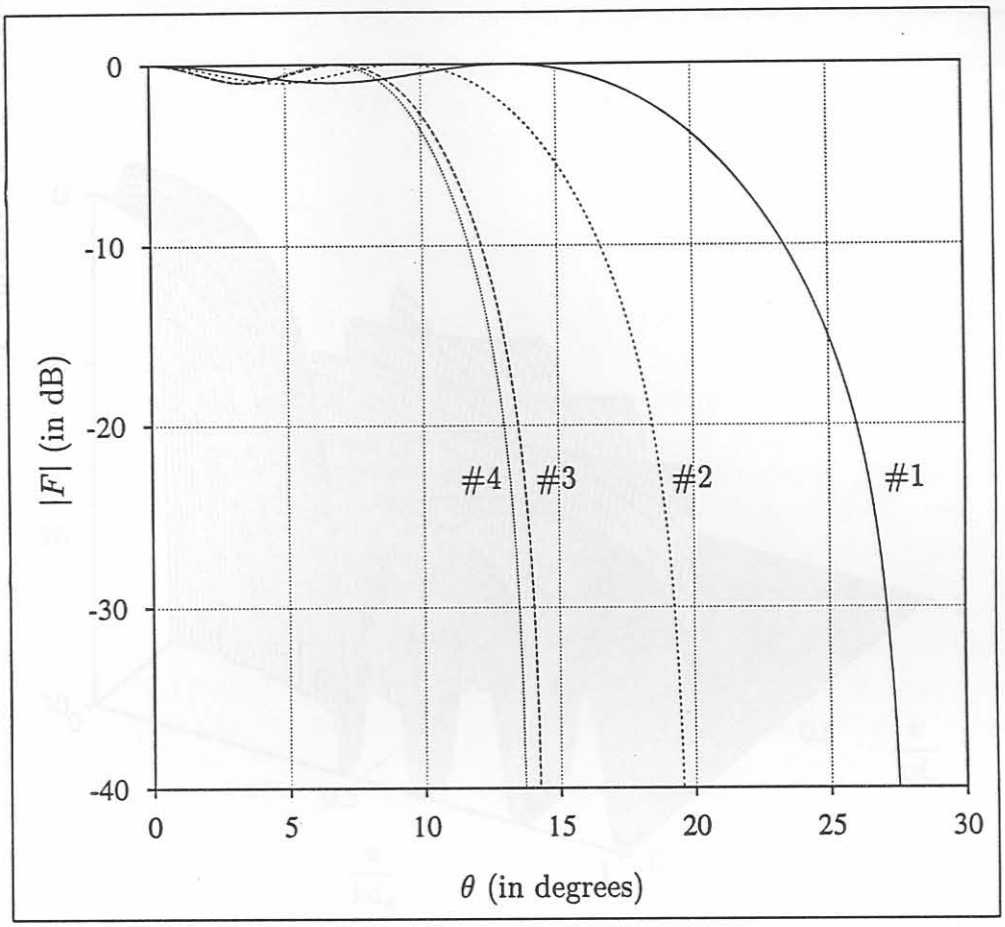

Figure 3.18: The principal cuts for the 10 by 19 element array:#1 ϕ=0° cut, #2 ϕ=90° cut of Kim and Elliot, #3 $\phi = 90°$ cut of optimal transformation function and #4 the pattern of a 19 element linear array.

in Figure 3.18), and the fall-off is steeper, the $\phi = 90°$ cut is not as narrow as it would be for a 19 element linear array with two roots off the Shelkunoff unit circle (trace #4 in Figure 3.18). The beamwidth in the $\phi = 90°$ cut is 42% wider than that of the 19 element linear array. With a better selection of transformation coefficients the beam in the $\phi = 90°$ cut can be much narrower. The optimum transformation coefficients are $t'_{10} = 0.1142$ and $t'_{12} = 0.8858$; this results in a much narrower beam in the $\phi = 90°$ cut (trace #3 in Figure 3.18). Compared to the 19 element linear array factor the optimum transformation coefficients result in an increased first-null beamwidth of only 3.6% in the ϕ=90° pattern cut. The planar array factor is displayed in Figure 3.19

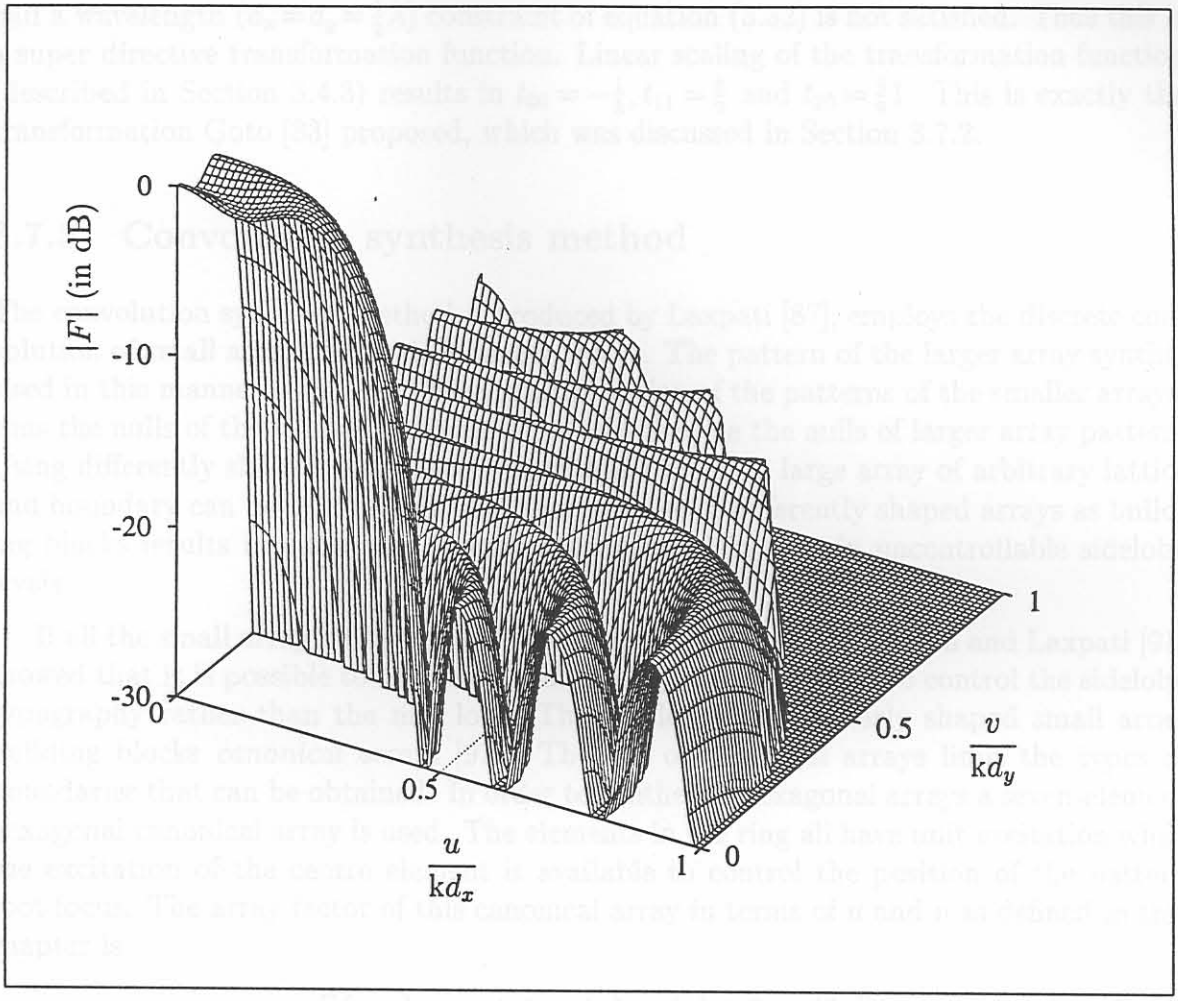

Figure 3.19: Surface plot of the 10 by 19 element planar array factor with optimum transformation coefficients (north-east quadrant only).

3.7.4 Extension of Baklanov Transformation by Kim

Kim [86] proposed an alternative transformation to Goto for the synthesis of hexagonal arrays (again using the the terms defined in this chapter):

$$w^2 = w_0^2 \cos(u) \cos\left(\tfrac{1}{2}u + \tfrac{1}{2}v\right) \cos\left(\tfrac{1}{2}u - \tfrac{1}{2}v\right) = w_0^2 cos^2(\tfrac{1}{2}\psi) \tag{3.61}$$

After some manipulation the transformation (3.61) can be written as

$$\cos(\psi) = -\tfrac{1}{2} + \cos(u)\cos(v) + \tfrac{1}{2}\cos(2u) \tag{3.62}$$

This is equal to the odd case transformation function with $I=2$ and $J=1$ and $-t_{00} = t_{20} = \frac{1}{2}$, $t_{11}=1$ and $t_{01}=t_{10}=t_{21}=0$.

In his paper Kim used $d_t = \frac{1}{2}\lambda$, thus $d_x = \frac{1}{4}\lambda$ and $d_y = \frac{\sqrt{3}}{4}\lambda$; for these values the constraint of equation (3.32) is satisfied. However, if the inter-element spacings are set to

half a wavelength ($d_x = d_y = \frac{1}{2}\lambda$) constraint of equation (3.32) is not satisfied. Thus this is a super directive transformation function. Linear scaling of the transformation function (described in Section 3.4.3) results in $t_{00} = -\frac{1}{3}, t_{11} = \frac{8}{9}$ and $t_{20} = \frac{4}{9}1$. This is exactly the transformation Goto [83] proposed, which was discussed in Section 3.7.2.

3.7.5 Convolution synthesis method

The convolution synthesis method, introduced by Laxpati [87], employs the discrete convolution of small arrays to produce larger arrays. The pattern of the larger array synthesised in this manner is obtained from multiplication of the patterns of the smaller arrays; thus the nulls of the smaller array patterns will then be the nulls of larger array pattern. Using differently shaped small arrays as building blocks a large array of arbitrary lattice and boundary can be synthesised. However, the use of differently shaped arrays as building blocks results in differently shaped root loci and in turn in uncontrollable sidelobe levels.

If all the small array building blocks have the same geometry Shelton and Laxpati [91] showed that it is possible to use the convolution synthesis method to control the sidelobe typography rather than the null loci. They called these similarly shaped small array building blocks *canonical arrays* [91]. The use of canonical arrays limit the types of boundaries that can be obtained. In order to synthesise hexagonal arrays a seven-element hexagonal canonical array is used. The elements in the ring all have unit excitation while the excitation of the centre element is available to control the position of the pattern root locus. The array factor of this canonical array in terms of u and v as defined in this chapter is

$$F_i(u,v) = a_i + 4\cos(u)\cos(v) + 2\cos(2u) \tag{3.63}$$

By using a linear mapping the authors were able to relate the value of a_i to the roots of a polynomial. In this case they selected a Chebyshev polynomial. In order to compare the transformation based method with the convolution based method the transformation function must be set equal to the canonical array factor. This transformation function will need scaling; linear scaling has the same purpose as linear mapping. Using equation (3.34) we obtain $C_1 = \frac{2}{9}$ and $C_2 = \frac{2a_i}{9} + \frac{1}{3}$; and the scaled transformation function is

$$H(u,v) = -\frac{1}{3} + \frac{8}{9}\cos(u)\cos(v) + \frac{4}{9}\cos(2u) \tag{3.64}$$

This is exactly the same transformation as that proposed by Goto [84], as mentioned in Section 3.7.2.

In order to synthesise square arrays a nine square canonical array is used. The corner elements have an excitation of α, the centre element a_i and the rest have unit excitation. Two degrees of freedom are available to control the position and shape of the pattern root locus. The array factor for a nine-element square canonical array is

$$F_i(u,v) = a_i + 4\alpha\cos(u)\cos(v) + 2\left(\cos(u) + \cos(v)\right) \tag{3.65}$$

The transformation function must be set equal to the canonical array factor. Linear scaling gives $C_1 = \frac{1}{2+4\alpha}$ and $C_2 = \frac{c_i+2}{2+4\alpha}$; and the scaled transformation function is

$$H(u,v) = -\frac{1}{1+2\alpha} + \frac{1}{1+2\alpha}\left(\cos(u)+\cos(v)\right) + \frac{2\alpha}{1+2\alpha}\cos(u)\cos(v) \tag{3.66}$$

The convolution synthesis method is again a special case of the transformation based synthesis method. For $\alpha_i = \frac{1}{2}$ the non-zero transformation function coefficients are $-t_{00} = t_{01} = t_1 = t_{11} = 1$. This is identical to the Baklanov transformation. Although α gives some control over the contour shape, the inner contours are always near circular. Shelton and Laxpati changed the value of α to optimise the dynamic range of the distribution. Although only symmetrical patterns were used it is possible to use canonical arrays with a complex excitation to obtain non-symmetrical patterns. In order to have control over the sidelobe level only the excitation of the centre element must be changed. This would be a special case of the transformation based synthesis technique for arbitrary contours. In fact the planar array formed by the transformation function (described in Section 3.4.6) is identical to the canonical array.

3.8 General Remarks and Conclusions

The transformation based synthesis technique has firstly been extended to the synthesis of planar arrays with arbitrary contoured footprint patterns, secondly to allow the synthesis of planar arrays with non-rectangular boundaries and thirdly to enable the synthesis of planar arrays with non rectangular boundaries.

The transformation based synthesis technique utilises a transformation that divides the problem into two decoupled sub-problems. One sub-problem (the contour transformation problem) involves the determination of certain coefficients in order to achieve the required footprint contours. The number of contour transformation coefficients which must be used depends on the complexity of the desired contour, but is very small in comparison to the number of planar array elements. The other sub-problem consists of a prototype linear array synthesis, for which powerful methods for determining appropriate element excitations, already exist. The size required for the prototype linear array depends on the amount of allowable ripple in the coverage area number and the detail of the beam shape through any cross-section. Simple recursive formulas then determine the final planar array excitations from the information forthcoming from the above two sub-problem solutions. The final planar array size is linked to the number of contour transformation coefficients and the prototype linear array size.

The synthesis technique can also be seen as a division of the synthesis problem into two separate problems; one pertaining to the shape of the footprint and the other involving the sidelobe level and main beam ripple. The separation of the synthesis procedure into two sub-problems not only is good from a computation point of view, but aids understanding by highlighting which considerations will finally determine the required array size (viz. contour complexity; allowed coverage ripple; final planar array directivity). The technique

in effect provides a structured procedure for spreading out the linear array excitations, thereby eliminating any guesswork that may otherwise be required.

The transformation coefficients can be seen as an inverse Fourier transformation of the desired contour shape. Thus as the complexity of the strictly controlled contour increases or its "beamwidth" decreases, more transformation function coefficients must be used to approximate the contour. If the number of transformation coefficients increase, the number of elements in the prototype linear array must decrease to keep the planar array size fixed. This in turn limits the performance (beamwidth, sidelobe ratio and main beam ripple) obtainable with the prototype linear array. Furthermore, the transformation function coefficients can be calculated such that the beamwidth is the narrowest possible in any particular pattern cut. The method works well when synthesising reconfigurable arrays where neither a change in inter-element spacing nor a change in the number of array elements is easily incorporated.

The biggest advantage of the transformation based synthesis technique is its computational efficiency. The final planar array excitations are very simply related to the prototype linear array excitation, which means that the technique can be used to rapidly synthesise even very large arrays. The computer time required to perform such a synthesis is relatively short, using very little system resources; making it feasible to conduct parametric studies of array performance and to perform design tradeoff studies even for very large arrays.

Numerical optimisation methods usually solve a set of equations (represented as a matrix in computer memory) subject to a set of constraints, where the number of equations is generally more than the number of unknowns. The memory requirements and the processing power needed for these optimisation methods increase rapidly as the number of array elements increase. In contrast, the memory requirement for the transformation based synthesis technique is small, approximately 12 times the total number of elements in real numbers and the total number of elements in complex numbers. In addition the processing time to synthesise large arrays with the transformation based synthesis technique is in the order of seconds, even on a modest personal computer.

However, the transformation based synthesis method does suffer some limitations. The foremost shortcoming is that the method can not be used to synthesise planar arrays with an arbitrary number of elements. For instance, consider the transformation for the odd case planar array. If a transformation of say $I = J = 2$ is required to obtain the desired contour shape then only planar arrays with $M = N = 2Q+1$ can be synthesised. This limitation is even worse for the even case planar arrays.

More detailed contour shapes can be achieved if all the planar array excitations are optimised by using a numerical optimisation synthesis technique. However, no phase information is known about the planar array factor; this has severe consequences on the optimisation procedures. These optimisation techniques are usually unstable and very dependent on good starting values. If the requirements on the planar array factor can not be met with the transformation based synthesis method, this method can be used to get a good set of starting values for optimisation techniques.