Real-time DoA Estimation for Automotive Radar

Yubo Wu, Chengzhang Li, Y. Thomas Hou, Wenjing Lou
Virginia Tech, Blacksburg, VA, USA

***Abstract*— For automotive radar, direction-of-arrival (DoA) estimation is the most challenging component in the target detection problem. To cope highly dynamic driving conditions and to achieve full autonomy, there are stringent requirements on the processing time and DoA estimation resolution. None of the state-of-the-art methods can accomplish both at the same time: FFT-based algorithm is computationally fast but cannot provide high resolution, while subspace-based algorithms such as MUSIC and ESPRIT can achieve super-resolution but cannot meet the timing requirement. In this paper, we present MARS – a real-time super-resolution algorithm based on maximum likelihood (ML) estimation. In contrast to traditional ML estimation, MARS exploits the intrinsic correlation between the input data of adjacent time slots to reduce the search space. To further reduce computation time, MARS decomposes the problems in each step into independent sub-problems that can be efficiently executed on GPU parallel computing platform. Simulation experiments show that MARS can achieve 1° super-resolution in DoA estimation under 1 ms.**

I. Introduction

The field of autonomous driving has shown a potential to revolutionize future transportation. A critical component to achieve a high level of automation is to obtain accurate road information in real time. Although camera and lidar can offer a fine resolution, their performances are sensitive to their environments. In contrast, radars are robust to the environmental conditions and are the most reliable. It is expected that radar will remain an indispensable component in advanced driver assistant systems (ADASs).

A fundamental problem in automotive radar is to achieve a high resolution direction-of-arrival (DoA) estimation in real time. For a high resolution, a 1° resolution (a.k.a super-resolution) is needed [1], [7]. For real time requirement, although there is no consensus in the literature on the exact number, we argue that $\sim$1 ms is needed eventually. This is because the end-to-end time delay from receiving signals at the radar to taking specific actions on the vehicle should be no more than 10 ms. So taking into account of data transfer among the sub-systems, data fusion, decision making, etc., the available time for DoA estimation can only be $\sim$1 ms. In summary, a fundamental challenge in automotive radar is to achieve 1° super-resolution in $\sim$1 ms.

DoA estimation has been investigated for many years. The maximum likelihood (ML) method enjoys the high estimation accuracy and robustness, but its computational time is prohibitively high. Fast Fourier transform (FFT) based method is computationally efficient, but requires a very large number of antennas to achieve 1° super-resolution. Subspace-based methods such as MUSIC [2] and ESPRIT [3] can achieve 1° super-resolution with a small number of antennas, but at the cost of much higher computation time and suffers from poor performance under low SNR.

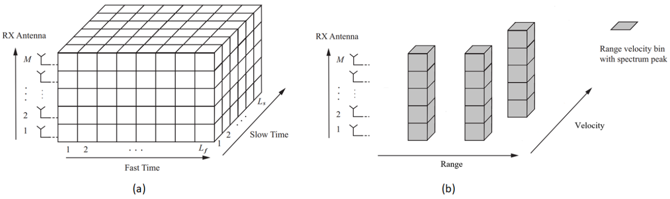


Fig. 1. (a) Data Cube. (b) Range-velocity bin with spectrum peak.

There are a number of recent works devoted to DoA estimation of automotive radar. The authors of [4] designed a modified DFT-based DoA estimation method by adding virtual signals to improve accuracy. However, this method cannot achieve 1° resolution requirement. In [5], the range, velocity and angular estimation were modeled as a multi-dimensional frequency estimation problem. The computation time is high due to the number of iterations required to refine the results and cannot meet the 1 ms time requirement. In [6], the MUSIC algorithm was implemented on graphics processing unit (GPU) and other multi-core processors. Although the authors showed the computation of the GPU-based algorithm is faster than the others, it is still on the order of $\sim$100 ms. To the best of our knowledge, none of the existing algorithms can consistently achieve 1° super-resolution within 1 ms for most realistic scenarios.

In this paper, we present MARS–a real-time super-resolution DoA estimation algorithm for automotive radar. We start with the same formulation for the DoA estimation problem as ML estimation. Unlike ESPRIT and MUSIC whose performances deteriorate under low SNR, ML estimation can find the optimal DoA that meets the 1° super-resolution requirement in most cases. To cut down the computation time associated with ML, we exploit the correlation between the input data of adjacent time slots. Since the range, velocity and DoA of each target only have modest change over two consecutive time slots, we propose to use the estimation results from the previous time slot to form a promising and reduced searching space. To further reduce computation time, we propose to decompose the problems at each step into independent sub-problems and employ GPU parallel computing. Through simulation experiments, we find that MARS is the only algorithm which can handle up to 100 bins of reflection points with 1° super-resolution within 1 ms.

II. Signal Model for DoA Estimation

Consider a moving target with a relative range R (in meter) and relative velocity V (in m/s) to a frequency modulated continuous wave (FMCW) radar that is equipped with an array of M antenna elements. The space between adjacent antenna elements is d m. The carrier frequency of the FMCW radar is f_c Hz. Denote C as the speed of the light. Then the corresponding wavelength λ satisfies $\lambda = C/f_c$. Denote T_c (in second) as the chirp duration and S (in Hz/s) as the chirp slope of the radar. Denote the sampling rate of the radar receiver as f_s Hz. Denote l_f as the sampling index within one chirp (a.k.a. the fast time index), i.e., $l_f = 1, \cdots, L_f$, where $L_f = f_s \cdot T_c$. Denote l_s as the index of chirps within one coherent processing interval (CPI) (a.k.a. the slow time index), i.e., $l_s = 1, \cdots, L_s$. Denote $m = 1, \cdots, M$ as the index of receive antenna element. Denote ϕ as the Azimuth angle of the target to the receive antenna. Assume the additive white Gaussian noise (AWGN) is $\omega[l_f, l_s, m] \in \mathcal{CN}\left(0, \sigma^2\right)$ and ignore the distortion of reflected signal. Denote $\xi[l_f, l_s, m]$ as the attenuation caused by the path loss, the antenna gain and the radar cross section (RCS) of the target. With the above notations, the output of the radar receiver within one frame can be represented as:

$$\begin{aligned} x[l_f, l_s, m] \approx \xi[l_f, l_s, m] \exp\{j2\pi[(f_b + f_d)\frac{l_f}{f_s} + \\ \frac{f_c m d \sin(\phi)}{C} + f_d l_s T_c + \frac{2 f_c R}{C}]\} + \omega[l_f, l_s, m] \end{aligned} \quad (1)$$

In (1), $f_b = \frac{2SR}{c}$ is beat frequency and is dependent on the range of target; $f_d = \frac{2V}{\lambda}$ is the Doppler shift due to the movement of the target.

In (1), the measurement $x[l_f, l_s, m]$ spans over the fast time, the slow time and the space dimension as illustrated in Fig. 1(a). For velocity estimation, the Doppler shift f_d can be extracted by L_s-point FFT across the slow time dimension to obtain the velocity V. Similarly, f_b can be extracted by L_f-point FFT across the fast time dimension and then range R can be obtained. The spectrum after 2-D FFT spans over a 2-dimension (2D) range-velocity grid, if there are multiple targets then multiple spectrum peaks may exist and they can be specified by R and V as shown in Fig. 1(b). Moreover, each peak may be associated with multiple incident signals from a clutter of targets with the same range and velocity but different DoAs. Denote N_e as the total number of bins containing the spectrum peaks. Denote N_i as the number of DoAs to measure in the i-th bin. Then the peak in the i-th bin corresponds to a linear combination of N_i incident signals. Its measurement for DoA estimation at the m-th antenna can be obtained as:

$$x_i[m] = \sum_{n=1}^{N_i} \xi_n^i[m] e^{j2\pi \frac{m d \sin(\phi_n^i)}{\lambda}} + \omega_n^i[m]. \quad (2)$$

For M antennas, we have a set $\mathcal{X}_i = \{x_i[1], \cdots, x_i[M]\}$, and for a total of N_e bins, we have $\mathcal{X} = \{\mathcal{X}_1, \cdots, \mathcal{X}_{N_e}\}$.

The goal of DoA estimation is to design a method (denoted as Ψ) to estimate ϕ_n^i via $\tilde{\phi}_n^i = \Psi(\mathcal{X}_i)$ such that the following objective function is minimized:

$$\min_{\Psi} \sum_{i=1}^{N_e} \sum_{n=1}^{N_i} |\phi_n^i - \tilde{\phi}_n^i|^2 . \quad (3)$$

III. MARS: A Novel Maximum-Likelihood based Real-time Super-resolution Algorithm

A. Main Idea

The goal of this work is to achieve $1°$ super-resolution in DoA estimation in real-time (under 1 ms). The main idea of MARS is to use ML estimation as our objective function because it can provide optimal solution and can be solved in parallel. To address the prohibitive computational time associated with the ML estimation, we exploit intrinsic correlation in range, velocity and DoA of a target between successive time slots. Specifically, we use the DoA estimation result from the previous time slot as the center of the search space and identify a much smaller promising search space for the current time slot (based on the target's range and velocity). In each step of MARS, we decompose a problem into a larger number of independent sub-problems and use GPU parallel computing to accelerate computational time. We briefly describe these ideas in the rest of this section.

B. Objective function

In (3), since ϕ_n^i only depends on $\mathcal{X}_i$, the complete problem (3) can be decomposed into N_e sub-problems based on each bin with spectrum peak. In each sub-problem, under ML estimation, the complex Gaussian noise at each antenna is assumed to be temporally and spatially independent. To solve the i-th sub-problem, the ML objective function can be reduced to the following:

$$\min_{\tilde{\phi}_n^i} \sum_{m=1}^{M} \left| x_i[m] - \sum_{n=1}^{N_i} \xi_n^i \exp\left[j2\pi\left(\frac{f_c m d \sin(\tilde{\phi}_n^i)}{c}\right)\right] \right|^2 \quad (4)$$

Thus, the search space of the sub-problem becomes (ϕ) and the size of the search space is $|(\phi)|^{N_i}$ which is determined by three parameters (the fields of view of radar, granularity and the number of targets in the same range-velocity bin). Fields of view differ between different types of radars. For example, a short-range radar may have an azimuthal filed of view from $-80°$ to $80°$ due to its wide beam-width. Granularity is determined by the required resolution and estimation accuracy, which is $1°$ in our case. The number of targets falling in the same range-velocity bin corresponds to the number of reflection points (from the same or different objects).

C. Reducing Search Space

Since the search space of (4) is prohibitively large, we propose to reduce its size by exploiting the correlation between estimation results from adjacent time slots. To do this, we need to associate estimation results from the last time slot with bins in the current time slot.

The Euclidean distance between the i-th bin in the current time slot t (with coordinate $(R_{t,i}, V_{t,i})$) and the j-th bin in the previous time slot $t-1$ (with coordinate $(R_{t-1,j}, V_{t-1,j})$) is:

$$d_{i,j}(t) = \sqrt{\left(\frac{R_{t,i} - R_{t-1,j}}{R_r}\right)^2 + \left(\frac{V_{t,i} - V_{t-1,j}}{V_r}\right)^2}, \quad (5)$$

where R_r and V_r are the range and velocity resolutions, respectively. Define k_i in $(t-1)$ as:

$$k_i = \arg\min_{j} \; d_{i,j}(t) \; . \tag{6}$$

To find k_i, we can start from a center point $(R_{t,i}, V_{t,i})$ with a radius Δ. If $d_{i,k_i}(t) \leq \Delta$, then bin i can be associated with bin k_i and we can use reduced search space for DoA estimation. Otherwise, we cannot find k_i and the search space cannot be reduced.

If we can find k_i, then by using $\tilde{\phi}_n^{k_i}$ in (4), we can find a sub search space as follows:

$$s_n^{k_i} = \{\phi | \tilde{\phi}_n^{k_i} - e_n^{k_i} \leq \phi \leq \tilde{\phi}_n^{k_i} + e_n^{k_i}\} \tag{7}$$

where $n = 1, \cdots N_{k_i}$ and $e_n^{k_i}$ is the expected DoA deviation between two adjacent time slots. Since the movement of any target on the road is continuous, $e_n^{k_i}$ (in degrees) can be calculated by the following equation:

$$e_n^{k_i} = a\frac{180|(V_{t,i} + V_{t-1,k_i})\tan(\tilde{\phi}_n^{k_i})|T_s}{(R_{t,i} + R_{t-1,k_i})} + b \; , \tag{8}$$

where T_s is the time interval between adjacent estimations. a and b are scale parameters to guarantee that the estimation solution exists in the reduced space. The first term represents potential DoA deviation caused due to mobility while the second term accounts for potential estimation error due to noise. For a far-distant target, the change in DoA is relatively small (between consecutive estimations) but the impact of the noise is large (because of attenuation over large distance). For a nearby target, the reserve is true. Thus, the size of the search space can be reduced substantially and the actual DoA is guaranteed to fall in the reduced search space.

By combining all $s_n^{k_i}$'s, the new search space for the current i-th sub-problem is:

$$S_i = s_1^{k_i} \cup s_2^{k_i} \cdots \cup s_{N_{k_i}}^{k_i} \; . \tag{9}$$

The complete algorithm for MARS is given in Algorithm 1.

D. Parallel Implementation

To accelerate computation time, we implement our algorithm based on parallelism and use GPU platform as follows:

- Step 1. We transfer measurement $\mathcal{X}$ from the range-velocity bins from host to the GPU memory. A total of $N_e M$ signals will be transferred.
- Step 2. The DoA estimation problem (3) is decomposed into N_e independent sub-problems that can be solved in parallel. On a GPU platform, N_e blocks will need to be allocated to all bins respectively. Each block runs one sub-problem independently.
- Step 3. Within each sub-problem, all elements in the reduced search space of (4) are calculated in parallel. On a GPU platform, in the i-th block, there are $|S_i|^{N_i}$ threads, each allocated to compute one element of (4).
- Step 4. we use parallel reduction technique to find a DoA estimation solution from all elements.
- Step 5. Save the estimation result in GPU memory for further use and also transfer it from GPU memory to the host.

[1]This exhaustive search for all bins may take ∼100s ms. But it only occurs once when the radar is turned on.

[2]A new bin is typically detected by a long-range radar with a relatively small DoA search space.

Algorithm 1 MARS

Initialization:
Use 2-D FFT to estimate R and V for all range-velocity bins with spectrum peaks.
Use ML estimation to estimate DoAs for all targets based on exhaustive search.[1]
Save the coordinate of each range-velocity bin (R and V) with spectrum peak and the corresponding estimated $\{\phi\}$
while receiving new input data **do**
    Use 2-D FFT to estimate R and V for all range-velocity bins with spectrum peaks
    for each bin **do**
        if it is a new bin **then**
            Use ML estimation to estimate $\{\phi\}$ based on exhaustive search.[2]
        else
            Reduce search space
            Estimate DoA
        end if
    end for
    Save the current estimation results
end while

IV. Performance Evaluation

In this section, we evaluate the performance of MARS through simulation experiments. We implement MARS on a NVIDIA DGX Station, which is equipped with an Intel Xeon E5-2698 v4 2.2 GHz CPU and a Tesla V100 GPU. We employ NVIDIA CUDA Toolkit 11.2 for implementation on GPU.

A. Setting

We set $M = 16$, which is a typical configuration in radars for autonomous driving applications. Consider a medium-range FMCW radar operating at 77 GHz. The range and velocity resolutions are 30 cm and 3 m/s, respectively [7]. The azumithal filed of view is $[-50°, 50°]$. T_s is set to be 10 ms and contains 10 frames. Denote N as the number of DoAs in one bin.

For MARS, we set $\Delta = 2, a = 2, b = 1$ and the granularity used in the search space is $1°$. We compare results from MARS with those from MUSIC and ESPRIT, both of which are state-of-the-art super-resolution algorithms. For MUSIC and ESPRIT, 10 frames are needed to calculate the covariance matrix [2][3].

B. Performance of DoA Estimation

The RMSEs (Root-mean-square error) of estimated DoA as a function of SNR for different DOA settings are shown in

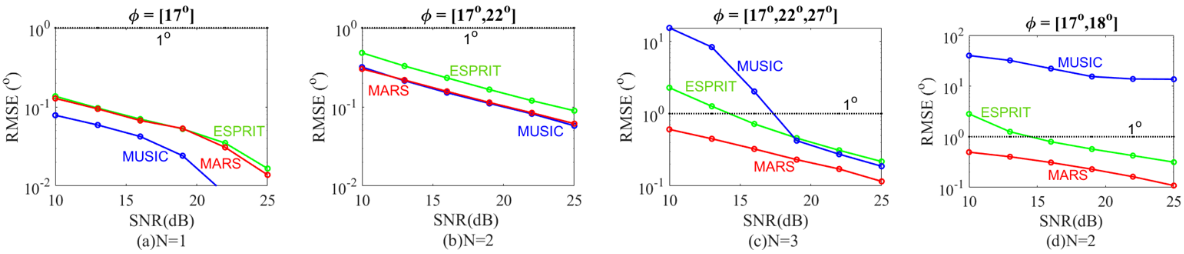
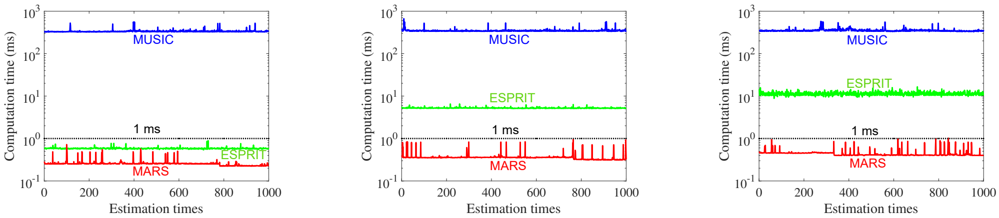


Fig. 2. RMSE vs. SNR for different DoA settings.



(a) N=1 (b) N=2 (c) N=3

Fig. 3. Computation time of MARS vs. ESPRIT and MUSIC under different DoA settings.

Fig. 2. Each point in the figure is obtained by averaging over 1,000 simulations. In Fig. 2(a)–(c), RMSE decreases as SNR increases for all three algorithms (MARS, ESPRIT, MUSIC), which is intuitive. From Fig. 2(a) to (c), the RMSE under each algorithm increases as the number of DoAs increases from 1 to 3. In Fig. 2(a) and (b), the performance of all three algorithms meet the 1° requirement. But in Fig. 2(c), only MARS can meet the 1° requirement over the entire range (SNR $\in [10, 25]$) while ESPRIT requires SNR > 15 and MUSIC requires SNR > 19.

In Fig. 2(d), RMSE versus SNR under 2 DoAs with 1° spacing is shown. We find that MARS is the only algorithm among the three to meet 1° resolution requirement over the entire range (SNR $\in [10, 25]$). On the other hand, ESPRIT requires SNR > 15 to achieve 1° resolution while MUSIC is unable to do so even when SNR increases to 25 dB.

C. Processing time of DoA estimation

We now evaluate the processing time of MARS and compare it to ESPRIT and MUSIC. We set the total number of bins, N_e, to 100. The mobility model is generated by SUMO under an urban scenario and the relative velocity of each target follows a uniform distribution from 0 to 160 km/h. Fig. 3 presents average computation time (over 100 bins) for DoA estimation over 1,000 estimations (over a 10-second period). In Fig. 3(a) to (c), the number of DoAs in each bin N varies from 1 to 3. In all figures, the computation time of MARS is always under 1 ms. In contrast, the computation times of ESPRIT and MUSIC are substantially higher than MARS. In particular, MUSIC algorithm requires 100s more times of computational time than MARS while ESPRIT requires 10s more times of computational time than MARS when $N = 2$ or 3.

V. CONCLUSIONS

In this paper, we presented MARS – a novel real-time super-resolution DoA estimation algorithm for automotive radar. By exploiting the correlation between input data and results of adjacent time slots, we reduce the search space of ML estimation. We also exploited problem decomposition and GPU parallel computing to accelerate the estimation process. Simulation results demonstrated that MARS can achieve 1° super-resolution within 1 ms, which is the first known algorithm to achieve this resolution-time performance under most testing scenarios.

ACKNOWLEDGMENT

This research was supported in part by NSF under grant CNS-1837519, Virginia Commonwealth Cyber Initiative (CCI), and Virginia Tech Institute for Critical Technology and Applied Science (ICTAS).

REFERENCES

[1] F. Roos, D. Kellner, J. Dickmann and C. Waldschmidt, "Reliable orientation estimation of vehicles in high-resolution radar images," in *IEEE Trans. Microw. Theory Techn.*, vol. 64, no. 9, pp. 2986-2993, Sept. 2016.

[2] R. Schmidt,"Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276-280, Mar. 1986.

[3] R. Roy and T. Kailath,"ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 7, pp. 984-995, July 1989.

[4] B. Kim, S. Kim, J. Lee, "A novel DFT-based DOA estimation by a virtual array extension using simple multiplications for FMCW radar" in *Sensros*, MDPI, pp. 1560-1576, 2018.

[5] F. Engels, P. Heidenreich, A. M. Zoubir, F. K. Jondral and M. Wintermantel, "Advances in automotive radar: a framework on computationally efficient high-resolution frequency estimation," in *IEEE Signal Process. Mag.*, vol. 34, no. 2, pp. 36-46, Mar. 2017.

[6] M. W. Majid and M. M. Jamali, "Parallel implementation of MUSIC algorithm on single-core, multi-core and NVIDIA's GPU," in *Proc. IEEE National Aerospace and Electronics Conference (NAECON)*, Dayton, OH, 2011, pp. 146-149.

[7] K. Ramasubramanian and B. Ginsburg, "AWR1243 sensor: highly integrated 76–81-GHz radar front-end for emerging ADAS applications," [Online] Available at http://www.ti.com/lit/wp/spyy003/spyy003.pdf