Proceedings of Machine Learning Research vol 245:1–12, 2024 2024 International Conference on Machine Learning and Intelligent Computing

A Sound Source Location Method Based on Time Difference of Arrival with Improved Dung Beetle Optimizer

Chunning Song* SCN206@GXU.EDU.CN
School of Electrical Engineering, Guangxi University, Nanning, 530004, China

Jindong Zhang YAN2011JIU@126.COM
School of Electrical Engineering, Guangxi University, Nanning, 530004, China

Editors: Nianyin Zeng and Ram Bilas Pachori

Abstract

In microphone array sound source localization based on Time Difference of Arrival (TDOA), traditional methods for solving the nonlinear equations of TDOA lead to significant deviations and lower accuracy. To address this issue, this paper proposes a TDOA-based sound source localization method using an Improved Dung Beetle Optimizer (IDBO) algorithm. This method enhances the performance of the Dung Beetle Optimizer (DBO) by employing strategies such as chaotic mapping, golden sine, and adaptive t-distribution, and applies it to sound source localization. To evaluate the performance of the IDBO, it is compared with DBO, Harris Hawk Optimizer (HHO), Gray Wolf Optimizer (GWO), Bald Eagle Search (BES) algorithm, and Whale Optimization Algorithm (WOA). The results showed that in solving benchmark functions and localization models, it demonstrates faster convergence speed, higher localization accuracy, and better stability.

Keywords: Sound source localization, Microphone array, Time difference of arrival, Dung beetle optimizer algorithm, Golden sine, Adaptive t-distribution

1. Introduction

Sound source localization technology has extensive applications in various fields such as social life (Liaquat et al., 2021), industry (Du et al., 2024), and military (Feng et al., 2023). Sound source localization algorithms can be categorized based on principles into methods like Maximum Output Power Beamforming, High-Resolution Spectrum Estimation, and Time Difference of Arrival (TDOA) based localization. Maximum Output Power Beamforming requires prior information on background noise and target sound sources to achieve optimal accuracy, making real-time localization challenging to ensure in practical scenarios where such information is limited (Kocsis and Horváth, 2024). The High-Resolution Spectrum Estimation technique is primarily designed for narrowband signals. However, when applied to broadband signals such as speech, it necessitates the decomposition of the wideband signal into narrowband signals. This process may introduce deviations due to conversion losses (Wang et al., 2023). On the other hand, TDOA-based localization algorithms have a simple structure, fast computational speed, and are commonly used for estimating the three-dimensional coordinates of sound sources (Zhang et al., 2023).

When employing the TDOA method for sound source localization, the initial step involves estimating the time delay, typically accomplished through techniques such as Generalized Cross Correlation (GCC) (Guan et al., 2024) and Adaptive Time Delay Estimation algorithms based on Least Mean Square (LMS) (Sun et al., 2023). Among these methods, GCC offers the advantage of low computational complexity and fast operation speed. Subsequently, the estimated delays can be utilized for sound source localization; however, direct calculation approaches often lack robustness and accuracy. Ma et al. (2020) proposed an Improved Harris Hawk Optimizer (IHHO) algorithm to enhance TDOA localization accuracy; nevertheless, its intricate structure necessitates initial values computed using the Chan algorithm that are then assigned to IHHO's initial population. Another study by Ghorpade et al. (2021) introduced a novel Gray Wolf Optimization (GWO) localization model applied to TDOA-based three-dimensional localization but with increased computational time. This paper presents an Improved Dung Beetle Optimizer (IDBO) by incorporating strategies like chaotic mapping, golden sine function, and adaptive t-distribution to improve the performance of DBO, thereby achieving rapid and high-precision three-dimensional sound source localization. Finally, simulation experiments compare the IDBO algorithm with DBO, HHO, GWO, BES, and WOA algorithms in order to validate its performance in TDOA-based three-dimensional sound source localization.

2. Localization model

The 3D sound source near-field positioning model is taken as an example to illustrate. Let 'm' represent the number of microphone arrays to be measured, (x, y, z) denote the unknown position of the sound source, and (X_i, Y_i, Z_i) indicate the position of the i microphone. Consequently, the distance di between the i microphone and the unknown sound source can be mathematically expressed:

$$d_i = \sqrt{(X_i - x)^2 + +(Y_i - y)^2 + (Z_i - z)^2} + \varepsilon_i \quad (i = 1, 2, \cdots, m) \tag{1}$$

The time delay utilized by the sound emitted from the sound source to reach microphone i is:

$$\tau_i = \frac{d_i}{c} \tag{2}$$

The sound velocity, typically 340 m/s, denoted as c, and the measurement noise ε are considered in the following expression. The reference microphone is designated as Microphone 1. By employing the generalized transformation correlation method, we can obtain the actual measured delay difference τ_{i1} between Microphone i and the reference microphone.

$$\tau_{i1} = \tau_i - \tau_1 \tag{3}$$

$$\begin{aligned} d_{i,1} &= c\tau_{i1} = d_i - d_1 + \varepsilon_{i,1} \\ &= \sqrt{(X_i - x)^2 + +(Y_i - y)^2 + (Z_i - z)^2} \\ &-\sqrt{(X_1 - x)^2 + +(Y_1 - y)^2 + (Z_1 - z)^2} + \varepsilon_{i,1} \end{aligned} \tag{4}$$

Hypothesis:

$$\begin{aligned} \underset{(m-1)*1}{\Delta D} &= [d_{2,1}, d_{3,1}, \cdots, d_{m,1}]^T \\ \underset{(m-1)*1}{D} &= [d_1, d_2, \cdots, d_m]^T \\ \underset{(m-1)*1}{D_1} &= [d_1, d_1, \cdots, d_1]^T \\ \underset{(m-1)*1}{N} &= [\varepsilon_{2,1}, \varepsilon_{3,1}, \ldots, \varepsilon_{m,1}]^T \end{aligned}$$

According to Equation 4, a system of equations in matrix form can be obtained:

$$\Delta D = D - D_1 + N \tag{5}$$

The position of the unknown sound source (x, y) can be solved by the maximum likelihood method, and its likelihood function is:

$$\begin{gathered}\prod_{i=1}^{M-1}\left\{\frac{1}{\sqrt{2\pi\delta}}\exp\left[-\frac{(d_{i,1}-d_i+d_1)^2}{2\delta^2}\right]\right\}\\ =\left[\frac{1}{\sqrt{2\pi\delta}}\right]^{M-1}\exp\left[-\frac{(\Delta D-D+D_1)^T(\Delta D-D+D_1)}{2\delta^2}\right]\end{gathered} \tag{6}$$

where $\varepsilon_{i,1}$ is an independent Gaussian white noise random variable with a mean of 0 and a variance of δ^2. For solving an unknown sound source coordinate that maximizes the likelihood function, it can be converted to an objective function that solves the following equation:

$$f\left(x, y, z\right) = \arg\left\{min\left[\left(\Delta D - D + D_1\right)^T\left(\Delta D - D + D_1\right)\right]\right\} \tag{7}$$

Equation 7 is highly nonlinear, which is relatively complicated to solve using general mathematical methods and has low precision. The intelligent algorithm can be used and (8) can be employed as an adaptation function, and the optimal solution of the objective function can be obtained by searching for the position with the highest fitness, that is, the location of the unknown sound source.

$$\text{fitness } = \left[\left(\Delta D - D + D_1\right)^T\left(\Delta D - D + D_1\right)\right] \tag{8}$$

3. Dung beetle optimizer algorithm

3.1. Primitive dung beetle optimizer algorithm

DBO is an optimization algorithm that simulates the behavior of dung beetles in nature. It abstracts the biological behaviors of dung beetles, such as rolling, breeding, foraging, and stealing, into operational processes in the algorithm (Xue and Shen, 2023). Each behavior of dung beetles updates the position with a specific strategy to find a global optimal solution or a good local optimal solution in the problem space, which can effectively solve complex search and optimization problems in practical applications. Although the algorithm is relatively new, its application in multiple fields shows its potential and value as an optimization tool.

3.1.1. BALL-ROLLING DUNG BEETLE.

When there are no obstacles in their path, the beetles use the sun to navigate, and the intensity of the sun's light affects their position. The formula for updating the position of the dung beetle when it rolls the ball is:

$$\begin{aligned} x_i\left(t+1\right) &= x_i\left(t\right) + \alpha \times k \times x_i\left(t-1\right) + b \times \Delta x \\ \Delta x &= \left|x_i\left(t\right) - X^w\right| \end{aligned} \tag{9}$$

where $x_i\left(t\right)$ represents the position of the i dung beetle at the t iteration, $k \in (0,\ 0.02)$ is the deviation coefficient, α is either -1 or 1, $b \in (0,\ 1)$ is a random number. Δx is used to simulate the variation in solar irradiance intensity, and X^w denotes the globally worst position.

When a dung beetle encounters an obstacle, it reorients itself by dancing to get a new route. The dung beetle's dancing behavior is simulated using the tangent function, where the position is updated as follows:

$$x_i\left(t+1\right) = x_i\left(t\right) + \tan\theta \cdot \left|x_i\left(t\right) - x_i\left(t-1\right)\right| \tag{10}$$

where $\theta \in [0, \pi]$, if θ is equal to 0, $\pi/2$, or π, the beetle's position will not be updated.

3.1.2. DUNG BEETLE BREEDING.

In nature, dung balls are hidden in a safe place by dung beetles, who then choose a suitable spot to lay their eggs. The security environment is defined as follows:

$$\left\{ \begin{array}{l} Lb^* = max\left(X^* \times \left(1-R\right), Lb\right) \\ Ub^* = min\left(X^* \times \left(1-R\right), Ub\right) \end{array} \right. \tag{11}$$

where X^* represents the current best position of the population. Lb denotes the lower bound of the search space, and Ub denotes the upper bound of the search space. Lb^* and Ub^* represent the lower and upper bounds of the oviposition area, respectively. The inertia weight $R = 1 - t/T_{max}$, where T_{max} is the maximum number of iterations. The female dung beetle will lay eggs by pushing the ootheca to the determined oviposition area. Each female dung beetle lays only one egg, and the position update of the ootheca is as follows:

$$B_i\left(t+1\right) = X^* + b_1 \times \left(B_i\left(t\right) - Lb^*\right) + b_2 \times \left(B_i\left(t\right) - Ub^*\right) \tag{12}$$

where $B_i(t)$ represents the position of the i ootheca at the t iteration, b_1 and b_2 are two independent random vectors of $1 \times D$, where D is the dimensionality of the optimization problem.

3.1.3. DUNG BEETLES ON THE PROWL.

When young dung beetles grow up, they will climb out of the ground to feed, and the young dung beetles are more likely to find food in the best feeding areas. The boundaries of the best feeding areas are defined as follows:

$$\left\{ \begin{array}{l} Lb^b = max\left(X^b \times \left(1-R\right), Lb\right) \\ Ub^b = min\left(X^b \times \left(1-R\right), Ub\right) \end{array} \right. \tag{13}$$

where X^b represents the global optimal position, while Lb^b and Ub^b denote the lower and upper bounds of the best foraging area. Once the optimal foraging area is determined, the updating of the position of the firefly during foraging is shown in Equation 14.

$$X_i\left(t+1\right) = X_i\left(t\right) + C_1 \times \left(X_i\left(t\right) - Lb^b\right) + C_2 \times \left(X_i\left(t\right) - Ub^b\right) \tag{14}$$

Here, $X_i(t)$ represents the position of the i firefly at the t iteration, where C_1 is a random number following a normal distribution, and C_2 is a random number ranging from 0 to 1.

3.1.4. DUNG BEETLE STEALING.

The global optimal location is the best place to steal, where thieving dung beetles steal other dung beetles' dung balls. The thieving dung beetle's location is updated as follows:

$$x_i\left(t+1\right) = X^b + S \times g \times (\Big|x_i\left(t\right) - X^* |+| x_i\left(t\right) - X^b\Big| \tag{15}$$

Here, $x_i(t)$ represents the position of the i thief firefly at the t iteration, where g is a $1 \times D$ random vector following a normal distribution, and S is a constant.

3.2. Improved dung beetle optimizer algorithm

3.2.1. TENT CHAOTIC MAPPING.

The initial population of the original dung beetle algorithm is randomly generated, which may result in an uneven distribution of the initial population, low population diversity, and susceptibility to falling into local optimal solutions, ultimately leading to incorrect sound source location identification. To address this issue, replacing random numbers with tent chaotic mapping for initializing the population not only affects the entire algorithm process but also yields better optimization results compared to random numbers (Chen et al., 2024). The iteration formula for tent chaotic mapping is as follows:

$$x_{i+1} = \{ \begin{array}{c} x_i/\alpha, 0 < x_i \leq \alpha \\ (1 - x_i)/(1 - \alpha), \alpha < x_i \leq 1 \end{array} \tag{16}$$

Here, x_i is the value at the i iteration, and α is a constant between 0 and 1.

3.2.2. INTEGRATION OF GOLDEN SINE INTO ROLLING BALL DUNG BEETLE.

The billiard-ball behavior of the dung beetle algorithm serves as its initial stage, during which it explores the global position and determines the overall optimization capability of the algorithm. By incorporating the golden section number from the golden sine algorithm into the position update process, all values on the sine function can be traversed while continuously narrowing down the search interval. This results in a high convergence speed and accuracy [13]. The billiard-ball behavior of dung beetles is enhanced by integrating the golden sine algorithm, leading to an updated formula for position calculation.

$$\left\{ \begin{array}{c} X_i(t+1) = X_i(t) + \alpha k X_i(t-1) + b\Delta X, \delta < S_T \\ X_i(t+1) = X_i(t)\left|\sin R_1\right| - R_2 \sin R_1 \left|\theta_1 X^b - \theta_2 X_i(t)\right|,\ \delta \geq S_T \end{array} \right. \tag{17}$$

where $X_i(t)$ represents the position of the i dung beetle at the sound source point during the t iteration, and X^b is the position of the global best sound source point where the dung beetle is located. $\delta = \mathrm{rand}(1)$, $S_T \in (0.5, 1]$, when $\delta < S_T$, it indicates that the dung beetle rolls freely to search for the sound source point without obstacles, and when $\delta \geq S_T$, it indicates that the dung beetle encounters obstacles and adjusts its rolling direction to search for the sound source point. θ_1 and θ_2 are coefficients introduced with the golden ratio, with values:

$$\theta_1 = -\pi + 2\pi(1 - \tau) \tag{18}$$

$$\theta_2 = -\pi + 2\pi\tau \tag{19}$$

where $\tau = \left(\sqrt{5} - 1\right)/2$ is the golden ratio coefficient.

3.2.3. ADAPTIVE T-DISTRIBUTED PERTURBATION STRATEGY.

The probability density function of the t-distribution (Zhang and Jia, 2024) is:

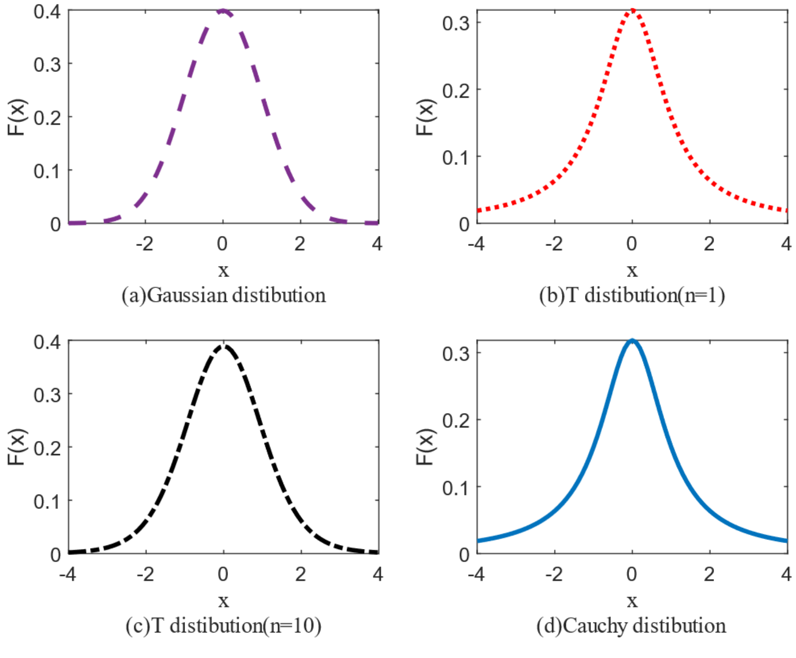
$$p_t(x) = \frac{\Gamma\left(\frac{n+1}{2}\right)}{\sqrt{n\pi} \times \Gamma\left(\frac{m}{2}\right)} \times \left(1 + \frac{x^2}{n}\right)^{-\frac{n+1}{2}}, -\infty < x < \infty \tag{20}$$

Here, n is the degree of freedom parameter, also serving as its decision parameter. $\Gamma(x)$ represents the gamma function, which is the second Euler integral. Its expression is:

$$\Gamma\left(\frac{n+1}{2}\right)=\int_0^{+\infty} x^{\frac{n+1}{2}-1} e^{-x} d x \tag{21}$$

The number of algorithm iterations is set to its degree of freedom n. When n=1, $t(n=1) \rightarrow C(0,1)$, indicating that the t distribution is equivalent to the Cauchy distribution. When $n=\infty$, $t(n=\infty) \rightarrow N(0,1)$, suggesting that the t distribution is equivalent to the Gaussian distribution. Both distributions are illustrated in Figure 1. Consequently, during the initial stage of algorithm search, the t distribution exhibits a wide range similar to that of the Cauchy distribution, facilitating large-scale exploration of sound source points. In contrast, during later stages of algorithm search, the Gaussian-like t distribution demonstrates a more concentrated range which enhances local development capabilities and improves accuracy in solving sound source positions. Additionally, incorporating an adaptive t-distribution disturbance strategy further enhances the Dung Beetle algorithm's ability to escape local optimal sound source points. The position variation formula is as follows:

$$X_{\text{new }}=X_i+X_i \cdot t(\text{ iter }) \tag{22}$$


(a)Gaussian distibution

(b)T distibution(n=1)

(c)T distibution(n=10)

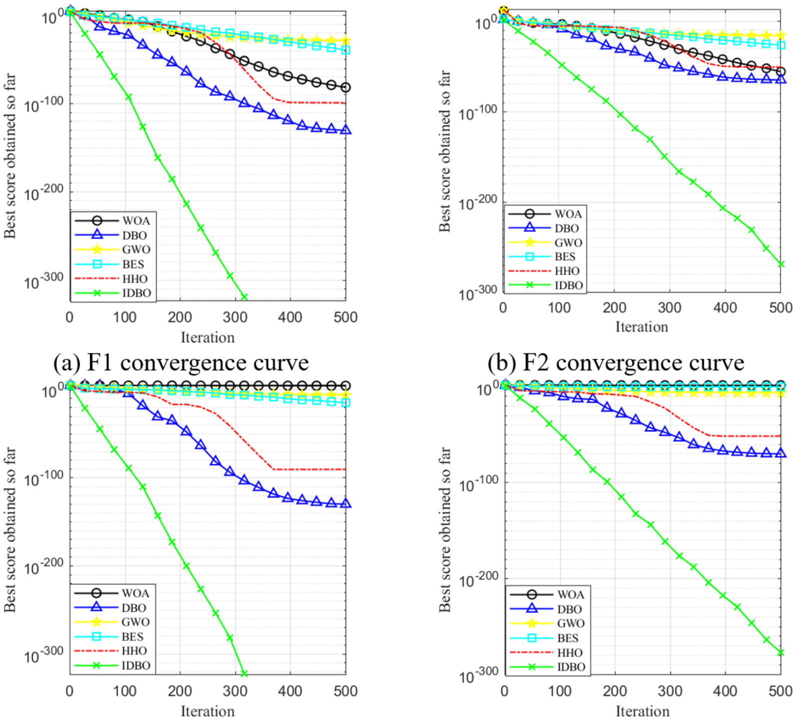
(d)Cauchy distibution

Figure 1: T distribution, Gaussian distribution, and Cauchy distribution.

In the equation, X_i represents the position of the i little firefly at the source, $X_{\text{new }}$ denotes the new position after the t-distribution mutation, and $t(\text{iter })$ represents the t-distribution operator, with the degrees of freedom parameter being the iteration count, iter. To retain certain information about the original firefly's position at the source, a disturbance probability P is set to determine whether the firefly at the current source position undergoes t-distribution mutation:

$$\begin{cases} X_{new} = X_i + X_i \cdot t\,(\,iter\,)\,, R < P \\ X_{new} = X_i,\ R \geq P \end{cases} \tag{23}$$

In the equation, $R = rand(0,1)$ is a random number, and the disturbance probability P is set to 0.5.



(a) F1 convergence curve

(b) F2 convergence curve

(c) F3 convergence curve

(d) F4 convergence curve

Figure 2: Test curve of the benchmark function.

4. Simulation experiment and analysis

4.1. Performance test of IDBO algorithm

The performance of the IDBO algorithm and the effectiveness of the proposed improvement strategy are tested by comparing it with several similar meta-heuristic intelligence algorithms on a benchmark function. These algorithms include the DBO, GWO, BES, WOA, and HHO. The parameters of the benchmark function are presented in Table 1.

The initial population size of each algorithm is set to 30 in order to ensure evaluation accuracy, and the test is conducted over 500 iterations. The convergence curves are compared and presented in Figure 2. To mitigate potential contingencies caused by random parameters or specific operations within the algorithm, a total of 30 independent runs were performed for the comparative experiment. The minimum value among these results was selected, and the average value along with the standard deviation across all 30 runs were calculated as shown in Table 2.

Table 1: Test function parameters.

Function	**Function name**	**Dimensionality**	**Value range**	**Optimal value**
F_1	Sphere	30	[-100,100]	0
F_2	Schwefel2.22	30	[-10,10]	0
F_3	Schwefel1.22	30	[-100,100]	0
F_4	Schwefel2.21	30	[-100,100]	0

Table 2: Test function results of each algorithm.

Test Function	Indicators	IDBO	DBO	GWO	BES	WOA	HHO
F1	min	0	1.7251×10^{-167}	7.9080×10^{-35}	2.7855×10^{-48}	1.4779×10^{-94}	4.6468×10^{-116}
F1	std	0	9.4261×10^{-107}	5.7417×10^{-33}	5.0554×10^{-38}	6.6201×10^{-82}	4.0945×10^{-99}
F1	avg	0	1.7210×10^{-107}	3.4412×10^{-33}	9.9318×10^{-39}	1.2739×10^{-82}	7.4890×10^{-100}
F2	min	5.1840×10^{-247}	9.7976×10^{-80}	1.1485×10^{-20}	1.7275×10^{-30}	7.5619×10^{-63}	8.1781×10^{-62}
F2	std	0	6.0111×10^{-60}	5.2157×10^{-20}	5.6865×10^{-27}	1.2122×10^{-52}	4.4380×10^{-52}
F2	avg	5.9890×10^{-282}	1.1179×10^{-60}	7.2687×10^{-20}	3.3647×10^{-27}	2.7592×10^{-53}	1.1354×10^{-52}
F3	min	0	2.7345×10^{-148}	9.7452×10^{-11}	1.1168×10^{-19}	5.2135×10^{3}	7.1061×10^{-111}
F3	std	0	5.9604×10^{-24}	1.7136×10^{-07}	9.4679×10^{-06}	9.9899×10^{3}	3.8903×10^{-78}
F3	avg	0	1.0882×10^{-24}	4.3831×10^{-08}	2.3530×10^{-06}	2.9211×10^{4}	7.1082×10^{-79}
F4	min	5.4299×10^{-240}	3.2329×10^{-89}	1.5042×10^{-09}	6.3886×10^{-14}	1.1375	9.1743×10^{-61}
F4	std	0	2.3628×10^{-51}	1.4825×10^{-08}	0.2246	29.2606	9.5747×10^{-51}
F4	avg	8.3244×10^{-288}	4.3196×10^{-52}	1.8501×10^{-08}	0.1428	31.4187	1.9370×10^{-51}

The convergence speed and solution accuracy of IDBO are superior to other intelligent algorithms on the benchmark test function, as depicted in Figure 1. Additionally, Table 1 demonstrates that the IDBO algorithm exhibits the lowest minimum value, average value, and standard deviation throughout the iterative process. In functions F1 and F3, all three evaluation indexes of IDBO yield a perfect score of 0, indicating consistent attainment of accurate solutions over 30 repeated operations. Moreover, in functions F2 and F4, the three evaluation indexes of IDBO outperform those of other algorithms significantly with a noticeable disparity in magnitudes.

After conducting the aforementioned analysis, IDBO demonstrates exceptional performance in the given benchmark functions. The substantial disparities in minimum, average, and standard deviation compared to other algorithms highlight its robust capability in locating global optimal solutions and ensuring high stability and reliability throughout the iterative process. Simultaneously, the proposed enhancement strategy's feasibility and superiority have been validated.

4.2. Localization simulation analysis

The experimental scene for sound source location simulation was set up as a three-dimensional space measuring 300 cm in length, 300 cm in width, and 300 cm in height. The arrangement of microphones is illustrated in Figure 3. To assess the accuracy of the algorithm's positioning, the mean square error was employed to quantify the deviation between the estimated sound source point position and its actual location.

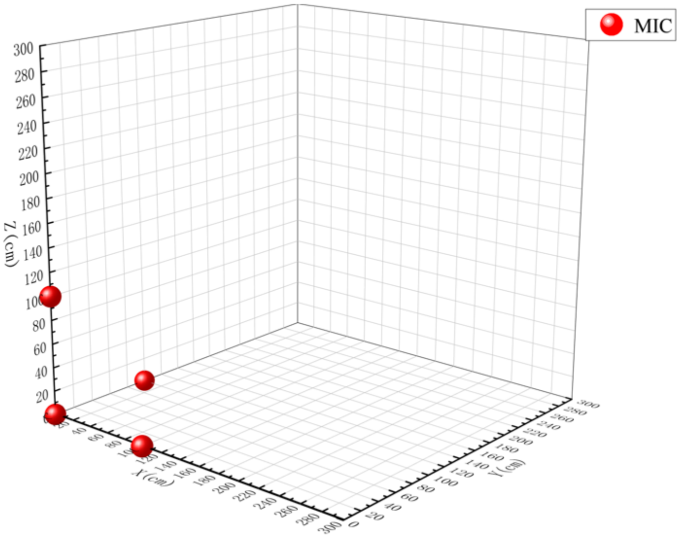


Figure 3: Microphone array distribution.

$$\text{RMSE } = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - x_{pi}\right)^2 + \left(y_i - y_{pi}\right)^2 + \left(z_i - z_{pi}\right)^2} \tag{24}$$

where n is the number of test sound sources, (x_i, y_i, z_i) represents the estimated position of the sound source, and (x_{pi}, y_{pi}, z_{pi}) represents the true position of the sound source. The population size

M for each algorithm is uniformly set to 30, and Gaussian white noise with standard deviations of $\sigma = 1,\ 2,\ 3,\ 4,\ 5$ cm is added separately. The comparison of mean square error for each algorithm under different noise intensities is shown in Figure 4.

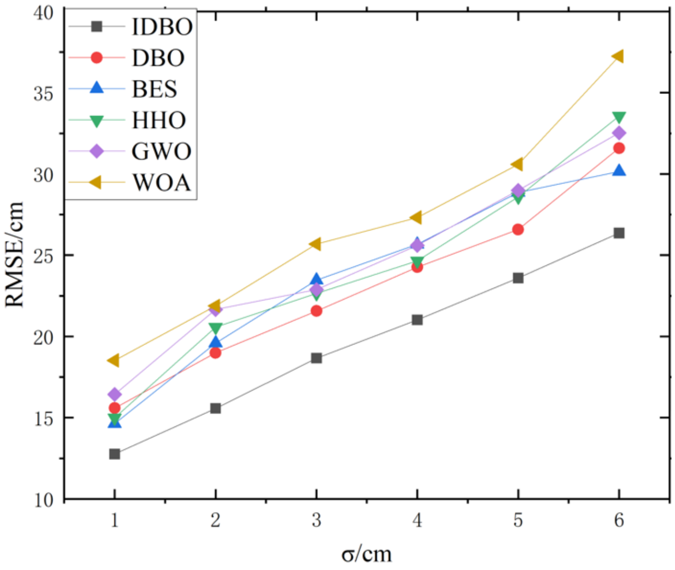


Figure 4: Comparison of localization accuracy of different intensity noise.

The comparison results reveal that as the noise gradually increases, the error of each algorithm also increases. Among them, the IDBO algorithm consistently exhibits the lowest RMSE throughout the entire range of noise levels, indicating its superior positioning accuracy under identical conditions. Furthermore, the curve slope of the IDBO algorithm is relatively small, implying that even with an increase in noise level, its performance remains stable and possesses excellent anti-noise capabilities. Figure 5 illustrates the error histogram for each algorithm under varying numbers of microphones when subjected to a noise condition $\sigma = 3\ cm$.

The comparison results revealed that the standard deviation of each algorithm decreases as the number of microphones increases, indicating that utilizing redundant measurement values can enhance the positioning accuracy for each algorithm. Among all algorithms, IDBO consistently exhibits the lowest error across different numbers of microphones. However, as the number of microphones continues to increase, the improvement in IDBO accuracy gradually diminishes. In practical applications, it is crucial to select an appropriate number of microphones based on specific requirements.

5. Conclusion

The DBO algorithm in this study is enhanced through the incorporation of chaos mapping, golden sine, adaptive t-distribution, and other strategies. The performance of the IDBO algorithm is evaluated using benchmark functions. Test results demonstrate that the IDBO algorithm surpasses both the original DBO algorithm and other similar algorithms in terms of convergence speed and accuracy. Subsequently, IDBO is applied to solve TDOA equations for sound source localization, with

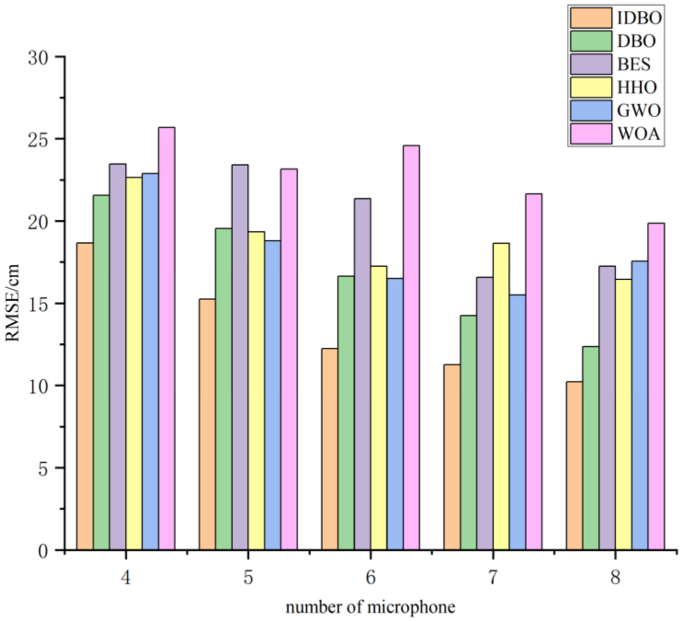


Figure 5: Error comparison of different microphones.

an analysis conducted on its performance under varying levels of noise and different numbers of microphones. The findings reveal that IDBO exhibits superior stability and location accuracy.

References

Q. Y. Chen, J. Shao, C. Q. Wang, and et al. Improved african vulture optimizer algorithm based on double dynamic adjustment. *Foreign electronic measurement technology*, 43(01):20–29, 2024. doi: 10.19652/j.cnki.femt.2305410.

Z.Y. Du, Z.Y. Hao, and P.F. Zhao. Study on ultrasonic guided wave propagation characteristics of gas-insulated transmission lines suitable for sound source location. *Electrotechnical newspaper*, 39(03):852–862, 2024. doi: 10.19595/j.cnki.1000-6753.tces.221902.

B. Feng, J. F. Zhao, and Q. Guo. Adaptive generalized cross correlation for burst location of acoustic arrays. *Journal of Detection and Control*, 45(05):60–65, 2023.

Sheetal N. Ghorpade, Marco Zennaro, and Bharat S. Chaudhari. Gwo model for optimal localization of iot-enabled sensor nodes in smart parking systems. *IEEE Transactions on Intelligent Transportation Systems*, 22(2):1217–1224, 2021. doi: 10.1109/TITS.2020.2964604.

Y. Guan, M. Dong, Y. J. Xi, and et al. Multi-source partial discharge localization method based on frequency domain threshold processing generalized cross-correlation and spatial screening. *Power grid technology*, pages 1–13, 2024. [J/OL].

Bálint Kocsis and Csaba Horváth. Further development of rotating beamforming techniques using asynchronous measurements. *Journal of Theoretical and Computational Acoustics*, 32(01): 2340006, 2024. doi: 10.1142/S2591728523400066.

Muhammad Usman Liaquat, Hafiz Suliman Munawar, Amna Rahman, Zakria Qadir, Abbas Z. Kouzani, and M. A. Parvez Mahmud. Localization of sound sources: A systematic review. *Energies*, 14(13), 2021. doi: 10.3390/en14133910.

Y. M. Ma, Z. D. Shi, K. Zhao, and et al. Tdoa localization based on improved harris eagle optimization algorithm. *Computer Engineering*, 46(12):179–184, 2020. doi: 10.19678/j.issn.1000-3428.0056965.

D. J. Sun, T. F. Huang, Q. Y. Peng, and et al. Gradient projection least-square multi-path channel high-resolution estimation method in real domain. *Acta Acoustitica*, 48(04):858–871, 2023. doi: 10.15949/j.cnki.0371-0025.2023.04.025.

X.H. Wang, Y.J. Hou, and X. Jin. Estimation method of poda orientation based on music algorithm reconstructed by toeplitz covariance matrix. *Acoustic technique*, 42(06):825–831, 2023. ISSN 1000-3630. doi: 10.16300/j.cnki.1000-3630.2023.06.017.

Jiankai Xue and Bo Shen. Dung beetle optimizer: a new meta-heuristic algorithm for global optimization. *The Journal of Supercomputing*, 79(7):7305–7336, may 2023.

D.G. Zhang, Z.F. Zhou, Y. Zhang, and L.D Wang. Tdoa sound source location method based on particle swarm optimization. *Electronic technology*, 36(09):21–28, 2023. doi: 10.16180/j.cnki.issn1007-7820.2023.09.004.

H. Y. Zhang and R. L. Jia. Chaotic adaptive vulture search algorithm based on t-distribution and spiral slime mold search. *Small microcomputer system*, pages 1–12, 2024. [J/OL].