Multiple-Target Localization by Millimeter-Wave Radars With Trapezoid Virtual Antenna Arrays

Wei Zhao, Student Member, IEEE, Jian-Kang Zhang, Senior Member, IEEE, Xiao-Ping Zhang, Fellow, IEEE, and Rong Zheng, Senior Member, IEEE

Abstract—We consider the problem of localizing multiple targets by millimeter wave (mmWave) radars with irregular antenna placement, i.e., trapezoid virtual antenna array. The goal is to estimate both the number of targets and their 3-D locations. While many well-known algorithms have been developed for either problems, they still suffer from several limitations, such as the need for a large amount of sampled radar data and high computation complexity. In this work, we develop an efficient solution by exploring the received signal structure in two steps: 1) estimating the number of targets and their ranges by extending Barone’s method to handle data from multiple antennas and 2) estimating the angle of arrival of each target by a Least-Square solution by exploring the received signal structure in two steps: 1) estimating the number of targets and their ranges by extending Barone’s method to handle data from multiple antennas and 2) estimating the angle of arrival of each target by a Least-Square algorithm optimization. The proposed algorithm has been evaluated through Monte-Carlo simulations and an indoor testbed. By comparing with baseline algorithms, including 2D-FFT and multiple signal classification (MUSIC), we find that the proposed algorithm has the best performance in high signal-to-noise ratio regimes.

Index Terms—Frequency modulated continuous waveform (FMCW) radar, multiple-target localization, Padé approximation, trapezoid virtual antenna array.

I. INTRODUCTION

MILLIMETER wave (mmWave) technologies, with wavelengths from 1 to 10 mm and frequency from 30 to 300 GHz [1], are gaining traction in the next generation of wireless communication (5G) and localization systems. Because of their short wavelength, many advantages emerge. For example, with significantly smaller antenna sizes, more antennas can be integrated in a very small area. Consequently, multiple-input multiple-output (MIMO), beamforming, and other well-established techniques can be employed, even on IoT devices with small form factors. One important application of mmWave technologies is radio detection and ranging (RADAR). Numerical techniques have been developed for target detection, localization, and tracking using radars. The location of a target can be represented in polar coordinates using its range, azimuth angle, and elevation angle. Range is typically estimated based on the time interval between signal transmission and reception [2], [3]. Modern radars use frequency modulated continuous waveform (FMCW) signals due to their pulse compression capability, where the range of a target is proportionally to the frequency of the intermediate frequency (IF) signal after the application of a matched filter to the received waveform.

With very few exceptions, target localization and target number estimation are performed in two separate steps in existing work. To estimate the number of targets, a variety of methods have been devised, including those using Akaike information criterion (AIC) [4], energy detection [5], machine learning models [6], etc. Once the number of targets is known, targets are separated according to their ranges and angles. To estimate the Angle of Arrivals (AoAs) of known number of targets, multiantenna systems, such as uniform linear array (ULA) [7] and uniform rectangular array (URA), are used and many algorithms have been proposed, including the estimation of signal parameters via rotational invariant techniques (ESPRIT) [8], [9], multiple signal classification (MUSIC) [10], Capon algorithm [11], [12], parallel factor analysis (PARAFAC)-based algorithm [13], matrix pencil method [14], and its enhancement [15], [16]. Jardak et al. [17] applied 1D-FFT, constant false alarm rate (CFAR) detection and peak grouping to the data matrix extracted from RADAR to estimate the number of targets and their range information.

An angle-FFT is then used to estimate the angles of remaining peaks, each corresponding to one target. Due to its usage of FFTs in range and angle domains, we call this approach 2D-FFT. In [18], a modified 3-D MUSIC algorithm was developed for 3-D target localization. Instead of choosing the largest few eigenvalues as the number of targets, the authors adopt the minimum description length (MDL) criterion [19], [20]. The three polar coordinates of all targets are estimated from the 3-D MUSIC pseudo-spectrum together.

The aforementioned approaches suffer from several limitations. In the FFT-based algorithm in [17], the number of sampled data from each received antenna has to be sufficiently large making the detection and localization of moving targets difficult. Moreover, the CFAR threshold needs to be tuned for target environments [21]. 1-D angular FFT assumes linear antenna arrays. Furthermore, the time complexity of super-resolution methods for AoA estimation such that the MUSIC algorithm is very high since they rely on a grid search.
TABLE I
FREQUENTLY USED SYMBOLS

<table>
<thead>
<tr>
<th>TX, RX</th>
<th>Transmit antenna, receiver antenna</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma$</td>
<td>Number of targets</td>
</tr>
<tr>
<td>$R_i, \theta_i, \phi_i$</td>
<td>Polar coordinates of target $i$: range, azimuth angle, elevation angle</td>
</tr>
<tr>
<td>$x_i, y_i, z_i$</td>
<td>Rectangular coordinates of target $i$</td>
</tr>
</tbody>
</table>

$x(t)$ | The transmitted waveform (chirp signal) from the virtual transmitter antenna at time $t$ |

$y_k(t)$ | The signal received by the $k$-th virtual received antenna at time $t$ |

$\omega_{a,i}$ | The phase shift between azimuth receiver antennas for target $i$ |

$\omega_{a,i}$ | The phase shift between one of the top 4 virtual receiver antennas and the corresponding virtual receiver antenna below it for target $i$ |

$\alpha_i$ | The backscatter coefficient of target $i$ |

$\tau_i$ | The interval between the time when a chirp signal is emitted and received for target $i$ |

$a_i, b_i$ | Steering vector of the real transmitter antennas and receiver antennas for target $i$ |

---

in pseudo-spectrum, which diminishes their applicability in real-time applications.

In this article, we introduce a new approach for multitarget localization using mmWave radars with trapezoid virtual antenna arrays. Trapezoid antenna arrays become increasingly common in COTS radar boards, e.g., IWR1443/1843/6843 from Texas Instruments [22], CAL77S244 from the Calterah Semiconductor Technology [23], to name a few in indoor and vehicle applications. We first separate received signals from multiple targets by exploiting the received signal structure. We cast the received signals on each antenna of FMCW radars into the special form and extend Barone’s method [24] to handle data from multiple antennas. Next, a Least-Square algorithm is employed to estimate the AoAs of each target. Simulation results show that the proposed method outperforms the 2D-FFT algorithm in ranging and localization. In the high signal-to-noise ratio (SNR) regimes, it achieves more accurate AoA estimations than both 2D-FFT and MUSIC algorithms. Furthermore, the proposed method is of low computation complexity and reaches good performance with as few as 100 samples when the SNR is at 30 dB.

The remainder of this article is organized as follows. Section II derives the radar system model for the trapezoid virtual antenna array; Section III gives the proposed algorithm to extend Barone’s method and apply the Least-Square algorithm in radar signal processing; Section IV presents the MATLAB simulation results by using 2D-FFT, MUSIC, and the proposed method. Section V shows the evaluation of the proposed algorithm on a testbed followed by conclusion in Section VI.

Notations: We use lower- and upper-case boldfaced symbols to denote a column vector and a matrix, respectively. $(\cdot)^T$ denotes the transpose operation and $(\cdot)^H$ is to obtain the Hermitian matrix. $(\cdot)$ denotes the estimated value, $\|\cdot\|$ denotes $L^2$-norm, and Re$(\cdot)$ denotes the real part of a complex number. $c^j_i$ represents a new vector consisting of the $i$th to the $j$th element of a vector $c$. Frequently used symbols are listed in Table I.

---

II. RADAR MODEL FOR MULTIPLE TARGETS

In this section, we present the antenna geometry of radars and the mathematical representations of transmitted and received signals.

A. System Model

For concreteness of the discussion, we consider a specific radar geometry [see Fig. 1(a)], where the distance between neighboring transmit antennas is $\lambda$, the carrier wavelength, and the middle transmit antenna is $\lambda/2$ higher than its two neighbors. The distance between neighboring receiver antennas is also $\lambda/2$. In order to separate transmitted signals, we let three transmit antennas take turns to transmit the same chirp signals, in the order of TX1, TX3, and TX2. Then, the reflected signals obtained on all the receiver antennas from the same target exhibit phase shifts from one another, caused by the distance differences between the target and each antenna. Equivalently, we can represent the system with $M = 1$ virtual transmit antenna located at TX1 and $N = 12$ virtual receiver antennas, among which RX1~4 are located at their original positions, and two replications of the virtual antenna array are located at the left side and on top of original four antennas, respectively [see Fig. 1(b)]. Thus, we call it “trapezoid virtual antenna array.” The eight virtual antennas in the bottom row are also called azimuth receiver antennas. This transformation greatly simplifies subsequent analysis.

Consider $T$ targets located at $(R_i, \theta_i, \phi_i)$ or $(x_i, y_i, z_i)$, $i = 1, 2, 3, \ldots, T$, where $\theta_i$ and $\phi_i$ are the azimuth and elevation angles of the $i$th target as shown in Fig. 2. To estimate $(x_i, y_i, z_i)$s, we first introduce some notations.
1) \(x(t)\): The transmitted waveform (chirp signal) from the virtual transmitter antenna at time \(t\).

2) \(y_k(t)\): The signal received by the \(k\)th virtual received antenna at time \(t\), where \(k = 1, 2, 3, \ldots, 12\).

For target \(i\),

1) \(\omega_{z,i}\): the phase shift between azimuth receiver antennas for target \(i\);

2) \(\omega_{x,i}\): the phase shift between one of the top 4 virtual receiver antennas and the corresponding virtual receiver antenna below it for target \(i\);

3) \(\alpha_i\): the backscatter coefficient of target \(i\);

4) \(R_t\): the range between target \(i\) and the center of radar antenna array, and we assume that the range meets far-field criterion;

5) \(\tau_i\) = \((2R_t/c)\): the interval between the time when a chirp signal is emitted and received for target \(i\). \(c\) is the velocity of the light.

Those two phase shifts can be derived by azimuth and elevation angles between the radar and the target as

\[
\begin{align*}
\omega_{x,i} &= \pi \sin \theta_i \cos \phi_i \\
\omega_{z,i} &= \pi \sin \phi_i.
\end{align*}
\]

In fact, the phase shifts can be represented using the steering vectors for each real transmit antenna and receive antennas, denoted by \(a_i(\omega_{x,i}, \omega_{z,i}) = [1, e^{j(2\omega_{x,i} - \omega_{z,i})}, e^{j(2\omega_{x,i} - \omega_{z,i})}, \ldots, e^{j(2\omega_{x,i} - \omega_{z,i})}]^T\). Hence, the equivalent steering vector for all 12 virtual receiver antennas is given by

\[
h_i = a_i(\omega_{x,i}, \omega_{z,i}) \otimes b_i(\omega_{x,i}) = [1, e^{j(2\omega_{x,i} - \omega_{z,i})}, e^{j(2\omega_{x,i} - \omega_{z,i})}, \ldots, e^{j(2\omega_{x,i} - \omega_{z,i})}]^T,
\]

where \(\otimes\) denotes the Kronecker product. Now, let \(\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_T]\), \(\omega_x = [\omega_{x,1}, \omega_{x,2}, \ldots, \omega_{x,T}]\), \(\omega_z = [\omega_{z,1}, \omega_{z,2}, \ldots, \omega_{z,T}]\), and \(\tau = [\tau_1, \tau_2, \ldots, \tau_T]\). We can write the received signal by all virtual antennas from all \(T\) targets as

\[
y(t; \alpha, \omega_x, \omega_z, \tau) = \sum_{i=1}^{T} \alpha_i h_i x(t - \tau_i) + v(t)
\]

where \(v(t)\) is a vector of Gaussian white noise at receiver side and each element in \(v(t)\) follows \(v(t) \sim \mathcal{N}(0, \sigma^2)\).

Let \(\mathbf{A} = \text{diag}([\alpha_1, \alpha_2, \ldots, \alpha_T])\), \(\mathbf{H} = [h_1, h_2, h_3, \ldots, h_T]\), \(x(t; \tau) = [x(t - \tau_1), x(t - \tau_2), \ldots, x(t - \tau_T)]^T\). The system model of the trapezoid virtual antenna array in matrix representation is given by

\[
y(t; \alpha, \omega_x, \omega_z, \tau) = \mathbf{H} \mathbf{A} x(t; \tau) + v(t).
\]

**B. IF Signals**

From the system model, we can analyze the waveform of the received signals for FMCW radars. Assume that the chirp signal emitted by a TX antenna is \(x(t) = e^{j2\pi f_c(t + (1/2)S\tau^2)}\), where \(f_c\) is the carrier frequency and \(S\) is the chirp slope. The received signals in a vector form at 12 virtual antennas are

\[
y(t; \alpha, \omega_x, \omega_z, \tau) = \sum_{i=1}^{T} \alpha_i h_i e^{j2\pi f_c(t(t - \tau_i) + (1/2)S\tau_i^2)} + v(t).
\]

After processed by a matched filter, the resulting IF signal is

\[
y_{IF}(t; \alpha, \omega_x, \omega_z, \tau) = y(t; \alpha, \omega_x, \omega_z, \tau)^* \cdot x(t)
\]

where

\[
y_{IF}(t) = v(t)^* \cdot x(t).
\]

Let \(x_{IF}(t; \tau_i) = e^{j2\pi f_c(t \tau_i + (1/2)S\tau_i^2)}\) and \(x_{IF}(t; \tau) = [x_{IF}(t; \tau_1), x_{IF}(t; \tau_2), \ldots, x_{IF}(t; \tau_T)]\). We can write the output IF signals for all 12 virtual antennas in a matrix form as

\[
y_{IF}(\alpha, \omega_x, \omega_z, \tau) = \mathbf{H}^\ast \mathbf{A} x_{IF}(t; \tau) + v_{IF}(t).
\]

**C. Sampling**

Let the sampling frequency be \(F_s\) and the total number be \(N_{SAMP}\). The sampling interval is \(T_s = (1/F_s)\) and the sampling start time is \(T_{Start}\). Define \(x_{IF,SAMP}(n; \tau_i) = e^{j2\pi f_c(n \tau_i + (1/2)S\tau_i^2)}\), \(n = 0, 1, 2, \ldots, N_{SAMP} - 1\) and let

\[
x_{IF,SAMP}(n; \tau) = \begin{bmatrix} x_{IF,SAMP}(n; \tau_1) \\ x_{IF,SAMP}(n; \tau_2) \\ \vdots \\ x_{IF,SAMP}(n; \tau_T) \end{bmatrix}.
\]

The \(n\)th sampled data for all 12 virtual antennas can be written as

\[
y_{IF,SAMP}(n; \alpha, \omega_x, \omega_z, \tau) = \mathbf{H}^\ast \mathbf{A} x_{IF,SAMP}(n; \tau) + v_{IF}(n).
\]

Next, let

\[
\mathbf{X}_{IF}(\tau) = \begin{bmatrix} x_{IF,SAMP}(0; \tau_1), \ldots, x_{IF,SAMP}(N_{SAMP} - 1; \tau_1) \\ x_{IF,SAMP}(0; \tau_2), \ldots, x_{IF,SAMP}(N_{SAMP} - 1; \tau_2) \\ \vdots \\ x_{IF,SAMP}(0; \tau_T), \ldots, x_{IF,SAMP}(N_{SAMP} - 1; \tau_T) \end{bmatrix}.
\]

We can then write the sampled radar matrix \(\mathbf{Y}_{IF}\) of size \(12 \times N_{SAMP}\) as

\[
\mathbf{Y}_{IF}(\alpha, \omega_x, \omega_z, \tau) = \mathbf{H}^\ast \mathbf{A} \mathbf{X}_{IF}(\tau) + \mathbf{v}_{IF}.
\]

Now, our problem can be stated as to estimate \(T, \omega_x, \omega_z, \) and \(\tau\) based on the sampled radar matrix \(\mathbf{Y}_{IF}\).

**III. APPROACH**

In this section, we present the proposed algorithm and analyze its property.
A. Overview of Barone’s Method

Before delving into the proposed method, we first review how Barone’s method works [24] and its connection to multitarget localization. The problem that Barone’s method solves is to estimate parameters from a sequence data $y$ of length $N_{SAMP}$, where $y = \{y_0, y_1, y_2, \ldots, y_{N_{SAMP}}\}$ and $y_n$ satisfies

$$y_n = s_n + v_n \quad (11)$$

for $n = 0, 1, 2, \ldots, N_{SAMP} - 1$. In (11), $s_n$ is a complex moment satisfying the following form: $s_n = \sum_{i=0}^{T} c_i \xi_n^i$, where $c_i$ and $\xi$ are complex numbers, $T$ is a nonnegative integer. $v_n$ is the complex additive Gaussian white noise with zero mean and known variance $\sigma^2$. To estimate $T$, $c_i$’s, and $\xi$’s from $y$, we summarize Barone’s method into the following five steps.

Step 1: Build $R$ independent pseudo replications of the original data sequence, as

$$y^{(r)}_n = y_n + v^{(r)}_n, \quad n = 0, 1, \ldots, N_{SAMP} - 1; \quad r = 1, \ldots, R \quad (12)$$

where, $\{v^{(r)}_n\} \sim \mathcal{N}(0, \sigma^2)$. Compute Padé poles and corresponding residuals based on the replicated data by Padé approximants [25]. The result is $R \cdot N_{SAMP}/2$ pairs of $(\xi^{(r)}_i, c^{(r)}_i)$.

Step 2: On a proper lattice $L$ on the complex plane, calculate the complex measure $\hat{S}_{N_{SAMP}, R}(\xi, \sigma^2)$ based on $(\xi^{(r)}_i, c^{(r)}_i)$s on those lattice points. Find all the local maxima of $|\hat{S}_{N_{SAMP}, R}(\xi, \sigma^2)|$.

Step 3: Count the number of Padé poles in the neighborhood of each local maxima.

Step 4: Select those local maxima whose number of associated Padé poles meet the prescribed threshold. The number of those local maxima is the estimated $\hat{T}$.

Step 5: For each of those local maxima, take the average of its corresponding $\xi^{(r)}_i$ and $c^{(r)}_i$ in its neighbor area. The results are the estimated $\hat{\xi}_i, \hat{c}_i$.

Though Barone’s method was originally developed as a mathematical tool in approximation theory, we find that the received signals from multiple targets in radar systems can be expressed in a form that the method can be applied. In particular, the received signal is the sum of reflected signals from unknown number of targets. Each summation of the $n$th sample of the reflected signal contains a target-dependent constant and a target-dependent term that changes over time. Finally, the received signal also contains a noise term. Thus, by representing the received radar signal in the form (11), we can apply Barone’s method to estimate the target parameters.

B. Transformation for Individual Antenna

In order to apply Barone’s method, we need to transform the radar data matrix $Y_{IF}$ to a suitable form as we discussed above. Define

$$c_{k,i} = \alpha_i e^{\omega_{k,i}} e^{j(k-1)\omega_{k,i}} e^{j(\omega_{k,i} - 2\omega_{k,i})} e^{2\pi (\tau_{i,ST_{Start} + f_{i, T_{1}} - 1/2} \tau_{i})} \quad (13)$$

for $k = 1, 2, 3, \ldots, 8$, and

$$c_{k,i} = \alpha_i e^{\omega_{k,i}} e^{-j(k-9)\omega_{k,i}} e^{2\pi (\tau_{i,ST_{Start} + f_{i, T_{1}} - 1/2} \tau_{i})} \quad (14)$$

for $k = 9, 10, 11, 12$, and $i = 1, 2, 3, \ldots, T$. Let

$$\xi_{i} = e^{2\pi \tau_{i} S_{I1}}. \quad (15)$$

We construct a Vandermonde matrix

$$V = \text{Vander}(\xi_1, \xi_2, \ldots, \xi_T)$$

$$= \begin{bmatrix} \xi_1^0 & \xi_1^1 & \cdots & \xi_1^{N_{SAMP}-1} \\ \xi_2^0 & \xi_2^1 & \cdots & \xi_2^{N_{SAMP}-1} \\ \vdots & \vdots & \ddots & \vdots \\ \xi_T^0 & \xi_T^1 & \cdots & \xi_T^{N_{SAMP}-1} \end{bmatrix}. \quad (16)$$

Now, we can write the system model (10) as

$$Y_{IF} = S + V_{IF}$$

$$= C \cdot V + V_{IF} \quad (17)$$

where the element in the $k$th row and $i$th column of the matrix $C$ is $c_{k,i}$. The size of $C$ and $V$ is $12 \times T$ and $T \times N_{SAMP}$, respectively. For the element in the $k$th row and $n$th column of $S$, we have $s_{k,n} = \sum_{i=1}^{T} c_{k,i} \xi_n^i$, where $k = 1, 2, 3, \ldots, 12$. And, this form is exactly the same as that mentioned by Barone.

To further analyze the sampling data from each virtual antenna, we write the sampling data from each virtual antenna as a column vector. Take the transpose to both sides of (17), we have

$$Y_{IF}^* = [Y_{IF,1}, Y_{IF,2}, Y_{IF,3}, \ldots, Y_{IF,12}]$$

$$= V^* C^T + V_{IF}^T. \quad (18)$$

The sampled data of the $k$th virtual antenna $y_{IF,k}$ is

$$y_{IF,k} = s_k + v_k$$

$$= \begin{bmatrix} s_{k,0} \\ s_{k,1} \\ \vdots \\ s_{k,N_{SAMP}-1} \end{bmatrix} + \begin{bmatrix} v_{k,1} \\ v_{k,2} \\ \vdots \\ v_{k,N_{SAMP}-1} \end{bmatrix}. \quad (19)$$

Now if we directly apply Barone’s method to $y_{IF,k}$ and get the estimated parameters for the $k$th virtual antenna. The results can be represented by the following.

1) Number of Targets: $\hat{T}_k$.
2) Residuals: $\hat{u}_k = [\hat{c}_{k,1}, \hat{c}_{k,2}, \ldots, \hat{c}_{k,T_{12}}]$.
3) Padé Poles: $\hat{\xi}_k = [\hat{\xi}_{k,1}, \hat{\xi}_{k,2}, \ldots, \hat{\xi}_{k,T_{12}}]$.

Repeating the above operation to the sampled data of all 12 virtual antennas, i.e., $Y_{IF,1}, Y_{IF,2}, \ldots, Y_{IF,12}$, we have

$$\hat{T} = [\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_{12}] \quad (20)$$

$$\hat{U} = [\hat{U}_1, \hat{U}_2, \ldots, \hat{U}_{12}] \quad (21)$$

$$\hat{\Xi} = [\hat{\xi}_{1}, \hat{\xi}_{2}, \ldots, \hat{\xi}_{12}] \quad (22)$$

However, due to the existence of random noise, $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_{12}$ may not be the same, and the locations of elements in $\hat{\Xi}$ do not coincide on the complex plane. One naive approach is to take the average of the estimated number of targets from each antenna and then perform clustering over the estimated locations. However, such a method fails to take advantage of the correlated nature of measurements at different antennas. Next, we propose a new algorithm that extends Barone’s method and performs joint estimations of the number of targets and key parameters.
C. Multitarget Localization Algorithm

In the algorithm, we first extend Barone’s method to a multiantenna case for estimating the number of targets and their ranges. Next, we employ a Least-Square algorithm to extract all the angle information. Finally, their coordinates are calculated. Fig. 3 shows the steps of the proposed algorithm.

Step 1: Compute \( \hat{T} \) and \( \tilde{T} \).

Recall the formation of the replication data in (12), which can be further expressed as

\[
y^{(r)}_n = \sum_{i=0}^{T} c_i \xi^n_i + v_n + v^{(r)}_n. \tag{23}
\]

Clearly, for each replicated data, adding more Gaussian white noise does not change the Padé pole \( \xi_i \)'s and the magnitude of residual \( c_i \)'s. From (19), the \( n \)th sampled data can be expressed as

\[
y_{n,k} = \sum_{i=1}^{T} c_{k,i} \xi^n_i + v_{n,k}. \tag{24}
\]

The key insight is, for different antennas (indexed by \( k \)), the only difference lies in the phase of \( c_{k,i} \), while the \( \xi_i \) and the magnitude of the \( c_{k,i} \) are the same among all antennas. Therefore, we can view these 12 sequences of sampled radar data (one from each antenna) as 12 replications of a data sequence whose element corresponds to

\[
\bar{y}_n = \sum_{i=1}^{T} c_i \xi^n_i. \tag{25}
\]

There is no noise component in this sequence. Instead, receiver side noise \( v_{k,n} \) can be seen as manually added noise (i.e., \( v^{(r)}_k \)) in (12) for each replication. From [24], when the replication data only differs in the manually added noise, the Padé poles and the residuals remain the same. Therefore, the positions of all the \( \xi_i \)'s do not change using the sampled radar data. Furthermore, the number of targets is determined by the number of local maxima that has enough estimated Padé poles in its neighbor area on the complex plane, which depends on the magnitude of corresponding residuals \( |\bar{y}_i| \). Though \( c_{k,i} \)'s differ in phase for different antenna, their magnitude are the same. Thus, the distribution of estimated Padé poles \( \xi_i^{(r)} \) remains the same. In other words, the estimated number of targets is not affected by the phase shift among antennas. Here, \( \tilde{T} \) is a placeholder and we will not compute its precise value.

Usually, people make tens of replicated data. If we only use 12 sequences of sampled radar data, the detection results may not be ideal. Thus, we can reuse some results we have already obtained to avoid making new pseudo data. We have got the \( \hat{T} \), estimated Padé poles set \( \hat{U} \) and corresponding residuals set \( \hat{Z} \) from 12 sequences of sampled radar data. By using these parameters, we can still continue to use steps 2 to 5 in Section III-A to get the final estimated \( \tilde{T} \) and \( \tilde{c} \).

For each element in \( \hat{\xi} \), we can compute an estimated \( \hat{\xi} \) from (15) as

\[
\hat{\xi}_i = \frac{1}{2\pi SI_i} \angle\hat{\xi}_i. \tag{26}
\]

The distance from radar to target \( i \) is computed by \( R_i = (c/2)\hat{\xi}_i/\hat{T} \).

Now, we summarize the detailed step 1 as follows.

Remark 1: Since \( \angle\hat{\xi}_i \in (0, 2\pi] \), the maximum \( \hat{\xi}_i \) is \( (1/SI_i) \), and the maximum estimated range of the target that can be uniquely determined satisfies \( \hat{R}_{\text{max}} = (c/2SI_i) \).

Step 2: Compute \( \hat{\omega}_x \) and \( \hat{\omega}_y \).

From the estimated Padé poles \( \hat{\xi} \), we construct a Vandermonde matrix

\[
\hat{V} = \text{Vander}(\hat{\xi}_1, \hat{\xi}_2, \ldots, \hat{\xi}_T) = \begin{bmatrix}
\hat{\xi}_0 & \hat{\xi}_1 & \cdots & \hat{\xi}_{N_{\text{SAMP}}-1} \\
\hat{\xi}_0 & \hat{\xi}_1 & \cdots & \hat{\xi}_{N_{\text{SAMP}}-1} \\
\vdots & \vdots & \ddots & \vdots \\
\hat{\xi}_0 & \hat{\xi}_1 & \cdots & \hat{\xi}_{N_{\text{SAMP}}-1}
\end{bmatrix} \tag{27}
\]

Let \( \hat{\xi}_k \) be the \( k \)th column of \( \hat{C}^T \), \( k = 1, 2, 3, \ldots, 12 \). We have \( y_{\text{IF},k} = \hat{V} \hat{\xi}_k + y_{\text{IF}} \). Due to the separable of parameters of each target and easy computation, we usually employ the Least-Square algorithm to resolve the remaining parameters. The close-form solution to the Least Square optimization problem \( \min_{[\\|y_{\text{IF},k} - \hat{V} \hat{\xi}_k\|]} \) is given by \( \hat{\xi}_k = (\hat{V}^T \hat{V})^{-1} \hat{V}^T y_{\text{IF},k} \). Repeating using the above solution for all \( k = 1, 2, 3, \ldots, 12 \), we can get the estimated \( \hat{C}^T \). Let \( \hat{c}i \) and \( ci \) be the \( i \)th column of \( \hat{C} \) (or the \( i \)th row of \( \hat{C}^T \)) and \( C \), respectively. \( ci \) is determined by the remaining parameters \( \alpha_i, \omega_{x,i}, \) and \( \omega_{z,i} \) of target \( i \), which can be estimated by solving another Least-Square optimization problem

\[
\min_{\alpha_i, \omega_{x,i}, \omega_{z,i}} f(\alpha_i, \omega_{x,i}, \omega_{z,i}) \tag{28}
\]
where $f(\alpha_i, \omega_{x,i}, \omega_{z,i}) = \|\tilde{c}_i - c_i\|^2$. The solution to the optimization problem can be transformed to the formula that we can get the value of estimated parameters directly

$$\hat{\gamma} = \arg\max_l \left| \hat{B}^{1:8}[l] + \hat{B}^{9:12}[l] \right|$$

(29)

where

$$\hat{B}^{1:8}[l] = \sum_{n=0}^{N_{DFT}-1} \tilde{d}_l^{1:8}[n] e^{-j\omega_{x,i}[l] n},$$

$$\hat{B}^{9:12}[l] = \sum_{n=0}^{N_{DFT}-1} \tilde{d}_l^{9:12}[n] e^{-j\omega_{z,i}[l] n},$$

(30)

(31)

in which, $\tilde{d}_l^{1:8} = \begin{bmatrix} \tilde{c}_l^{1:8H} \\ 0 \end{bmatrix}$ and $\tilde{d}_l^{9:12} = \begin{bmatrix} \tilde{c}_l^{9:12H} \\ 0 \end{bmatrix}$. The detailed transformation steps have been included in Appendix.

Let $\omega_{x,i} = (2\pi l / N_{DFT})$, where $N_{DFT}$ is the length of $\tilde{d}_l^{1:8}$ and $\tilde{d}_l^{9:12}$, $0 \leq l \leq N_{DFT} - 1$. $T$ can be determined by finding the maximum value within $[0, N_{DFT} - 1]$. Then, the phase shift between azimuth receiver antennas for target $i$ can be obtained as

$$\hat{\omega}_{x,i} = \frac{2\pi \hat{\gamma}}{N_{DFT}}.$$  

(32)

Replacing $\hat{\omega}_{x,i}$ in the definition of $B^{1:8}$ and $B^{9:12}$ (46), (47), and the phase of $\hat{B}^{1:8}$ and $\hat{B}^{9:12}$ can be obtained by

$$\hat{\theta}_{B^{1:8}} = \frac{\hat{\omega}_{B^{1:8}}}{\hat{\omega}_{B^{1:8}}}, \hat{\theta}_{B^{9:12}} = \frac{\hat{\omega}_{B^{9:12}}}{\hat{\omega}_{B^{9:12}}}.$$  

(33)

Finally, substitute $\hat{\omega}_{x,i}$, $\hat{\theta}_{B^{1:8}}$, and $\hat{\theta}_{B^{9:12}}$ in (48), we get

$$\hat{\omega}_{z,i} = 2\hat{\omega}_{x,i} + \hat{\theta}_{B^{1:8}} - \hat{\theta}_{B^{9:12}}.$$  

(34)

Now, we summarize the detailed step 2 as follows.

**Step 3:** Compute the coordinates of each target.

For target $i$, we can determine its location $\hat{\mathbf{P}}_i = [\hat{x}_i, \hat{y}_i, \hat{z}_i]$ from $\hat{R}_i$, $\hat{\omega}_{x,i}$ and $\hat{\omega}_{z,i}$ as

$$\hat{x}_i = \hat{R}_i \cos \hat{\theta}_i \sin \hat{\phi}_i = \hat{R}_i \hat{\omega}_{x,i} \pi,$$  

(35)

$$\hat{z}_i = \hat{R}_i \sin \hat{\theta}_i = \hat{R}_i \hat{\omega}_{z,i} \pi,$$  

(36)

$$\hat{y}_i = \sqrt{\hat{R}_i^2 - \hat{x}_i^2 - \hat{z}_i^2}.$$  

(37)

Besides that, according to (1) and (2), the azimuth angle and elevation angle of target $i$ can be computed as

$$\hat{\phi}_i = \arcsin \frac{\hat{\omega}_{x,i} \pi}{\hat{\phi}_i}, \hat{\theta}_i = \arcsin \frac{\hat{\omega}_{x,i} \pi \cos \phi_i}{\hat{\phi}_i}.$$  

(38)

**Remark 2:** The complexity of the proposed algorithm can be analyzed as follows. Step 1 of Barone’s method is $O(N_{SAMP}^2)$, followed by $O(2^T(N_{SAMP}/2 + 1))$ in step 2 where $l$ is the number of points along the lattice edge. Let the number of local maxima be $T$. The complexity of step 3 and 4 is $O(T(N_{SAMP}/2 + 1))$ and that of step 5 is $O(T^2)$. Thus, the overall complexity of Barone’s method is $O(N_{SAMP}^2 + 2^T(N_{SAMP}/2 + 1) + T^2(N_{SAMP}/2 + 1) + T^3)$. The complexity of Algorithm 2 is mainly contributed by calculating $\hat{\mathbf{C}}$ and $\hat{\mathbf{T}}$, which is $O(T^2 N_{SAMP} + T^3)$. Thus, the overall complexity of Algorithm 2 is $O(T^2 N_{SAMP} + T^3 + T^3 N_{SAMP} + N_{DFT} \log N_{DFT})$.

**IV. SIMULATION STUDY**

In this section, we conduct simulations to evaluate the performance of the proposed algorithm.

**A. Simulation Settings**

We have implemented the proposed algorithm in MATLAB. The simulation settings for the radar are as follows: 1) the start frequency and end frequency are 77 and 81 GHz, respectively; 2) the three transmit antennas take turns in sending chirp signals of 58-us length in the order of TX1, TX3, and TX2; 3) the IF signals at the 12 virtual receiver antennas are sampled between 7 and 57 us after a chirp has been sent. The total sampling number is 225.

For the targets and the environment, we vary the SNR from 0 to 30 dB in 5-dB increments with complex additive Gaussian white noise. Under each SNR situation, 10,000 Monte-Carlo experiments are repeated with randomly generated $T$ targets with ranges between 0.05 and 9 m, azimuth angles between $-28^\circ$ and $28^\circ$, and elevation angles between $-14^\circ$ and $14^\circ$.

Two metrics are used to evaluate the performance of different algorithms: the average number of detected targets under different SNRs, and target location estimation errors. To calculate the location errors, we need to first associate each estimated target with an actual target. Let $P_i = [x_i, y_i, z_i]$, $i = 1, 2, 3, \ldots, T$, be the actual locations of $T$ target, and $\hat{P}_j = [\hat{x}_j, \hat{y}_j, \hat{z}_j]$, $i = 1, 2, 3, \ldots, T$, be the estimated locations, where $T$ is the estimated number of targets. We perform target association using the Kuhn–Munkres (KM) algorithm by solving the following bipartite matching problem:

$$\min_{i,j} 1 \sum_{i=1}^{T} \sum_{j=1}^{T} S_{ij} x_{ij}$$

s.t.  

$$S_{ij} = \|P_i - \hat{P}_j\|,$$

$$x_{ij} \in \{0, 1\},$$

$$\sum_{j=1}^{T} x_{ij} \leq 1,$$

$$\sum_{i=1}^{T} x_{ij} \leq 1,$$

$$\sum_{i=1}^{T} \sum_{j=1}^{T} x_{ij} = T.$$  

(39)
where $\tilde{T} = \min(T, \bar{T})$. We denote the resulting pairs by $(\mathbf{P}_t, \bar{\mathbf{P}}_t)$, $t = 1, 2, 3, \ldots, \tilde{T}$. Finally, the location estimation error is then computed as

$$D = \frac{1}{\tilde{T}} \sum_{t=1}^{\tilde{T}} \left\| \mathbf{P}_t - \bar{\mathbf{P}}_t \right\|.$$

With $\tilde{T}$ pairs of associated targets, the MAE of all polar coordinate parameters (i.e., $R$, $\theta$, and $\phi$) are given by

$$\text{MAE}_R = \frac{1}{\tilde{T}} \sum_{t=1}^{\tilde{T}} |R_t - \bar{R}_t|,$$

$$\text{MAE}_\theta = \frac{1}{\tilde{T}} \sum_{t=1}^{\tilde{T}} |\theta_t - \bar{\theta}_t|,$$

$$\text{MAE}_\phi = \frac{1}{\tilde{T}} \sum_{t=1}^{\tilde{T}} |\phi_t - \bar{\phi}_t|.$$

### B. Baseline Algorithms

For comparison purposes, we have also implemented three baseline algorithms: 1) 2D-FFT; 2) 2D-MUSIC; and 3) 3D-MUSIC. In the 2D-FFT algorithm, we apply range-FFT to the sampled radar data matrix and find peaks in the spectrum. CFAR detection and group peaking are used to estimate the number of targets and their ranges. Next, we apply angle-FFT to the data at each detected peak from all antennas to obtain the angle information. In 2D-MUSIC, target detections and range estimation are the same as in 2D-FFT, but angle-FFT is replaced by the MUSIC algorithm to estimate the AoAs of potential targets.

The 3D-MUSIC algorithm is based on the approach in [18]. It precalculates range steering vectors, azimuth, and elevation angle steering vectors. The number of targeted is estimated by MDL [19]. To accelerate the computation time, after the angle information. In 2D-MUSIC, target detections and range estimation are the same as in 2D-FFT, but angle-FFT is replaced by the MUSIC algorithm to estimate the AoAs of potential targets.

The key parameters of the three algorithms are summarized in Table II and the time complexity analysis has been put in Table III. Note that $L_a$ is the CFAR noise averaging window length and $N_a$ is the number of angle bins. In our case, $N_a$ equals to $N_{\text{DFT}}$. All the baseline algorithms except 3D-MUSIC have been evaluated in 10,000 Monte-Carlo experiments for each SNR situation. We only run 100 experiments per SNR scenario for 3D-MUSIC due to its extremely high time complexity.

### C. Results and Analysis

1) Average Number of Detected Targets: Fig. 4 shows that with three or six targets, the average number of detected targets under different SNRs. It can be observed the proposed method under-estimate the number of targets in the low SNR regime but outperforms the 2D-FFT method when the SNR is above 8 dB for three targets and 18 dB for six targets. In addition, MDL, a method based on information-theoretic criteria, has the best performance in nearly all SNR regimes.

Recall that from Padé approximant theory, the candidate Padé poles are among the local maxima of complex measure $\tilde{S}_{N_{\text{DFT}}}(\zeta, \sigma^2)$. However, in the low SNR situation, the energy of the noise and the energy of the signal are comparable. Thus, the local maxima due to signal and those from noise are of similar values. From [24], the higher local maxima, the higher the probability for more Padé poles in its neighboring area. However, due to noise, the number of large local maxima increases on the lattice. As a result, the number of Padé poles around the local maxima corresponding to a real target may drop below the predefined percentage (e.g., >50% of $R$). In contrast, in the higher SNR regime, large local maxima typically corresponds to the signals reflected from real targets. They can attract sufficient number of Padé poles to exceed the threshold and thus can be detected.

The 2D-FFT method performs well when targets are well separated. However, its range resolution is determined by the sweeping bandwidth $B$ (e.g., 4 GHz) and its best angle resolution is given by $2/N$ at the boresight direction. If two targets fall in the same range and angle bins, they cannot be separated in the 2D-FFT method. The choice of the CFAR threshold also affects the number of detected targets in low SNRs by trading-off false positive rates with detection probabilities.

2) Location Errors: Fig. 5 shows the distance between estimated and true locations for three and six targets under different SNRs. As expected, as the SNR increases, average location errors decreases for all algorithms. In comparison, the proposed algorithm always achieves lower location errors than 2D-FFT and has better performance than 2D-MUSIC in a high
SNR regime. We further plot the MAE of range, azimuth, and elevation angle estimation errors for three and six targets in Fig. 6(b). It can be observed from Fig. 6(a) that the proposed algorithm has lower errors in estimating range parameters. However, in estimating azimuth and elevation angles, 2D-FFT and 2D-MUSIC algorithms have higher accuracy in low SNRs, but the proposed algorithm outperforms both of them under high SNRs.

While the AoA and localization accuracy of 2D-MUSIC consistently outperforms 2D-FFT in all scenarios, the AoA estimations of 3D-MUSIC are worse than those of 2D-FFT in the low SNR regime for both 3-target and 6-target cases. This can be attributed to two factors. First, the 3D-MUSIC algorithm in [18] only utilizes stacked-up auto-correlation matrices for each antenna element over time (instead of exploring the correlation in both space and time. Second, to approximate auto-correlation matrices in time, a large number of frames are needed. Otherwise, the resulting approximation is biased. This is especially true in the low SNR regime. It should be noted that due to the peak search approach adopted, the accuracy of 3D-MUSIC reported errs on the optimistic side. Nevertheless, the proposed method has comparable or lower localization errors as 3D-MUSIC in mid- to high-SNR regimes at significantly lower computation costs. This demonstrates the advantages of the proposed method.

3) Impacts of Data Samples: The purpose of this set of experiments is to understand how the number of samples affects the performance of the proposed method. In the experiments, the SNRs are set to 30 dB and the length of sampled data is chosen from 12, 42, 73, 103, 134, 164, 195, and 225. 10,000 experiments are performed for each length. Fig. 7 shows the results for three and six targets. As shown in those two figures, as the number of data samples becomes larger, the number of detected targets approaches the actual value while the estimated target locations get closer to the ground truth locations. For the previous experiments, we only use the first 80 sampled data from each virtual receiver antenna to achieve the results and any 80 continuous data samples can be used to get the similar results. In fact, about 100 data samples are sufficient to detect all targets accurately, and using less data samples is advantageous in detecting and locating mobile targets. We plan to derive this relationship mathematically as part of our future research.

V. TESTBED EVALUATION

We have implemented the proposed and baseline algorithms using a COTS mmWave radar, i.e., Taxes Instruments IWR6843ISK. The radar has three transmit antennas and four-receiver antennas, where the antenna arrangement is the same.
as that in Fig. 1(a). Three corner reflectors, each of which is constructed by three metal plates perpendicular to one another, are used to represent three-point targets. We use Optitrack, an optical motion capture system, to determine the ground truth locations of the radar and targets. The detailed parameters are summarized in Table IV and the experiment environment is shown in Fig. 8.

Table V summarizes the average range and directional errors of different approaches over multiple measurements. From the experiment results, we see that the proposed algorithm has lower average errors than baseline algorithms in locations, ranges, and azimuth angles. Nonetheless, we notice that the measured errors are far larger than those from simulations. The discrepancy between simulation and experimental results can be attributed to several sources. First and foremost, interference from surrounding objects, such as walls, pillars, and furniture, can significantly impact the performance. Second, the edge length of corner reflectors is 8 in, making them multisample point targets. Other reasons may include the limitations of the radar board, e.g., the accuracy of its mixer, ADC, and the cosine pattern of its antennas. In terms of the MUSIC algorithms, we only utilize one received chip in the analysis, which may not be sufficient to estimate covariance matrices accurately in such an indoor environment with high interference.

VI. Conclusion

In this article, we proposed a new algorithm to estimate the locations of multitargets by using the radar with irregular antenna placement. We first derive the radar system model of the trapezoidal antenna array. Then, we have a series of signal processing steps to obtain the sampled IF signal. Next, we give the detailed steps of the proposed algorithm which includes the application and extension of Barone’s idea and Least-Square algorithm. Finally, by using all the above technologies, we can get the estimated number of targets and their positions in the 3-D space.

According to the simulation results, we can conclude that the proposed algorithm can detect nearly all the targets especially in the high SNR regime, and the probability of losing targets is very low compared with 2D-FFT method. Besides, as for those detected targets, the location error is significantly lower than that of the 2D-FFT and MUSIC algorithm. We also find that the resolution and performance of the proposed algorithm does not heavily depend on the number of sampled data and hardware design. Compared with the 2D-FFT algorithm which needs hundreds of data samples to cover a large indoor area, the proposed algorithm only needs less than half of them to get an even better result.

APPENDIX

We now show the detailed transformation steps of the Least-Square optimization problem

$$\min_{\alpha_i,\omega_{x,i},\omega_{z,i}} f(\alpha_i, \omega_{x,i}, \omega_{z,i})$$ (44)

where $f(\alpha_i, \omega_{x,i}, \omega_{z,i}) = \|\hat{c}_i - c_i\|^2$. Since $\|c_i\|^2 = 12\alpha_i^2$

$$f(\alpha_i, \omega_{x,i}, \omega_{z,i}) = \|\hat{c}_i\|^2 + \|c_i\|^2 - \overline{c}_i^H \bar{c}_i - \overline{c}_i^H \bar{c}_i$$

$$= \|\hat{c}_i\|^2 + 12\alpha_i^2 - 2\text{Re}(\overline{c}_i^H \bar{c}_i).$$

Let $\tilde{c}_i = e^{j\psi} \bar{h}_s$, where $\psi = 2\pi(\tilde{c}_i S_{\text{Start}} + f_c \tilde{t}_i - (1/2)S_{\tilde{f}_i}^2)$. Then, $c_i = \alpha_i \overline{\tilde{c}_i}$; and

$$f(\alpha_i, \omega_{x,i}, \omega_{z,i}) = \|\tilde{c}_i\|^2 + 12\alpha_i^2 - 2\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i).$$ (45)

Take the derivative of (45) with respect to $\alpha_i$, we have $(\partial f/\partial \alpha_i) = 24\alpha_i - 2\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i)$. From the KKT conditions, to minimize (44), we have $\hat{\alpha}_i = (\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i))/12$. Next, we substitute $\hat{\alpha}_i$ in $f(\alpha_i, \omega_{x,i}, \omega_{z,i})$ to get

$$f(\alpha_i, \omega_{x,i}, \omega_{z,i})$$

$$= \|\tilde{c}_i\|^2 + 12 \left(\frac{\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i)}{12}\right)^2 - 2\frac{\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i)}{12}\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i)$$

$$= \|\tilde{c}_i\|^2 - \frac{1}{12}\left(\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i)\right)^2.$$ Clearly, since $\|\tilde{c}_i\|^2$ is a constant, in order to minimize $f(\alpha_i, \omega_{x,i}, \omega_{z,i})$, we need to maximize $\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i)$. Due to the phase shift differences, we consider the bottom eight virtual antennas and top 4 receiver antennas separately, and write $\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i)$ as $\text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i) = \text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i) + \text{Re}(\overline{\tilde{c}_i}^H \bar{c}_i)$. Let

$${B}^{1:8} = \left|{B}^{1:8}\right| e^{j\phi_{1:8}} = \left[1, e^{-j\omega_{x,1}}, \ldots, e^{-j\omega_{x,7}}\right]^T$$ (46)
\[
B^{9;12} = B^{1;8} e^{i \theta_{g;12}} = c_i^{9;12H} \left[ 1, e^{-j \omega_{c,i}}, e^{-j 2 \omega_{c,i}}, e^{-j 3 \omega_{c,i}} \right]^T.
\]

(47)

In order to maximize \(|\text{Re}(c_i^{1;8H} c_j^{9;12}) + \text{Re}(c_i^{9;12H} c_j^{1;8})|\), the following conditions must be achieved:

\[
\phi_{g;12} + \psi = 2k\pi \quad \text{and} \quad \phi_{g;12} + \psi + \omega_{c,i} - 2\omega_{c,i} = 2k\pi \quad (k \in \mathbb{Z}).
\]

(48)

Under the above conditions, (44) is equivalent to

\[
\max_{\omega_{c,i}, \psi} \left| B^{1;8} \right| + \left| B^{9;12} \right|.
\]

The form of \(B^{1;8}\) and \(B^{9;12}\) can be determined by taking discrete Fourier transform (DFT) of \(c_i^{1;8H}\) and \(c_i^{9;12H}\). Usually, to achieve a higher precision, we append 0 s to \(c_i^{1;8H}\) and \(c_i^{9;12H}\) to make their lengths to a preset value, e.g., \(\tilde{d}_i^{1;8} = \begin{bmatrix} c_i^{1;8H} \\ 0 \end{bmatrix}\)

and \(\tilde{d}_i^{9;12} = \begin{bmatrix} c_i^{9;12H} \\ 0 \end{bmatrix}\). Next, we can take DFT to \(\tilde{d}_i^{1;8}\) and \(\tilde{d}_i^{9;12}\) as

\[
\tilde{B}^{1;8}[l] = \sum_{n=0}^{N_{\text{DFT}}-1} \tilde{d}_i^{1;8}[n] e^{-j \omega_{c,i} [n] l},
\]

\[
\tilde{B}^{9;12}[l] = \sum_{n=0}^{N_{\text{DFT}}-1} \tilde{d}_i^{9;12}[n] e^{-j \omega_{c,i} [n] l}.
\]

Once we add the magnitude of the DFT results together, (44) has been transformed to

\[
\hat{\gamma} = \arg \max \left| \tilde{B}^{1;8}[l] \right| + \left| \tilde{B}^{9;12}[l] \right|.
\]

REFERENCES


Wei Zhao (Student Member, IEEE) received the B.Eng. degree in automation from Xidian University, Xi’an, China, in 2014, and the M.A.Sc. degree in electrical and computer engineering from McMaster University, Hamilton, ON, Canada, in 2017, where he is currently pursuing the Ph.D. degree with the Department of Computing and Software.

Jian-Kang Zhang (Senior Member, IEEE) received the B.S. degree in information science (mathematics) from Shaanxi Normal University, Xi’an, China, in 1983, the M.S. degree in information and computational science (mathematics) from Northwest University, Xi’an, in 1988, and the Ph.D. degree in electrical engineering from Xi’an Jiaotong University, Xi’an, in 1999.

Xiao-Ping Zhang (Fellow, IEEE) received the B.S. and Ph.D. degrees in electronic engineering from Tsinghua University, Beijing, China, in 1992 and 1996, respectively, and the M.B.A. degree (with Honors) in finance, economics, and entrepreneurship from the University of Chicago Booth School of Business, Chicago, IL, USA, in 2008.

Rong Zheng (Senior Member, IEEE) received the B.E. and M.E. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1990 and 1995, respectively, and the Ph.D. degree from the Department of Computer Science, University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2004.

Since Fall 2000, he has been with the Department of Electrical, Computer and Biomedical Engineering, Ryerson University, Toronto, ON, Canada, where he is currently a Professor and the Director of the Communication and Signal Processing Applications Laboratory.

Since Fall 2000, he has been with the Department of Electrical, Computer and Biomedical Engineering, Ryerson University, Toronto, ON, Canada, where he is currently a Professor and the Director of the Communication and Signal Processing Applications Laboratory.