MIMO Radar Transmit Beampattern Design Without Synthesising the Covariance Matrix

Sajid Ahmed *Senior Member, IEEE* and Mohamed-Slim Alouini *Fellow, IEEE*

Computer, Electrical, Mathematical Sciences and Engineering (CEMSE) Division

Mail: King Abdullah University of Science and Technology (KAUST)

Thuwal 23955-6900, Saudi Arabia

Emails: sajid.ahmed, slim.alouini@kaust.edu.sa

Phone: +966 (02) 808-0286, +966 (02) 808-0283

EDICS: NSP-APPL, SAM-RADR

Abstract

Compared to phased-array, multiple-input multiple-output (MIMO) radars provide more degrees-of-freedom (DOF) that can be exploited for improved spatial resolution, better parametric identifiability, lower side-lobe levels at the transmitter/receiver, and design variety of transmit beampatterns. The design of the transmit beampattern generally requires the waveforms to have arbitrary auto- and cross-correlation properties. The generation of such waveforms is a two step complicated process. In the first step a waveform covariance matrix is synthesised, which is a constrained optimisation problem. In the second step, to realise this covariance matrix actual waveforms are designed, which is also a constrained optimisation problem. Our proposed scheme converts this two step constrained optimisation problem into a one step unconstrained optimisation problem. In the proposed scheme, in contrast to synthesising the covariance matrix for the desired beampattern, n_T independent finite-alphabet constant-envelope waveforms are generated and pre-processed, with weight matrix $\mathbf{W}$, before transmitting from the antennas. In this work, two weight matrices are proposed that can be easily optimised for the desired symmetric and non-symmetric beampatterns and guarantee equal average power transmission from each antenna. Simulation results validate our claims.

Index Terms

MIMO radar, colocated antennas, beampattern design, geometric function.

I. Introduction

Multiple-input multiple-output (MIMO) radars allow each transmitting antenna to transmit independent waveforms and thus provide extra degrees-of-freedom (DOF) that can be exploited to improve system performance [1], [2]. For example, for given number of transmit and receive antennas, compared to phased-array using MIMO radar more number of targets can be identified. MIMO radars can be classified into two categories: widely distributed [3] and colocated [4]. In the widely distributed case the transmitting antennas are widely separated so that each antenna may view a different aspect of the target. This topology can increase the spatial diversity of the system. In colocated systems, the transmitting antennas are closely spaced to view the same aspect of the target. The colocated antenna radar cannot provide spatial diversity but can increase the spatial resolution of the system. The focus of this paper will be to achieve the desired beampattern by exploiting colocated MIMO radar.

The waveform design methods to achieve specific goals for widely distributed radars are discussed in [5] (and the references therein), while the waveform design methods for colocated radars to achieve desired beampatterns are discussed in detail [6]–[11]. Phased-array radar do an excellent job when the direction of the target is known and a very narrowly focused beam in that direction is required [12]. However, if the direction of the target is not known and the requirement is to have a uniform power in more than one direction, MIMO radar with colocated antennas can provide better solutions due to the extra DOF. In conventional methods, the design of waveforms to achieve a desired beampattern is a 2-step process:

- In the first step, for the desired beampattern, a covariance matrix, $\mathbf{R}$, of the waveforms is synthesised [9], [11]. Since $\mathbf{R}$ is a covariance matrix, it must be a positive semi-definite. Secondly, for the ease of design all of its diagonal elements that represent the transmitted average power from the corresponding antenna should be equal. For example, to design a given beampattern using multiple colocated antennas requires multiple radio-frequency-amplifiers (RFAs). It is well known that RFAs have limited dynamic range and cannot have maximum power efficiency at all power levels. If each antenna is required to transmit at a different power level then multiple different RFAs with different bias voltage levels will be required. The requirement of RFA for each antenna may change with respect to the beampattern, therefore a change in desired beampattern may require a change

in hardware. Constraining each antenna to transmit at the same power level provide flexibility to design different desired beampatterns without changing the hardware.

- In the second step, the actual waveforms are designed to realise the synthesised covariance matrix [10], [13]. Due to the limited dynamic range, RFAs cannot have maximum power efficiency at all power levels. Therefore, the designed waveforms need to have low peak-to-average power ratios (PAPRs). Ideally all the waveforms need to have constant-envelope, which is again a challenging problem.

To reduce the computational burden of two step constrained optimisation, to design the desired beampatterns, in [14] and [15] independent waveforms are pre-processed with complex weights to form multi-rank beamformers to achieve the desired beampatterns. These schemes are very simple and the complex weights for pre-processing can be found in closed form. However, the existing pre-processing schemes to achieve the desired beampatterns have few drawbacks, such as they do not guarantee equal power transmission from each antenna and variety of optimal beampatterns cannot be obtained. Both of these requirement are crucial.

In our proposed algorithm n_T independent finite alphabet constant-envelope (FACE) waveforms, such as binary-phase shift-keying (BPSK) and quadrature-phase shift-keying (QPSK), are generated, a process which is very easy. Next, the waveform p before transmitting from the antenna q is multiplied with the complex weight, w_{pq}, and the values of these weights are optimised to obtain the desired beampattern. To transmit the same average power from each antenna, the weights corresponding to the waveforms transmitted from antenna q are parametrised by the coordinates of hyper-sphere that guarantee equal power transmission from each antenna. Therefore, the proposed scheme converts the two step constrained optimisation problem for the desired beampattern design into a one step simple optimisation problem, fulfills the equal elemental power constraint, and variety of beampatterns can be achieved.

The remainder of this paper is organised as follows. In the following section the problem formulation and some background are given. The proposed algorithm is developed in section III. The computational complexity of the proposed algorithm is described and compared with the synthesising covariance matrix and designing waveforms method in Section IV. Section V discusses the PAPR of the proposed algorithm. Simulation results are given in Section VI, followed by our conclusions in Section VII.

Notation: Bold upper case letters, $\mathbf{X}$, and lower case letters, $\mathbf{x}$, respectively denote matrices and vectors. The (m,n)th element of a matrix $\mathbf{X}$ is denoted by x_{mn} and mth column-vector of a matrix $\mathbf{X}$ is denoted by $\mathbf{x}_m$. Transpose, conjugate, and conjugate transposition of a matrix are respectively denoted by $(.)^T$, $(.)^*$, and $(.)^H$. E$\{.\}$ denotes the statistical expectation and $\Re(\cdot)$ denotes the real part. The column-vector of N zeros is denoted by $\mathbf{0}_N$. Modulo-N operation on m is denoted by $\langle m \rangle_N$. Finally, the set of numbers $\{p, p+1, \ldots, q\}$ is denoted by $p:q$.

II. Problem Formulation and Previous Work

The problem considered in this paper is to transmit uniform power at a number of given target locations and minimise in all other locations. This is also called beampattern matching. To achieve this a uniform linear array of n_T antenna elements with half-wavelength inter-element spacing is used. Let $x_q(n)$ be the baseband transmitted signal from antenna q at time index n. The baseband received signal at spatial location θ_k can be written as

$$r_k(n) = \sum_{p=1}^{n_T} e^{-j(q-1)\pi \sin(\theta_k)} x_q(n), \quad n = 1, 2, \ldots, N, \tag{1}$$

where N is the total number of transmitted symbols. By defining, $\mathbf{a}_t(\theta_k) = \begin{bmatrix} 1 & e^{j\pi \sin(\theta_k)} & \cdots & e^{j(n_T-1)\pi \sin(\theta_k)} \end{bmatrix}^T$, a transmit steering vector, and $\mathbf{x}(n) = \begin{bmatrix} x_1(n) & x_2(n) & \cdots & x_{n_T}(n) \end{bmatrix}^T$, a vector of transmitted symbols, the received signal in (1) can be written as

$$r_k(n) = \mathbf{a}_t^H(\theta_k)\mathbf{x}(n). \tag{2}$$

Following (2), the power received at location θ_k can be written as

$$\begin{aligned} P(\theta_k) &= \mathrm{E}\Big\{\mathbf{a}_t^H(\theta_k)\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{a}_t(\theta_k)\Big\} \\ &= \mathbf{a}_t^H(\theta_k)\mathbf{R}\mathbf{a}_t(\theta_k), \end{aligned} \tag{3}$$

where $\mathbf{R}$ is the covariance matrix of the waveforms. In order to achieve the desired beampattern, an appropriate covariance matrix $\mathbf{R}$ has to be found. The matrix $\mathbf{R}$ can be optimised by minimising the difference between the desired and designed values of the beampattern. However, as mentioned in the introduction, the matrix $\mathbf{R}$ should be a positive semi-definite and all of its diagonal elements must be equal.

In order to match the desired beampattern and satisfy the first constraint, the iterative algorithms proposed in [6]–[8], [16] synthesise $\mathbf{R}$ using its square-root factorization. If the matrix $\mathbf{R}$ is found indirectly using its square-root matrix, $\mathbf{U}$, then it can be guaranteed to be positive semi-definite.

The algorithm proposed in [6] exploits the Cholesky factorization of $\mathbf{R}$ to achieve a positive definite $\mathbf{R}$. In this algorithm, to optimise the matrix $\mathbf{U}$ for the desired beampattern, following cost-function is used

$$J_1(\mathbf{U}) = \frac{1}{K}\sum_{k=1}^{K}\left(|\mathbf{a}_t^H(\theta_k)\mathbf{U}| - \sqrt{P_d(\theta_k)}\right)^2, \tag{4}$$

where $P_d(\theta_k)$ is the desired beampattern and $|\mathbf{a}_t^H(\theta_k)\mathbf{U}|$ is the square-root of the designed beampattern at location θ_k. In this algorithm the authors perform a LS fit to the square-root of the desired power, while in their later work the LS fit is directly performed on the desired power [7]. The approximation of the beampattern depends heavily on the choice of cost-function. In [8] a modified LS error criterion is used to find the square-root matrix $\mathbf{U}$, which in discrete form can be written as

$$J_2(\mathbf{U}) = \frac{1}{K}\sum_{k=1}^{K}\left(|\mathbf{a}_t^H(\theta_k)\mathbf{U}|^2 - P_d(\theta_k)\right)^2. \tag{5}$$

In this algorithm, the LS fit is directly performed on the desired power $P_d(\theta_k)$. However, in contrast to the algorithm in [6], the matrix $\mathbf{U}$ is non-triangular. Here, to find $\mathbf{U}$, a gradient-descent algorithm is used to minimize the cost-function with respect to $\mathbf{U}$. The main contribution in [8] is to convert the constrained optimization problem into an unconstrained optimization problem, which is achieved by replacing the equal elemental power constraint with the total power constraint.

The beampattern matching in [9] uses semi-definite quadratic programming (SQP) to optimise the following cost-function

$$J(\mathbf{R}) \;=\; \frac{1}{K}\sum_{k=1}^{K}\left(\mathbf{a}_t^H(\theta_k)\mathbf{R}\mathbf{a}_t(\theta_k) - \alpha P_d(\theta_k)\right)^2, \tag{6}$$

subject to the constraints

$$\mathbf{C}_1. \quad \mathbf{R} \geq 0,$$

$$\mathbf{C}_2. \quad \mathbf{R}(q,q) = c, \quad \text{for} \quad q = 1, 2, \ldots, n_T,$$

where α is a scaling factor and c is the transmitted power from each antenna. The SQP requires that the cost-function should be real and in quadratic form, which make it difficult to design non-symmetric beampatterns. This algorithm yields optimum beampattern match using 2-norm cost function.

To find the optimum value of $\mathbf{R}$ and α, our previous work in [11], converts the constrained optimisation problem in (6) into an unconstrained optimisation problem, which implicitly fulfills the constraints and can be applied to optimise the 1- and 2-norm cost-functions. In [11], it is also shown that optimising the 1-norm cost-function yields better performance than optimising the 2-norm cost-function.

In all of the above mentioned algorithms, covariance matrix is synthesised first, while in the second step, user needs to generate waveforms to realise this covariance matrix.

To generate actual waveforms to realise the covariance matrix for pre-defined PAPR, the algorithm in [10] uses an iterative algorithm. For PAPR $= 1$, the algorithm generates constant-envelope waveforms that do not exactly match the beampattern of covariance matrix. In other words, the waveforms cannot exactly realise the covariance matrix. To exactly match the beampattern of the covariance matrix, the waveforms require higher values of PAPR. The other drawback of the algorithm is the generation of non-finite alphabet waveforms, which are challenging to use in practice.

To reduce the computational complexity of synthesising the covariance matrix and then the generation of FACE waveforms for the desired beampattern, in our previous work [13], [17], a one step unconstrained iterative algorithm is proposed. This algorithm generates the best possible BPSK and QPSK waveforms for the desired beampattern. However, the MSE between the desired and the designed beampattern using BPSK and QPSK is higher than the non-finite alphabet waveforms proposed in [10].

The recent work on beampattern matching in [14] and [15], instead of synthesising the covariance matrix and then designing the actual waveforms, use simple sub-optimal approaches. In [15] multiple rank-1 beamformers are designed such that the resultant of these beamformers yields the desired beampattern. While in [14] a sub-optimal covariance matrix is decomposed into its eigenvalues and eigenvectors, eigenvectors corresponding to the significant eigenvalues are used to form a weight matrix for pre-processing the independent transmitted waveforms. The performance of these algorithms compared to synthesising the covariance matrix is poor. They have lower role-off in the transition band and do not guarantee equal average power constraint.

The motivation of the proposed work is to provide an unconstrained one-step algorithm to obtain the symmetric and non-symmetric desired beampattern that can optimally match the desired beampattern under the equal elemental power constraint.

III. Proposed Algorithm

In the proposed scheme n_T independent FACE waveforms are generated. The waveform p before transmitting from the antenna q is multiplied with the weight w_{pq} as shown in Fig. 1. Therefore, as a result, the antenna q transmits a sum of n_T weighted waveforms. Using the proposed model, the baseband received signal at location θ_k can be written as

$$r_k(n) = \sum_{q=1}^{n_T} \sum_{p=1}^{n_T} w_{pq} x_p(n) e^{j(q-1)\pi \sin(\theta_k)}, \ n = 1, 2, \ldots, N. \tag{7}$$

In vector form the received signal can be written as

$$r_k(n) \ = \ \mathbf{a}_t^T(\theta_k) \mathbf{W}^T \mathbf{x}(n), \tag{8}$$

where

$$\mathbf{W} \ = \ \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1n_T} \\ w_{21} & w_{22} & \cdots & w_{2n_T} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n_T1} & w_{n_T2} & \cdots & w_{n_Tn_T} \end{bmatrix}.$$

The received power at location θ_k using (8) can be written as

$$P(\theta_k) \ = \ \mathrm{E}\{\mathbf{a}_t^T(\theta_k) \mathbf{W}^T \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{W}^* \mathbf{a}_t^*(\theta_k)\}.$$

Since the transmitted waveforms are fully uncorrelated, $\mathrm{E}\{\mathbf{x}(n)\mathbf{x}^H(n)\} = \mathbf{I}$ and the received power can be written as

$$P(\theta_k) \ = \ \mathbf{a}_t^H(\theta_k) \mathbf{W}^H \mathbf{W} \mathbf{a}_t(\theta_k).$$

To obtain desired power $P_d(\theta_k)$ for $k = 1, 2, \ldots, K$, the matrix $\mathbf{W}$ can be optimised using the following cost-function

$$J(\mathbf{W}) = \frac{1}{K} \sum_{k=1}^{K} \left(\mathbf{a}_t^H(\theta_k) \mathbf{W}^H \mathbf{W} \mathbf{a}_t(\theta_k) - \alpha P_d(\theta_k) \right)^2. \tag{9}$$

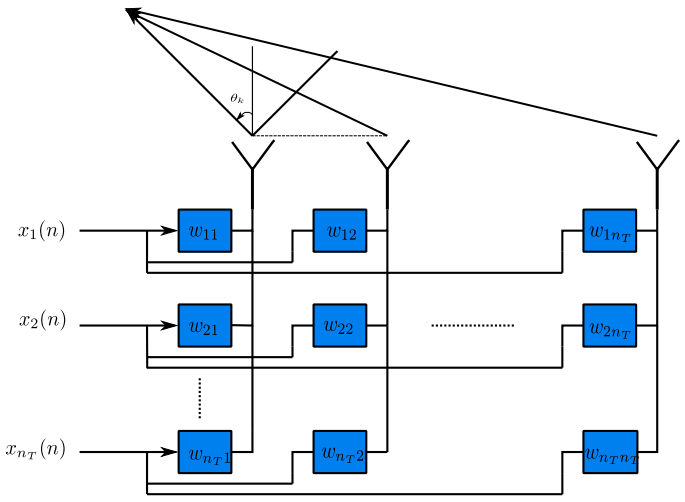

Fig. 1. Proposed model of the transmitter.

The use of this scheme does not impose any condition on $\mathbf{W}$ to be positive semi-definite. It only requires the generation of fully independent FACE waveforms, which is very easy. Next, we want to transmit the same average power from each antenna, which was the second constraint in synthesising the covariance matrix $\mathbf{R}$. Using our proposed scheme, the average transmitted power from antenna q can be derived as

$$\begin{aligned} P_{\text{ave}}(q) &= \mathrm{E}\left\{\left|\sum_{p=1}^{n_T} w_{pq}x_p(n)\right|^2\right\} \\ &= \mathrm{E}\left\{\mathbf{w}_q^H\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w}_q\right\} \\ &= \mathbf{w}_q^H\mathbf{w}_q. \end{aligned} \tag{10}$$

It can be noted in (10) that the transmitted power from antenna q is equal to the norm of the qth column-vector of $\mathbf{W}$. Optimising $\mathbf{W}$ freely for the desired beampattern does not guarantee equal power transmission from each antenna. To transmit the same power from each antenna, $\mathbf{w}_q$ must have the same norm for each value of q. Therefore, if (10) can be a fixed constant for any value of q, equal power transmission from each antenna can be guaranteed. To fulfill this constraint, all columns of weight matrix $\mathbf{W}$ must have equal norm. This can be achieved by parametrising the columns of $\mathbf{W}$ by multi-dimensional spherical co-ordinate system, which is explained in the following paragraph.

A point on a q-dimensional sphere can be specified by q spherical or Cartesian coordinates. If the spherical coordinates of a point are known, which are typically used to specify a point on the sphere,

then the corresponding Cartesian coordinates can be easily found using the simple relationships between them [18]. At any point on the sphere, the norm of the Cartesian coordinates is always equal to the radius, r, of the sphere, and it does not change with the change in spherical coordinates. This property allows Cartesian coordinates to form a fixed norm variable column-vector, whose elements can be freely changed by changing the spherical coordinates. In the proposed work, this property is exploited to form a matrix $\mathbf{W}$ of fixed norm column-vectors. In the following, we review the method of finding the Cartesian coordinates of a point on multi-dimensional spheres.

A 1-dimensional sphere of radius r (a single point at a distance r from the origin) can have only one Cartesian coordinate and it can be found as

$$w_{11} = r, \tag{11}$$

where w_{11} is the Cartesian coordinate and is equal to the radius of the 1-dimensional sphere. The Cartesian coordinates of a 2-dimensional sphere of radius r (a circle) can be found as

$$\begin{aligned} w_{12} &= r\sin(\psi_{21}), \\ w_{22} &= r\cos(\psi_{21}), \end{aligned} \tag{12}$$

where w_{12} and w_{22} are the Cartesian coordinates, while ψ_{21} is the spherical coordinate of the 2-dimensional sphere and represents the angular location of the point on the circle. Similarly, the Cartesian coordinates of a 3-dimensional sphere of radius r (a conventional sphere) can be found as

$$\begin{aligned} w_{13} &= r\sin(\psi_{31})\sin(\psi_{32}), \\ w_{23} &= r\sin(\psi_{31})\cos(\psi_{32}), \\ w_{33} &= r\cos(\psi_{31}), \end{aligned} \tag{13}$$

where w_{13}, w_{23}, and w_{33} are the Cartesian coordinates, while ψ_{31} and ψ_{32} are the spherical coordinates and represent the zenith and azimuth angles of a sphere. These results can be easily extended to find the Cartesian coordinates of a point on high dimensional spheres. It can be seen in (11), (12), and (13) that the norm of the Cartesian coordinates of a point on any of these spheres is equal to r. If the radius of the sphere $r = 1$, then the norm will also be equal to one. The Cartesian coordinates of a q-dimensional sphere can be used to form an $n_T \geq q$ element unit norm column-vector. To do so, its q elements can

be parametrised by the q Cartesian coordinates and the remaining $n_T - q$ elements can be set to zero. Similarly, n_T independent unit norm column-vectors can be generated to form an $n_T \times n_T$ weight matrix, $\mathbf{W}$. In the following, we will discuss two methods to form such weight matrices.

A. Method-1

In the first method, to generate an $n_T \times n_T$ weight matrix, $\mathbf{W}$, for $q = 1, 2, \ldots, n_T$, parametrise the q elements of the qth column-vector of $\mathbf{W}$ by the Cartesian coordinates of q-dimensional sphere and set the remaining $n_T - q$ elements to zero. The corresponding matrix is shown in (14)

$$\mathbf{W}(\boldsymbol{\psi}) = \begin{bmatrix} 1 & \sin(\psi_{21}) & \sin(\psi_{31})\sin(\psi_{32}) & \cdots & \prod_{m=1}^{n_T-1}\sin(\psi_{n_T m}) \\ 0 & \cos(\psi_{21}) & \sin(\psi_{31})\cos(\psi_{32}) & \cdots & \prod_{m=1}^{n_T-2}\sin(\psi_{n_T m})\cos(\psi_{n_T,n_T-1}) \\ \vdots & \ddots & \cos(\psi_{31}) & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \sin(\psi_{n_T 1})\cos(\psi_{n_T 2}) \\ 0 & \cdots & \cdots & 0 & \cos(\psi_{n_T 1}) \end{bmatrix}, \tag{14}$$

where $\boldsymbol{\psi} = \{\psi_{21}, \psi_{31}, \psi_{32}, \cdots, \psi_{n_T,n_T-1}\}$. It can be noted in (14) that the qth column of $\mathbf{W}$ contains $(q-1)$ free angles $\{\psi_{q1}, \psi_{q2}, \ldots, \psi_{q,q-1}\}$ and for any value of these angles the qth column of $\mathbf{W}$ has unit norm.

B. Method-2

In the second method, to generate an $n_T \times n_T$ weight matrix, $\mathbf{W}$, parameterize each of its column-vector by the independent Cartesian coordinates of P-dimensional sphere, where $P \leq n_T$, and set the remaining elements in each column-vector to zero. The generated matrix using this method is shown in (15)

$$\mathbf{W}(\boldsymbol{\psi}) = \begin{bmatrix} \prod_{m=1}^{P-1}\sin(\psi_{1m}) & \cdots & \prod_{m=1}^{n_T-2}\sin(\psi_{n_T m})\cos(\psi_{n_T,P-1}) \\ \prod_{m=1}^{P-2}\sin(\psi_{1m})\cos(\psi_{1,P-1}) & \cdots & \vdots \\ \vdots & \vdots & \sin(\psi_{n_T 1})\cos(\psi_{n_T 2}) \\ \sin(\psi_{11})\cos(\psi_{12}) & \cdots & \cos(\psi_{n_T 1}) \\ \cos(\psi_{11}) & \cdots & \mathbf{0}_{n_T-P} \\ \mathbf{0}_{n_T-P} & & \prod_{m=1}^{P-1}\sin(\psi_{n_T m}) \end{bmatrix}, \tag{15}$$

where $\boldsymbol{\psi} = \{\psi_{11}, \ldots, \psi_{1(P-1)}, \ldots, \psi_{n_T 1}, \ldots, \psi_{n_T,P-1}\}$. Here, each column-vector contains $(P-1)$ free angles and for the qth column-vector, the corresponding P Cartesian coordinates are assigned to

$\big\langle (q-1:P+q-2) \big\rangle_{n_T} + 1$ elements. Generating the weight matrix using this method, in-contrast to the method-1, insures that each antenna transmits the same number of waveforms. It should be noted here that the DOF using the weight matrix of the first method is $\sum_{q=1}^{n_T}(q-1) = \frac{n_T^2 - n_T}{2}$ while the DOF using the weight matrix of second method is $n_T(P-1)$. For both matrices to design similar beampatterns, the DOF should be the same, which yields $P \approx \frac{n_T}{2} + 1$. Therefore, the the optimal value of P for the symmetric beampattern is $n_T/2 + 1$. As a safe margin we suggest to use $P = n_T/2 + 2$ as an optimum value.

In both of the above methods, if $\mathbf{W}(\boldsymbol{\psi})$ is real, $\mathbf{W}^H(\boldsymbol{\psi})\mathbf{W}(\boldsymbol{\psi})$ will also be real and only capable of generating symmetric beampatterns. To obtain a non-symmetric beampattern, $\mathbf{W}^H(\boldsymbol{\psi})\mathbf{W}(\boldsymbol{\psi})$ should be a complex matrix. This requires the weight matrix to be a complex and to fulfill the equal elemental power constraint the norm of each of its column should be equal to one. To generate complex weight matrices for the non-symmetric beampatterns, a theorem is defined in the following.

Theorem 1: If $\mathbf{W}(\boldsymbol{\psi})$ and $\mathbf{W}(\boldsymbol{\phi})$ are real matrices each of unit norm column-vectors for any $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$, then the matrix

$$\mathbf{W}(\boldsymbol{\psi}, \boldsymbol{\phi}) = \frac{1}{\sqrt{2}}\Big(\mathbf{W}(\boldsymbol{\psi}) + j\mathbf{W}(\boldsymbol{\phi})\Big) \tag{16}$$

will be complex and the norm of each of its column-vectors will be equal to one.

Proof: It is known that

$$\mathbf{W}^H(\boldsymbol{\psi}, \boldsymbol{\phi})\mathbf{W}(\boldsymbol{\psi}, \boldsymbol{\phi}) = \frac{1}{2}\Big(\mathbf{W}^T(\boldsymbol{\psi})\mathbf{W}(\boldsymbol{\psi}) + \mathbf{W}^T(\boldsymbol{\phi})\mathbf{W}(\boldsymbol{\phi})\Big) + \frac{j}{2}\Big(\mathbf{W}^T(\boldsymbol{\psi})\mathbf{W}(\boldsymbol{\phi}) - \mathbf{W}^T(\boldsymbol{\phi})\mathbf{W}(\boldsymbol{\psi})\Big). \tag{17}$$

In (17), it can be noted that the diagonal elements of both matrices in the upper term are equal to one, while the diagonal elements of both matrices in the lower term are equal. Therefore, the diagonal elements of the matrix $\mathbf{W}^H(\boldsymbol{\psi}, \boldsymbol{\phi})\mathbf{W}(\boldsymbol{\psi}, \boldsymbol{\phi})$ are all equal to one. This proves that the norm of each column of the complex matrix $\mathbf{W}(\boldsymbol{\psi}, \boldsymbol{\phi})$ is also equal to one.

It should be noted here that, in the weight matrix of method-1, if all the angles in $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ are equal to zero then $\mathbf{W}(\boldsymbol{\psi}, \boldsymbol{\phi})$ will become a diagonal matrix and each of its diagonal elements will be equal

to $(1+j)/\sqrt{2}$. In this case, the proposed system model with the weight matrix of method-1 will act as a MIMO radar. On the other hand, in the weight matrix of method-1, if all the angles in $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ are equal to $\pi/2$ then all the elements in the first row of $\mathbf{W}(\boldsymbol{\psi}, \boldsymbol{\phi})$ will be equal to $(1+j)/\sqrt{2}$, while all the remaining elements will become equal to zero. This converts the proposed model into a phased-array, where all the antennas transmit fully correlated waveforms.

The advantage of using these weight matrices is that angles in $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ can be freely optimised to minimise the cost-function in (9) for the desired beampattern. To optimise $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ any gradient-based algorithm can be used. An important point to be noted here is that the cost-function given in (9) is a function of $\mathbf{W}(\boldsymbol{\psi}, \boldsymbol{\phi})$, which is a function of $\sin(\cdot)$ and $\cos(\cdot)$ terms. Therefore, there can be a question of the convexity of the cost-function. In the appendix A, we prove that the cost-function is convex. Moreover, in the simulation section, we show that with different initialisation values the algorithm converges to the same minimum point. In the sequel for brevity $\mathbf{W}$ will be used without its parameters $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$.

The overall benefits of the proposed algorithm are:

- There is no need to synthesise the waveform covariance matrix nor the waveforms to realise it for the desired beampattern.
- To obtain non-symmetric beampatterns the covariance matrix should be a complex. The SQP cannot work with complex numbers, therefore it requires the problem to be converted into a real problem before optimisation. The proposed method does not have any problem in optimising complex cost-functions; therefore non-symmetric beampatterns can also be easily obtained.
- A simple gradient-based algorithm can be used to find the weight matrix $\mathbf{W}$, without imposing any constraints.
- The equal elemental average power constraint is implicitly fulfilled.

In this work, to optimise the weight matrix, $\mathbf{W}$, for the desired beampattern, the Newton gradient algorithm is used due to its fast convergence speed. The derivations of the Newton gradient algorithm for this problem are given in the Appendix A.

IV. Computational Complexity

For the symmetric beampattern, in each iteration for given ψ, the formation of weight matrix of method-1 requires $(n_T-1)(n_T-2)$ real multiplications, while the weight matrix of method-2 requires $2n_T(P-2)$ real multiplications. In applying Newton-gradient algorithm, the main computational complexity involved is in finding $P(\Theta,\theta_k)=\mathbf{a}_t^H(\theta_k)\mathbf{W}^H\mathbf{W}\mathbf{a}_t(\theta_k)$, which for weight matrix of method-1 requires $n_T(n_T+1)$ and method-2 requires $2n_T(P+1)$ real multiplications. In each iteration, $P(\Theta,\theta_k)$ is required for each value of θ_k. The second main computational complexity is in finding the Newton gradient, $\mathbf{\Delta}(\Theta^i)$, which can be computed in $\mathcal{O}\left(\left(\frac{n_T^2-n_T}{2}\right)^3\right)$ complex multiplications. Therefore, the approximate overall computational complexity of our proposed algorithm *per iteration* for the weight matrix of method-1 is $\mathcal{C}_1 = \mathcal{O}\left(Kn_T(n_T+1)\right)+\mathcal{O}\left(4\left(\frac{n_T^2-n_T}{2}\right)^3\right)$ and for the weight matrix of method-2 is $\mathcal{C}_2 = \mathcal{O}(2Kn_T(P+1))+\mathcal{O}\left(4\left(\frac{n_T^2-n_T}{2}\right)^3\right)$ real multiplications.

On the other hand, to synthesise the covariance matrix, $\mathbf{R}$, for the desired beampattern algorithm in [9] uses SQP, whose complexity for a prefixed accuracy of η is $\mathcal{O}(\log(1/\eta)n_T^{3.5})$ [19]. Once the covariance matrix $\mathbf{R}$ is synthesised, for the next step, to design the actual waveforms the algorithm in [10], for each iteration requires the singular-value-decomposition (SVD) of an $n_T\times N$ matrix, which require $\mathcal{O}(n_T^2N+N^3)$ complex multiplications [20]. After the SVD, this algorithm finds a unitary matrix and updates the initial random variables, which again requires $\mathcal{O}(n_T^2N)$ complex multiplications. Then the second stage of this algorithm from [21] at each iteration requires $\mathcal{O}(N)$ complex multiplications. Therefore, the overall computational complexity *per iteration* of the waveforms design, *per iteration*, is $\mathcal{O}\left(N+3n_T^2N+N^3\right)$ complex multiplications.

By comparing the computational complexity of both algorithms, it can be noticed that if $N>(\frac{n_T^2-n_T}{2})$ the computational complexity of our proposed algorithm will be much lower than synthesising the covariance matrix and then designing the waveforms.

V. Peak-to-Average Power Ratio at an Antenna

The PAPR at the transmit antenna q is defined as

$$\text{PAPR}=\frac{P_{\text{max}}(q)}{P_{\text{ave}}(q)}.$$

Both of the weight matrices guarantee unit average power. However, the instantaneous power may vary at each antenna, therefore, the PAPR at antenna q depends only on the maximum value of instantaneous power. One drawback associated with this scheme is the possibility of more than unity PAPR at the antennas. In this section, an upper bound on the PAPR is derived for each antenna. The instantaneous power transmitted from the antenna q can be written as

$$\begin{aligned} P_{\text{inst}}(q) &= \left| \sum_{p=1}^{n_T} w_{pq} x_p(n) \right|^2, \\ &= \mathbf{w}_q^H \mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w}_q, \\ &= |\mathbf{w}_q^H \mathbf{x}(n)|^2 \leq |\mathbf{w}_q|^2 |\mathbf{x}(n)|^2. \end{aligned} \tag{18}$$

Since $|\mathbf{w}_q|^2 = 1$, and for unit power BPSK or QPSK symbols $|\mathbf{x}(n)|^2 = n_T$, the $P_{\text{inst}}(q)$ will be less than n_T.

A. PAPR Using the Weight Matrix of Method-1

In the weight matrix of method-1, first column has only one non-zero element and $w_{p1} = 0$ for $p > 1$. Therefore, first antenna can transmit only one waveform and using (18) the instantaneous power at the first antenna can be written as

$$P_{ins}(1) = |w_{11} x_1(n)|^2 \leq |w_{11}|^2 |x_1(n)|^2 = 1.1. \tag{19}$$

Second column of the weight matrix of method-2 has only two non-zero elements, and $w_{p2} = 0$ for $p > 2$. Therefore, second antenna can transmit only two waveform and the instantaneous power at the second antenna can be written as

$$\begin{aligned} P_{ins}(2) &= \left| \sum_{p=1}^{2} w_{p2} x_p(n) \right|^2, \\ &= \left| \begin{bmatrix} w_{12} & w_{22} \end{bmatrix} \begin{bmatrix} x_1(n) \\ x_2(n) \end{bmatrix} \right|^2, \\ &\leq |\mathbf{w}_2|^2 |\mathbf{x}_2(n)|^2 = 1.2, \end{aligned} \tag{20}$$

where $\mathbf{x}_2(n) = \begin{bmatrix} x_1(n) & x_2(n) \end{bmatrix}^T$. Similarly, the last column of weight matrix of method-1 has n_T elements, therefore, the instantaneous power at the n_Tth antenna can be written as

$$\begin{aligned} P_{ins}(n_T) &= \left| \sum_{p=1}^{n_T} w_{pn_T} x_p(n) \right|^2, \\ &= \left| \begin{bmatrix} w_{1n_T} & \cdots & w_{n_T n_T} \end{bmatrix} \begin{bmatrix} x_1(n) \\ \vdots \\ x_{n_T}(n) \end{bmatrix} \right|^2, \\ &\leq |\mathbf{w}_{n_T}|^2 |\mathbf{x}(n)|^2 = 1.n_T. \end{aligned} \tag{21}$$

Therefore, the upper bound on the maximum value of the transmitted power from antenna q can be written as

$$P_{\max}(q) = \leq q. \tag{22}$$

In (22), equality will hold only when $\mathbf{w}_q = \mathbf{x}(n)$, which is only possible at the first antenna. Therefore, the maximum PAPR at transmit antenna one is 1, at transmit antenna two is less than 2, at transmit antenna three is less than 3, and so on. To further appreciate this point, for BPSK symbols the instantaneous transmitted power from antenna q, using (18), can be written as

$$P_{\text{inst}}(q) = \tilde{\mathbf{w}}_q^H \begin{bmatrix} x_1(n)x_1^*(n) & \cdots & x_1(n)x_m^*(n) \\ \vdots & \ddots & \vdots \\ x_q(n)x_1^*(n) & \cdots & x_q(n)x_q^*(n) \end{bmatrix} \tilde{\mathbf{w}}_q, \tag{23}$$

where $\tilde{\mathbf{w}}_q = [w_{1q}\ w_{2q}\ \ldots\ w_{qq}]^T$. Since $x_m(n)x_m^*(n) = 1$ for $m = 1, \ldots, q$ and $x_m(n)x_n^*(n) = \pm 1$ for $m \neq n$, the instantaneous transmitted power from antenna q can be written as

$$\begin{aligned} P_{\text{inst}}(q) &= \mathbf{w}_q^H \mathbf{w}_q + 2\Re\Big(\pm w_{1q}^* w_{2q} \cdots \pm w_{1q}^* w_{qq} + \cdots \pm w_{(q-1)q}^* w_{qq} \Big) \\ &= 1 + 2\Re\Big(\pm w_{1q}^* w_{2q} \cdots \pm w_{1q}^* w_{qq} + \cdots \pm w_{(q-1)q}^* w_{qq} \Big). \end{aligned} \tag{24}$$

It should be noted here that the maximum value of $\sin(\cdot)$ or $\cos(\cdot)$ is equal to one. Since w_{nm} is the function of the product of at-least $(m-n)$ $\sin(\cdot)$ and/or $\cos(\cdot)$ terms for $m \neq n$, it is expected that its value will be much less than one and the product of two such numbers w_{nq} and w_{mq} will be even less than the individual numbers. Moreover, as the index q increases due to the greater number of terms in

w_{nq}, its value can be close to zero. On the basis of these observations, it can be expected with high probability that the maximum value of PAPR will be even less than q and for higher antenna index, q, it will be even lower.

B. PAPR Using the Weight Matrix of Method-2

In the weight matrix of method-2, each column-vector contains only P non-zero elements. Therefore, using (18), the instantaneous power at any transmit antenna q can be written as

$$\begin{aligned} P_{ins}(q) &= \left| \sum_{p=1}^{P} w_{pq} x_{qp}(n) \right|^2, \\ &= \left| \begin{bmatrix} w_{1q} & \cdots & w_{Pq} \end{bmatrix} \begin{bmatrix} x_{q1}(n) \\ \vdots \\ x_{qP}(n) \end{bmatrix} \right|^2, \\ &\leq |\mathbf{w}_q|^2 |\mathbf{x}_q(n)|^2 = 1.P, \end{aligned} \tag{25}$$

where $\mathbf{x}_q(n) = \begin{bmatrix} x_{q1}(n) & \cdots & x_{qP}(n) \end{bmatrix}^T$ and the subscript $qp = \langle q + p - 2 \rangle_{n_T} + 1$. Here, in contrast to the weight matrix of method-1, the upper bound on the maximum power transmitted from any antenna is P. Due to the reasons discussed in the previous section, it can be said that the actual value of PAPR will be much less than P.

Perfectly orthogonal BPSK and QPSK waveforms can be very easily generated by exploiting Hadamard codes [22]. For example, a Hadamard matrix of order 8 can be generated as

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}. \tag{26}$$

Similarly, high order Hadamard matrices can be easily generated. For BPSK waveforms the pth column of $\mathbf{H}$ can be considered as the symbols of waveform p. To generate n_T waveforms, each of N BPSK symbols, can be obtained by assigning first n_T columns of the Hadamard matrix of order N to the corresponding waveforms. Orthogonal BPSK waveforms can also be easily generated by first generating a matrix of $N \times n_T$ independent Gaussian random variables and then taking the sign of it.

VI. Simulation

In this section, to validate the performance of the proposed algorithm, some numerical examples are presented. In all of the following simulations to achieve the desired beampattern a uniform linear array of 10 elements with half-wavelength inter-element spacing is used. For the proposed scheme, to generate n_T independent and orthogonal BPSK waveforms, a 64×64 Hadamard matrix is generated and the first n_T columns are assigned to n_T waveforms. To optimise the weight matrix $\mathbf{W}$ for the desired beampattern, the Newton gradient algorithm is used. The results of the proposed algorithm are compared with the algorithm in [9] and [10]. The algorithm in [9], uses SQP to synthesise the covariance matrix for the desired beampattern. The algorithm in [10] generates non-finite alphabet constant-envelope (N-FACE) waveforms to realise the synthesised covariance matrix, which are challenging to use in practice. In all of the simulation, for each waveform 64 symbols are generated.

In the first simulation, a symmetric beampattern of 60-degrees beam-width is desired. To achieve this beampattern a covariance matrix is synthesised and to realise this covariance matrix, waveforms of different PAPRs are generated. In Fig. 2, the beampattern obtained using the covariance matrix method is compared with the beampattern of our proposed algorithm, which is obtained using the weight matrix of method-1. It can be seen in the figure that both algorithms identically match the desired beampattern. This is the optimum beampattern matching that can be achieved using the 2-norm optimisation. In Fig. 3, the beampattern of the N-FACE waveforms generated to realise the covariance matrix is compared with our proposed algorithm. In the figure, it can be seen that for PAPR $= 1$, the beampattern of the waveforms has higher side-lobe-levels. By increasing the PAPR $= 2$, the waveforms match the optimum beampattern but do not retain constant-envelope. Thus, our proposed algorithm provides the optimum beampattern matching in only one step unconstrained optimisation.

Similarly, in the second simulation, the beampattern of three main-lobes each of 20 degrees beam-width at locations -40, 0, and 40 degrees is desired. In Fig. 4, the beampattern obtained using covariance matrix method is compared with the beampattern of our proposed algorithm, which is again obtained using the weight matrix of method-1. It can be seen in the figure that both of the algorithms identically match the desired beampattern. In Fig. 5, the beampattern of N-FACE waveforms designed to realise the covariance matrix with PAPR $= 1$ and PAPR $= 2$ is compared with our proposed algorithm. It can be seen here again that with PAPR $= 1$, the beampattern of waveforms does not match the optimum designed beampattern.

For the desired beampatterns, in both of the above simulations, the proposed algorithm using the weight matrix of method-2, with the value of $P = 7$, also exactly match the optimum designed beampattern. The corresponding beampatterns are not shown, as they will not provide any extra information.

In the third simulation, a non-symmetric beampattern of beam-width 40 degrees, located at -30 degrees, is desired. This beampattern is difficult to design using SQP method of synthesising the covariance matrix for the desired beampattern. In Fig. 6, the beampattern obtained using the weight matrix of method-1 is compared with the beampattern obtained using the weight matrix of method-2. It can be seen in the figure that with $P = 5$, due to the less DOF used, the designed beampattern is not a good match of the desired beampattern. However, the beampattern obtained using the weight matrix of method-1 and the beampattern obtained using the weight matrix of method-2 with $P = 8$ and $P = 10$ are almost similar. The MSE between the desired and designed beampatterns using the weight matrix of method-1 is 8.35, while using the weight matrix of method-2 with the value of $P = 8$ and 10 respectively is 8.29 and 8.27. Here, compared to symmetric beampattern, weight matrix of method-2 yields slightly better MSE. The reason for this is that for non-symmetric beampatterns, the extra DOF of the weight matrix of method-2 provides better beampattern match. On the basis of these observations, we can say that for non-symmetric beampatterns the optimal value of $P = n_T$. However, the use of extra DOF of the weight matrix of method-2 does not have any benefit in the design of symmetric beampatterns.

In Fig. 7, the convergence behavior of the proposed algorithm for the desired beampatterns, in the above simulations, using the weight matrix of method-1 and -2 is shown. For the weight matrix of method-2, the value of $P = 7$. The algorithm is initialised with all values of $\boldsymbol{\psi}, \boldsymbol{\phi}$ and α equal to zero for all of the

desired beampatterns. It can be seen in the figure that the proposed algorithm converges in less than 100 iterations. It can be seen in the figure that for symmetric beampatterns both weight matrices converge at the same point, while for non-symmetric beampattern weight matrix of method-2 performs slightly better than weight matrix of method-1. The reason for this is already mentioned above. It should be noted here that the points of convergence of the proposed algorithm for the symmetric beampatterns are the same as obtained using the convex optimisation in [9].

In the appendix A, it is proved that the cost-function given in (9) is convex. To show this point by simulations, Figures 8 and 9 shows the convergence behavior of the proposed algorithm with 50 independent initial values. Here, the weight matrix of method-1 is optimised for three main-lobe and non-symmetric beampatterns. It can be seen in the figures that the proposed algorithm converges to the same minimum point, although the cost-function is a function of non-linear functions. Same convergence behavior is achieved, when weight matrix of method-2 is optimised for the desired beampatterns with independent initial values.

In the final simulation, the PAPR of the proposed algorithm at different antennas is determined. Fig.10 shows the corresponding PAPRs obtained using both of the weight matrices, here for the weight matrix of method-2 the value of $P = 7$. It can be seen in the figure that the PAPR, using the weight matrix of method-1, irrespective of the desired beampattern at the first antenna is equal to one, at the second antenna is less than two, and so on. Actually, as explained in Sec. V-A and can also be seen in the figure, the PAPR is significantly lower than the antenna index. On the other hand, the PAPR using the weight matrix of method-2 never exceeds 7 at any antenna. In most of the cases, the PAPR using weight matrix of method-2 is also significantly lower than the value of P.

To further explore the PAPR for the large number of antennas, a simulation for the desired symmetric beampattern of three main-lobes is performed with 50 antennas. In the simulation weight matrix of method-1 and method-2 is optimised for the desired beampattern, both weight matrices converge to the same point. For the weight matrix of method-2 the value of $P = 30$. The corresponding PAPRs are shown in Fig. 11. It can be seen in the figure that the PAPR using the weight matrix of method-1 has lower values at the lower and higher index antennas, while it has moderate values at the middle index antennas, as it was expected in our numerical results. However, compare to the upper bound on the

antenna index the value of PAPR is significantly lower. Similarly, the value of PAPR using the weight matrix of method-2 is identical throughout. Here, again the PAPR at any antenna is very less than the upper bound P.

VII. Conclusions

In this work a simple algorithm to obtain a desired beampattern using MIMO radar is proposed. The proposed algorithm converts the conventional two step constrained optimisation problem of constant-envelope waveform design, for the desired beampattern, into a one step unconstrained optimisation problem. In the proposed scheme n_T independent practical finite-alphabet constant-envelope waveforms are generated and pre-processed before transmitting from the antennas. It also facilitates the use of Newton-gradient techniques for the optimization of cost-functions. The weight matrices of the proposed scheme guarantee equal average power transmission from each antenna and can yield symmetric and non-symmetric beampatterns. Simulation results show that the proposed scheme outperforms current methods in both computational complexity and performance. The only drawback of the proposed scheme is that the resulting PAPR may be greater than unity. However, it is shown that this PAPR is bounded and is very less than the antenna index with high probability.

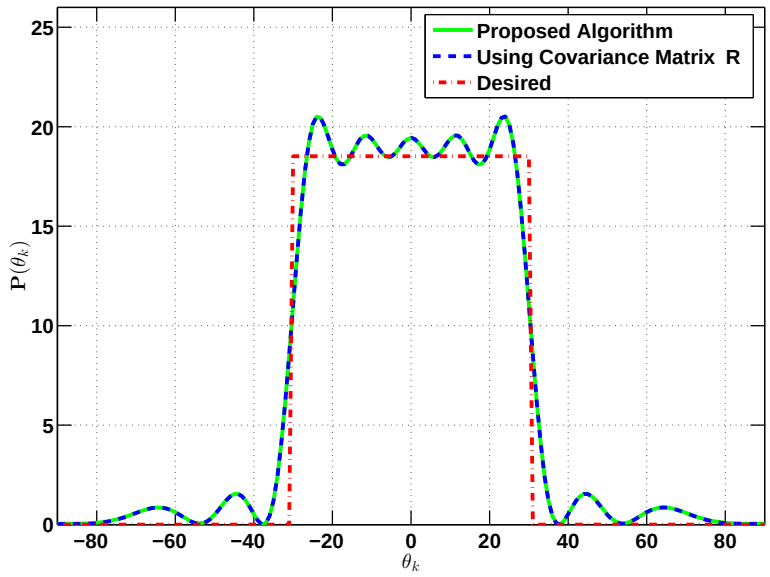

Fig. 2. Beampattern of the proposed algorithm is compared with the beampattern obtained by synthesising the covariance matrix, $\mathbf{R}$, for the desired beampattern, using the algorithm in [9].

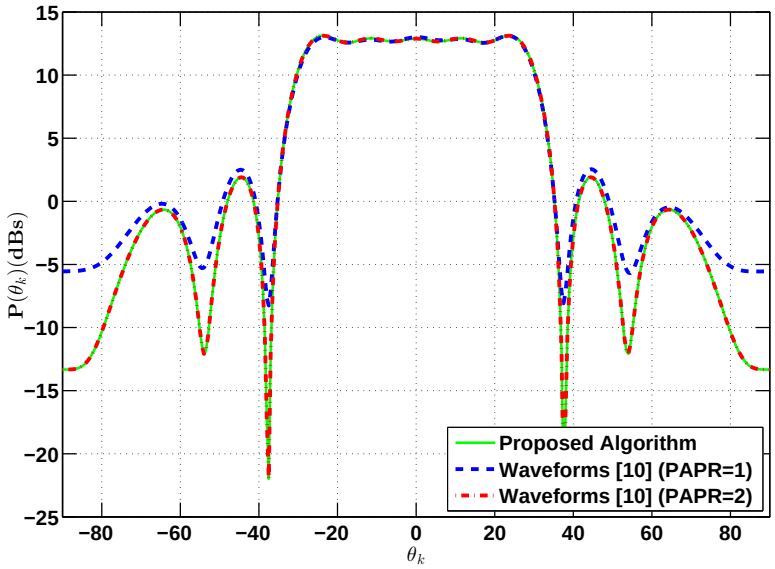

Fig. 3. Beampattern of the proposed algorithm is compared with the beampattern of the waveforms of PAPR $= 1$ and PAPR $= 2$ generated using [10] to realise $\mathbf{R}$. For both algorithms each waveform transmit 64 symbols.

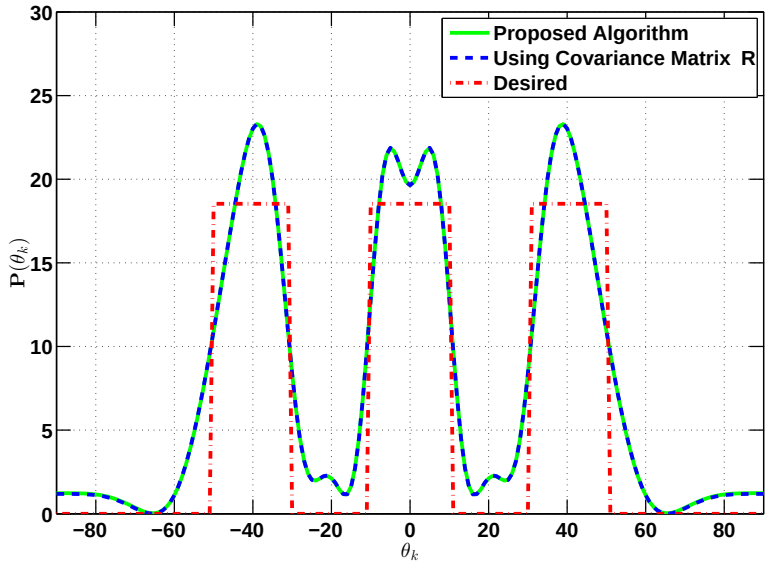

Fig. 4. Beampattern of the proposed algorithm is compared with the beampattern obtained by synthesising the covariance matrix, $\mathbf{R}$, for the desired beampattern, using the algorithm in [9].

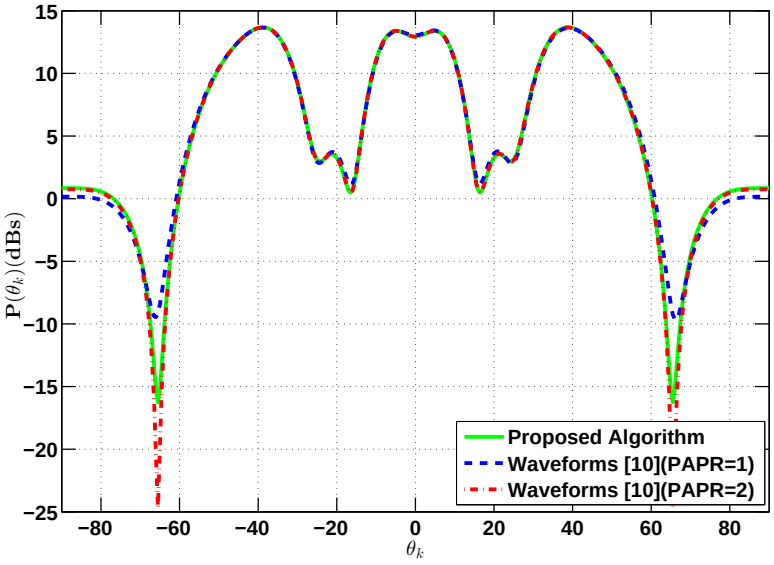

Fig. 5. Beampattern of the proposed algorithm is compared with the beampattern of the waveforms of PAPR $= 1$ generated using [10] to realise $\mathbf{R}$. For both algorithms each waveform transmit 64 symbols.

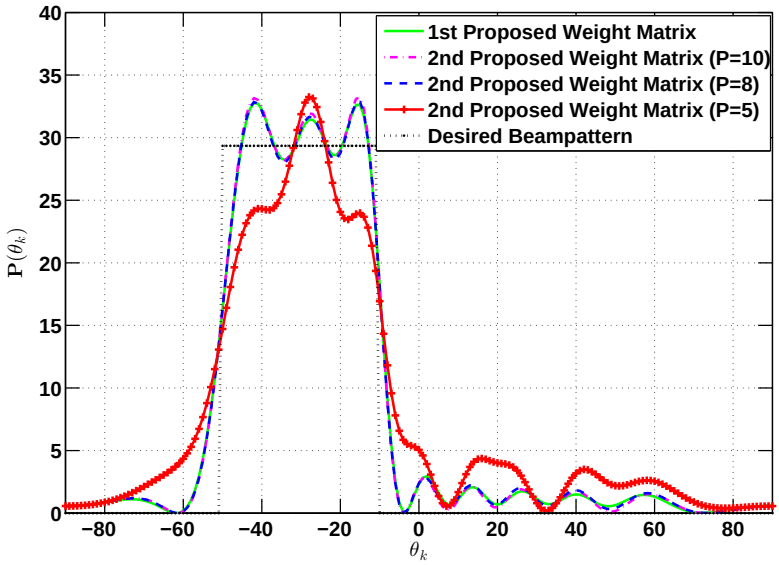

Fig. 6. Comparison of non-symmetric beampatterns obtained using the weight matrix of method-1 and method-2. The impact of increasing the value of P can also be seen in the figure.

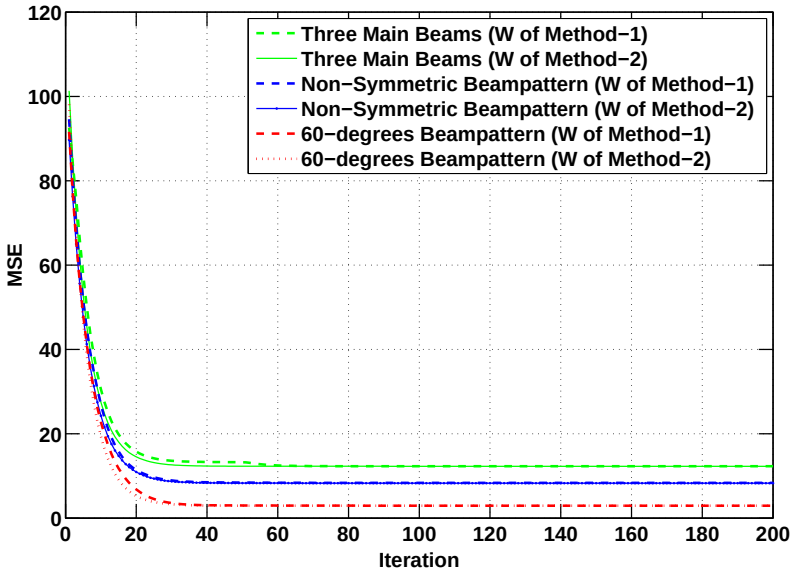

Fig. 7. Convergence behavior of the Newton gradient algorithm for all of the three desired beampatterns using the 1st and 2nd proposed weight matrices. For the weight matrix of method-2, the value of $P = 7$.

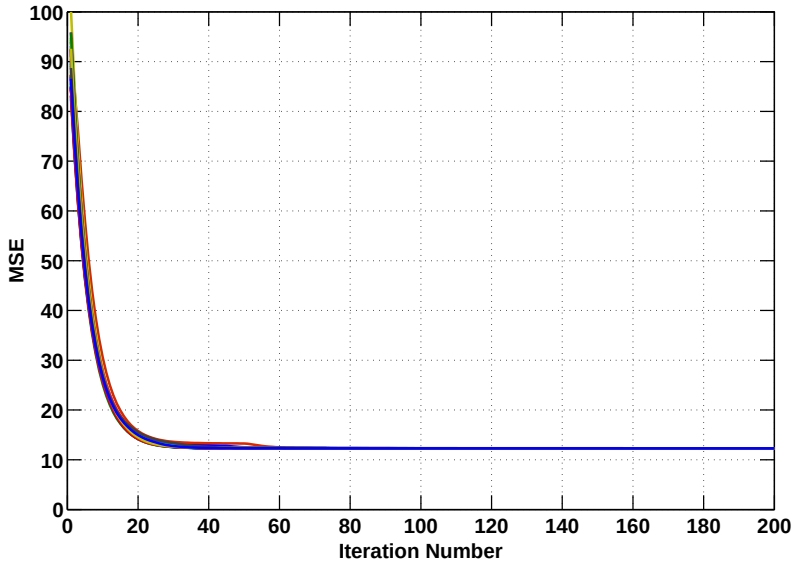

Fig. 8. Convergence behavior of the Newton gradient algorithm with 50 independent different initialisations for three main beam desired beampattern.

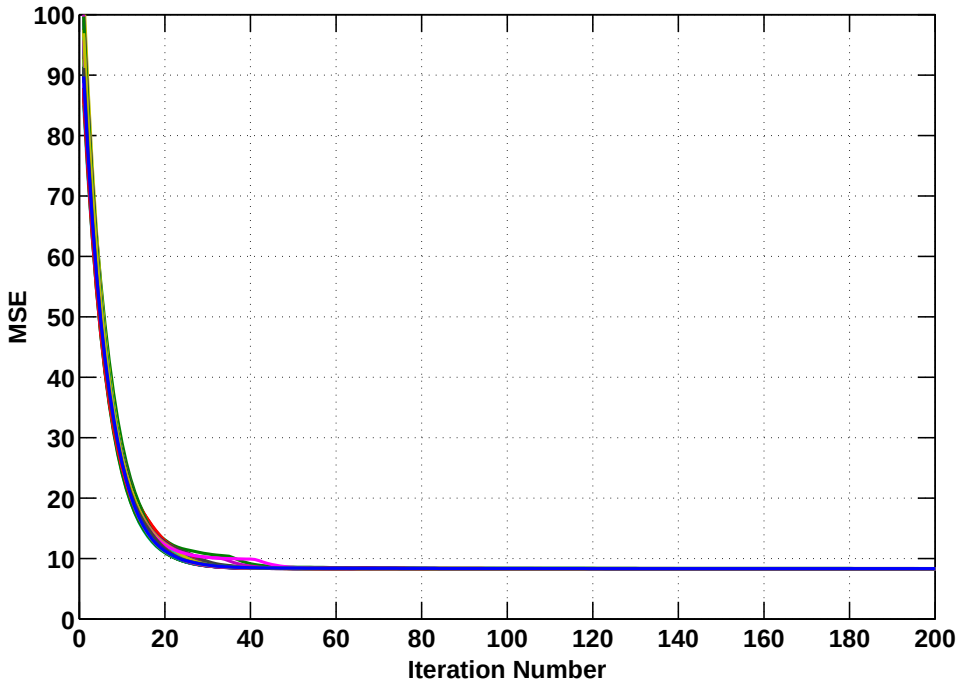

Fig. 9. Convergence behavior of the Newton gradient algorithm with 50 independent different initialisations for non-symmetric desired beampattern.

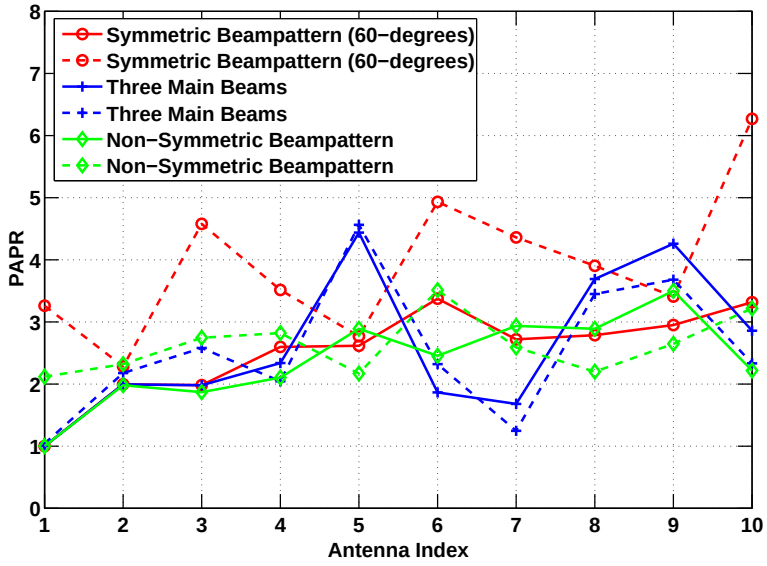

Fig. 10. The PAPR using the weight matrix of method-1 ($-$) and weight matrix of method-2 ($--$). Here, for the weight matrix of method-2 $P = 7$.

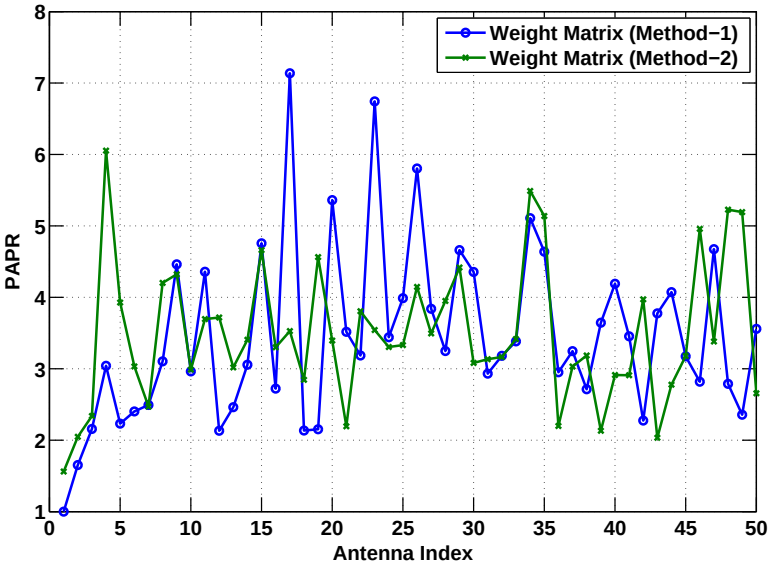

Fig. 11. The trend of PAPR using the weight matrix of method-1 and method-2. Here, the number of transmit antenna is 50 and for the weight matrix of method-2 the value of $P = 30$.

APPENDIX A
COST-FUNCTION CONVEXITY PROOF

The convexity condition for a function $f(x)$ is given by [23]

$$\lambda f(x_1) + (1-\lambda) f(x_2) \quad \geq \quad f(\lambda x_1 + (1-\lambda) x_2)$$

or

$$\lambda f(x_1) + (1-\lambda) f(x_2) \quad - \quad f(\lambda x_1 + (1-\lambda) x_2) \geq 0,$$

where $1 \geq \lambda \geq 0$. Therefore, for (9) to be convex, the following condition must be satisfied

$$\lambda J(\mathbf{W}(\Theta_1)) + (1-\lambda) J(\mathbf{W}(\Theta_2)) - J(\lambda \mathbf{W}(\Theta_1) + (1-\lambda)\mathbf{W}(\Theta_2)) \geq 0, \tag{A.1}$$

where $\Theta_m = \{\boldsymbol{\psi}_m, \boldsymbol{\phi}_m\}$. It is since $\mathbf{W}^H(\Theta_m)\mathbf{W}(\Theta_m)$ is positive semi-definite, the term $\mathbf{a}_t^H(\theta_k)\mathbf{W}^H(\Theta_p)\mathbf{W}(\Theta_p)\mathbf{a}_t(\theta_k)$ will be real and non-negative. Therefore, by assuming $\varrho(\theta_k, \Theta_p) = \mathbf{a}_t^H(\theta_k)\mathbf{W}^H(\Theta_p)\mathbf{W}(\Theta_p)\mathbf{a}_t(\theta_k)$, the first term of (A.1) can be written as

$$\begin{aligned}
&\lambda J(\mathbf{W}(\Theta_1)) + (1-\lambda) J(\mathbf{W}(\Theta_2)) \\
&= \frac{1}{K}\sum_{k=1}^{K} \Big(\lambda \Big(\varrho^2(\theta_k, \Theta_1) - 2\alpha P_d(\theta_k)\varrho(\theta_k, \Theta_1) + \alpha^2 P_d(\theta_k) \Big) \\
&\quad + (1-\lambda) \Big(\varrho^2(\theta_k, \Theta_2) - 2\alpha P_d(\theta_k)\varrho(\theta_k, \Theta_2) + \alpha^2 P_d(\theta_k) \Big) \Big) \\
&= \frac{1}{K}\sum_{k=1}^{K} \Big(\lambda \Big(\varrho^2(\theta_k, \Theta_1) - 2\alpha P_d(\theta_k)\varrho(\theta_k, \Theta_1) \Big) \\
&\quad + (1-\lambda) \Big(\varrho^2(\theta_k, \Theta_2) - 2\alpha P_d(\theta_k)\varrho(\theta_k, \Theta_2) \Big) + \alpha^2 P_d(\theta_k) \Big).
\end{aligned} \tag{A.2}$$

Similarly, the second term of (A.1) can be written as

$$\begin{aligned}
&J(\lambda \mathbf{W}(\Theta_1) + (1-\lambda)\mathbf{W}(\Theta_2)) \\
&\quad = \frac{1}{K}\sum_{k=1}^{K} \Big(\lambda \varrho(\theta_k, \Theta_1) + (1-\lambda)\varrho(\theta_k, \Theta_2) - \alpha P_d(\theta_k) \Big)^2 \\
&\quad = \frac{1}{K}\sum_{k=1}^{K} \Big(\lambda^2 \varrho^2(\theta_k, \Theta_1) + (1-\lambda)^2 \varrho^2(\theta_k, \Theta_2) + \alpha^2 P_d(\theta_k) \\
&\qquad + 2\lambda(1-\lambda)\varrho(\theta_k, \Theta_1)\varrho(\theta_k, \Theta_2) - 2\alpha\lambda P_d(\theta_k)\varrho(\theta_k, \Theta_1) \\
&\qquad - 2\alpha(1-\lambda) P_d(\theta_k)\varrho(\theta_k, \Theta_2) \Big).
\end{aligned} \tag{A.3}$$

Using (A.2) and (A.3), the convexity condition in (A.1) can be written as

$$\begin{aligned}
&\lambda J(\mathbf{W}(\Theta_1))+(1-\lambda)J(\mathbf{W}(\Theta_2))-J(\lambda\mathbf{W}(\Theta_1)+(1-\lambda)\mathbf{W}(\Theta_2))\\
=&\frac{1}{K}\sum_{k=1}^{K}\Bigg(\lambda\Big(\varrho^2(\theta_k,\Theta_1)-\cancel{2\alpha P_d(\theta_k)\varrho(\theta_k,\Theta_1)}\Big)\\
&+(1-\lambda)\Big(\varrho^2(\theta_k,\Theta_2)-\cancel{2\alpha P_d(\theta_k)\varrho(\theta_k,\Theta_2)}\Big)\\
&-\lambda^2\varrho^2(\theta_k,\Theta_1)-(1-\lambda)^2\varrho^2(\theta_k,\Theta_2)+\cancel{2\alpha\lambda P_d(\theta_k)\varrho(\theta_k,\Theta_1)}\\
&-2\lambda(1-\lambda)\varrho(\theta_k,\Theta_1)\varrho(\theta_k,\Theta_2)+\cancel{2\alpha(1-\lambda)P_d(\theta_k)\varrho(\theta_k,\Theta_2)}\Bigg)\\
=&\frac{\lambda(1-\lambda)}{K}\sum_{k=1}^{K}\Big(\varrho^2(\theta_k,\Theta_1)+\varrho^2(\theta_k,\Theta_2)-2\varrho(\theta_k,\Theta_1)\varrho(\theta_k,\Theta_2))\Big)\\
=&\frac{\lambda(1-\lambda)}{K}\sum_{k=1}^{K}\Big(\varrho(\theta_k,\Theta_1)-\varrho(\theta_k,\Theta_2))\Big)^2\geq 0,
\end{aligned}$$

which proves that the cost-function in (9) is convex.

APPENDIX B
NEWTON-GRADIENT ALGORITHM

To apply the Newton-gradient algorithm, the cost-function in (9) can be written as

$$J(\Theta)=\mathbf{\Gamma}^H(\Theta)\mathbf{\Gamma}(\Theta), \tag{B.1}$$

where

$$\mathbf{\Gamma}(\Theta)=\frac{1}{\sqrt{K}}\begin{bmatrix} P(\Theta,\theta_1)-\alpha P_d(\theta_1)\\ P(\Theta,\theta_2)-\alpha P_d(\theta_2)\\ \vdots\\ P(\Theta,\theta_K)-\alpha P_d(\theta_K)\end{bmatrix},$$

$P(\Theta,\theta_k)=\mathbf{a}_t^H(\theta_k)\mathbf{W}^H\mathbf{W}\mathbf{a}_t(\theta_k)$, and $\Theta=\{\boldsymbol{\psi},\boldsymbol{\phi}\}$. The updated parameter vector Θ that minimizes the cost-function $J(\Theta)$ can be found using the generalized Newton-gradient algorithm given by [24] as

$$\Theta^{i+1}=\Theta^i+\mu\mathbf{\Delta}(\Theta^i), \tag{B.2}$$

where μ is the adaptation gain and the Newton-gradient $\mathbf{\Delta}(\Theta)$ is defined as

$$\begin{aligned}
\mathbf{\Delta}(\Theta^i) &= -\left[\frac{d^2J(\Theta^i)}{d{\Theta^i}^2}\right]^{-1}\left[\frac{dJ(\Theta^i)}{d\Theta^i}\right]\\
&= -\left[\frac{d\mathbf{\Gamma}(\Theta^i)}{d\Theta^i}\left(\frac{d\mathbf{\Gamma}(\Theta^i)}{d\Theta^i}\right)^H\right]^{-1}\frac{d\mathbf{\Gamma}(\Theta^i)}{d\Theta^i}\mathbf{\Gamma}(\Theta^i).
\end{aligned}$$

The differentiation of $\mathbf{\Gamma}(\Theta^i)$ with respect to Θ^i can be written as

$$\frac{d\mathbf{\Gamma}(\Theta^i)}{d\Theta^i} = \frac{1}{\sqrt{K}} \begin{bmatrix} \frac{dP(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_1)}{d\boldsymbol{\psi}^i} & \cdots & \frac{dP(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_K)}{d\boldsymbol{\psi}^i} \\ \frac{dP(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_1)}{d\boldsymbol{\phi}^i} & \cdots & \frac{dP(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_K)}{d\boldsymbol{\phi}^i} \\ -P_d(\theta_1) & \cdots & -P_d(\theta_K) \end{bmatrix}, \tag{B.3}$$

where

$$\frac{dP(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{d\boldsymbol{\psi}^i} = \begin{bmatrix} \frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \psi^i_{21}} \\ \frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \psi^i_{31}} \\ \frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \psi^i_{32}} \\ \\ \vdots \\ \\ \frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \psi^i_{n_T,n_T-1}} \end{bmatrix}$$

$$\text{and} \quad \frac{dP(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{d\boldsymbol{\phi}^i} = \begin{bmatrix} \frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \phi^i_{21}} \\ \frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \phi^i_{31}} \\ \frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \phi^i_{32}} \\ \\ \vdots \\ \\ \frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \phi^i_{n_T,n_T-1}} \end{bmatrix}.$$

Here, the differentiation shown is with respect to the spherical coordinates of first weight matrix, for the differentiation with respect to the spherical coordinates of second weight matrix same strategy can be adopted. It is difficult to write a closed-form expression for $\mathbf{\Delta}(\Theta)$; therefore an approximate gradient was evaluated in [25] by finite element approximation. The same approach is considered in this paper. The approximation follows the lines given in TABLE I below :

REFERENCES

[1] E. Fishler, A. Haimovich, R. S. Blum, L. J. Cimini, D. Chizhik, and R. A. Valenzuela, "MIMO Radar: An idea whose time has come," *In Proc. International Radar Conference, Philadelphia, PA, USA*, pp. 71–78, Apr. 2004.

[2] E. Fishler, A. Haimovich, R. S. Blum, L. J. Cimini, D. Chizhik, and R. A. Valenzuela, "Spatial diversity in Radars – Models and detection performance," *IEEE Trans. Signal Processing*, vol. 54, pp. 823–838, Mar. 2006.

TABLE I

1. Start at some initial value Θ^0 and form $\mathbf{\Gamma}(\Theta^0)$.
2. for $i = 0 : N_i$

   The individual partial differentiation is evaluated using backward difference as

   $$\frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \psi^i_{mn}} \approx -\frac{1}{\varepsilon^i}\Big(P(\theta_k;\psi^i_{21},\phi^i_{21}\cdots,\psi^i_{mn}+\varepsilon^i,\cdots) - P(\theta_k;\psi^i_{21},\phi^i_{21},\cdots,\psi^i_{mn},\cdots)\Big),$$

   $$\frac{\partial P(\boldsymbol{\psi}^i,\boldsymbol{\phi}^i,\theta_k)}{\partial \phi^i_{mn}} \approx -\frac{1}{\varepsilon^i}\Big(P(\theta_k;\psi^i_{21},\phi^i_{21}\cdots,\phi^i_{mn}+\varepsilon^i,\cdots) - P(\theta_k;\psi^i_{21},\phi^i_{21},\cdots,\phi^i_{mn},\cdots)\Big),$$

   where ε^i is a small adaptive perturbation evaluated as

   $\varepsilon^i = \frac{J(\Theta^i)}{J(\Theta^0)}\varepsilon^0$

   while ε^0 is the initial perturbation.

   $$\mathbf{\Delta}(\Theta^i) = -\left[\frac{d\mathbf{\Gamma}(\Theta^i)}{d\Theta^i}\left(\frac{d\mathbf{\Gamma}(\Theta^i)}{d\Theta^i}\right)^H\right]^{-1}\frac{d\mathbf{\Gamma}(\Theta^i)}{d\Theta^i}\mathbf{\Gamma}(\Theta^i)$$

   $$\Theta^{i+1} = \Theta^i + \mu\mathbf{\Delta}(\Theta^i)$$

   end
3. Use Θ^{i+1} to form $\mathbf{W}$

[3] A. M. Haimovich, R. S. Blum, and L. J. Cimini, "MIMO radar with widely separated antennas," *IEEE Signal Processing Mag.*, vol. 25, pp. 116–129, Jan. 2008.

[4] P. Stoica, J. Li, and X. Zhu, "MIMO radar with co-located antenna: Review of some recent work," *IEEE Signal Processing Magazine*, vol. 24, pp. 106–114, Sep. 2007.

[5] Y. Yang and R. S. Blum, "MIMO radar waveform design based on mutual information and minimum mean-square error estimation," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 43, pp. 330–343, Jan. 2007.

[6] D. R. Fuhrmann and J. S. Antonio, "Transmit beamforming for MIMO radar systems using partial signal correlation," *In Proc. 38th Asilomar Conference on Signals, System and Computers*, pp. 295–299, Nov. 2004.

[7] D. R. Fuhrmann and J. S. Antonio, "Transmit beamforming for MIMO radar systems using signal cross-correlation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, pp. 171–185, Jan. 2008.

[8] T. Aittomaki and V. Koivunen, "Low complexity method for transmit beamforming in MIMO radar," *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-07), Honolulu, Hawaii, U.S.A*, pp. 305–308, Apr. 2007.

[9] P. Stoica, J. Li, and Y. Xie, "On probing signal design for MIMO radar," *IEEE Trans. Signal Processing*, vol. 55, pp. 4151–4161, Aug. 2007.

[10] P. Stoica, J. Li, and X. Zhu, "Waveform synthesis for diversity-based transmit beampattern design," *IEEE Trans. Signal Processing*, vol. 56, pp. 2593–2598, Jun. 2008.

[11] S. Ahmed, J. S. Thompson, and B. Mulgrew, "Unconstrained synthesis of covariance matrix for MIMO radar transmit beampattern," *IEEE Trans. on Signal Processing*, vol. 59, pp. 3837 – 3849, Aug. 2011.

[12] M. I. Skolnik, *Radar Handbook*. McGraw-Hill, Inc, 2nd Edition, 1990.

[13] S. Ahmed, J. S. Thompson, B. Mulgrew, and Y. Petillot, "Finite alphabet constant-envelope waveform design for MIMO radar beampattern," *IEEE Trans. on Signal Processing*, vol. 59, pp. 5326 – 5337, Nov. 2011.

[14] A. Hassanien and S. A. Vorobyov, "Transmit Energy Focusing for DOA Estimation in MIMO Radar With Colocated Antennas," *IEEE Trans. on Signal Processing*, vol. 59, pp. 2669–2682, Jun. 2011.

[15] B. Friedlander, "On transmit beamforming for MIMO radar," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 48, pp. 3376–3388, Oct. 2012.

[16] T. Aittomaki and V. Koivunen, "Signal covariance-matrix optimisation for transmit beamforming in MIMO radars," *In Proc. 41st Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA*, pp. 182–186, Nov. 2007.

[17] S. Jardak, S. Ahmed, and M. S. Alouini, "Generating correlated QPSK waveforms by exploiting Gaussian random variables," *In proceedings IEEE Conference on Signals, Systems and Computers (ASILOMAR-2012), California, USA*, Oct. 2012.

[18] W. Hayt and J. Buck, *Engineering Electromagnetics*. McGraw-Hill Series in Electrical Engineering.

[19] A. D. Maio, S. D. Nicola, A. Farina, and S. Iommelli, "Adaptive detection of a signal with angle uncertainty," *IET Journal Radar, Sonar and Navigation*, vol. 4, pp. 537–547, Aug. 2010.

[20] G. H. Golub and C. F. V. Loan, *Matrix computations*. Johns Hopkins University Press, 3rd Ed.

[21] J. A. Tropp, I. S. Dhillon, J. R. W. Heath, and T. Strohmer, "Designing structured tight frames via an alternating projection method," *IEEE Transaction on Information Theory*, vol. 51, pp. 188–209, Jan. 2005.

[22] S. Boussakta and J. G. A. Holt, "Fast algorithm for calculation of both Walsh-Hadamard and Fourier transforms (FWFTs)," *IET Electronic Letters*, vol. 25, pp. 1352–1353, Sep. 1989.

[23] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press.

[24] B. Widcolumn and S. D. Stearns, *Adaptive Signal Processing*. Printice Hall, Upper Saddle River, NJ, USA.

[25] E. Abd-Elrady and L. Gan, "Identification of Hammerstein and Wiener models using spectral magnitude matching," *Proceedings of the 17th International Federation of Automatic Control (IFAC), Seoul, Korea*, pp. 6440–6445, Jul. 2008.