

基于改进麻雀搜索算法模型的阵列优化

李震宇, 安浩, 刘励韬, 孙坤

上海理工大学机械工程学院, 上海

收稿日期: 2024年12月23日; 录用日期: 2025年1月16日; 发布日期: 2025年1月24日

摘要

声学相机常用于检测设备异响, 但现有技术面临传感器数量过多、定位精度不足等挑战。为应对这一问题, 急需提出科学合理的声学麦克风阵列排布方案。本文提出基于麻雀捕食仿生学的麻雀搜索算法, 以优化麦克风阵列中的阵元数量。该算法模拟麻雀在捕食过程中寻找最优资源分布的策略, 能够有效地在广阔搜索空间中选择最优阵列配置。优化后的阵列在减少传感器数量的同时, 依然能够提供高精度的声源定位, 从而为声学检测系统提供了更加高效且经济的解决方案。通过仿真计算, 优化后阵列的阵元数量较原始阵列减少了75.1%。然而, 传统麻雀搜索算法在计算过程中存在迭代方式较慢、时间过长等问题。为此, 本文提出采用软阈值迭代方法替代传统的迭代方式, 通过此改进将算法的运行速度提升了3419.3%以上, 并且阵列中阵元的位置依然保持优良的定位效果, 相较于原始算法改进后的矩阵客观评价指标, 方向图宽度提升了24.3%, 信噪比提高了7.8%, 波束尖锐度提升了7.5%。

关键词

麦克风阵列优化, 麻雀搜索算法, 软阈值迭代方法

Array Optimization Based on Improved Sparrow Search Algorithm Model

Zhenyu Li, Hao An, Litao Liu, Kun Sun

School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Dec. 23rd, 2024; accepted: Jan. 16th, 2025; published: Jan. 24th, 2025

Abstract

Acoustic cameras are often used to detect abnormal and abnormal noises in equipment, but the existing technology faces challenges such as too many sensors and insufficient positioning accuracy. In order to solve this problem, it is urgent to propose a scientific and reasonable arrangement scheme of acoustic microphone array. In this paper, a sparrow search algorithm based on sparrow

predation bionics is proposed to optimize the number of array elements in the microphone array. The algorithm simulates the strategy of finding the optimal resource distribution in the process of predation, and can effectively select the optimal array configuration in the vast search space. The optimized array provides a more efficient and cost-effective solution for acoustic inspection systems by reducing the number of sensors while still providing high-precision sound source localization. Through simulation calculation, the number of elements in the optimized array is reduced by 75.1% compared with the original array. However, the traditional sparrow search algorithm has some problems in the calculation process, such as slow iteration mode and long time. Compared with the improved objective evaluation index of the original algorithm, the pattern width is increased by 24.3%, the signal-to-noise ratio is increased by 7.8%, and the beam sharpness is increased by 7.5%.

Keywords

Microphone Matrix Optimization, Sparrow Search Algorithm, Soft-Threshold Iteration Method

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

麻雀搜索算法(SSA)作为一种基于自然界麻雀觅食行为的优化算法,已经被广泛应用于多种优化问题中。该算法模拟麻雀在寻找食物过程中,通过探索和利用相结合的策略来寻找最优解。在麻雀搜索算法中,麻雀分为探索者和利用者两类,探索者负责寻找新的食物源,利用者则通过对已知食物源的优化来提高效率。由于其灵活的搜索机制,麻雀搜索算法在处理复杂的优化问题时表现出较强的全局搜索能力。对于麦克风阵列优化问题,麻雀搜索算法能够有效地减少传感器数量,同时保证阵列的声源定位精度,提供一种高效且经济的解决方案。

然而,传统的麻雀搜索算法在计算过程中存在着迭代速度慢、计算时间长等问题。为此,本文提出了一种基于软阈值迭代方法的改进麻雀搜索算法。软阈值迭代方法通过在每次迭代过程中引入软阈值函数,对麻雀的位置进行适当的约束和调整,从而加速算法的收敛速度并避免过度的搜索波动。通过这种改进,麻雀搜索算法能够在保持全局搜索能力的同时,显著提高计算效率,特别适用于大规模优化问题。通过仿真验证,改进后的麻雀搜索算法在优化麦克风阵列配置时,运行速度得到了提高,同时仍能保持较高的定位精度,为声学检测系统的优化提供了更加高效的方案。

2. 传统模型的建立与求解

本文用到的符号及其含义如表 1 所示。

2.1. 传统麻雀搜索算法模型

传统的麻雀搜索算法(SSA)由薛建凯[1]于 2020 年提出,是一种全新的优化方法,其设计灵感来源于麻雀在自然界中的觅食和反捕食行为。该算法通过模拟这些自然行为并将其转化为数学模型,从而构建出一个能够在搜索空间内同时进行全局探索和局部开发的优化方法。SSA 的关键在于通过麻雀觅食与反捕食的行为规则来指导优化过程,这些规则帮助算法进行有效的优化。该算法特别适合解决复杂的优化问题,因为它不仅保持了全局搜索的能力,还能够进行精确的局部搜索,因此能够在特定条件下找到最优解。

Table 1. Symbol description
表 1. 符号说明

符号	含义
X	麻雀种群的位置信息
F_X	麻雀种群的适应度矩阵
$iter_{max}$	最大迭代次数
R_2	搜索预警值
ST	搜索安全值
Q	因子强度
X_i	麻雀的位置参数
K	麻雀移动方向与步长
BW	方向图宽度
SNR	信噪比
BS	波束尖锐度

2.1.1. 麻雀搜索算法的仿生构思

麻雀群体展现出高度灵活的行为策略，以适应不断变化的环境与资源条件[2][3]。首先，群体内的角色分工尤为明显且动态：部分个体作为探索者，主动寻找食物并引领群体前往资源丰富的区域；而其他个体则作为加入者，依赖探索者的发现进行觅食。这种角色的灵活转换大大提高了群体的觅食效率，充分展现了麻雀面对环境变化时的适应能力。

在觅食过程中，麻雀群体内的个体互动表现出竞争与合作并存的特点。攻击者会与食物摄入量较高的同伴争夺资源，以提高自身的能量摄取效率。然而，这种竞争并未破坏群体稳定，反而与合作行为共同维持了内部的动态平衡，使资源得以更有效地分配和利用。

此外，麻雀的警觉性是其生存的关键。个体对周围环境保持高度的敏锐和好奇心，能够迅速察觉潜在捕食者的存在。当发现危险时，麻雀会通过叫声发出警告，整个群体立即作出反应，迅速飞向安全区域。这种快速的集体响应机制显著提高了个体的生存率，也增强了整个群体的适应能力。

通过角色分工、竞争与合作的平衡及高度的警觉性，麻雀群体展现了高效的觅食策略与强大的环境适应力。

2.1.2. 麻雀搜索算法的模型构建

通常而言，优化就是在众多的方案中为一个特定的问题找到最佳的解决方案，或者对于一些数值求解问题得到一个全局的最优值。一个不失一般性的最优化问题的数学形式可以表达为如下：

$$\begin{aligned} \min & f(x) \\ \text{s.t. } & g_i(x) \leq 0, \quad i=1,2,\dots,m \\ & h_j(x) \leq 0, \quad j=1,2,\dots,l \end{aligned} \tag{1}$$

其中， $x = \{x_1, x_2, \dots, x_n\}^T \in R^n$ 表示 n 维决策向量，也就是问题的搜索空间， $x_i \in [x_i^L, x_i^U]$ ， x_i^L 和 x_i^U ，分别表示为每一维决策变量的上界和下界， $f(x)$ 是目标函数， $g_i(x) \leq 0 (i=1,2,\dots,m)$ 是 m 个不等式约束，

$h_j(x)=0(j=1,2,\cdots,l)$ 是 l 个等式约束。

大多数传统的优化技术依赖于梯度信息来寻找解决方案,这通常需要目标函数是连续的并且具有可微分的特性。但是,随着实际问题变得越来越复杂和庞大,它们往往包含大量的非线性元素,并且不具备明显的连续性或可微性。在应对这些挑战时,传统的优化手段常常表现不佳,甚至在某些情况下根本无法在给定条件下找到解决方案。

根据 2.1.1 的生物特性可以构建出算法的数学模型,为了使算法更加简洁明了,对麻雀的生物行为做出以下理想化规范,主要规则如下:

- 1) 探索者凭借高能量储备和适应度值(Fitness Value)领导觅食,指引加入者寻找食物,其适应度值决定了能量储备和领导地位。
- 2) 麻雀遇捕食者时发出警报,警报值高时探索者带领加入者转移到安全区域,保护群体免受威胁。
- 3) 探索者与加入者身份可动态转换,但比例固定,保持种群平衡,一只麻雀成为探索者必有另一只转变为加入者。
- 4) 加入者因能量低而觅食位置差,饥饿时可能离群觅食,以获取更多能量,改善自身状况,这个情况下,加入者也就变为了探索者。
- 5) 加入者跟随探索者觅食,争夺资源以提高捕食率,通过监控探索者行为寻找食物机会。
- 6) 面对危险,边缘麻雀迅速移向安全区,中心麻雀则随机移动靠近同伴,减少危险区域,增强群体安全。

在仿真实验中,我们需要将麻雀虚拟化进行食物的寻找,则由 n 只麻雀组成的种群可表示为如下形式:

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,d} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,d} \end{bmatrix} \quad (2)$$

其中, d 表示待优化问题变量的维数, n 则是麻雀的数量。那么,所有麻雀的适应度值可以表示为如下形式:

$$F_X = \begin{bmatrix} f([x_{1,1} & x_{1,2} & \cdots & x_{1,d}]) \\ f([x_{2,1} & x_{2,2} & \cdots & x_{2,d}]) \\ \vdots \\ f([x_{n,1} & x_{n,2} & \cdots & x_{n,d}]) \end{bmatrix} \quad (3)$$

在 SSA 中,那些适应度值较高的探索者在觅食时会首先获得资源。同时,由于这些探索者肩负着为整个麻雀群体寻找食物源并指引觅食方向的职责,它们因此被赋予了比加入者更广阔的搜索范围。根据规则(1)和规则(2),在每次迭代的过程中,探索者的位置更新描述如下:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \cdot \exp\left(-\frac{i}{\alpha \cdot iter_{max}}\right) & \text{if } R_2 < ST \\ X_{i,j}^t + Q \cdot L & \text{if } R_2 \geq ST \end{cases} \quad (4)$$

其中, t 表示当前迭代数, j 表示待优化问题的维度,取值为 $1,2,3,\cdots,d$ 。 $iter_{max}$ 表示种群更新的最大迭代次数,是计算过程中设定的常数。 X_{ij} 代表的是第 i 个麻雀在第 j 维中的位置信息参数。 α 是 $(0, 1]$ 范围内

的一个随机数。 R_2 麻雀在搜索范围内的预警值,取值范围为[0.1]。 ST 安全值,取值范围为[0.5, 1]。同时,种群在搜索过程中的预警值安全值也是可以人为设置的。 Q 是一个服从正态分布的随机数。 L 表示一个 $1 * d$ 的矩阵,并且该矩阵内每个元素全部为 1。

在探索者与加入者位置进行更新时,存在两种情况,一种是无捕食者时($R < ST$),探索者可以在较大的范围内搜索食物,位置更新通过指数衰减因子来调整,表现为较为广泛的探索,另一种是有捕食者时($R \geq ST$),当探索者发现捕食者时,所有麻雀都需要迅速飞到安全区域,探索者的位置将根据随机扰动和常数 L 进行调整,保证群体整体安全。

并且加入者的行为依据探索者的行动来做出调整,它们会观察探索者的表现,并根据探索者发现更好食物源时,立即放弃原位,争夺新发现的食物。如果加入者在竞争中失败,则按照加入者转变为探索者的规则进行位置更新,同时会产生同样数量的探索者变为加入者。如果加入者发现探索者的食物源更优质,它们会迅速改变自己的位置,如果加入者无法获取探索者找到的食物,它们会根据规则(5)调整位置,通常是远离当前的位置,以寻找新的食源。加入者的位置更新描述如下:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_w - X_{i,j}^t}{i^2}\right) & \text{if } i > n/2 \\ X_p^{t+1} + |X_{i,j}^t - X_p^{t+1}| \cdot A^+ \cdot L & \text{otherwise} \end{cases} \quad (5)$$

其中, X_p 表示当前探索者所处的最优位置,代表探索者找到的最佳食物源位置。 X_w 表示当前全局最差的位置,也即搜索范围的边缘位置,通常指的是适应度最差的解。 A 表示一个 $1 * d$ 的矩阵,其中每个元素随机赋值为 1 或 -1,且赋值矩阵的更新规则为 $A^+ = A^T (AA^T)^{-1}$ 。当 $i > n/2$ 时,这当第 i 个加入者的适应度值较低时,表示它没有找到食物,处于极度饥饿状态,因此需要移动到其他位置寻求食物,以增加能量。

在仿真实验中,假设那些意识到危险的麻雀只占种群总数的少部分。所有麻雀的初始位置是在种群的安全搜索范围内随机生成的。这种假设意味着大部分麻雀初始时处于相对安全的区域,而只有少数麻雀会在过程中检测到潜在的危险,从而采取相应的反应。通过将规则(6)数学建模可以表示为如下形式:

$$X_{i,j}^{t+1} = \begin{cases} X_b^t + \beta \cdot |X_{i,j}^t - X_b^t| & \text{if } f_i > f_g \\ X_{i,j}^t + K \left(\frac{|X_{i,j}^t - X_w^t|}{(f_i - f_w) + \varepsilon} \right) & \text{if } f_i = f_g \end{cases} \quad (6)$$

其中, X_b 是当前的全局最优位置。 $K \in [-1, 1]$ 是一个随机数, f_i 则是种群中第 i 麻雀个体也就是当前计算麻雀个体的适应度值。 f_g 和 f_w 分别是当前全局最佳和最差的适应度值。 ε 是最小的常数,以避免分母出现零。

为简单起见,当 $f_i > f_g$ 表示第 i 个麻雀的适应度超出安全范围,即此时的麻雀处于种群搜索范围的边缘,遭受捕食者的攻击的概率大。 X_b 表示当前种群中适应度最优的麻雀个体的位置信息。在算法中, X_b 用来指导其它麻雀的位置调整。它代表了当前解空间中最好的解,也是种群趋向的目标。 $f_i = f_g$ 时,表明当前麻雀处于全局最优位置附近。在算法中,这一状态通常触发特殊的行为逻辑,比如防止算法陷入局部最优。在位置更新过程中,步长控制参数 K 和 β 对麻雀的运动范围和方向起着决定性作用。 K 控制麻雀与最差位置的距离,而 β 则通过随机扰动控制探索的步幅。通过这些参数的调节,麻雀能够根据当前环境的不同需求调整行为策略。

以上述模型构建流程可以得出传统的麻雀搜索算法的数学流程如图 1 所示:

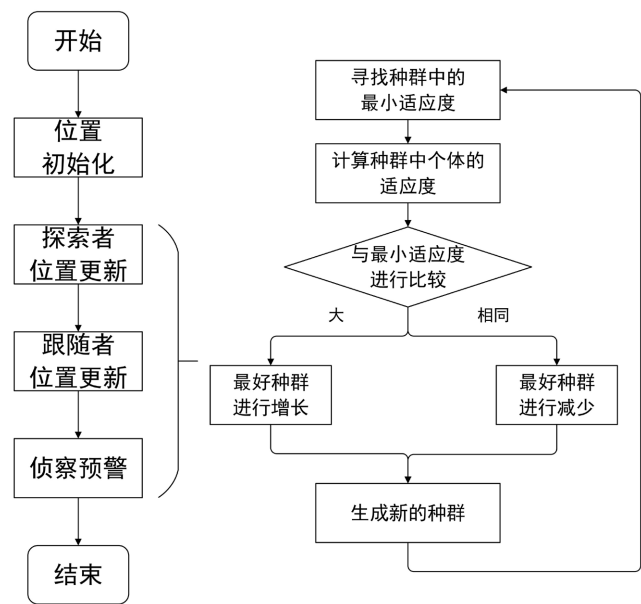


Figure 1. Flowchart of the sparrow search algorithm
图 1. 麻雀搜索算法流程图

2.2. 阵列仿真效果

对于阵列优化问题，即将阵列中的阵元进行路径规划，使其每个阵元都能到达最优位置，从而做到阵列在数量以及构型上的优化，本文对原始麦克风矩阵进行优化，最终得到数量以及阵形的最优化结果，原始麦克风阵列参数如表 2 所示。

Table 2. Original microphone matrix parameters
表 2. 原始麦克风矩阵参数

参数名称	数量
麦克风阵元个数	385 个
阵列臂数	10 臂
矩阵阵形	同心圆
麦克风阵元相邻最小距离	0.06869 m

2.2.1. 原始矩阵的仿真结果

原始矩阵示意图以及该矩阵通过波束形成对空间中3个点声源进行接收仿真示意图如图2，图3所示，点声源在空间中声速均为340 m/s，频率f均在3~4 kHz，坐标分别为(0.3, 0.1) (0.2, 0.4) (0.1, -0.3)。

从图中可以观测到三个点声源的位置被清晰的标识出来，并且在声源的主瓣位置与旁瓣位置的区分相对明显，这表明主旁瓣值的差距较大，从而证明该矩阵的效果比较好，而通过客观矩阵评估值——方向图宽度(Beamwidth, 简称BW)，信噪比(Signal-to-noise ratio, 简称SNR)，波束尖锐度(Beam Sharpness, 简称BS)三个指标进行分析，指标计算公式如下：

1) 方向图宽度BW

方向图宽度通常指天线辐射方向图中的半功率点(也叫3 dB点)之间的角度距离。方向图宽度反映了天线的辐射集中度，通常用于描述天线的辐射图形的开阔程度，方向图宽度越小，意味着阵列的指向性越强，表示阵列性能越好。计算公式为

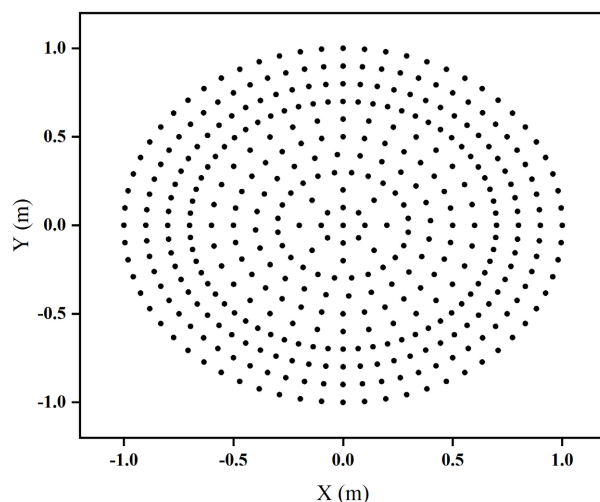


Figure 2. The original microphone matrix

图 2. 原始麦克风矩阵

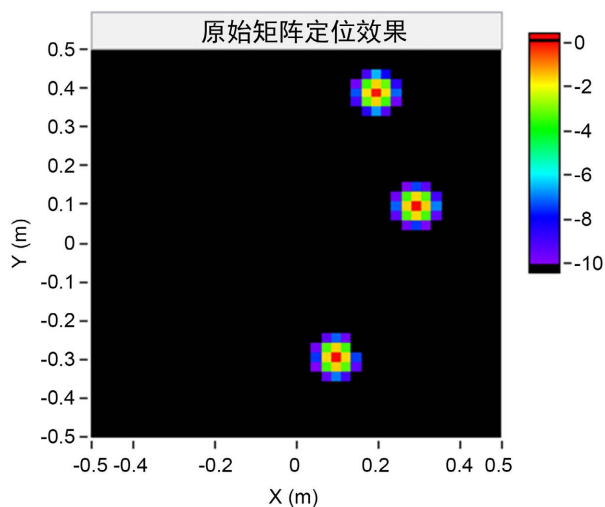


Figure 3. The localization effect of the sound source

图 3. 声源定位效果

$$\theta_{3dB} = \frac{70\lambda}{D} \quad (7)$$

其中 θ_{3dB} 表示3 dB波束宽度，单位为度， λ 是信号波长， D 表示天线的直径，又称传感器有效孔径。

2) 信噪比SNR

信噪比是衡量信号强度与噪声强度之比的度量，常用于评估通信系统的质量，信噪比越高，表示阵列的接收性能越好。信噪比的常用公式为：

$$\text{SNR} = \frac{P_{\text{signal}}}{P_{\text{noise}}} \quad (8)$$

其中 P_{signal} 是信号的功率， P_{noise} 是噪声的功率。

3) 波束尖锐度BS

波束尖锐度反映了天线的指向性，即天线将辐射能量集中到某个方向的程度。波束尖锐度可以用天

线的方向性或增益来表示，阵列定位声源时产生越小的主瓣宽度表示波束越尖锐。天线的方向性和增益之间有如下关系：

$$D = \frac{4\pi \cdot A_e}{\lambda^2}$$

(9)

其中， D 是天线的方向性(无单位)，也可以理解为波束的尖锐度， A_e 是天线的有效孔径， λ 是信号的波长。

通过以上计算公式，可以得出原始麦克风矩阵在对声源进行定位时的客观指标，结果为 $BW = 929.00$ 度， $BS = 18300.00$ 度， $SNR = 36.57$ dB。

2.2.2. 麻雀搜索算法优化后矩阵结果

由客观数据上可以清晰看出原始麦克风矩阵的性能优异，但原始矩阵的缺点在于阵元数量过多，存在冗余麦克风阵元，这些麦克风阵元在声源定位过程中发挥的作用并不明显，从而产生经济成本以及空间成本的增加。

根据麻雀搜索算法对原始麦克风矩阵中阵元进行位置优化，可以得出图4优化后麦克风矩阵示意图。

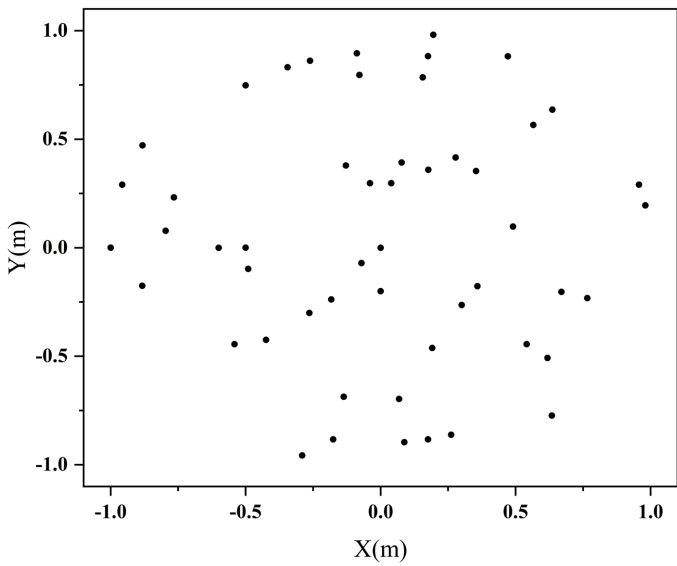


Figure 4. Optimized rear microphone matrix
图 4. 优化后麦克风矩阵

此时麦克风矩阵参数如表3所示。

Table 3. Original microphone matrix parameters
表 3. 优化麦克风矩阵参数

参数名称	数量
麦克风阵元个数	53 个
阵列臂数	10 臂
矩阵阵形	同心圆
麦克风阵元相邻最小距离	0.06869 m
运行时间	257.2216 s

同样使优化后的麦克风矩阵对上述三个点声源进行定位仿真，可以得出如图5所示定位效果示意图。

由上示意图可以看出，虽然麻雀搜索算法将原始矩阵中的多数麦克风去除，但对于声源的定位效果仍然优秀，且旁瓣的产生对于声源主瓣的距离较远，对其影响较小；从客观数值上分析，优化后矩阵的BW = 1107.00度，BS = 17307.00度，SNR = 17.99 dB，从客观数据对比上可以看出，虽然三个指标参数绝对值均有不同程度的波动，但幅度并不大，均在可接受范围内，所以经过数据对比可以判定经过麻雀搜索算法后对原始矩阵中存在的冗余阵元进行了去除，保证了麦克风矩阵性能的同时，节约了经济成本与空间成本。

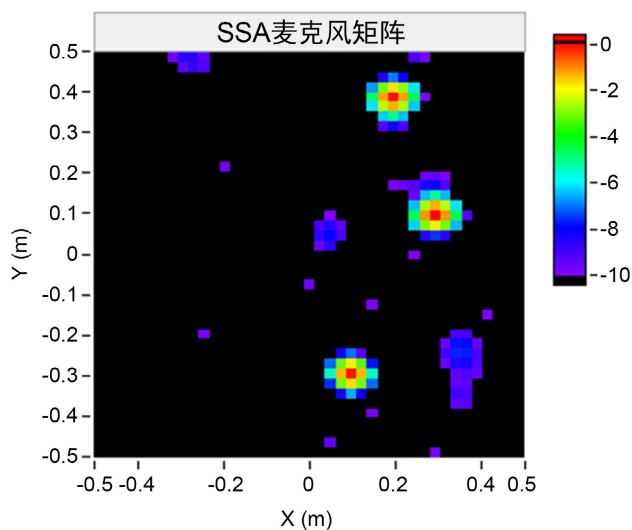


Figure 5. The localization effect of the sound source
图 5. 声源定位效果

3. 改进模型的建立与求解

3.1. 优化麻雀搜索算法模型

通过多次仿真实验分析，传统的麻雀搜索算法在计算过程中存在着迭代速度慢、计算时间长等问题。而软阈值迭代方法通过在每次迭代过程中引入软阈值函数，对麻雀的位置进行适当的约束和调整，从而加速算法的收敛速度并避免过度的搜索波动。通过这种改进，麻雀搜索算法能够在保持全局搜索能力的同时，显著提高计算效率，特别适用于大规模优化问题。通过仿真验证，改进后的麻雀搜索算法在优化麦克风阵列配置时，运行速度得到了提高，同时仍能保持较高的定位精度。

3.1.1. 快速软阈值迭代算法模型的构建

对于一个基本的线性逆问题：

$$y = Ax + w \quad (10)$$

其中 $A \in M * N$ ， $y \in M$ 且是已知的， w 是未知噪声。

(10)式最常见的方式使用最小二乘法(Least Squares)来求解：

$$\hat{x}_{LS} = \arg_x \min \|Ax - y\|_2^2 \quad (11)$$

其中，当矩阵维度 $M = N$ 且 A 为非奇异矩阵时，最小二乘法的解等价于 $A^{-1}y$ 。

线性逆问题可通过最小二乘法求解，但当系统病态时，最小二乘法容易受到微小扰动的影响，导致

结果不稳定。在这种情况下,最小二乘法并不适用,需要采用其他方法来解决病态问题,确保解的稳定性[4]。

基于上述情况,为应对病态线性系统,安德烈·尼古拉耶维奇·吉洪诺夫[4][5]提出了吉洪诺夫正则化并引入了正则化项,以实现方差和偏差的权衡。其核心思想是通过加入一个 ℓ_2 范数的正则化项来提高解的稳定性,避免病态问题导致的过大估计方差。吉洪诺夫正则化求解病态问题可以表示为:

$$\hat{x}_T = \arg \min_x \|Ax - y\|_2^2 + \lambda \|x\|_2^2 \quad (12)$$

其中 $\lambda > 0$ 为正则化参数。

在解决病态线性系统的逆问题时,通常有两种解决方法,一种求解方法时岭回归即采用 ℓ_2 范数作为正则项,LASSO (Least Absolute Shrinkage and Selection Operator)是一种优化方法,使用 ℓ_1 范数作为正则化项,有助于在求解过程中实现特征选择。其转化为二阶锥规划问题,可以通过内点法求解,但在数据维度较大时,由于计算复杂度较高,求解效率较低。LASSO问题的数学模型可以如下表示:

$$\hat{x} = \arg \min_x \|Ax - y\|_2^2 + \lambda \|x\|_1 \quad (13)$$

基于梯度的方法其算法结构简单,容易操作,并算法复杂度小,整个算法的计算量主要集中在测量矩阵 A 与导向向量 y 的数学运算上。基于上述原因,很多研究者研究通过简单的基于梯度的方法来求解凸优化问题。

在众多基于梯度的算法中,基于梯度的算法(如Iterative Shrinkage Thresholding Algorithm,简称ISTA)因其简单易操作、计算复杂度低而受到青睐。ISTA通过软阈值操作每次更新解,并使用梯度下降法最小化目标函数,在大规模数据下表现出较高的计算效率,其具体迭代格式如下:

$$x_{k+1} = \text{soft}_{\lambda t} (x_k - 2tA^T (Ax_k - y)) \quad (14)$$

其中 $\text{soft}_{\lambda t}(\cdot)$ 是软阈值操作函数:

$$\text{soft}_T = \text{sign}(x_i)(|x_i| - T) \quad (15)$$

通过这种迭代公式结合梯度下降法求解一个无约束最小化问题,可以将原本的软阈值迭代模型化简为:

$$x_k = \text{soft}_{\lambda tk} (x_{k-1} - t_k \nabla f(x_{k-1})) \quad (16)$$

文献[4]已经证明,ISTA的收敛速度为次线性,具体为 $O(1/k)$,其中 k 是迭代次数。通过选择适当的步长,ISTA能够在解决 ℓ_1 正则化问题时保持较好的收敛性能,尽管其收敛速度相较于其他方法较慢。

在长期实验过程中发现,在使用ISTA算法时运算速度仍有提升空间,为了ISTA算法的快速收敛,文献[4]的作者采用了著名的梯度加速策略Nesterov加速技术,Nesterov加速技术通过预先向前“走”一步计算梯度,并基于该位置更新参数,相比传统的梯度下降法,能显著加速收敛。其收敛速度提升至 $O(1/k^2)$,有效减少了计算时间,并在优化问题中表现优异[6]。梯度加速策略Nesterov的加速技术数学模型[5][7]如下所示:

$$\begin{aligned} v_{t+1} &= u_t v_t - \alpha_t \nabla g(\theta_t + u_t v_t) \\ \theta_{t+1} &= \theta_t + v_{t+1} \end{aligned} \quad (17)$$

ISTA不仅仅是一个特定的算法,而是软阈值操作算法的一类,包括FISTA和TWISTA等变种。它们在求解LASSO等 ℓ_1 范数正则化问题时具有较强的优势,能快速收敛并在高维数据处理中有效应用。

3.1.2. 快速软阈值迭代麻雀搜索模型的构建

对于麦克风阵列优化问题可以近似看作是对每个阵元的适应度进行无约束最小化问题，在原始麻雀搜索算法中的迭代过程分为，1) 对原始种群中的加入者与探索者的适应度在每次位置更新后适应度的重新规划，2) 每次种群更新后的探索者与加入者的身份转换后种群的更新迭代，基于公式(4)~(6)可以得知两种更新在每一次迭代过程中都会对探索者与加入者的位置信息进行重新计算，并且在种群更新后需要对种群中的麻雀重新进行适应度计算，而在上述不论是位置信息还是适应度的计算更新都会在计算过程中对种群中所有的个体进行计算，而在这些个体中并不是所有的个体都存在计算价值，有些个体在搜索最优位置信息时处于怠惰状态，在算法中表达为这些个体的位置信息以及适应度值较为理想，并不需要进行更新，类似于生物界中有些麻雀会盲从优秀的麻雀个体的跟随种群中的大部队，而这些优秀个体在短时间内并不会被淘汰或变为加入者，因为存在上述问题原始算法本身会产生巨大的计算量，算法的运行速度会逐步减缓，从而导致运行时间的增强。

而加入了FISTA算法后，会对原始麻雀搜索算法中的计算量进行软阈值操作，省去不必要的个体的计算，但这些个体并不是从种群中别剥离，两者的对比可以通过以下流程图图6更为直观的观察。

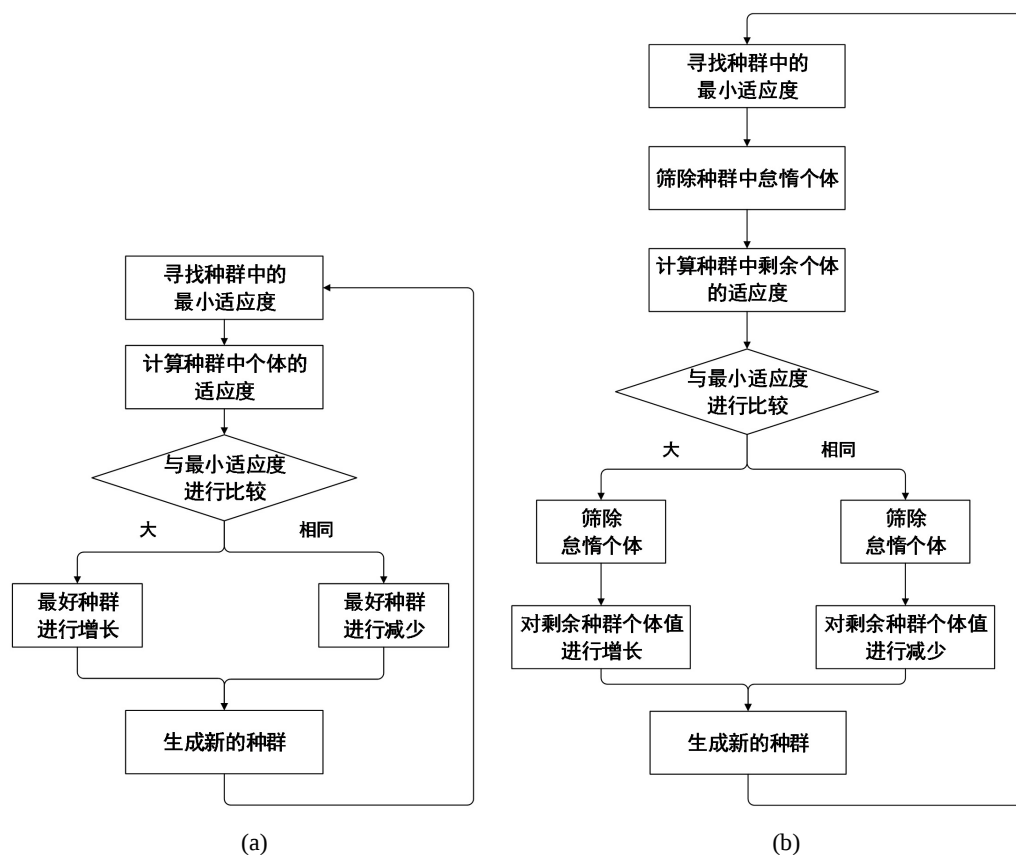


Figure 6. Iterative update process: (a) The original sparrow search algorithm (b) The soft threshold iterative sparrow search algorithm

图 6. 迭代更新流程 (a) 原始麻雀搜索算法 (b) 软阈值迭代麻雀搜索算法

从上述图中可以观察到，虽然软阈值迭代的加入使流程图变得更加复杂，但实际算法在运行过程中省略了一部分计算，从而提高了算法的运行速度，并且软阈值迭代的加入，由于部分个体的适应度在迭代过程中并没有发生改变，所以在算法运行过程中并不需要对这些个体重新计算，所以需要在种群中筛除。

通过对比原始麻雀搜索算法数学模型进行启发,改进后的麻雀搜索算法的数学模型仅对加入者与探索者的更新迭代部分进行改进,改进后数学模型如下:

探索者位置信息更新迭代模型:

$$X_{i,j}^{t+1} = \begin{cases} \text{soft}_{\lambda t} \left(X_{i,j}^t \cdot \exp \left(-\frac{i}{\alpha \cdot \text{iter}_{\max}} \right) \right) & \text{if } R_2 < ST \\ \text{soft}_{\lambda t} (X_{i,j}^t + Q \cdot L) & \text{if } R_2 \geq ST \end{cases} \quad (18)$$

加入者位置信息更新迭代模型:

$$X_{i,j}^{t+1} = \begin{cases} \text{soft}_{\lambda t} \left(Q \cdot \exp \left(\frac{X_{\text{worst}} - X_{i,j}^t}{i^2} \right) \right) & \text{if } i > n/2 \\ \text{soft}_{\lambda t} (X_p^{t+1} + |X_{i,j}^t - X_p^{t+1}| \cdot A^+ \cdot L) & \text{otherwise} \end{cases} \quad (19)$$

其中 $\text{soft}_{\lambda t}(\cdot)$ 是软阈值操作函数。

3.2. 仿真实验结果对比

同 2.2 以相同情况下对于阵列优化问题进行仿真实验,本节将以改进后的麻雀搜索算法对原始麦克风矩阵进行优化,最终得到数量,阵形以及算法运行时间的最优化结果,原始麦克风阵列参数如表 4 所示,原始麦克风阵列示意图如图 7 所示。

Table 4. Original microphone matrix parameters

表 4. 原始麦克风矩阵参数

参数名称	数量
麦克风阵元个数	385 个
阵列臂数	10 臂
矩阵阵形	同心圆
麦克风阵元相邻最小距离	0.06869 m

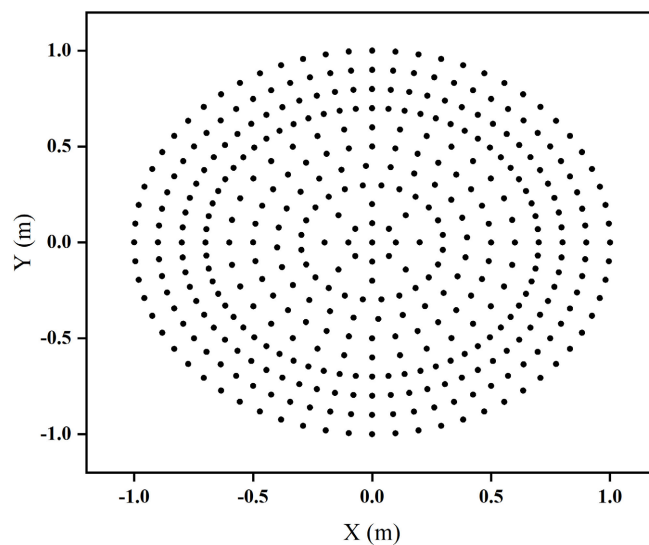


Figure 7. The original microphone matrix

图 7. 原始麦克风矩阵

实验所用麦克风阵列收集点声源在空间中声速均为340 m/s, 频率 f 均在3~4 kHz, 坐标分别为(0.3, 0.1) (0.2, 0.4) (0.1, -0.3), 实验结果如下表5所示, 优化后麦克风阵列图如图8所示。

Table 5. Original microphone matrix parameters

表 5. 优化麦克风矩阵参数

参数名称	数量
麦克风阵元个数	51 个
阵列臂数	10 臂
矩阵阵形	同心圆
麦克风阵元相邻最小距离	0.06869 m
运行时间	7.3088 s

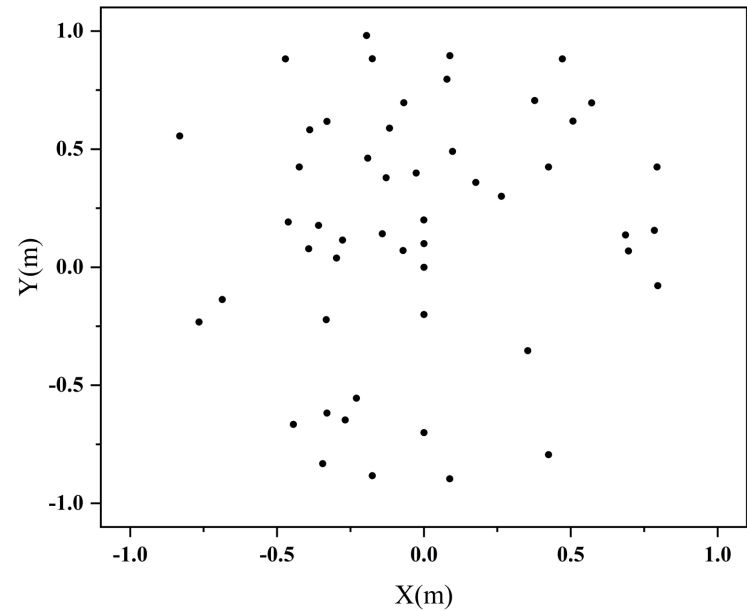


Figure 8. Optimized rear microphone matrix

图 8. 优化后麦克风矩阵

由图上可以看出, 经过软阈值迭代后的麻雀搜索算法在对原始麦克风矩阵进行阵列优化时, 相较于原始麻雀搜索算法, 不仅排布上空间利用率更高, 并且数量上优化的结果也比原始算法的效果好。

优化后的阵列示意图如图9所示, 从图中可以观察到三个声源被清晰的定位出来, 表现出优化后矩阵的声源定位效果优秀, 并且产生的旁瓣并不影响主瓣的精度; 通过客观数据对比发现, 该矩阵产生的波束宽度(BW)为1377.00度, 信噪比(SNR)达19.40 dB, 波束尖锐度(BS)为18611.00度。相较于原始算法改进后的矩阵参数, 方向图宽度提升了24.3%, 信噪比提高了7.8%, 波束尖锐度提升了7.5%。尽管方向图宽度有一定程度的增大, 但并不影响声源的定位效果。更为重要的是, 改进后算法产生的优化矩阵的信噪比优于原始算法。此外, 新算法对原始阵列进行优化后生成的新阵列中阵元数较原始算法产生的阵列小, 这进一步验证了该算法对阵列优化后生成的新阵列的高性能。最后, 从算法运行时间上看, 优化后算法的运行时间较原始算法运行时间提升了3419.3%, 在运算速度上取得巨大提升。

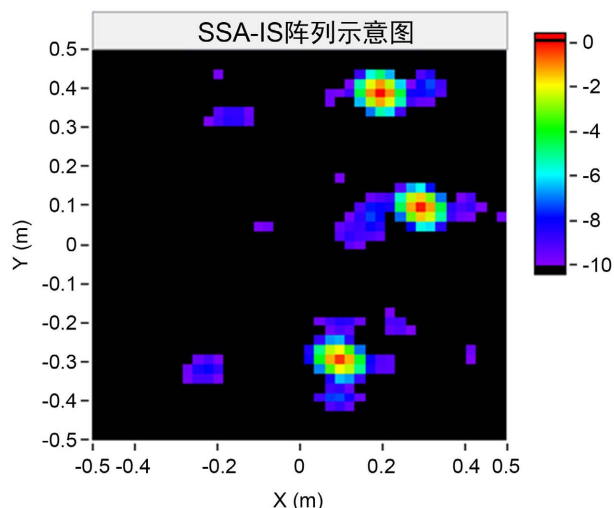


Figure 9. The localization effect of the sound source
图9. 声源定位效果

4. 小结

麦克风阵列在设备异响检测中的应用需要在信噪比(SNR)较低的环境下对声源进行高精度定位。然而,过多的传感器数量会导致数据处理复杂、成本增加,并对实时性要求提出更高的挑战。因此,如何在阵元数量和定位精度之间取得平衡,成为麦克风阵列优化中的关键问题。为了解决以上问题,本文借鉴了麻雀觅食行为,利用麻雀搜索算法(SSA)对麦克风阵列进行优化设计,并进一步结合软阈值迭代方法对传统SSA进行改进。

本文模拟麻雀群体觅食过程,探索者负责寻找最优食物源(全局搜索),加入者则对已知区域进行跟踪和补充(局部搜索),根据适应度函数对阵元分布进行迭代优化建立起相应的数学模型,并通过引入适合矩阵优化的适应度函数,计算当前麦克风阵列配置的性能指标,包括方向图宽度、信噪比增益和波束尖锐度等。

为克服传统SSA迭代速度慢、陷入局部最优等问题,本文在每次迭代中引入软阈值函数,通过动态约束对麻雀位置调整,从而加速收敛过程。软阈值迭代方法不仅显著提高了计算效率,还改善了阵列优化的结果稳定性。

通过仿真实验,本文验证了改进SSA在麦克风阵列优化中的有效性。主要成果如下:

- 1) 优化后的麦克风阵列相比原始阵列,阵元数量减少了75.1%,实现了显著的硬件资源节约。
- 2) 优化阵列的方向图主瓣宽度缩小了24.3%,提高了声源定位的分辨能力。
- 3) 波束尖锐度和信噪比分别提升了7.5%和7.8%,表明改进算法在复杂噪声环境下具有更强的抗干扰能力。
- 4) 最大旁瓣电平显著降低,进一步提高了定位的精确性。
- 5) 改进SSA的收敛速度提升了3419.3%,极大地降低了计算时间,使其在实时性要求较高的场景中更具应用价值。

综上所述,本文提出的基于改进SSA的麦克风阵列优化方法,成功实现了阵元数量的有效削减,同时保持甚至提升了声源定位性能。优化后的阵列在方向图性能、抗干扰能力和计算效率上均表现优异,为声学相机在异响检测领域的应用提供了一种高效、经济的解决方案。此外,本文的研究成果不仅能够应用于设备异响检测,还可推广至其他领域,如环境声源监测、医疗声学成像等。改进后的麻雀搜索算法

也为其他优化问题提供了新的解决思路，其通用性和灵活性值得进一步探讨。

参考文献

- [1] 薛建凯. 一种新型的群智能优化技术的研究与应用[D]: [硕士学位论文]. 上海: 东华大学, 2020.
- [2] Barnard, C.J. and Sibly, R.M. (1981) Producers and Scroungers: A General Model and Its Application to Captive Flocks of House Sparrows. *Animal Behaviour*, **29**, 543-550. [https://doi.org/10.1016/s0003-3472\(81\)80117-0](https://doi.org/10.1016/s0003-3472(81)80117-0)
- [3] Lendvai, Á.Z., Barta, Z., Liker, A. and Bókonyi, V. (2004) The Effect of Energy Reserves on Social Foraging: Hungry Sparrows Scrounge More. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, **271**, 2467-2472. <https://doi.org/10.1098/rspb.2004.2887>
- [4] Beck, A. and Teboulle, M. (2009) A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences*, **2**, 183-202. <https://doi.org/10.1137/080716542>
- [5] 王铁松, 王振杰. 用吉洪诺夫正则化方法研究自由网平差问题[C]//山东省测绘学会. 山东省测绘学术年会论文集. 2006: 76-77.
- [6] Nesterov, Y.E. (1983) A Method for Solving the Convex Programming Problem with Convergence Rate $O(\frac{1}{k^2})$. *Proceedings of the USSR Academy of Sciences*, **269**, 543-547.
- [7] 张晓丹, 苟海洋, 刘东晓, 等. 一种基于 Nesterov 加速梯度法的最小二乘逆时偏移成像研究[J]. 地球物理学进展, 2022, 37(6): 2498-2507.