Research Article

Low-Complexity Design of Frequency-Hopping Codes for MIMO Radar for Arbitrary Doppler

Badrinath S., Anand Srinivas, and V. U. Reddy

Communication Research Center, IIIT-H, Hyderabad 500032, India

Correspondence should be addressed to V. U. Reddy, vur@iiit.ac.in

There has been a recent interest in the application of Multiple-Input Multiple-Output (MIMO) communication concepts to radars. Recent literature discusses optimization of orthogonal frequency-hopping waveforms for MIMO radars, based on a newly formulated MIMO ambiguity function. Existing literature however makes the assumption of small target Doppler. We first extend the scope of this ambiguity function to large values of target Doppler. We introduce the concept of hit-matrix in the MIMO context, which is based on the hit-array, which has been used extensively in the context of frequency-hopping waveforms for phased-array radars. We then propose new methods to obtain near optimal waveforms in both the large and small Doppler scenarios. Under the large Doppler scenario, we propose the use of a cost function based on the hit-matrix which offers a significantly lower computational complexity as compared to an ambiguity based cost function, with no loss in code performance. In the small Doppler scenario, we present an algorithm for directly designing the waveform from certain properties of the ambiguity function in conjunction with the hit-matrix. Finally, we introduce "weighted optimization" wherein we mask the cost function used in the heuristic search algorithm to reflect the properties of the required ambiguity function.

1. Introduction

MIMO radar is a recent evolution of radar that utilizes multiple transmitters and receivers [1, 2]. MIMO radar waveforms can have any degree of coherence with each other, ranging from complete coherence (in which case it is equivalent to a phased-array radar) to complete incoherence (orthogonality). The choice of radar waveforms [3] plays an important role in the resolution characteristics of the radar. The optimization of radar waveforms for the phased-array radar, which is also viewed as single-input multiple-output (SIMO) radar, focuses on obtaining a desirable ambiguity function in terms of range and Doppler resolutions. On the other hand, MIMO radars provide spatial resolution and spatial diversity in addition to range and Doppler resolution.

Frequency-hopping codes have been used in pulse compression radars [4] because of their highly desirable ambiguity properties. The design of frequency-hopping codes for SIMO radars to obtain desired ambiguity functions [5] has been well studied. A method to design discrete frequency-coding waveforms for the netted radar has been proposed in [6] and a modified Genetic algorithm to design orthogonal discrete frequency-coding waveforms for MIMO radar has been presented in [7]. The hit-array [8, 9] has also been extensively used for waveform design in the SIMO context.

Recently, Chen and Vaidyanathan [10, 11] have dealt with the design of frequency-hopping codes for MIMO case based on the optimization of a newly formulated MIMO ambiguity function [12]. Their approach to designing radar waveforms is to first parameterize these waveforms and then apply simulated annealing to find a near-optimal set of parameters using a "cost function" that allows comparison of different parameter sets. The formulation assumes small target Doppler resulting in a simpler cost function. This formulation, however, is inapplicable in the presence of large target Doppler.

In this paper, we first extend the scope of this ambiguity function to large values of target Doppler, and then introduce the concept of hit-matrix in the MIMO context. Next, we propose new methods to obtain optimal waveforms in both the large Doppler and small Doppler scenarios.
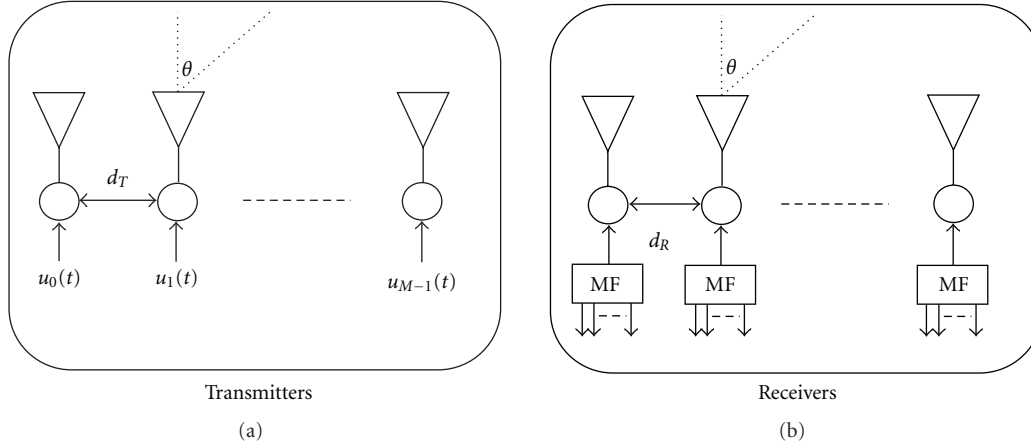
FIGURE 1: Transmitters and receivers in a MIMO radar (MF = Matched filter).

In the case of large Doppler, we propose a cost function based on the hit-matrix which offers a significantly lower computational complexity as compared to an ambiguity-based cost function, with no loss in code performance. In the small Doppler case, we present an algorithm for directly designing the waveform from certain properties which can be obtained from the ambiguity function in conjunction with the hit-matrix. Finally, we introduce a "weighted optimization" wherein we weight the cost function, used in the heuristic search algorithm, to reflect the properties of the required ambiguity function.

In Section 2, we present our model for MIMO radar and frequency-hopping waveforms. In Section 3, we reformulate Chen and Vaidyanathan's MIMO ambiguity function such that it is applicable to any general value of Doppler. In Section 4, we describe the use of hit-matrix as an optimization tool. The hit-matrix corresponds to a digitized version of the ambiguity function which is relatively simple to compute. In Section 5, using a hit-matrix-based cost function, we use simulated annealing to search for the best frequency-hopping codes under the large Doppler scenario. In Section 6, we present an algorithm for directly obtaining a waveform corresponding to a good ambiguity function in the small Doppler case. In Section 7, we propose a method to obtain waveforms satisfying certain conditions on the ambiguity function. A heuristic search performed using a weighted cost function, with the weights representing conditions on ambiguity function, illustrates our proposed method. Section 8 concludes this paper.

2. System Model

2.1. MIMO Radar Model. Consider a monostatic MIMO radar that contains M transmitters and N receivers with their antennas configured as uniform linear arrays, as shown in Figure 1. We assume a point target and also that the target, transmitters and receivers lie in the same 2D plane. Let d_T and d_R represent the spacing between consecutive transmitters and receivers, respectively, and let $\gamma = d_T/d_R$.

We define the *spatial frequency* of the target as

$$f = \frac{d_R \sin(\theta)}{\lambda},\qquad(1)$$

where θ is the target angle with respect to the broadside direction and λ is the wavelength of the RF carrier of the transmitted waveforms. Let τ and ν be the target delay (which is a measure of target range) and Doppler frequency (a measure of target velocity), respectively. The spatial frequency f corresponds to the angular location of the target with respect to the arrays of the radar. Let $\{u_m(t)\}$, $m \in \{0,\dots,M-1\}$ represent the M transmitter waveforms. Then, the waveform received at the nth receiver antenna can be expressed as [11]

$$y_n(t)\big|_{\tau,\nu,f} \approx \sum_{m=0}^{M-1} u_m(t-\tau)e^{j2\pi\nu t}e^{j2\pi f(\gamma m+n)}.\qquad(2)$$

2.2. Frequency-Hopping Waveforms. Frequency-hopping signals are good candidates for the radar waveforms because they are easily generated and have constant modulus. The waveforms can be represented as (see Figure 2)

$$u_m(t) = \sum_{l=0}^{L-1} \phi_m(t-T_l),\qquad(3)$$

where

$$\phi_m(t) = \sum_{q=0}^{Q-1} e^{j2\pi c_{m,q}\Delta f t}s(t-q\Delta t),$$

$$s(t) = \begin{cases} 1, & \text{if } 0 < t < \Delta t, \\ 0, & \text{otherwise.} \end{cases}\qquad(4)$$

Here, $c_{m,q}$ is the (m,q)th element of the matrix $[C]_{M\times Q}$ and it can assume values from the set $\{1,\dots,K\}$, where K is the total number of frequency hops available. As shown
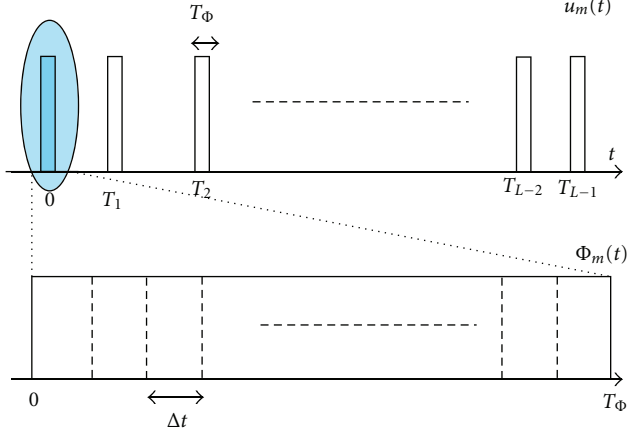
FIGURE 2: The structure of frequency-hopping waveforms.

in Figure 2, each transmitter waveform $u_m(t)$ consists of a stream of L identical pulses $\phi_m(t)$. Each pulse in turn contains Q constant amplitude frequency subpulses each having width Δt and frequency $c_{m,q}\Delta f$. Here we only concern ourselves with waveforms that are orthogonal at zero Doppler and zero delay mismatch, that is,

$$\int_{-\infty}^{\infty} u_m(t)u_{m'}^*(t)\mathrm{d}t = 0, \quad \forall m \neq m'. \tag{5}$$

To attain orthogonality, we impose the following conditions on the waveforms [11]:

$$\Delta f \Delta t = 1,$$
$$c_{m,q} \neq c_{m',q}, \quad \forall m \neq m', \; \forall q. \tag{6}$$

Orthogonal waveforms result in a uniform gain in all directions, which is a key aspect of detection using MIMO radars. For fixed Δt and Δf, these waveforms can be completely described by the code matrix:

$$C = \left[c_{m,q} \right]_{M \times Q} \tag{7}$$

and the pulse spacing vector:

$$\vec{T} = \left[T_0 \; T_1 \; \cdots \; T_{L-1} \right]_{1 \times L}. \tag{8}$$

The pulse spacing vector plays a role in shaping the Doppler resolution of the waveforms. In this paper, however, we will not be dealing with optimization of this vector.

3. MIMO Radar Ambiguity Function

The resolution of a radar system is determined by the response to a point target in the matched filter output. This response can be characterized by a function called the ambiguity function. The traditional Woodward ambiguity function for a SIMO radar is given as

$$\chi(\tau, \nu) = \int_{-\infty}^{\infty} u(t)u^*(t-\tau)e^{-j2\pi\nu t}dt. \tag{9}$$

In the above expression, τ and ν represent the delay and Doppler mismatch at the receiver, respectively. The ideal ambiguity function should be sharp around the region of zero-mismatch, that is, $(\tau, \nu) = (0,0)$. This idea has been extended to the MIMO case in [11]. Consider the expression for the received signal in a MIMO radar given in (2). Let (τ_1, ν_1, f_1) represent the true parameters of a target, and let (τ_2, ν_2, f_2) be the assumed parameters at the receiver. The summed match filter output is given as

$$\sum_{n=0}^{N-1} \int_{-\infty}^{\infty} \left(y_n(t)\big|_{\tau_1, \nu_1, f_1} \right)\left(y_n(t)\big|_{\tau_2, \nu_2, f_2} \right)^* dt$$

$$= \left(\sum_{n=0}^{N-1} e^{j2\pi(f_1 - f_2)n} \right)$$

$$\times \left(\sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \int_{-\infty}^{\infty} u_m(t-\tau_1)u_{m'}^*(t-\tau_2) \right.$$

$$\left. \times e^{j2\pi(\nu_1 - \nu_2)t}e^{j2\pi(f_1 m - f_2 m')\gamma}dt \right). \tag{10}$$

In the above expression, the second term corresponds to the ambiguity function when only one receiver is present, while the first term brings out the effect of having multiple receivers. To simplify the ambiguity function and the waveform design problem, the first term can be decoupled from the above expression. The resulting expression is termed the "MIMO ambiguity function". We can now consider $(\tau_1 - \tau_2)$ to be the delay mismatch and $(\nu_1 - \nu_2)$ to be the Doppler mismatch, and rewrite the expression. The MIMO radar ambiguity function is thus given as [11]

$$\chi(\tau, \nu, f, f') = \sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \chi_{m,m'}(\tau, \nu)e^{j2\pi(fm - f'm')\gamma}, \tag{11}$$

where τ and ν represent the Doppler and delay mismatch at the receiver, f represents the target's true spatial frequency, f' represents the assumed spatial frequency at the receiver, and $\chi_{m,m'}(\tau, \nu)$ represents the cross-ambiguity function between the waveforms $u_m(t)$ and $u_{m'}(t)$, given by

$$\chi_{m,m'}(\tau, \nu) = \int_{-\infty}^{\infty} u_m(t)u_{m'}^*(t+\tau)e^{j2\pi\nu t}dt. \tag{12}$$

For frequency-hopping waveforms (see Figure 2), (12) becomes

$$\chi_{m,m'}(\tau, \nu) = \sum_{l=0}^{L-1} \sum_{l'=0}^{L-1} \chi_{m,m'}^\phi(\tau + T_l - T_{l'}, \nu)e^{j2\pi\nu T_l}, \tag{13}$$

where

$$\chi_{m,m'}^\phi(\tau, \nu) = \int_0^{Q\Delta t} \phi_m(t)\phi_{m'}^*(t+\tau)e^{j2\pi\nu t}dt. \tag{14}$$

We assume that the maximum expected target delay is less than the time interval between any two consecutive pulses of each transmitter waveform as they are colocated. This

assumption implies that $|\tau| < (\min_{l,l'}(|T_l - T_{l'}|) - Q\Delta t)$ due to which

$$\chi_{m,m'}^{\phi}(\tau + T_l - T_{l'}, \nu) = 0 \quad \text{for } l \neq l' \tag{15}$$

and hence

$$\chi_{m,m'}(\tau, \nu) = \chi_{m,m'}^{\phi}(\tau, \nu) \sum_{l=0}^{L-1} e^{j2\pi\nu T_l}. \tag{16}$$

$\chi_{m,m'}^{\phi}(\tau, \nu)$ is the cross-ambiguity between two individual pulses of different waveforms ($\phi_m(t)$ and $\phi_{m'}(t)$) and can be expanded as

$$\chi_{m,m'}^{\phi}(\tau, \nu)$$
$$= \sum_{q=0}^{Q-1} \sum_{q'=0}^{Q-1} \chi_{\text{rect}}\left(\tau + (q - q')\Delta t, \nu + \left(c_{m,q} - c_{m',q'}\right)\Delta f\right)$$
$$\times e^{j2\pi(\nu + (c_{m,q} - c_{m',q'})\Delta f)q\Delta t} e^{-j2\pi c_{m',q'}\Delta f\tau}, \tag{17}$$

where

$$\chi_{\text{rect}}(\tau, \nu)$$
$$= \int_0^{\Delta t} s(t)s(t + \tau)e^{j2\pi\nu t}dt$$
$$= \begin{cases} (\Delta t - |\tau|)\text{sinc}(\nu(\Delta t - |\tau|))e^{j\pi\nu(\Delta t - |\tau|)}, & \text{if } |\tau| < \Delta t, \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

represents the ambiguity function of the rectangular pulse $s(t)$. In [11], the Doppler is assumed to be small ($\nu\Delta t \approx 0$) due to which (17) reduces to

$$\chi_{m,m'}^{\phi}(\tau, \nu) = \sum_{q=0}^{Q-1} \sum_{q'=0}^{Q-1} \chi_{\text{rect}}\left(\tau + (q - q')\Delta t, \left(c_{m,q} - c_{m',q'}\right)\Delta f\right)$$
$$\times e^{j2\pi\Delta f(c_{m,q} - c_{m',q'})q\Delta t} e^{-j2\pi c_{m',q'}\Delta f\tau}$$
$$= \chi_{m,m'}^{\phi}(\tau, 0). \tag{19}$$

In the subsequent sections, we do not assume small Doppler and work with (17). We next describe the hit-matrix formalism in Section 4 and waveform optimization for the large Doppler case in Section 5.

4. The Hit-Matrix Formalism

The hit-array has been introduced as a tool to analyze frequency-hopping waveforms in [9]. The central concept in this formulation is that of a "hit", which occurs when the received pattern has been shifted in the time-frequency

space in such a way that it overlaps with the original pattern at exactly one time-frequency position. In this section, we extend the hit-array to the hit-matrix, which is applicable to frequency-hopping codes for MIMO radar under the large Doppler scenario. We define the hit-matrix for the code matrix C as

$$H = [h_{k,l}]_{(2Q-1)\times(2K-1)}, \tag{20}$$

$$-Q < k < Q, \quad -K < l < K, \quad k, l \in \mathbb{Z}, \tag{21}$$

where

$$h_{k,l} = \begin{cases} \sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \sum_{q=0}^{Q-k-1} \delta\left(c_{m,q} - c_{m',(q+k)} + l\right), & \text{if } k \geq 0, \\ \sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \sum_{q=0}^{Q+k-1} \delta\left(c_{m,q} - c_{m',(q-k)} - l\right), & \text{otherwise,} \end{cases} \tag{22}$$

and $\delta(\cdot)$ refers to the Kronecker delta function. The concept of hit-array was used in phased array as well as multiple access radars for approximating auto- and cross-ambiguity functions. The hit-array is either auto-hit-array ($\hat{H}_{m,m}$) or cross-hit-array ($\hat{H}_{m,m'}$) as defined below (see (24)). The hit-matrix H can be obtained by summing the individual auto- and cross-hit arrays as follows:

$$H = \sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \hat{H}(m, m'), \tag{23}$$

where

$$\hat{H}(m, m') = \left[\hat{h}_{k,l}(m, m')\right]_{(2Q-1)\times(2K-1)},$$

$$\hat{h}_{k,l}(m, m') = \begin{cases} \sum_{q=0}^{Q-k-1} \delta\left(c_{m,q} - c_{m',(q+k)} + l\right), & \text{if } k \geq 0, \\ \sum_{q=0}^{Q+k-1} \delta\left(c_{m,q} - c_{m',(q-k)} - l\right), & \text{otherwise.} \end{cases} \tag{24}$$

As an example, consider the code matrix:

$$C = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{bmatrix}. \tag{25}$$

Here $M = 2$, $Q = 3$, and $K = 3$. The cross-hit arrays for this code are

$$\hat{H}(0,0) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \hat{H}(1,0) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 1 & 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix},$$

$$\hat{H}(0,1) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad \hat{H}(1,1) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 3 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \tag{26}$$

and the resulting hit-matrix is

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 3 & 2 & 1 & 1 \\ 1 & 2 & 6 & 2 & 1 \\ 1 & 1 & 2 & 3 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix}. \tag{27}$$

Ambiguity Function and the Hit-Matrix. We now describe how the hit-matrix of a frequency-hopping code relates to its ambiguity function. Consider the MIMO radar ambiguity function in (11) which, in view of (16) and (17), can be expressed as

$$\chi(\tau, \nu, f, f') = \Omega(\tau, \nu, f, f') \left[\sum_{l=0}^{L-1} e^{j2\pi\nu T_l} \right], \tag{28}$$

where

$$\Omega(\tau, \nu, f, f')$$
$$= \left[\sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \sum_{q=0}^{Q-1} \sum_{q'=0}^{Q-1} G_{m,m',q,q'}(\tau, \nu) e^{j2\pi(fm - f'm')\gamma} \right] \tag{29}$$

represents the ambiguity between two pulses and

$$G_{m,m',q,q'}(\tau, \nu) = \chi_{\text{rect}}\left(\tau + (q - q')\Delta t, \nu + \left(c_{m,q} - c_{m',q'}\right)\Delta f\right)$$
$$\cdot e^{j2\pi(\nu + (c_{m,q} - c_{m',q'})\Delta f)q\Delta t} e^{-j2\pi c_{m',q'}\Delta f\tau} \tag{30}$$

is the cross-ambiguity between the qth subpulse of $u_m(t)$ and the (q')th subpulse of $u_{m'}(t)$. A plot of $|G_{m,m',q,q'}(\tau, \nu)|$ is shown in Figure 3. The position of its peak is $(\tau_{\text{peak}}, \nu_{\text{peak}}) = (-(q - q')\Delta t, -(c_{m,q} - c_{m',q'})\Delta f)$. Along the τ-axis, this function drops to zero at a shift of Δt from τ_{peak}, and along the ν-axis, it falls by roughly 6 dB relative to the
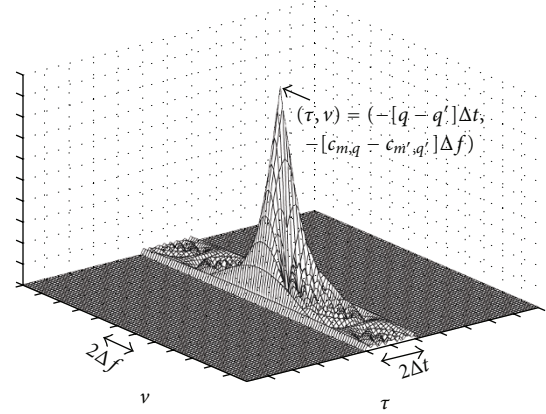

FIGURE 3: Cross-ambiguity between subpulses, $|G_{m,m',q,q'}(\tau, \nu)|$.

peak at a shift of Δf from ν_{peak}. Hence most of the energy of each subpulse ambiguity function is located within an area of $4\Delta f\Delta t$ around $(\tau_{\text{peak}}, \nu_{\text{peak}})$. For different values of (m, m', q, q'), the shape of $G_{m,m',q,q'}(\tau, \nu)$ remains the same, and only the position of its peak in the delay-Doppler space changes. The set of possible values of $(\tau_{\text{peak}}, \nu_{\text{peak}})$ is given as

$$\left(\tau_{\text{peak}}, \nu_{\text{peak}}\right) \in \{(k\Delta t, l\Delta f)\}, \quad k, l \in \mathbb{Z},$$
$$-Q < k < Q, \quad -K < l < K. \tag{31}$$

The number of 4-tuples (m, m', q, q') for which $G_{m,m',q,q'}(\tau, \nu)$ peaks at the position (τ_0, ν_0) is

$$P(\tau_0, \nu_0) = \begin{cases} \sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \sum_{q=0}^{Q-\tau_0/\Delta t-1} \delta\left(c_{m,q} - c_{m',(q+\tau_0/\Delta t)} + \dfrac{\nu_0}{\Delta f}\right), \\ \qquad\qquad\qquad \text{if } \tau_0 \geq 0, \\ \sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \sum_{q=0}^{Q+\tau_0/\Delta t-1} \delta\left(c_{m,q} - c_{m',(q-\tau_0/\Delta t)} - \dfrac{\nu_0}{\Delta f}\right), \\ \qquad\qquad\qquad \text{otherwise}, \end{cases}$$
$$= h_{k,l}\big|_{k=\tau_0/\Delta t, l=\nu_0/\Delta f}. \tag{32}$$

Thus each element of the hit-matrix corresponds to the number of functions $G_{m,m',q,q'}(\tau, \nu)$ centered at the same point in the delay-Doppler space. The $(MQ)^2$ different functions $G_{m,m',q,q'}(\tau, \nu)$ for different values of the 4-tuple (m, m', q, q') are the building blocks of the overall MIMO ambiguity function, and the distribution of the positions of their peaks in the delay-Doppler space plays a key role in determining the shape of the overall ambiguity function. The value of $h_{0,0}$ corresponds to the height of the mainlobe in the ambiguity function and is constant for a given code matrix size. The values of $h_{k,l}$ outside of $(k, l) = (0, 0)$ correspond to sidelobes in the ambiguity function centered at $(\tau, \nu) = (k\Delta t, l\Delta f)$. The higher these values of $h_{k,l}$, the higher will be the corresponding sidelobes. Also, since the spatial frequency parameters f and f' appear as complex

exponential weights in the summation of (29), a reduction in the values of the hit-matrix outside of $(k, l) = (0, 0)$ will result in a corresponding reduction in sidelobes along the spatial frequency dimensions as well.

5. Waveform Design for Large Doppler

We now describe how frequency-hopping codes can be optimized under the large Doppler scenario to yield a desired ambiguity function. Since the second product term in the right-hand-side of (28) is not dependent on the choice of code matrix C, we only concern ourselves with the optimization of the first term $\Omega(\tau, \nu, f, f')$. To apply heuristic search algorithms like simulated annealing, we require a cost function that allows the desirability of different codes to be compared. Following the formulation of [11], a cost function for the large Doppler case is given as follows:

$$f_p(C) = \int_{-Q\Delta t}^{Q\Delta t} \int_{-K\Delta f}^{K\Delta f} \int_0^{1/\gamma} \int_0^{1/\gamma} |\Omega(\tau, \nu, f, f')|^p \, df \, df' \, d\nu \, d\tau.$$

(33)

It can be shown that the height of the peak in $\Omega(\tau, \nu, f, f')$ at $(\tau, \nu, f, f') = (0, 0, f, f)$ is constant, and hence this cost function favors codes that have their sidelobes flattened out over the delay, Doppler and spatial frequency dimensions. Increasing the value of p increases the penalty on higher sidelobes. $f_p(C)$ can be evaluated for each code using Riemann sums with a sufficient number of bins instead of integrations. However, this results in a high computational complexity which increases as $O(M^2 Q^2 \mathrm{BT})$, where BT is the time-bandwidth product of the code.

Now, consider the elements of the hit-matrix $\mathbf{H}$. From (22), $h_{0,0} = MQ$, and as explained in the previous section, it corresponds to the height of the mainlobe in the ambiguity function, and $h_{k,l}$ outside of $(k, l) = (0, 0)$ correspond to sidelobes in the ambiguity function. The higher these values of $h_{k,l}$, the higher will be the corresponding sidelobes. This motivates us to choose the following cost function based on the hit-matrix:

$$g_p(C) = \sum_{k=-Q+1}^{Q-1} \sum_{l=-K+1}^{K-1} (h_{k,l})^p,$$

(34)

which favors the codes that have their sidelobes flattened out over the delay, Doppler and spatial frequency dimensions. Evaluation of $g_p(C)$ is far less computationally intensive than $f_p(C)$, and increases only as $O(M^2 Q^2)$. This allows the heuristic search algorithms using this cost function to rapidly traverse the code space, thereby allowing good codes to be found faster.

We now describe how we apply simulated annealing using $g_p(C)$. We use a slightly modified form of simulated annealing called quantum-simulated annealing, which allows faster convergence. The quantum-simulated annealing parameters are temperature (T), rate of decrease of temperature (α), jump size (J) and rate of decrease of jump size (β). The algorithm is initialized with a value of $T > 0$ and $J > 0$, choosing α and β from $(0, 1)$. The steps of the algorithm are as follows.

(1) Randomly draw a code matrix C from $\{0, 1, \ldots, K - 1\}^{MQ}$ such that the code is orthogonal, that is, $c_{m,q} \neq c_{m',q}$ for $m \neq m'$.

(2) Randomly draw j from $\{1, 2, \ldots, \lceil J \rceil\}$.

(3) Set $C' = C$, and repeat steps 3(a) to 3(c) j times.

    (a) Randomly draw m from $\{0, \ldots, M - 1\}$ and q from $\{0, \ldots, Q - 1\}$.

    (b) Select k from $\{0, \ldots, K - 1\}$ with $k \neq \{c'_{m,q}, \forall m\}$.

    (c) Set $c'_{m,q} = k$.
    (At this point, we will have the original code C, and C' obtained from the above steps.)

(4) Randomly draw U from $[0, 1]$.

(5) If $U < \exp((g_p(C) - g_p(C'))/T)$, then set $C = C'$.

(6) Set $T = \alpha T$ and $J = \beta J$.

(7) If a sufficiently small value of $g_p(C)$ has been obtained, terminate the algorithm. Otherwise, return to step 2.

Consider as an example the optimization of frequency-hopping codes for $(M, Q, K) = (4, 6, 24)$. The total number of possible codes of this size is $(K!/(K - M)!)^Q = 2.8 \times 10^{32}$. We found the number of Riemann sum bins, required for reasonable accuracy in the evaluation of $f_p(C)$ for this code size, to be 100 each along the delay and Doppler dimensions, and 20 each along the two spatial frequency dimensions. Hence, while $f_p(C)$ requires the computation and summation of 2.3×10^9 terms per iteration, $g_p(C)$ only needs 576 computations per iteration, resulting in a significant decrease in complexity.

5.1. Simulation Results. We present a number of examples to demonstrate the effectiveness of the hit-matrix-based cost function in designing good frequency-hopping codes. We have generated codes of various sizes by simulated annealing using both $f_p(C)$ and $g_p(C)$ at $p = 3$. The code parameters used were $M = 4$, $Q \in \{6, \ldots, 10\}$ and $K \in \{MQ/2, MQ, 2MQ, 3MQ\}$ for each value of Q. Other parameters used were $\Delta t = 1$, $\Delta f = 1$ and $\gamma = 1$. For simulated annealing, we set the parameters $T = 10$, $J = 12$, $\alpha = 0.9$ and $\beta = 0.95$.

First, we consider the code obtained for $(M, Q, K) = (4, 6, 24)$ using $g_3(C)$. The decrease in the cost with respect to the iterations of simulated annealing is shown in Figure 4. The final code obtained is

$$C = \begin{bmatrix} 20 & 2 & 24 & 5 & 6 & 1 \\ 22 & 14 & 4 & 24 & 1 & 20 \\ 18 & 1 & 12 & 6 & 23 & 22 \\ 11 & 24 & 9 & 2 & 17 & 14 \end{bmatrix}.$$

(35)

A plot of the hit-matrix of this code is shown in Figure 5 and a plot of $|\Omega(\tau, \nu, f, f')|$ at $(f, f') = (0, 0)$ is shown in Figure 6. We observe that the ambiguity function of the
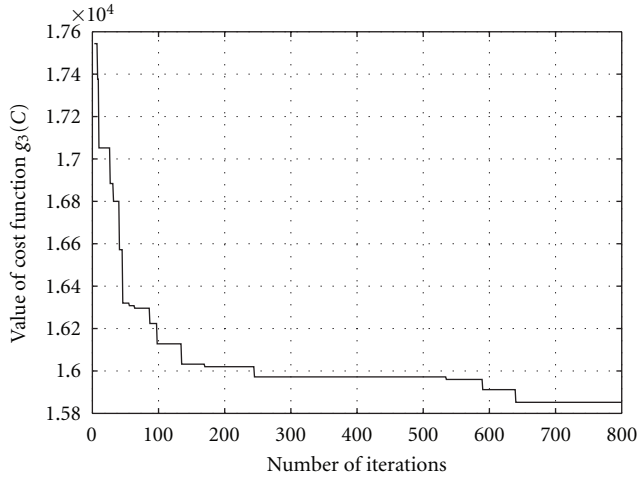
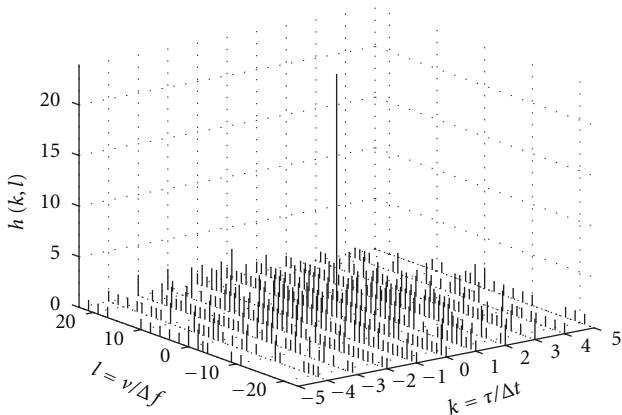FIGURE 4: Decrease in $g_3(C)$ versus iterations of simulated annealing.

FIGURE 5: Plot of the hit-matrix of a code obtained from simulated annealing using $g_3(C)$.
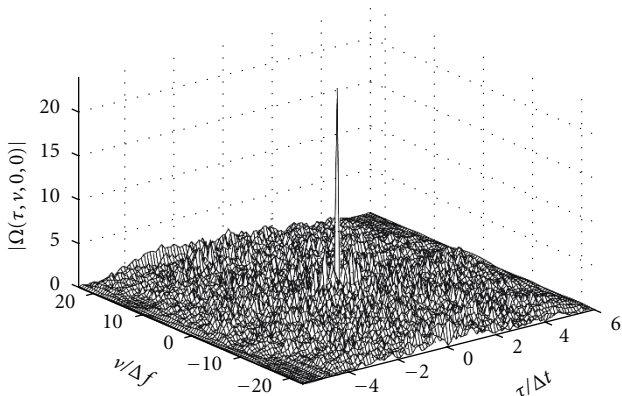

FIGURE 6: $|\Omega(\tau, \nu, 0, 0)|$ for the code whose hit-matrix is shown in Figure 5.

above code has a very sharp mainlobe along the delay and Doppler dimensions, with 98% of the sidelobes lying 10 dB below the peak. Also note the visual similarity between the plots of hit-matrix and ambiguity function. Although it may appear that optimization using $g_p(C)$ involves just the time and Doppler dimensions, it takes the spatial frequency into consideration as well. To show this, we consider values of $\Omega(\tau, \nu, f, f')$ at different values of (τ, ν).

(1) $(\tau, \nu) = (0, 0)$. The Ω function can be expressed from (29) and (30) as follows:

$$\Omega(0, 0, f, f') = \sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \delta_{m,m'} e^{j2\pi(fm - f'm')\gamma}$$
$$= \sum_{m=0}^{M-1} e^{j2\pi(f - f')m\gamma}. \tag{36}$$

The above expression is independent of the code matrix, and thus the ambiguity function in this form cannot be optimized along the spatial frequency dimensions at $(\tau, \nu) = (0, 0)$.

(2) $(\tau, \nu) \neq (0, 0)$. Noting that each peak, given by $|G_{m,m',q,q'}(\tau, \nu)|$, corresponds to a hit which is reflected in (34), and the ambiguity function consists of a weighted sum of hits with the weights being of the form $e^{j2\pi(fm - f'm')\gamma}$, optimizing the hit-matrix is analogous to optimizing the upper bound on the ambiguity function along the spatial frequency dimensions.

We may point out here that the ambiguity function has a very low magnitude when $(\tau, \nu) \neq (0, 0)$ for all mismatched values of f and f'. To show this, various cuts of $|\Omega(\tau, \nu, f, f')|$ are shown in Figure 7 (at $\gamma = 1$). We now compare various codes obtained using $f_p(C)$ and $g_p(C)$. We take samples from the function $|\Omega(\tau, \nu, f, f')|$ and plot their empirical cumulative distribution function (ECDF), which show the percentage of samples of $|\Omega(\tau, \nu, f, f')|$ less than specified magnitudes. The highest peak has been normalized to 0 dB. Figure 8 shows the ECDF curves for various codes obtained at $Q = 6$. Note that use of either cost function yields codes with similar ECDF curves. In Figure 9, we have plotted the magnitude of 95th percentile of $|\Omega(\tau, \nu, f, f')|$ for different codes, as a function of the time bandwidth products BT $= (Q\Delta t \times K\Delta f)$. Note that up to BT $= 392$, $f_p(C)$ and $g_p(C)$ both yield codes with a similar number of undesirable peaks. The curve corresponding to $f_p(C)$ could not be extended beyond BT $= 392$ because of increasing computational complexity. However, using $g_p(C)$, codes with BT as high as 1200 (and higher) could be easily generated.

A large Doppler scenario will arise when a MIMO radar operating in the K_a band (27 GHz to 40 GHz) is expected to detect 5000 km class missile targets. To see how the cost function in [11], which assumes low Doppler, performs in comparison to the large Doppler cost function in (33),

$f = 0.25, f' = 0.05$

(a)

$f = 0.25, f' = 0.25$

(b)

$f = 0.25, f' = 0.45$

(c)

$f = 0.5, f' = 0.3$

(d)

$f = 0.5, f' = 0.5$

(e)

$f = 0.5, f' = 0.7$

(f)

$f = 0.75, f' = 0.55$

(g)

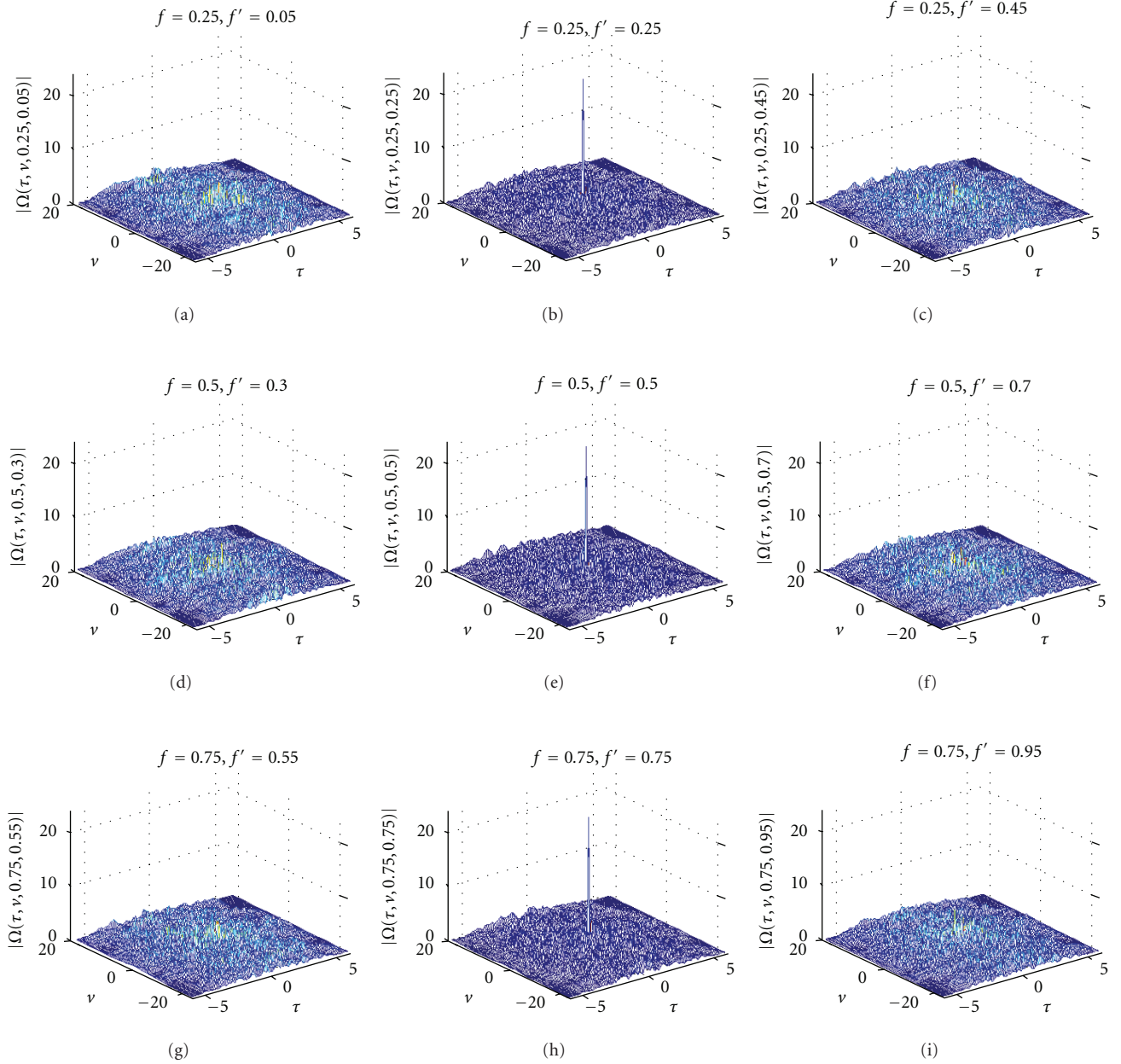$f = 0.75, f' = 0.75$

(h)

$f = 0.75, f' = 0.95$

(i)

FIGURE 7: Various cuts of $|\Omega(\tau, \nu, f, f')|$ for the code whose hit-matrix is shown in Figure 5.

when Doppler frequency is a significant percentage of the signal bandwidth, we conducted the following simulation. We obtained the codes from the minimization of the two cost functions at $p = 3$, choosing the Doppler frequency as 25% of the transmitter waveform bandwidth for three different values of K. Figure 10 gives the ECDF curves of the corresponding $|\Omega(\tau, \nu, f, f')|$. Note from the plots that the large Doppler cost function yields an improvement of 0.5 dB for $K = 12$ and 1.0 dB for $K = 48$ at 90%. We may point out here that the bandwidth increases with increasing K, and accordingly the Doppler frequency goes up as it is 25% of the bandwidth.

6. Waveform Optimization for Small Doppler Case ($\nu \ll \Delta f$)

As derived in [11], for the small Doppler condition ($\nu \Delta t \approx 0$) and under the assumption of no range folding, we have from (16) and (19)

$$\chi_{m,m'}(\tau, \nu) \approx \chi_{m,m'}^{\phi}(\tau, 0) \sum_{l=0}^{L-1} e^{j2\pi \nu T_l}. \quad (37)$$

In this case, we can write the ambiguity function from (11),
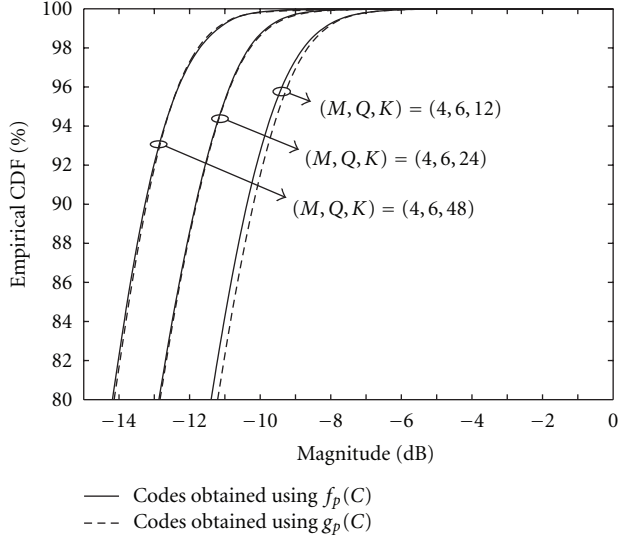
Figure 8: Empirical CDF of $|\Omega(\tau, \nu, f, f')|$.
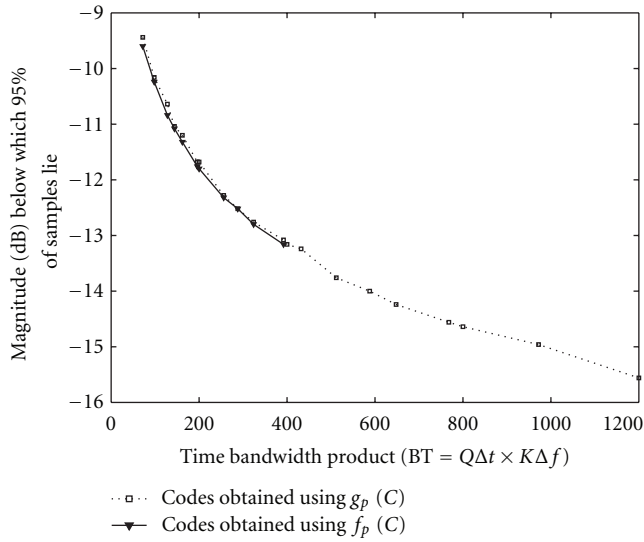
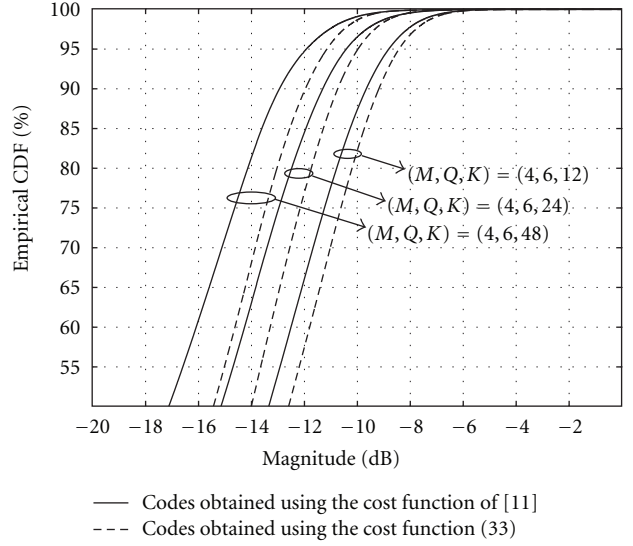Figure 9: Magnitude of 95th percentile of $|\Omega(\tau, \nu, f, f')|$.

(16) and (19) as follows:

$$
\chi(\tau, \nu, f, f')
$$
$$
= \left[\sum_{m=0}^{M-1}\sum_{m'=0}^{M-1} \chi_{m,m'}^{\phi}(\tau, 0)e^{j2\pi(fm-f'm')\gamma}\right] \cdot \left[\sum_{l=0}^{L-1}e^{j2\pi\nu T_l}\right],
$$
$$
(38)
$$

which, in view of (28), becomes

$$
\chi(\tau, \nu, f, f') = [\Omega(\tau, 0, f, f')] \cdot \left[\sum_{l=0}^{L-1}e^{j2\pi\nu T_l}\right]. \quad (39)
$$

Note from (38) that the $\phi_m(t)$ terms (corresponding to the waveform pulses), which are contained in $\chi_{m,m'}^{\phi}(\tau, 0)$,

Figure 10: Empirical CDF of $|\Omega(\tau, \nu, f, f')|$ for the codes obtained with two different cost functions.

do not affect the Doppler resolution. The objective behind waveform design is to obtain a set of waveforms with a desirable MIMO ambiguity function. In [11] a heuristic search (simulated annealing) is performed over the space of all code-matrices, to acquire a code C which minimizes the cost function.

In the SISO case, we expect the ambiguity function to be sharp about the point of zero-mismatch, that is, $(\tau, \nu) = (0, 0)$. Similarly, in the MIMO scenario, we want the ambiguity function to be sharp around the region of zero-mismatch, which corresponds to values of the function $\Omega(\tau, 0, f, f')$ over the line $\{(\tau, f, f') \mid \tau = 0, f = f'\}$. As the waveforms are assumed to be orthogonal, we can write

$$
\Omega(0, 0, f, f' \mid f = f')
$$
$$
= \sum_{m=0}^{M-1}\sum_{m'=0}^{M-1} \delta(m - m')e^{j2\pi f\gamma(m-m')} = M. \quad (40)
$$

Thus, at zero-mismatch $\Omega(0, 0, f, f' \mid f = f')$ is a constant value proportional to the number of transmitters and is independent of the chosen code-matrix. The objective now is to suppress all the peaks not lying on this line. Therefore, the code optimization problem under the small Doppler scenario (as given in [11]) reduces to the minimization of the cost function:

$$
f_p'(C) = \int_{-Q\Delta t}^{Q\Delta t} \int_0^{1/\gamma} \int_0^{1/\gamma} |\Omega(\tau, 0, f, f')|^p \, df \, df' \, d\tau \quad (41)
$$

In the next subsection, we explain our proposed algorithm.

The Hit-Matrix under the Small Doppler Assumption. In this scenario, the hit-matrix reduces from a $(2Q - 1) \times (2K - 1)$ matrix to a $(2Q - 1) \times (1)$ array. We define the hit-matrix under small Doppler, H^{sd}, as follows

$$H^{\text{sd}} = [h_{k,0}]_{(2Q-1)\times(1)},$$
$$-Q < k < Q, \quad k \in \mathbb{Z},$$
(42)

where $h_{k,0}$ is given by

$$h_{k,0} = \sum_{m=0}^{M-1} \sum_{m'=0}^{M-1} \sum_{q=0}^{Q-|k|-1} \delta\left(c_{m,q} - c_{m',(q+|k|)}\right), \quad (43)$$

and $\delta(\cdot)$ refers to the Kronecker delta function.

Note that the hit-matrix H^{sd} will also have a close correlation with the ambiguity function under the low Doppler scenario. Further, it is easy to see that $h_{0,0}$ is a constant equal to MQ and is independent of the values in the code matrix. Our objective in code design can now be described by the following two conditions.

(1) *Condition A.* Within the constraints defining the code matrix $(M, Q, \Delta f, \Delta t, K)$, the sidelobe levels must be reduced. This means that the total number of hits, given by

$$S = \sum_{k=-Q+1}^{Q-1} h_{k,0}, \quad (44)$$

should be made as small as possible.

(2) *Condition B.* A key aspect of waveform design is to achieve high mainlobe to peak sidelobe ratio in its ambiguity function. This is reflected in $h_{k,0}$. We know from the previous condition that the sum total of all the elements of $h_{k,0}$ over all values of k equals S. Also, $h_{0,0}$ is constant and is equal to MQ. Our objective, therefore, is to spread out the remaining $(S - MQ)$ elements corresponding to $(S - MQ)$ peaks over the $2Q - 2$ summation terms of $h_{k,0}$ (excluding $k = 0$) thereby minimizing the peak sidelobe. This can be expressed in compact form as follows

$$\text{Minimize}: \max_{k; k \neq 0} h_{k,0}. \quad (45)$$

6.1. Waveform Design. In this section, we propose a direct waveform design algorithm that yields codes that satisfy both Conditions A and B described above. This algorithm requires the number of frequencies K to be constrained to $MQ/2$ (the consequences of such a restriction are discussed in the next subsection).

Towards finding codes that satisfy *Condition A*, we define

$$S' = \sum_{k=1}^{Q-1} h_{k,0}. \quad (46)$$

Since $h_{0,0} = MQ$ and $h_{k,0} = h_{-k,0}$, we can write $S = 2S' + MQ$. Hence, minimizing S for a given value of MQ is equivalent to

minimizing S'. Now, from (43) we note that if the pair of entries $c_{m,q}$ and $c_{m',q'}$ in the code matrix C are equal, they will contribute a value of 1 (or one "hit") to $h_{k,0}$ at $k = |q - q'|$. Thus we can say that S' will equal the number of pair of entries with identical values that can be found in the code matrix.

Using this alternate interpretation of S', we proceed to show that under the constraint $K = MQ/2$, S' will be minimized only when each of the K usable frequency values occur exactly twice in the code matrix. To see this, consider an example with $MQ = 6$, which means that the code matrix has 6 entries and 3 frequency choices are available. Let X, Y and Z correspond to the 3 frequencies. If we choose each of these three frequencies twice, that is, we make the selection $\{X, X, Y, Y, Z, Z\}$, the possible pairs of entries with identical frequencies will be $\{X, X\}$, $\{Y, Y\}$ and $\{Z, Z\}$, thus giving $S' = 3$. Suppose we pick $\{X, X, Y, Z, Z, Z\}$, the number of possible same-frequency pairs will be four: one $\{X, X\}$ pair and three $\{Z, Z\}$ pairs giving $S' = 4$, one higher than the case where each frequency was used twice. Other combinations can also be shown to yield a higher value of S'. A similar argument can be extended to other values of M and Q as well.

Next, we provide the conditions under which *Condition B* will be satisfied. Any code matrix with $K = MQ/2$ and with each frequency used twice will have $S' = MQ/2$. In order to satisfy *Condition B*, we require that these $MQ/2$ hits be spread as uniformly as possible among the $Q - 1$ summation terms of (46). This leads to the condition

$$N_m \leq h_{k,0} \leq (N_m + 1), \quad \forall k \in \{1, 2, \dots, Q - 1\}, \quad (47)$$

where

$$N_m = \left\lfloor \frac{(MQ/2)}{Q - 1} \right\rfloor, \quad (48)$$

which implies that all the values of $h_{k,0}$ for $k > 0$ must equal either N_m or $N_m + 1$.

For a given value of M and Q, there will exist several code matrices that satisfy Conditions A and B simultaneously. We now describe an algorithm that will randomly generate one such code matrix in each run. Initially, we start with a code matrix that has all entries "vacant", and hence $h_{k,0} = 0$ for all $k \in \{1, \dots, Q - 1\}$. The algorithm has two steps. In Step 1 (see Figure 11), we first randomly select a pair of vacant entries $c_{m,q}$ and $c_{m',q'}$ such that $|q - q'| = Q - 1$, and fill these entries with frequency index 1. This increments the value of $h_{k,0}$ at $k = (Q - 1)$ by 1, while leaving it unchanged for other values of $k > 0$. We repeat this procedure another $N_m - 1$ times, using a different frequency index each time, until we have $h_{k,0} = N_m$ at $k = (Q - 1)$. Next we proceed to similarly increment the other values of $h_{k,0}$ to N_m, in the order $k = (Q - 2), (Q - 3), \dots, 1$. Step 1 is completed when $h_{k,0} = N_m$ at $k = 1$.

In the cases where $(MQ/2)/(Q - 1)$ is not an integer, the code matrix will still have some vacant entries after Step 1. Step 2 of the algorithm (see Figure 12) aims to fill the remaining entries, while ensuring that

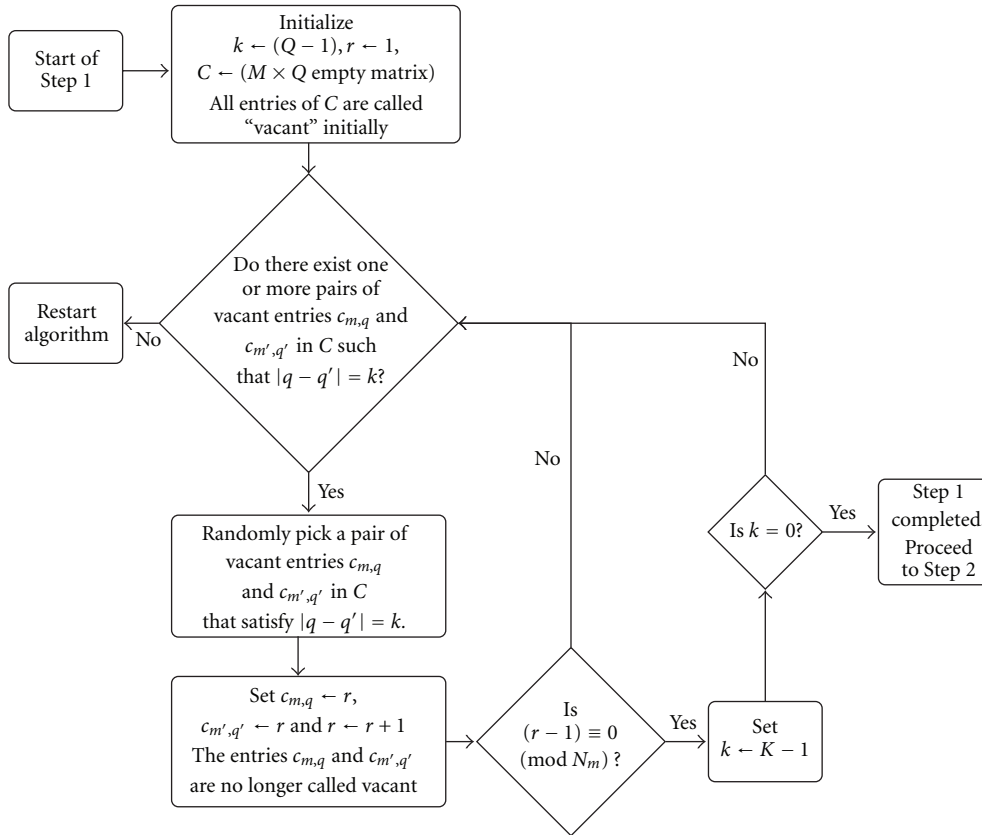$$\max_{k \in \{1,\dots,Q-1\}} h_{k,0} \leq (N_m + 1). \quad (49)$$
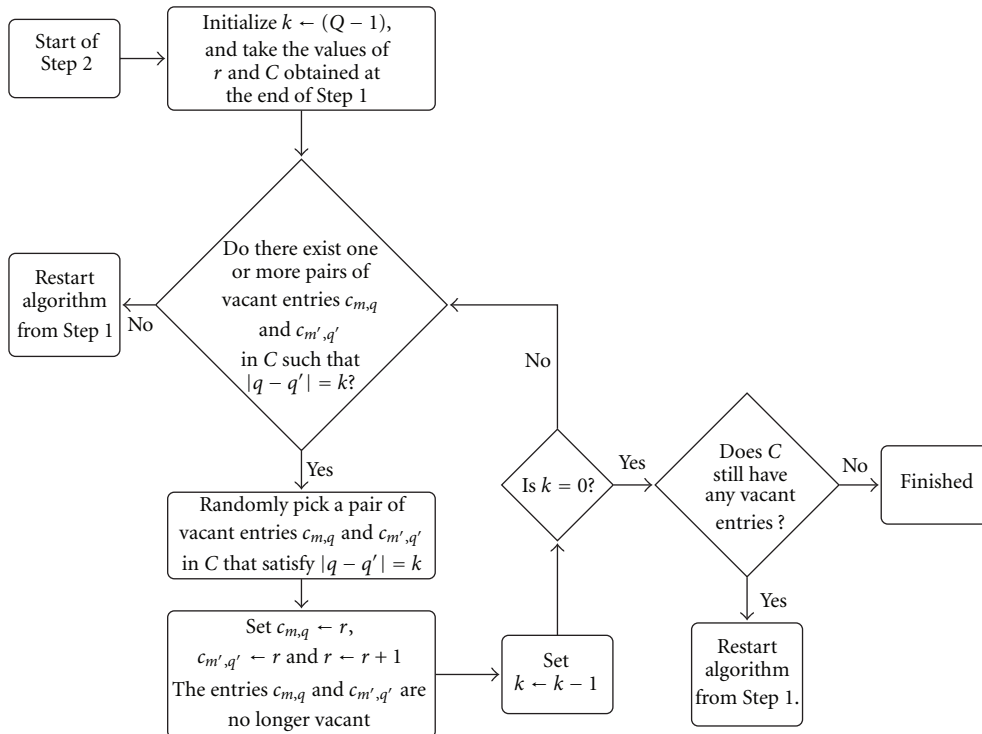
FIGURE 11: Step 1 of the algorithm.
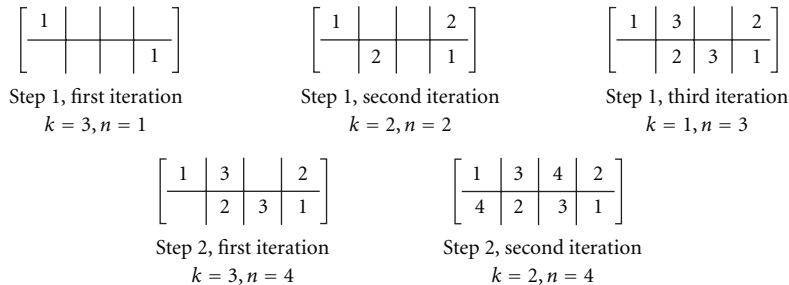

FIGURE 12: Step 2 of the algorithm.

FIGURE 13: A possible sequence of code matrices obtained at various stages of the algorithm ($M = 2$ and $Q = 4$).

This is achieved by again traversing through the values of $h_{k,0}$ in the order $k = (Q-1), (Q-2), \ldots, 1$, and incrementing by one hit wherever possible. Flow charts providing the details of the two steps are shown in Figures 11 and 12. As can be seen from the algorithm flow charts, there are certain cases under which we need to discard the code matrix obtained after a few iterations and restart from the beginning of Step 1. This case arises when, from a partially filled code matrix, it is not possible to fill the remaining positions in such a way that the resulting code matrix will generate orthogonal waveforms. However, the probability of such an event occurring is very low. Through extensive simulations (for various values of M and Q), we have observed that the probability of a code being generated with three or less restarts is over 99%.

Consider a simple example for $M = 2$ and $Q = 4$. Since $(MQ/2)/(Q-1) = 4/3$ is not an integer, both Step 1 and Step 2 will need to be performed. The one possible sequence of code matrices obtained at various stages of the algorithm are shown in Figure 13.

6.2. Related Discussion. For frequency-hopping waveforms, the parameters which can be controlled are - $\{M, Q, K, \Delta t, \Delta f\}$. The proposed algorithm assumes one of these parameters K to be equal to $MQ/2$. One possible limitation of this method is the loss of flexibility in designing codes for a required time-bandwidth product, BT. Table 1 lists all possible time-bandwidth products between 0 and 400 for which codes can be designed from the proposed algorithm for $M = 4$. Note that the bandwidth of the waveform is $K\Delta f$, the duration of each pulse is $Q\Delta t$ and, hence, BT $= KQ\Delta t\Delta f$. From the entries of the table, we may state that constraint on K is not a serious limitation on the flexibility in designing the code. On the other hand, the algorithm hardly needs any computations in comparison to those required by a heuristic search-based method such as simulated annealing.

6.3. Simulation Results. We will now present simulation results to demonstrate the effectiveness of our proposed algorithm. We have generated codes of various sizes using both methods, the heuristic search proposed by [11] with the cost function (41) at $p = 3$ and our proposed algorithm. The code parameters used were $M = 4$, $Q \in \{6, \ldots, 14\}$ and $K = MQ/2$ for each value of Q. From the code matrices

TABLE 1: Values of BT for which codes can be designed with the proposed algorithm ($M = 4$ and $\Delta f \Delta t = 1$).

32	160	288
50	162	294
64	192	300
72	196	324
96	200	338
98	216	350
100	224	360
128	242	384
144	250	392
150	256	400

obtained from both methods for $M = 4$, $Q = 10$, we plotted the Empirical CDFs of their respective $|\Omega(\tau, 0, f, f')|$ functions (see Figure 14). In order to see how a randomly generated code performs on average, we generated randomly 100 orthogonal code matrices of same size and plotted ECDF of the code corresponding to the median cost. Note from the plots that both methods are similar in terms of performance and they show a marked improvement over a randomly generated code.

In order to compare the performance of the two methods with increasing code size, we have plotted the magnitude below which 95% of the samples of $|\Omega(\tau, 0, f, f')|$ lie. The lower the curve, the lower is the corresponding sidelobe level, which corresponds to a better code. From Figure 15, we see that the performance of the proposed algorithm is either very similar or slightly better than that of the heuristic search-based method for all Q. The marginal performance loss of the heuristic method for some values of Q, as seen in the figure, may be because the simulated annealing has converged to one of the local minima.

7. Weighted Optimization ($\Delta f \leq \nu < K\Delta f$)

When optimizing frequency-hopping codes using either $f_p(C)$ and $g_p(C)$, the aim is to minimize the sidelobe levels over the entire range of delay-Doppler space, that is, for $-Q\Delta t < \tau < Q\Delta t$ and $-K\Delta f < \nu < K\Delta f$. However, it is
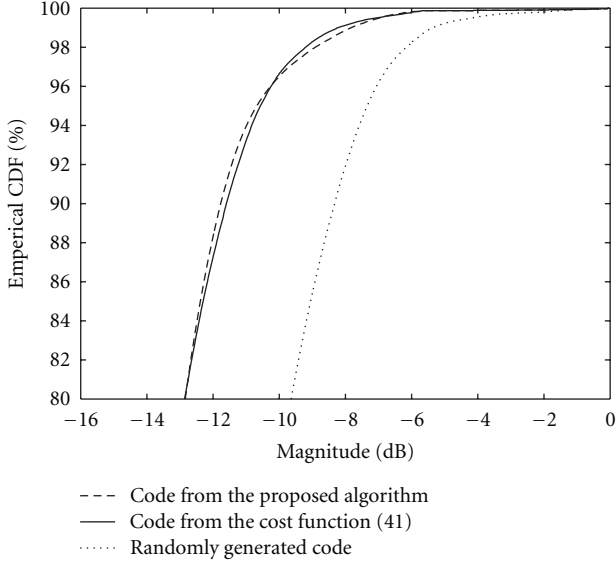
FIGURE 14: Empirical CDFs of the codes from the two methods ($M = 4$, $Q = 10$, and $K = 20$).

FIGURE 16: CDFs of the optimized codes with and without weighting in the specified region ($M = 4$, $Q = 10$, $K = 80$, $l_{max} = 8$, $k_{max} = 2$).
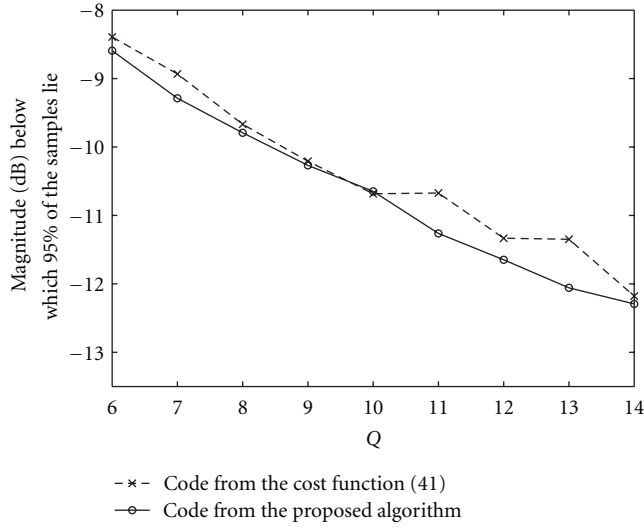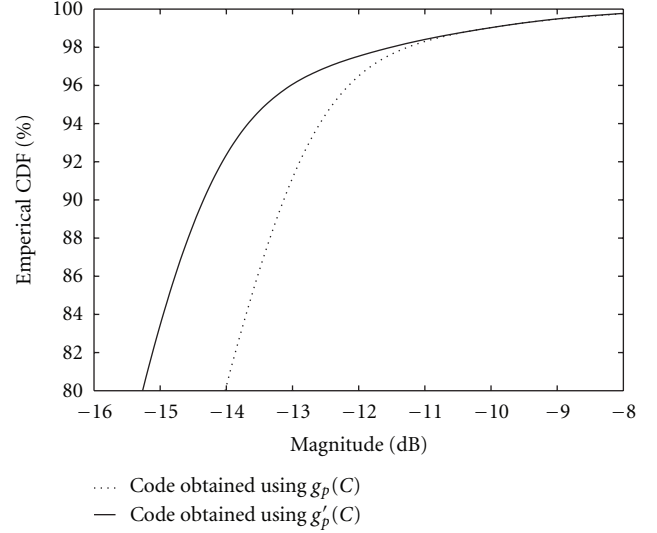

FIGURE 15: Magnitude of 95th percentile of $|\Omega(\tau, 0, f, f')|$ ($M = 4$).

possible to place a higher importance on the minimization of sidelobe peaks in a subregion of the delay-Doppler space, by using a weighted cost function defined as follows:

$$g'_p(C) = \sum_{k=-Q+1}^{Q-1} \sum_{l=-K+1}^{K-1} \lambda_{k,l}(h_{k,l})^p, \qquad (50)$$

where $\lambda_{k,l}$ is the weight applied to (k, l)th element of H (see (20)). We now discuss possible applications of such a weighted cost function.

Example 1. When the range of target Doppler is smaller than the overall bandwidth of the transmitted signals ($\max(|\nu|) < K\Delta f$), optimization can be performed for a subset of the Doppler values using the weights:

$$\lambda_{k,l} = \begin{cases} 1, & \text{if } 0 < |l| < l_{max}, \\ 0, & \text{otherwise}, \end{cases} \qquad (51)$$

where $l_{max} = \lceil \max(|\nu|)/\Delta f \rceil$.

Example 2. Using this method, it is possible to find waveforms with their ambiguity sidelobes constrained to particular regions of the the delay-doppler space. Suppose we wish to find codes with very few ambiguity sidelobes occurring in a region close to the mainlobe. Consider the following mask

$$\lambda_{k,l} = \begin{cases} 1, & \text{if } 0 < |l| < l_{max}, \ 0 < |k| < k_{max}, \\ 0, & \text{otherwise}, \end{cases} \qquad (52)$$

where l_{max} and k_{max} are bounds on the regions of delay-Doppler space. Thus, a penalty is applied on sidelobes occurring inside the region of interest and the cost is unaffected by sidelobes occurring outside this bounded region. To show the gain this method could have, we now compare the final ambiguity functions obtained by using $g_p(C)$ and $g'_p(C)$. In order to compare them, we consider the delay-Doppler subspace of the corresponding ambiguity functions bounded by l_{max} and k_{max} and plot their respective CDFs. The parameters for this simulation were $M = 4$, $Q = 10$, $K = 80$, $l_{max} = 8$ and $k_{max} = 2$. Figure 16 shows the gain achievable within the region of interest.

8. Conclusion

In this paper, we have shown how the MIMO radar ambiguity function for orthogonal frequency-hopping waveforms can be extended to general values of Doppler. We have also presented the hit-matrix as an analysis tool for these waveforms. To enable the optimization of these waveforms under the large Doppler scenario using simulated annealing, cost functions have been presented based on the ambiguity function as well as the hit-matrix. The codes obtained using both cost functions are shown to have similar performance based on their ECDF curves. The hit-matrix-based cost function has a significantly lower computational complexity, and can be useful when searching for codes with high time-bandwidth products, where using a ambiguity-based cost function is infeasible.

Under the small Doppler scenario, an algorithm has been proposed which directly computes the code matrix of a given size. It has been shown to perform as well as the heuristic search proposed by Chen and Vaidyanathan [11]. The use of weighted cost functions to optimize the ambiguity function within a subregion of the delay-Doppler space has also been explored. This method of "Weighted Optimization" also addresses the problem of waveform design for intermediate Doppler.

Acknowledgments

References

[1] E. Fishler, A. Haimovich, R. Blum, D. Chizhik, L. Cimini, and R. Valenzuela, "MIMO radar: an idea whose time has come," in *Proceedings of the IEEE Radar Conference*, pp. 71–78, April 2004.

[2] J. Li and P. Stoica, "MIMO radar with colocated antennas," *IEEE Signal Processing Magazine*, vol. 24, no. 5, pp. 106–114, 2007.

[3] J. Li and P. Stoica, *MIMO Radar Signal Processing*, Wiley-IEEE Press, New York, NY, USA, 2008.

[4] S. W. Golomb, "Constructions and properties of costas arrays," *Proceedings of the IEEE*, vol. 72, no. 9, pp. 1143–1163, 1984.

[5] N. Levanon and E. Mozeson, *Radar Signals*, Wiley-IEEE Press, New York, NY, USA, 2004.

[6] H. Deng, "Discrete frequency-coding waveform design for netted radar systems," *IEEE Signal Processing Letters*, vol. 11, no. 2, pp. 179–182, 2004.

[7] B. Liu and Z. He, "Orthogonal discrete frequency-coding waveform design for MIMO radar," *Journal of Electronics*, vol. 25, no. 4, pp. 471–476, 2008.

[8] J. R. Bellegarda, S. V. Maric, and E. Titlebaum, "The hit array: a synthesis tool for multiple access frequency hop signal," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 3, pp. 624–635, 1993.

[9] J. R. Bellegarda and E. L. Titlebaum, "The hit array: an analysis formalism for multiple access frequency hop coding," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 27, no. 1, pp. 30–39, 1991.

[10] C.-Y. Chen and P. P. Vaidyanathan, "Properties of the MIMO radar ambiguity function," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 2309–2312, April 2008.

[11] C.-Y. Chen and P. P. Vaidyanathan, "MIMO radar ambiguity properties and optimization using frequency-hopping waveforms," *IEEE Transactions on Signal Processing*, vol. 56, no. 12, pp. 5926–5936, 2008.

[12] G. San Antonio, D. R. Fuhrmann, and F. C. Robey, "MIMO radar ambiguity functions," *IEEE Journal on Selected Topics in Signal Processing*, vol. 1, no. 1, pp. 167–177, 2007.