

A Fast Distributed Direct Position Determination Algorithm Based on Delay and Doppler

Zhuo Qiu

Science and Technology On Blind Signal Processing Laboratory .Chengdu, China

*Corresponding author

Abstract—In the moving sensor network, the direct positioning method (DPD) can greatly improve the positioning performance of the system, but it also brings the problem of high computational complexity and poor stability of the algorithm. In this paper, a distributed fast DPD algorithm based on time-frequency difference is proposed. The complex large-scale matrix operation and the maximal eigenvalue solution in the original DPD algorithm are replaced by the cross ambiguity function (CAF) and the matrix norm respectively. An effective distributed computing method is designed to make full use of the sensor network Of the computation resources. The simulation results show that the proposed method can greatly simplify the computational complexity, improve the stability of the algorithm , while the location performance is equivalent to the original DPD algorithm.

Keywords- DPD; matrix norm; time-frequency difference; cross Ambiguity Function (CAF)

I. INTRODUCTION

In the passive sensor passive positioning method, the most commonly used method is based on time-frequency difference estimation, usually through two steps to achieve: first, collecting the pairs sensor signal sampling data, using CAF to estimate the time-frequency difference parameter. then, using the parameter estimating results to find the geographical location of the target. The model of the received signal is: $r(t) = be^{2\pi f t} s(t - \tau) + \omega(t), 0 \leq t \leq T$. Where $s(t)$ is the transmitted signal, f is the Doppler, τ is the signal's delay, b is path attenuation, $\omega(t)$ is complex Gaussian noise . The time-Frequency difference between two signals can be effectively estimated by finding the peak of the CAF given by:

$$CAF_{f_1, f_2}(\tau, f) = \int_{-\infty}^{+\infty} r_1(t) r_2^*(t + \tau) e^{-j2\pi f t} dt \quad (1)$$

Under the condition of high signal to noise ratio (SNR), the time-frequency difference based on CAF are estimated Effective and unbiased. It even can be approaching CRLB. However, since the parameter estimation and position resolution are separated in the traditional method, resulting in information loss in data process, so the location performance should not be optimal. Moreover, it can not work in the lower SNR condition.

In recent years, many scholars have studied how to directly obtain the target location information from the signal sampling

data. They proposed a direct location method based on the time-frequency difference system^{[1][2]}. By constructing the maximum likelihood function, the signal sampling data can be processed directly, the target position can be searched through geographic grid directly. The DPD method avoids the loss of information caused by two-step separation in the traditional method, and has better location performance, which reduces the requirement of SNR compared with the traditional method. But the original DPD algorithm needs large-scale matrix operation, and it is necessary to transmit all sensor's signal data to the central sensor to complete all the process.

In this paper, through analyzing the basic model of the original DPD algorithm, we improves the DPD algorithm by combining the knowledge of the CAF and matrix theory. Then ,we propose a fast distributed DPD algorithm based on time-frequency difference, Which can effectively reduce the computational load and can effectively balance the computational load in the sensor network. Finally, the validity of the algorithm is examed by simulation.

II. PROBLEM FORMULATION

Similarly to [1], We consider a stationary radio transmitter located at position p_e and L moving receivers. The receivers are synchronized in frequency and time. Each receiver intercepts the transmitted signal at K short intervals along its trajectory. Let $p_{l,k}$ and $v_{l,k}$ where $(l = 1, \dots, L; k = 1, \dots, K)$ denote the position and velocity vectors of the l -th receiver at the k -th interception interval. The complex signal observed by the l -th receiver at the k -th interception interval at time t is :

$$r_{l,k}(t) = b_{l,k} s_k(t - \tau_{l,k}) e^{j2\pi f_{l,k} t} + \omega_{l,k}(t), 0 \leq t \leq T \quad (2)$$

Assume that we collect N time samples, sampled with sampling interval T_s , Define:

$$\begin{aligned} \mathbf{r}_{l,k} &= [r_{l,k}(t_1), \dots, r_{l,k}(t_N)]^T & \mathbf{w}_{l,k} &= [\omega_{l,k}(t_1), \dots, \omega_{l,k}(t_N)]^T \\ \mathbf{s}_k &= [s_k(t_1), \dots, s_k(t_N)]^T & \mathbf{A}_{l,k} &= \text{diag}\{e^{j2\pi f_{l,k} t_1}, \dots, e^{j2\pi f_{l,k} t_N}\} \end{aligned} \quad (3)$$

Given this, (2) can be rewritten in vector format as

$$\mathbf{r}_{l,k} = b_{l,k} \mathbf{A}_{l,k} \mathbf{F}_{l,k} \mathbf{s}_k + \mathbf{w}_{l,k} \quad (4)$$

where $\mathbf{F}_{l,k}$ is a down shift operator, represents delay shift.

We construct the following ML function as

$$C(p) = \sum_{k=1}^K \sum_{l=1}^L \left\| \mathbf{r}_{l,k} - b_{l,k} \mathbf{A}_{l,k} \mathbf{F}_{l,k} \mathbf{s}_k \right\|^2 \quad (5)$$

Maximizing the (5) by least squares, we get

$$\begin{aligned} \hat{b}_{l,k} &= \left[(\mathbf{A}_{l,k} \mathbf{F}_{l,k} \mathbf{s}_k)^H \mathbf{A}_{l,k} \mathbf{F}_{l,k} \mathbf{s}_k \right]^{-1} (\mathbf{A}_{l,k} \mathbf{F}_{l,k} \mathbf{s}_k)^H \mathbf{r}_{l,k} \\ &= (\mathbf{A}_{l,k} \mathbf{F}_{l,k} \mathbf{s}_k)^H \mathbf{r}_{l,k} \end{aligned} \quad (6)$$

We assume $\|\mathbf{s}_k\|^2 = 1$, Substituting (6) in (5), we obtain

$$C_1(p) = \sum_{k=1}^K \sum_{l=1}^L \left\| \mathbf{r}_{l,k} \right\|^2 - \left| (\mathbf{A}_{l,k} \mathbf{F}_{l,k} \mathbf{s}_k)^H \mathbf{r}_{l,k} \right|^2 \quad (7)$$

Since $\|\mathbf{r}_{l,k}\|^2$ is independent of the parameters, so that minimizing (7) is equivalent to maximize following equation:

$$C_2(p) = \sum_{k=1}^K \sum_{l=1}^L \left| (\mathbf{A}_{l,k} \mathbf{F}_{l,k} \mathbf{s}_k)^H \mathbf{r}_{l,k} \right|^2 = \sum_{k=1}^K \mathbf{s}_k^H \mathbf{Q}_k \mathbf{s}_k \quad (8)$$

Where $\mathbf{Q}_k \cong \mathbf{V}_k \mathbf{V}_k^H$ and $\mathbf{V}_k \cong [\mathbf{F}_{1,k}^H \mathbf{A}_{1,k}^H \mathbf{r}_{1,k}, \dots, \mathbf{F}_{L,k}^H \mathbf{A}_{L,k}^H \mathbf{r}_{L,k}]$, as the signal waveform is unknown and then the maximization of (9) can be regarded as a maximizing quadratic form with respect to \mathbf{s}_k . The resulting of this maximization is that, \mathbf{s}_k should be selected as the eigenvector corresponding to largest eigenvalue of matrix \mathbf{Q}_k . Meanwhile, for a matrix \mathbf{X} , both $\mathbf{X}^H \mathbf{X}$ and $\mathbf{X} \mathbf{X}^H$ have same nonzero eigenvalues; therefore, the matrix \mathbf{Q}_k with $N \times N$ dimension can be replaced by $\bar{\mathbf{Q}}_k \cong \mathbf{V}_k^H \mathbf{V}_k$ with $L \times L$ dimension, which will reduce the computation cost. Then ML estimation of (8) is equivalent to the ML estimation of the following expression:

$$C_3(p) = \sum_{k=1}^K \lambda_{\max} \{ \bar{\mathbf{Q}}_k \} \quad \bar{\mathbf{Q}}_k \cong \mathbf{V}_k^H \mathbf{V}_k \quad (9)$$

Using a grid-search approach, we can find the maximum value of (9), Which is the location of the emitter.

$$\hat{p} = \arg \max_p \{ C_3(p) \} \quad (10)$$

III. ALGORITHM IMPROVEMENT

The original algorithm must transmit all the signal data to a sensor for a centralized large-scale serial matrix operation, while the remaining sensors are in idle state. If there is data loss, it will directly affect the matrix eigenvalue solution, which led to the location mismatching geographical grid, resulting in large location bias. In addition, the location processing of the original DPD algorithm is complex, it is difficult to achieve the computation load balancing, system stability is weak. In this section, we develop a method to improve the weakness as above.

A. Matrix $\bar{\mathbf{Q}}_k$ Computation

It can be seen from the above model analysis that the cost function $C_3(p)$ at any geographical grid position p is directly related to $\bar{\mathbf{Q}}_k$, and the computation of $\bar{\mathbf{Q}}_k$ involves generation, storage and computation of three N-dimensional matrices (\mathbf{V}_k , $\mathbf{A}_{l,k}$, $\mathbf{F}_{l,k}$). Assuming that the signal sampling rate is 10 MHz and the sampling time is 10 ms, the number of sampling points is $N=10^5$. Then, number of matrix elements becomes $N \times N = 10^{10}$. Therefore, the number of matrix elements increases with the number of sampling points, which consumes a lot of computing resources in practical engineering applications.

According to (9), the (i, j) th element of the matrix $\bar{\mathbf{Q}}_k$ is:

$$\begin{aligned} \bar{Q}_{k,i,j}(p) &= \mathbf{V}_{i,k}^H(p) \mathbf{V}_{j,k}(p) \\ &= \int_0^T r_{i,k}^*(t + \tau_{i,k}(p)) r_{j,k}(t + \tau_{j,k}(p)) e^{-j2\pi(\bar{f}_{j,k}(p) - \bar{f}_{i,k}(p))t} dt \end{aligned} \quad (11)$$

we can see the (i, j) th element of the matrix $\bar{\mathbf{Q}}_k$ is the CAF of signal $r_{i,k}$ and $r_{j,k}$:

$$\begin{aligned} &\int_0^T r_{i,k}^*(t + \tau_{i,k}(p)) r_{j,k}(t + \tau_{j,k}(p)) e^{-j2\pi(\bar{f}_{j,k}(p) - \bar{f}_{i,k}(p))t} dt \\ &= CAF_{i,j}(\tau_{j,k}(p) - \tau_{i,k}(p), \bar{f}_{j,k}(p) - \bar{f}_{i,k}(p)) \\ &= CAF_{k,i,j}(p) \end{aligned} \quad (12)$$

Thus, the matrix $\bar{\mathbf{Q}}_k$ can be replaced by:

$$\bar{Q}_{k,i,j} = CAF_{k,i,j} \quad (13)$$

According to the knowledge of numerical linear algebra, we know that the inner product $x^T y$ of vector-vector $x, y \in R^n$ requires n multiplication and $n-1$ addition. If the multiplication term is retained, the computation of the inner product is n . Similarly, the matrix-vector $y = Ax, A \in R^{m \times n}$ multiplication is calculated as mn ; Matrix-matrix multiplication ($C = AB, A \in R^{m \times n}, B \in R^{n \times p}$) is calculated as mnp .

It is clear that ,the computation of the matrix \mathbf{V}_k in the original DPD algorithm is about $(N^3 + N^2)L$, and the computation of the matrix \bar{Q}_k is about NL^2 . So, the coputation of original DPD algorithm is about:

$$f_Q = N^3L + N^2L + NL^2 \quad (14)$$

The computation of CAF_k is about:

$$f_{CAF} = NL^2 \quad (15)$$

The method of calculating the CAF avoids the large-scale matrix operation and greatly reduces the computational complexity.

B. Computation of Matrix Eigenvalue

The cost function in (9) shows that, each grid point must calculate eigenvalues of the matrix \bar{Q}_k , which is complicated and can not be distributed. we should try to avoid the eigenvalue computation of the matrix \bar{Q}_k directly. According to the knowledge of the compatible norm in the matrix theory^[5],we can learn that:

Theorem: If $\|\bullet\|_m : C^{n \times n} \rightarrow R$ is a compatible norm,Then, for any $A \in C^{n \times n}$ there is $|\lambda_i| \leq \|A\|_m$,Where λ_i is an eigenvalue of A .

It means that the largest eigenvalue λ_{\max} of matrix A , existing $|\lambda_{\max}| \leq \|A\|_m$.Thus, the matrix norm can be used to approximate the equivalent maximum eigenvalue. According to the matrix theory knowledge, the following matrix norm are all compatible matrix norm:

$$\begin{aligned} \|A\|_{m_1} &= \sum_{j=1}^n \sum_{i=1}^n |a_{i,j}| \dots m_1 \text{ norm} \\ \|A\|_{m_2} &= \left(\sum_{j=1}^n \sum_{i=1}^n |a_{i,j}|^2 \right)^{1/2} \dots m_2 \text{ norm} \\ \|A\|_1 &= \max_j \sum_{i=1}^n |a_{i,j}| \dots 1 \text{ norm} \end{aligned} \quad (16)$$

In this paper, we consider the use of the relatively simple norm as cost function, such as m_1 norm, m_2 norm, 1 norm. The cost function (9) can be replaced by

$$C_4(p) = \sum_{k=1}^K \|\bar{Q}_k\|_m, m = m_1, m_2, 1 \quad (17)$$

we can see the cost function $C_4(p)$ is calculated by directly superimposing the element modulus or modulo square of

\bar{Q}_k .Now, there is no need to focus all the matrix elements to a single sensor for eigenvalue computation. Each sensor can complete the computation and stacking, then transmits data to the center sensor. The center sensor can obtain the final result by accumulating each sensor's data.

In the square matrix $A \in R^{n \times n}$, the computation of the eigenvalue is n^3 .Thus, the eigenvalue computation of the matrix \bar{Q}_k is about L^3 .In the method of the norm computation, it's computation just include addition ,no multiplication,the computation can be approximated to zero.

If there is some data loss in the actual engineering application. In the original algorithm, it will directly lead to the loss of some of the matrix elements in the matrix \bar{Q}_k , which will have a great influence on the eigenvalue computation, resulting in the mismatch of the geographical grid points and lead to large localization bias. But , in the improved algorithm, due to the avoidance of eigenvalue computation, data loss will only lead to a slight decline in location accuracy, but will not cause a large positioning bias, the algorithm is more robust.

So, the algorithm using the norm can not only reduce the computational complexity, but also improve the robustness of the localization algorithm.

C. Distributed Computing Method

Since the computation of the cost function $C_4(p)$ has no matrix eigenvalue computation, only the module value of matrix elements are superimposed. The algorithm can be distributed to each sensor for distributed parallel computation, the target location result can be obtained by superposition of the center sensor. The basic steps of the distributed computing method are as follows:

a) Each sensor node in the sensor network broadcasts the signal data over the entire network and simultaneously receives signal data from other sensor nodes.

b) Each sensor node calculates and accumulates the $CAF_{k,i,j}(p)$ of each geographic grid point (p) at the same time.

We can get the result $\bar{Q}_{s,j} = \sum_{i=1}^n |CAF_{k,i,j}(p)|$

c) Each sensor node transmits the $\bar{Q}_{s,j}$ to the central sensor node for accumulation, we can get the cost function distribution map $\|\bar{Q}_k\|_m$ of a single sampling interval.

d) When there are multiple samples, repeat the above steps and accumulate, we can obtain the geographical distribution map of the cost function under multi-sensor multi-sample scenario.Then,the peak in the map is the target position coordinate.

According to the analysis of the previous sections, we can see that , in the original DPD algorithm, the computational load required for the central sensor is about:

$$f_o = N^3L + N^2L + NL^2 + L^3 \quad (18)$$

According to the analysis of(15) ,the improved algorithm's computation of each sensor node is about:

$$f_d = f_{CAF}/L = NL \quad (19)$$

Therefore, the improved fast distributed DPD algorithm has the advantages of simple computation and high stability.

IV. SIMULATION

A. Complexity of Algorithms

The improved fast distributed DPD algorithm has a significant computation reduction compared with the original DPD method. In order to have a more intuitive comparison, we have a comparison of the computation of one sensor network. The original algorithm uses the computational complexity of the central node, the improved algorithm uses the computation of a single ordinary node, The simulation results are as follows:

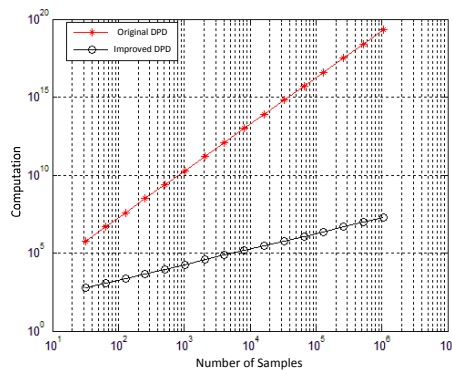


FIGURE I. COMPUTATION COMPARISON

It can be seen that ,compared with the original algorithm, the improved fast distributed DPD algorithm's computational load of the single sensor is significantly reduced, when it uses distributed parallel computing. Improvements in system operation efficiency are obvious.

B. Positioning Performanc

In order to examine the location performance of the fast distributed DPD localization algorithm. We compare the original method, the m_1 norm method, the m_2 norm method and the 1 norm method ,using Monte Carlo computer simulations (with 500 runs each time).we also provide the positioning performance of traditional two-step process under the same signal condition.

In this simulation, we assumed that a set of 9 moving sensors receive the signal from one stationary emitter. The sampling frequency is 20 kHz and the number of samples is equal to 200.

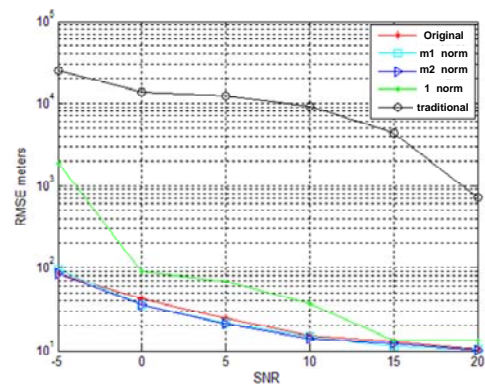


FIGURE II. LOCATION PERFORMACE COMPARATION

The simulation results show that the improved method is equivalent to the original DPD method. And location performance of all methods is superior to the traditional two-step method.

V. CONCLUSION

Compared with the traditional two-step method, the DPD method has the advantages of low signal-to-noise ratio, but requires complex large-scale matrix computation. If the original DPD method is applied directly in the sensor network, the location process is complicated, the computation load of central sensor is very high, while the remaining sensor is in the idle state, the system's localization processing efficiency is low. if there is data loss, the system localization processing is poor stability.

In this paper, a fast distributed DPD algorithm is designed to improve the performance of the original DPD algorithm, which greatly reduces the computational complexity of the algorithm and improves the stability of the positioning process. The algorithm can also realize the load balance of network computing through distributed deployment, improve the efficiency of system processing, and provide effective technical ideas for the subsequent engineering application.

REFERENCES

- [1] A.J.Weiss. Direct Geolocation of Wideband EmittersBased on Delay and Doppler[J]. IEEE Transactions onSignal Processing, 2011, 59(6) : 2513-2521.
- [2] A.J.Weiss, A.AMAR. DIRECT GEOLOCATION OF STATIONARY WIDEBAND RADIO SIGNALBASED ON TIME DELAYS AND DOPPLER SHIFTS[C].IEEE/SP 15th workshop on statistical signal processing, 2009,101-104.
- [3] Glenn D.Hartwell. Improved Geo-Spatial Resolution Using a Modified Approach to the Complex Ambiguity Function[D]. NAVAL POSTGRADUATE SCHOOL.2005.9
- [4] Andrew Moss.Temporally Adjusted Complex Ambiguity Function Mapping Algorithm for Geolocating Radio Frequency Signals[D]. NAVAL POSTGRADUATE SCHOOL.2014.12
- [5] Zhang xianda.Matrix analysis and applications[M].2013.11