

Research Article

An Improved Crayfish Optimization Algorithm with Multi-Strategy Collaboration

Jin Bo*, Wang Weimin

School of Computer, Jiangsu University of Science and Technology, Zhenjiang, China

Abstract

The crayfish optimization algorithm (COA) is competitive in solving continuous optimization problems, but its performance deteriorates in high-dimensional and multimodal environments because random initialization may provide uneven population coverage, exploration relies excessively on the current best solution, and exploitation lacks effective multi-source information interaction. To address these limitations, this study proposes a multi-strategy improved crayfish optimization algorithm (ICOA). First, an enhanced Kent chaotic map with random perturbation is used to initialize the population, thereby improving ergodicity and spatial diversity. Second, an adaptive grid-based multi-cave strategy incorporating Monte Carlo sampling dynamically constructs an elite cave set and assigns target caves according to load-aware probability distributions. This probabilistic mechanism broadens global exploration and reduces premature population aggregation. Third, a multi-source crossover operator combining adaptive frequency rank encoding with chaotic self-attention reconstructs interactions among individuals while preserving the physical scale and boundary feasibility of candidate solutions. The contribution of each strategy is examined through ablation experiments. ICOA is further compared with COA, PSO, GWO, MVO, and BOA on six challenging benchmark functions over 30 independent runs, and statistical significance is evaluated using the Wilcoxon rank-sum test. The proposed method is also applied to tension/compression spring and pressure vessel design problems. Experimental results show that ICOA consistently improves solution accuracy, convergence behavior, robustness, and the ability to escape local optima, while achieving the best optimization results in both constrained engineering applications. These findings demonstrate that the collaborative strategies provide an effective balance between global exploration and local exploitation for complex continuous optimization tasks.

Keywords

Crayfish Optimization Algorithm, Chaotic Map, Adaptive Grid, Monte Carlo Sampling, Self-Attention Mechanism, Multi-Source Crossover Operator

基于多策略协同的改进小龙虾优化算法

金波*, 王卫民

江苏科技大学计算机学院, 镇江, 中国

*Correspondence: Jin Bo (1773822281@qq.com)

Received: 2 July 2026; Accepted: 28 August 2026; Published: 14 September 2026



Copyright: © The Author(s), 2026. Published by Science Publishing Group. This is an **Open Access** article, distributed under the terms of the Creative Commons Attribution 4.0 License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution and reproduction in any medium, provided the original work is properly cited.

摘要

针对小龙虾优化算法过度依赖极值、探索不足及易陷入局部最优的缺陷，提出一种改进的小龙虾优化算法。采用改进混沌映射初始化种群以提升解的遍历性；设计自适应网格多洞穴策略，融合蒙特卡洛抽样以概率分布代替确定性选择，增强全局勘探广度；引入基于自适应排位编码与混沌自注意力的多源交叉算子，重构特征交互网络，提升高维复杂环境下的极值逃逸能力。在 6 个极具挑战性的基准函数上开展消融与对比实验，并进行 Wilcoxon 秩和检验；进一步将其应用于拉压弹簧与压力容器设计问题。结果表明，改进算法在求解精度、收敛速度与极值逃逸概率上均获显著提升，在处理强约束的实际工业优化问题时展现出卓越鲁棒性

关键词

小龙虾优化算法，混沌映射，自适应网格，蒙特卡洛抽样，自注意力机制，多源交叉算子

1. 引言

小龙虾优化算法（COA）是Jia [1]等人于2023年提出的一种新型群体智能优化算法，因数学模型简单、参数极少及寻优高效等优势，近年来被广泛应用于复杂医学与工业图像的多阈值分割[2]、深度神经网络的超参数自动化调优[3]以及无人机（UAV）复杂环境下的航迹规划[4]等前沿工程领域。相较于粒子群[5]、灰狼[6]、多元宇宙[7]和蝴蝶[8]等经典群智能算法，COA在探索与开发的自适应切换上展现出了极强的竞争力。为进一步提升其性能上限，诸多学者通过引入对立学习[9]、适应度-距离平衡[10]及聚集效应[11]等策略对其进行了有效改进。然而，现有COA算法在处理高维、多峰等复杂非凸优化问题时仍面临固有缺陷：一是伪随机初始化极易导致初始解分布不均；二是探索阶段过度依赖单一最优解，确定性的引导方式严重削弱了全局勘探广度；三是开发阶段缺乏多源信息的协同与记忆牵引，迭代后期极易陷入局部极值停滞。针对上述瓶颈，本文提出一种多策略协同改进的小龙虾优化算法（ICOA）。首先，通过引入改进的Kent混沌映射[12]显著提升初始种群的空间遍历性；其次，设计自适应网格多洞穴策略并巧妙融合蒙特卡洛[13]抽样思想，利用概率分布彻底打破种群对单一极值的绝对依赖；最后，创新性地提出基于自适应排位编码[14]与混沌自注意力[15]的多源交叉算子，重构个体特征的交互网络，从底层逻辑上赋予算法极值逃逸的动能。本文通过在6个精选的基准测试函数上开展严格的消融与对比实验，并将其应用于拉压弹簧[16]与压力容器[17]两项经典的工业约束设计问题中，验证了所提算法在提升求解精度与极值逃逸概率方面的有效性及实际工程应用价值。

2. 小龙虾优化算法

小龙虾优化算法（Crayfish Optimization Algorithm, COA）是由Jia等人[1]提出的一种新型元启发式智能优化算法。该算法的灵感来源于自然界中小龙虾的避暑、竞争和觅食三大核心生存行为。算法通过引入“环境温度”作为全局状态变量，动态调控种群在探索（避暑行为）与开发（竞争与觅食行为）阶段之间的平衡。标准COA的寻优过程可归纳为以下五个主要步骤：

2.1. 种群初始化

种群初始化是算法搜索的起点，初始解的质量与空间分布直接影响算法的收敛效率。设算法的种群规模为 N ，优化问题的解空间维度 D 。标准的COA算法采用随机分布的方式在给定的搜索空间边界内生成初始种群。第 i 只小龙虾在第 j 维空间的位置 $X_{i,j}$ 初始化数学模型如下：

$$X_{i,j} = lb_j + (ub_j - lb_j) \times rand \quad (1)$$

其中， $i \in \{1, 2, \dots, N\}$ 为个体索引， $j \in \{1, 2, \dots, D\}$ 为维度索引； lb_j 和 ub_j 分别代表第 j 维搜索空间的下界和上界； $rand$ 表示(0,1)之间的随机数。

完成初始化后，计算种群中所有个体的适应度值。算法将当前适应度最优的个体位置记录为初始全局最优位置 X_G ，并记录局部最优位置 X_L ，为后续迭代提供方向引导。

2.2. 环境温度驱动机制

小龙虾的生理机能受环境温度的严格制约，COA算法在每次迭代时，通过随机函数模拟当前环境的温度状态 $temp$ ：

$$temp = rand \times 15 + 20 \quad (2)$$

由上式可知, 模拟温度 $temp$ 在 $20 \sim 35^\circ\text{C}$ 之间无规则波动。该变量不直接参与坐标计算, 而是作为状态切换的核心阈值: 当 $temp > 30^\circ\text{C}$ 时, 触发小龙虾的避暑或竞争行为; 当 $temp \leq 30^\circ\text{C}$ 时, 触发觅食行为。

2.3. 避暑阶段

当环境温度 $temp > 30^\circ\text{C}$ 且随机概率 $rand < 0.5$ 时, 小龙虾为了躲避高温会进行大范围迁移寻找洞穴, 进入避暑阶段 (全局探索阶段)。理想洞穴位置 X_{shade} 由全局最优位置 X_G 和局部最优位置 X_L 共同决定。

$$X_{shade} = \frac{X_G + X_L}{2} \quad (3)$$

个体向洞穴迁移的位置更新公式为:

$$X_{i,j}^{t+1} = X_{i,j}^t + C \times rand \times (X_{shade} - X_{i,j}^t) \quad (4)$$

其中, t 为当前迭代次数, C 为递减曲线, 定义为:

$$C = 2 - \frac{t}{T} \quad (5)$$

其中, T 为最大迭代次数。

2.4. 竞争阶段

当环境温度 $temp > 30^\circ\text{C}$ 但随机概率 $rand \geq 0.5$ 时, 表明多个个体同时竞争同一个优质洞穴, 进入竞争阶段 (开发过渡阶段)。此时, 当前个体 Xi 会随机选取种群中的另一竞争个体 X_z ($z \in \{1, 2, \dots, N\}$ 且 $z \neq i$) 作为对抗参考, 其位置更新模型为:

$$X_{i,j}^{t+1} = X_{i,j}^t - X_{z,j}^t + X_{shade} \quad (6)$$

该公式利用个体间的位移差产生空间扰动, 并在洞穴 X_{shade} 的吸引下进行位置重构。

2.5. 觅食阶段

当环境温度适宜 (即 $temp \leq 30^\circ\text{C}$) 时, 小龙虾群体集中搜寻食物, 进入觅食阶段 (开发阶段)。此过程为 COA 深度局部开发的最核心环节。首先, 算法将当前全局最优个体位置视为食物源 X_{food} :

$$X_{food} = X_G \quad (7)$$

接着, 计算食物的相对大小 Q , 来决定后续的进食策略:

$$Q = 3 \times rand \times \left(\frac{f_i}{f_{food}} \right) \quad (8)$$

其中, f_i 和 f_{food} 分别为第 i 个体和全局最优个体适应度值。

在自然界中, 小龙虾进食活跃度依赖温度。为此, 算法利用高斯分布模型引入摄食量因子 p , 调控局部开发强度:

$$p = 0.2 \times \frac{1}{\sqrt{2\pi} \times 3} \times \exp \left(-\frac{(temp - 25)^2}{18} \right) \quad (9)$$

随后, 算法根据食物大小 Q 执行位置更新:

1) 撕裂与交替摄食机制:

当 $Q > 2$ 时, 食物规模过大, 个体需先执行撕裂操作。算法通过指数衰减使食物体积在数学上“缩小”:

随后, 模拟双步足的交替进食动作, 利用正余弦振荡进行极其精细的空间挖掘:

$$X_{i,j}^{t+1} = X_{i,j}^t + X_{food} \times p \times (\cos(2\pi \cdot rand) - \sin(2\pi \cdot rand)) \quad (10)$$

2) 直接摄食机制:

当 $Q \leq 2$ 时, 食物大小适中, 个体直接吞咽。此时去除三角函数扰动, 简化为快速向目标收敛的模型:

$$X_{i,j}^{t+1} = (X_{i,j}^t - X_{food}) \times p + p \times rand \times X_{i,j}^t \quad (11)$$

3. 改进的小龙虾优化算法

3.1. 改进 Kent 混沌映射初始化种群

标准小龙虾优化算法在初始化阶段采用伪随机数生成初始种群。这种方式在面对高维、多峰值等复杂优化景观时, 极易导致解空间覆盖率低、初始个体出现“中心扎堆”现象。初始解分布的不均匀直接削弱了算法迭代前期的全局探索能力, 进而导致后期易陷入局部最优。

混沌序列因其对初值极端敏感、遍历性强且分布均匀等数学特性, 已成为提升群智能算法初始化多样性与搜索广度的重要手段。相较于常用的 Logistic[18]、Tent[19] 等映射机制, Kent[12]混沌映射不仅参数结构简单, 且在大

规模序列生成中展现出更优的遍历均匀性与迭代速度。标准 Kent混沌映射的数学模型如下所示：

$$z_{k+1} = \begin{cases} \frac{z_k}{a}, & 0 < z_k < a \\ \frac{1-z_k}{1-a}, & a \leq z_k < 1 \end{cases} \quad (12)$$

其中， z_k 为第 k 次迭代生成的混沌变量， $a \in (0,1)$ 为分段控制参数。

然而，标准Kent混沌映射在有限字长精度的计算机中执行时，若参数 a 为简单有理数，系统轨道极易落入短周期循环（即“周期锁相”效应）。这种现象会导致生成的序列丧失随机性，从而削弱对种群多样性的增益效果。

为打破短周期循环并进一步拓宽混沌映射的扰动范围，本文引入随机扰动 $\Delta = 2 \times \text{rand}(0,1) - 1/N$ 对标准 Kent映射进行改进。引入该随机扰动项后，不仅能有效破坏原有的周期锁相路径，还能增加混沌序列在相空间中的微观跃迁能力。改进后的Kent混沌映射公式如下所示：

$$z_{k+1} = \begin{cases} \left(\frac{z_k}{a} + \Delta \right) \bmod 1, & 0 < z_k < a \\ \left(\frac{1-z_k}{1-a} + \Delta \right) \bmod 1, & a \leq z_k < 1 \end{cases} \quad (13)$$

其中， N 为种群规模； rand 表示 $(0,1)$ 之间的随机数； $\bmod 1$ 为取小数部分操作，旨在确保扰动后的混沌变量 z_{k+1} 始终被严格限制在 $(0,1)$ 的有效定义域内。

在获取到分布均匀且随机性极强的改进Kent混沌序列后，需将其逆映射至算法的实际决策空间中，以完成第 i 只小龙虾在第 j 维空间的位置 X_{ij} 的初始化：

$$X_{i,j} = \text{lb}_j + z_{i,j} \times (\text{ub}_j - \text{lb}_j) \quad (14)$$

3.2. 融入蒙特卡洛思想的自适应网格-多洞穴策略

在标准 COA 算法的高温避暑阶段，个体仅受单一全局洞穴 X_{shade} 的吸引。这种高强度的确定性单一节点引导极易导致种群在迭代前期快速聚集，致使探索范围严重萎缩；一旦 X_{shade} 陷入局部极值，整个种群将面临早熟收敛的风险。

为突破这一寻优瓶颈，本文提出一种自适应网格-多洞穴策略（Grid-Caves）。从概率统计与随机搜索的数学本质来看，该策略深刻汲取了蒙特卡洛[13]（Monte Carlo）方法的抽样哲学：彻底摒弃了传统算法中确定性的贪婪导向机制，将连续的解空间转化为离散的网格状态空间，并

依靠动态构建的概率分布模型驱动个体的随机探索。该策略通过动态调整精英洞穴数量，并利用负载感知的蒙特卡洛轮盘赌进行目标指派，天然维持了种群的空间分散性，从根源上杜绝了局部扎堆现象。Grid-Caves策略的数学执行流程如下：

3.2.1. 目标值归一化与体积测度

为保证不同量纲的目标函数值在同一尺度上具有可比性，首先对种群的适应度目标值进行线性归一化：

$$f_i = \frac{f_i - f_{\min}}{\max(f_{\max} - f_{\min}, \varepsilon)} \quad (15)$$

其中， f_i 为归一化后的第 i 个体的目标函数值，其取值映射至区间[1]； $f_{\min}$ 和 $f_{\max}$ 为当前种群在历史迭代中搜索到的目标函数最小与最大边界值； $\varepsilon = 10^{-12}$ 为防零偏置常数。

3.2.2. 动态网格划分

基于归一化目标区间的宽度（即体积测度），动态自适应计算当前迭代步的网格总数 $K(t)$ ：

$$K(t) = \left\lceil \frac{\alpha \cdot N}{\max(f_{\max} - f_{\min}, \varepsilon)} \right\rceil \quad (16)$$

其中， N 为种群总规模； α 为网格划分缩放因子， $K(t)$ 用于控制网格的基础密度。当目标区间宽度逐渐变小（即种群发生收敛）时，会自适应放大以实现网格细化。

3.2.3. 网格索引与唯一标识映射

第 i 个个体的离散网格编号 b_i 计算模型为：

$$b_i = \min \left\{ \text{floor}(f_i \cdot K(t)), K(t) - 1 \right\} \quad (17)$$

其中， $b_i \in \{0, 1, \dots, K(t) - 1\}$ ，该索引号即为个体对应网格的唯一标识。

3.2.4. 候选洞穴提取

针对划分出的每一个非空网格 G ，仅提取该网格内部适应度最优的个体作为本区域的代表性候选洞穴：

$$X_G^{\text{cave}} = \arg \min_{i \in G} f_i \quad (18)$$

其中, $\mathbf{X}_G^{\text{cave}}$ 为网格 G 内的最佳决策向量。遍历所有非空网格提取的候选洞穴, 共同构成当前代的全局候选洞穴矩阵 $\mathbf{C}_{\text{all}}$ 。

3.2.5. 精英洞穴的动态截断机制

为在迭代的不同阶段自适应地平衡算法的全局探索与局部开发能力, 本文对提取出的候选洞穴集合 $\mathbf{C}_{\text{all}}$ 实施了基于保留比例指数衰减的动态截断机制。

在迭代初期, 算法需要维持较高的多洞穴数量以支持广域的拓扑勘探; 而随着迭代的深入, 为了加速收敛并聚焦于高质解区, 予以保留的精英洞穴数量应当逐步缩减。规定当代予以保留的精英洞穴数量 $M_c(t)$ 随迭代次数呈指数级衰减:

$$\eta(t) = \eta_0 \cdot \exp\left(-\frac{\kappa \cdot t}{T}\right) \quad (19)$$

$$M_c(t) = \max(M_{\min}, \text{ceil}(\eta(t) \cdot N_{\text{all}})) \quad (20)$$

其中, t 与 T 为当前迭代次数与最大迭代次数; $\eta(t)$ 为随迭代次数动态变化的洞穴保留比例; η_0 为初始保留比例基数, 取值区间为 $[0.4, 0.6]$; N_{all} 为当前代候选洞穴集合 $\mathbf{C}_{\text{all}}$ 的元素总数; $\text{ceil}(\cdot)$ 为向上取整函数; $M_{\min}$ 为保障种群多样性的最低保留洞穴数量底线, 通常设为 $\max(1, \text{ceil}(0.02N))$ 。

完成上述指数衰减截断后, 将剩余的高置信度解按适应度排序, 构建出本代的核心引导库, 即精英洞穴集合 $\mathbf{C}_{\text{elite}}$ 。

3.2.6. 基于蒙特卡洛抽样的负载感知目标指派

对于进入避暑阶段的普通个体 i , 本文创新性地引入蒙特卡洛抽样机制为其分配目标洞穴。

首先计算个体与第 k 个精英洞穴在归一化空间的欧氏距离 $d_{i,k}$, 并定义受收缩半径 $R(t)$ 约束的局部可见集 Ω_i :

$$\Omega_i = \{k \mid d_{i,k} \leq R(t)\} \quad (21)$$

$$R(t) = \rho \cdot \sqrt{D} \cdot \left(1 - \frac{t}{T}\right)^{\beta_R} \quad (22)$$

其中, D 为决策维度; ρ 与 β_R 分别为初始基数与收缩指数。随后, 计算可视域内各洞穴的负载惩罚概率质量函数 (PMF):

$$p_{i,k} = \frac{\left(\frac{1}{1 + \beta\mu_k}\right)^\gamma}{\sum_{j \in \Omega_i} \left(\frac{1}{1 + \beta\mu_j}\right)^\gamma} \quad (23)$$

其中, $p_{i,k}$ 为第 k 个洞穴被选中的概率; μ_k 为记录洞穴当前已被选中次数的负载计数器; β 为拥挤排斥惩罚系数; γ 为概率放大指数。

完成概率分布构建后, 个体 i 的避暑洞穴不再依据确定性原则进行指派, 而是通过多项式分布 (Categorical) 进行蒙特卡洛[13]轮盘赌抽样:

$$\begin{aligned} \mathbf{X}_i^{\text{shade}} &= \mathbf{X}_{k^*}^{\text{cave}}, \\ k^* &\sim \text{Categorical}(p_{i,1}, \dots, p_{i,|\Omega_i|}) \end{aligned} \quad (24)$$

通过引入蒙特卡洛抽样机制, Grid-Caves 策略使得个体在大概率向优质且空闲洞穴收敛的同时, 依然保留了小概率探索次优或稍拥挤洞穴的试探性。这种概率分布驱动的随机游走极大地丰富了种群避暑轨迹的多样性, 从根本上阻断了单一解造成的群体坍缩效应。

3.3. 基于自适应排位编码与混沌自注意力的多源智能交叉算子

在将交叉算子引入小龙虾优化算法时, 传统机制多依赖于随机个体间的双亲基因互换。这种盲目的匹配缺乏对种群全局拓扑结构的感知, 在处理高维复杂空间时易破坏有效基因并引发早熟收敛。Transformer[14]架构中的自注意力机制[15] (Self-Attention, SA) 具备卓越的多源特征协同能力, 但将其直接应用于启发式算法时面临三大底层冲突: 种群无序性与 SA 序列化要求的冲突; 算法无反向传播梯度导致网络投影权重无法更新的困境; 以及真实物理宽量纲易导致注意力计算中隐特征发生“数值淹没”的量纲灾难。为破解上述瓶颈, 本文摒弃传统机制, 创新性地提出一种融合自适应排位编码与表征解耦的多源智能交叉算子 (MSA-Crossover), 在 COA 的局部开发后期重构个体间的特征交互网络。

3.3.1. 种群序列化与自适应频率排位编码 (AFPE)

为使自注意力机制获取对个体优劣的感知能力, 本文构建了序列化排位编码机制。在算法的每次交叉操作前, 计算当前种群个体的适应度, 并按贪婪法则由优至劣进行严格升序排列, 生成有序种群矩阵 $\mathbf{X}_{\text{sorted}} \in \mathbb{R}^{N \times D}$ 。(其中 N 为种群规模, D 为决策变量的空间维度)。

为规避宽量纲坐标导致排位编码特征产生“数值淹没”现象,引入双通道表征解耦机制,首先对 X_{sorted} 沿各特征维度进行极值归一化:

$$\hat{X}_{i,j} = \frac{X_{i,j} - lb_j}{ub_j - lb_j} \times 2 - 1 \quad (25)$$

其中, $\hat{X}_{i,j}$ 表示归一化后第 i 个个体在第 j 维度的标度无关意图表征值; $X_{i,j}$ 为对应的真实物理坐标; ub_j 和 lb_j 分别为第 j 维搜索空间的物理上界与下界。

随后,构建排位位置编码。标准Transformer编码公式依赖10000的常数底数,在连续优化问题的低维空间(如 $D \leq 100$)中会引发高维频率极速衰减,导致不同排位精英个体的特征向量趋于常数化。因此,本文提出自适应频率排位编码(Adaptive Frequency Positional Encoding, AFPE),以空间维度 D 为缩放基底:

$$PE_{pos,j} = \begin{cases} \sin\left(\pi \cdot \frac{pos}{N} \cdot 2^{\frac{j-1}{D}}\right), & j \text{ 为奇数} \\ \cos\left(\pi \cdot \frac{pos}{N} \cdot 2^{\frac{j}{D}}\right), & j \text{ 为偶数} \end{cases} \quad (26)$$

其中, $PE_{pos,j}$ 表示排位索引为 $pos \in [1, N]$ 的个体在第 $j \in [1, D]$ 维的编码值; pos/N 将绝对排位映射为相对排位。该分段三角函数确保了不同排位的个体特征在低维隐空间中具有显著的非线性区分度。

最后,将归一化表征矩阵 $\hat{X}$ 与排位编码矩阵 PE 进行逐元素叠加,生成输入矩阵 X_{in} :

$$X_{in} = \hat{X} + PE \quad (27)$$

3.3.2. 物理量纲保真的多源交叉自注意力机制 (MSA-Crossover)

在获取意图感知输入矩阵 X_{in} 后,算法需构建完整的查询(Query)、键(Key)与值(Value)映射体系。受限于启发式算法无法通过反向传播更新网络权重,本文利用混沌映射的遍历性动态生成投影权重方阵 $W_Q \in \mathbb{R}^{D \times D}$ 与 $W_K \in \mathbb{R}^{D \times D}$,进而获取查询矩阵 Q 与键矩阵 K 。

$$Q = X_{in} W_Q, \quad K = X_{in} W_K \quad (28)$$

在标准的自注意力架构中,特征融合需将通过 Q, K 算得的注意力概率与值矩阵 V 进行点积。通常 $V = X_{in} W_V$ 。然而,考虑到启发式算法中个体的基因坐标具备严格的物理边界约束,若将其映射至无量纲的隐空间,将彻底破坏交叉后新生个体的物理合法性。为此,本文在值获取通道

中强制解除归一化约束,实施物理量纲保真策略,即将值投影矩阵 W_V 设定为恒等单位矩阵 I 。由此,值矩阵 V 严格等价于种群当前的真实物理坐标矩阵 X_{sorted} :

$$V = X_{sorted} W_V = X_{sorted} I = X_{sorted} \quad (29)$$

在确立了物理保真的 V 矩阵后,本文引入方差缩放机制(以空间维度 D 为缩放基底)防Softmax梯度饱和,执行完整的多亲加权自注意力特征融合。融合后的注意力特征引导矩阵 $X_{att} \in \mathbb{R}^{N \times D}$ 的计算模型如下所示:

$$\begin{aligned} X_{att} &= \text{Attention}(Q, K, V) \\ &= \text{Softmax}\left(\frac{QK^T}{\sqrt{D}}\right)V \end{aligned} \quad (30)$$

3.3.3. 基于残差网络结构的自适应位置更新

为避免个体瞬间向引导质心跃迁导致自我遗传记忆丧失,本文借鉴残差连接(Residual Connection)[20]架构,将 X_{att} 视作引导扰动量叠加至个体当前状态,并施加边界反弹约束:

$$X_{new} = X_{sorted} + \omega \cdot (X_{att} - X_{sorted}) \quad (31)$$

$$X_{new} = \max(\min(X_{new}, ub), lb) \quad (32)$$

其中, X_{new} 为完成交叉重组后的新生代种群; $\omega = 1 - t/T$ 为自适应融合因子(t 为当前迭代次数, T 为最大迭代次数),随迭代进程线性减以平衡算法的全局探索与局部开发。 $\max$ 与 $\min$ 为逐元素极值截断函数,确保新生个体的坐标严格落于物理可行域 $[lb, ub]$ 内部。

3.4. 算法流程

算法1 改进小龙虾优化算法

输入: 种群规模 N , 最大迭代 T , 空间维度 D 及边界 lb_j 和 ub_j , 目标函数 $f(x)$ 。

输出: 全局最优解 X_G 及目标值 f_G 。

1.初始化: 利用式(13)-(14)的改进 Kent 混沌映射生成初始种群; 计算所有个体的初始适应度,记录当前全局最优 X_G 与局部最优 X_L 。

2.主循环迭代寻优:

While $t \leq T$ do

用式(2)模拟当前环境温度 $temp$

if $temp \leq 30^\circ\text{C}$ then

利用式(15)-(20)提取体积测度,执行自适应网格划分并构建精英洞穴库 C_{elite} 。

for $i = 1$ to N do

利用式(21)-(23)计算可视域内各洞穴的负载惩罚概率 $p_{i,k}$ 。

利用式(24)执行蒙特卡洛轮盘赌抽样，确定个体目标洞穴 X_i^{shade} 。

```
if rand < 0.5 then
    执行长距离避暑迁跃位置更新。
else
    随机选个体，执行竞争位置更新。
end if
end for
else( temp ≤ 30 °C )
for i = 1 to N do
    计算食物规模 Q 与高斯摄食概率 p
    if Q > (C3 + 1) / 2 then
        撕裂食物，交替振荡摄食更新。
    else
        执行直接收敛摄食更新。
    end if
end for
    对当前觅食后种群执行适应度升序排列，生成  $X_{sorted}$ 。
    利用式(25)-(27)对种群特征进行极值归一化与自适应频率排位编码 (AFPE)。
    利用式(28)-(30)执行方差受控的混沌矩阵投影，构建物理量纲保真的 Q-K-V 架构并计算自注意力引导矩阵  $X_{att}$ 。
    利用式(31)-(32)执行多源交叉融合与残差连接，生成并截断新生代种群  $X_{new}$ 。
end if
    计算新生代种群适应度和全局最优  $X_G$ 。
    t = t + 1
end while
return 全局最优解  $X_G$  及目标值  $f_G$ 。
```

4. 实验结果与讨论

为验证多策略协同改进小龙虾优化算法 (ICOA) 的综合寻优性能，本章开展了严格的消融实验与横向对比测

试。鉴于篇幅限制并避免同质化冗余评估，本文从国际经典基准测试集中精选了6个极具挑战性的极值景观函数作为测试床：包括用于检验极限开发精度与收敛速率的单峰函数 F_6, F_7 ；用于探测全局勘探广度与惩罚边界逃逸能力的高维复杂多峰函数 F_8, F_9 ；以及考核深层极值微调与探索开发动态平衡的复合函数 F_{20}, F_{21} 。精选测试函数的具体数学表达式及参数空间界限详见表1。

4.1. 实验环境与参数设置

本研究所有仿真实验均在统一的软硬件平台下独立完成。硬件配置为Intel Core i7-6700HQ(2.60 GHz)处理器与 8GB 内存；软件环境为 64位 Windows 10 操作系统与 MATLAB R2021b。为保证公正，ICOA与小龙虾算法 (COA) [1]、粒子群算法 (PSO) [5]、灰狼算法 (GWO) [6]、多元宇宙算法 (MVO) [7]及蝴蝶算法 (BOA) [8]的种群规模统一设为 $N = 30$ ，最大迭代次数 $T = 500$ 。所有算法均在各函数上独立运行 30 次，以平均适应度 (Avg) 量化寻优精度，标准差 (Std) 量化鲁棒性。各基线算法的核心超参数均采用原始文献推荐的最佳值，参数配置明细如表2所示。

4.2. ICOA 改进策略的有效性分析

为验证 ICOA所引入策略的独立有效性与协同增益，以标准COA为基线，构建三种变体：仅引入改进Kent混沌映射的ICOA1；仅引入网格蒙特卡洛抽样的ICOA2；仅引入自注意力多源交叉算子的ICOA3。各变体在6个精选函数上的统计结果见表3。分析可知，单一策略在特定适应度景观下均表现出显著的定向增益。在复杂惩罚边界的多峰函数 F_8, F_9 中，ICOA2的精度大幅优于基线，证明了基于蒙特卡洛[13]抽样的多洞穴机制有效打破了单一极值依赖，强制种群分散从而规避了早熟坍塌；在复合函数中，ICOA3展现出深度的微观挖掘能力，印证了自注意力交叉[15]算子通过提取群体多源特征牵引，成功弥补了原算法无记忆性的短板。最终，全策略融合的ICOA在6个函数上均包揽了最优均值与极小标准差，实现了前期广域勘探、中期抗早熟与后期精准微调的协同闭环。

表 1 精选基准测试函数明。

测试函数	维度	范围	最优值
$f_6(x) = \sum_{i=1}^n (x_i + 0.5)^2$	30	[-100,100]	0
$f_7(x) = \sum_{i=1}^n ix_i^4 + rand(0,1)$	30	[-1.28,1.28]	0

测试函数	维度	范围	最优值
$f_8(x) = \sum_{i=1}^n -x_i \sin(\sqrt{ x_i })$	30	[-500,500]	-12569.5
$f_9(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[-5.12,5.12]	0
$f_{20}(x) = -\sum_{i=1}^4 c_i \exp\left(-\sum_{j=1}^6 a_{i,j}(x_j - p_{i,j})^2\right)$	6	[1]	-3.32
$f_{21}(x) = -\sum_{i=1}^5 [(x - a_i)(x - a_i)^T + c_i]^{-1}$	4	[10]	-10.15

表 2 对比算法参数设置。

算法	参数设置
ICOA	$a = 0.4$
	$\alpha = 0.8, \eta_0 = 0.5, \kappa = 1.5, \rho = 0.15,$
	$\beta_R = 1.0, \beta = 0.1, \gamma = 2.0$
COA	$\omega = 1.0 \rightarrow 0$
	$temp_{th} = 30^{\circ}C, C_3 = 3.0$
	$\mu = 25, \sigma = 3, C_1 = 0.2$
PSO	$c_1 = 2.0, c_2 = 2.0$
	$w = 0.9 \rightarrow 0.4$
GWO	$a = 2 \rightarrow 0$
MVO	$WEP_{min} = 0.2, WEP_{max} = 1.0$
	$p = 6.0$
BOA	$c_0 = 0.01, p = 0.8$

4.3. 与其他先进算法的对比与分析

为进一步确立 ICOA 的横向竞争优势，将其与 COA[1]、PSO[5]、GWO[6]、MVO[7]、BOA[8]算法展开综合寻优精度对比（见

表4），并提取平均迭代数据绘制了典型收敛曲线（见图1）。

由

表4数据可知，传统的PSO与BOA在高维复杂空间中易丧失多样性，求解精度偏低；标准COA虽在单峰函数表现尚可，但在复合函数上陷入严重的极值停滞。相比之下，ICOA展现出了压倒性的寻优统治力：在单峰函数 F_6, F_7 上，其精度逼近理论极限，印证了自注意力[15]多源交叉算子赋予个体的极致微观开采能力；在极具欺骗性的多峰函数 F_8 与复合函数 F_{20}, F_{21} 中，ICOA取得了跨越对比算法数个量级的代际优势，确凿证明了融入蒙特卡洛[13]抽样的多洞穴机制彻底打破对单一极值的盲目趋附，强制种群规避了早熟坍塌。

结合图1的收敛行为进一步剖析可知：在迭代初期，ICOA曲线呈现断崖式下降，这表明改进 Kent[12]混沌映射赋予了种群极高的初始空间遍历性，大幅提升了前期勘探效率；在迭代中后期，当GWO[6]、MVO[7]等算法陷入局部极值导致曲线停滞时，ICOA仍能出现多次“阶梯状”的二次下跌。这一显著的极值逃逸现象，有力印证了自注意力多源交叉算子通过重构虚拟引力，成功引导个体跳出局部深渊，实现了全局探索与局部开发的完美平衡，展现出卓越的收敛精度与鲁棒性。

4.4. Wilcoxon 秩和检验分析

为从统计学层面验证 ICOA 的性能优势并非源于随机误差，本节采用非参数 Wilcoxon 秩和检验对各算法独

立运行30次的结果进行显著性分析。设定统计显著性水平 $\alpha = 0.05$,若 p 值小于0.05，则拒绝性能无差异的原假设，表明 ICOA 显著优于对比算法；若 $p \geq 0.05$ （在表5中以下划线标注），或输出 NaN（同时求得理论极值），则判定两者性能无显著差异。

由表5可知，在与COA[1]、PSO[5]、GWO[6]、MVO[7]、BOA[8]算法的较量中，ICOA 的 p 值几乎全部远小于0.05。这一压倒性的概率证据确凿排除了偶然性干扰，充分印证了引入的混沌初始化、Grid-Caves策略及自注意力交叉算子在底层逻辑上实现了实质性的性能跃升，赋予了ICOA 极其卓越的寻优鲁棒性。

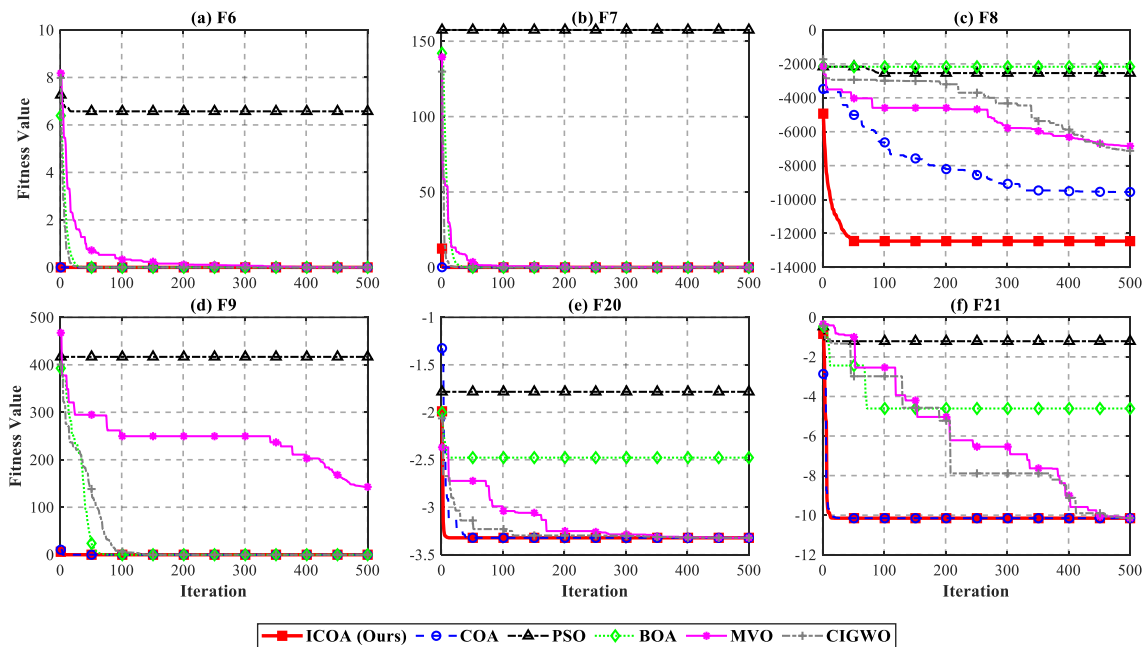


图 1 多算法收敛曲线对比。

表 3 算法消融实验性能对比。

函数		ICOA	COA	ICOA1	ICOA2	ICOA3
f_6	Ave	0.00E+00	3.94E-01	1.39E-03	7.34E-07	2.86E-05
	Std	0.00E+00	2.20E-01	2.06E-03	2.50E-06	4.54E-05
f_7	Ave	3.73E-06	8.68E-05	8.93E-05	4.01E-05	5.63E-05
	Std	2.74E-06	9.90E-05	5.95E-05	3.63E-05	5.89E-05
f_8	Ave	-1.21E+04	-8.50E+03	-5.89E+03	-1.21E+04	-6.62E+03
	Std	2.66E+02	8.47E+02	6.11E+02	2.66E+02	5.32E+02
f_9	Ave	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00

函数		ICOA	COA	ICOA1	ICOA2	ICOA3
f_{20}	Ave	-3.32E+00	-3.27E+00	-3.17E+00	-3.30E+00	-3.29E+00
	Std	1.50E-08	6.99E-02	1.12E-01	5.36E-02	6.16E-02
f_{21}	Ave	-1.02E+01	-6.45E+00	-1.02E+01	-1.02E+01	-1.02E+01
	Std	7.17E-15	2.77E+00	2.13E-04	1.79E-08	7.17E-15

表 4 多算法寻优精度对比。

函数		ICOA	COA	PSO	BOA	MVO	GWO
f_6	Ave	0.00E+00	5.89E+00	6.24E+04	6.22E+00	2.74E+02	2.29E+01
	Std	0.00E+00	9.82E-01	5.57E+03	5.46E-01	1.15E+02	9.19E+00
f_7	Ave	3.83E-06	1.31E-03	1.34E+02	7.41E-05	5.92E-01	5.94E-02
	Std	2.94E-06	9.94E-04	2.18E+01	8.34E-05	1.63E-01	2.13E-02
f_8	Ave	-1.23E+04	-4.21E+03	-2.71E+03	-2.48E+03	-6.99E+03	-5.31E+03
	Std	2.08E+02	1.06E+03	4.16E+02	3.82E+02	5.89E+02	1.25E+03
f_9	Ave	0.00E+00	0.00E+00	4.20E+02	2.41E-11	2.14E+02	7.83E+01
	Std	0.00E+00	0.00E+00	2.57E+01	1.88E-12	3.59E+01	3.87E+01
f_{20}	Ave	-3.32E+00	-3.27E+00	-1.35E+00	-2.24E+00	-3.27E+00	-3.22E+00
	Std	2.26E-02	6.63E-02	5.51E-01	3.14E-01	7.43E-02	1.12E-01
f_{21}	Ave	-8.05E+00	-5.54E+00	-7.94E-01	-3.99E+00	-6.28E+00	-7.48E+00
	Std	2.79E+00	2.75E+00	4.20E-01	6.06E-01	3.51E+00	3.23E+00

表 5 Wilcoxon 秩和检验结果。

函数	COA	PSO	BOA	MVO	GWO
f_6	1.21e-12	1.21e-12	1.21e-12	1.21e-12	1.21e-12
f_7	2.37e-10	3.02e-11	3.16e-10	3.02e-11	3.02e-11
f_8	2.86e-11	2.86e-11	2.86e-11	2.86e-11	2.86e-11
f_9	NAN	1.21e-12	1.21e-12	1.21e-12	1.21e-12
f_{20}	7.71e-07	1.79e-11	1.79e-11	1.79e-11	1.79e-11
f_{21}	2.65e-05	8.07e-11	2.70e-11	2.56e-03	4.31e-02

4.5. 经典工程设计问题

4.5.1. 拉压弹簧设计问题

拉压弹簧设计[16]问题的核心目标是在满足最小挠度、剪切应力、震动频率及几何外形等4个严格的不等式力学约束前提下，寻找 3个连续设计变量——线径 d 、平均卷径 D 与有效圈数 N 的最优组合，从而使弹簧的整体重量（制造成本）最小化。

设设计变量向量 $x = [x_1 \ x_2 \ x_3] = [D \ d \ N]$ ，其相应的搜索空间边界分别为： $0.05 \leq x_1 \leq 2.0$,

$$0.25 \leq x_2 \leq 1.3, 2.0 \leq x_3 \leq 15.0。$$

目标函数：

$$f(x) = (x_3 + 2) \cdot x_2 \cdot x_1^2$$

约束条件：

$$g_1(x) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \leq 0$$

$$g_2(x) = \frac{4x_2^2 - x_1 x_2}{12566(x_2 x_1^3 - x_1^4)} + \frac{1}{5108 x_1^2} - 1 \leq 0$$

$$g_3(x) = 1 - \frac{140.45 x_1}{x_2^2 x_3} \leq 0$$

$$g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \leq 0$$

本文将 ICOA 与 COA[1]、PSO[5]、GWO[6]、MVO[7]、BOA[8]算法共同应用于拉压弹簧设计[16]问题进行了对比。表6的实验统计结果中可以看出，ICOA 在解决拉压弹簧设计问题方面排名第一。

表 6 拉压弹簧设计寻优结果对比。

算法	变量的最优值			优化成本
	D	d	N	
ICOA	0.0519	0.3609	11.0514	0.012667
COA	0.0521	0.3660	10.7781	0.012679
PSO	0.0701	0.8543	3.0331	0.021100
GWO	0.0521	0.3658	10.7886	0.012687
MVO	0.0557	0.4607	7.2100	0.013169
BOA	0.0500	0.3143	14.6252	0.013063

4.5.2. 压力容器设计问题

压力容器设计[17]问题旨在通过寻找 4个核心设计变量——球壳厚度 T_s 、圆柱壳厚度 T_h 、内半径 R 与圆柱体长度 L 的最优组合，从而使制造该容器的总成本最小化。

设数学模型 $x = [x_1 \ x_2 \ x_3 \ x_4] = [T_s \ T_h \ R \ L]$ ，该工程问题的搜索空间物理边界分别为： $0 \leq x_1, x_2 \leq 99$ ， $10 \leq x_3, x_4 \leq 200$ ，且 x_1 是 x_2 0.0625英寸的整数倍。

目标函数：

$$f(x) = 0.6224 \cdot x_1 \cdot x_3 \cdot x_4 + 1.7781 \cdot x_2 \cdot x_3^2 + 3.1661 \cdot x_1^2 \cdot x_4 + 19.84 \cdot x_1^2 \cdot x_3$$

约束条件：

$$g_1(x) = -x_1 + 0.0193 \cdot x_3 \leq 0$$

$$g_2(x) = -x_2 + 0.00954 \cdot x_3 \leq 0$$

$$g_3(x) = -\pi \cdot x_3^2 \cdot x_4 - \frac{4}{3} \pi \cdot x_3^3 + 1296000 \leq 0$$

$$g_4(x) = x_4 - 240 \leq 0$$

本文将 ICOA 与 COA[1]、PSO[5]、GWO[6]、MVO[7]、BOA[8]算法共同应用于压力容器设计[17]问题进行了对比。表7的实验统计结果中可以看出，ICOA 在解决压力容器设计问题方面排名第一。

表 7 压力容器设计寻优结果对比。

算法	变量的最优值				优化成本
	T_s	T_h	R	L	
ICOA	0.8125	0.4375	42.0984	176.6367	6059.7164
COA	0.8125	0.4375	42.0963	176.6723	6060.1921
PSO	4.9375	3.5000	67.9159	13.9993	65557.4017
GWO	0.8125	0.4375,	42.0958	176.6959	6060.6473
MVO	0.8750	0.4375	45.3041	140.6399	6095.6847
BOA	1.1875	0.6250	56.1947	56.6265	7686.2581

5. 结论

针对小龙虾优化算法 (COA) [1]易陷入局部最优、全局探索不足的缺陷, 本文提出一种多策略协同改进的 ICOA算法。首先, 引入改进 Kent[12]混沌映射初始化种群, 打破了随机生成的聚簇盲区; 其次, 提出自适应网格-多洞穴策略 (Grid-Caves), 融入蒙特卡洛[13]概率抽样, 强制分散避暑轨迹, 有效延缓了早熟收敛; 最后, 跨界引入基于 Transformer[14]架构的自注意力[15]多源交叉算子, 通过提取群体历史特征构建虚拟牵引, 彻底弥补了原算法无记忆性的短板, 赋予个体强大的极值逃逸动能。

在 6 个极具挑战性的基准测试函数上的消融与对比实验表明, ICOA 的收敛精度与鲁棒性全面超越 ICOA 与 COA[1]、PSO[5]、GWO[6]、MVO[7] 及 BOA[8], 且 Wilcoxon 秩和检验确证了其显著的统计学优势。此外, 在拉压弹簧与压力容器两项复杂工程设计问题中, ICOA 展现出了极强的约束跨越与极值逼近能力, 寻优成本最低。

鉴于 ICOA 卓越的寻优性能与极值逃逸能力, 未来的工作计划将其拓展至更为复杂的实际工程与离散优化场景, 重点研究方向包括:

- (1) 复杂医学与工业图像的多阈值精确分割[2];
- (2) 深层神经网络与机器学习模型的超参数自动化调优[3];
- (3) 无人机 (UAV) [4]在三维动态未知环境下的安全航迹规划;

参考文献

[1] Jia H, Rao H, Wen C, et al. Crayfish optimization algorithm [J]. Artificial Intelligence Review, 2023, 56(Suppl 2): 1919-1979. <https://doi.org/10.1007/s10462-023-10567-4>

[2] Hu P, Han Y, Zhang Z, et al. A multi-level thresholding image segmentation algorithm based on equilibrium optimizer [J].

Scientific reports, 2024, 14(1): 29728. <https://doi.org/10.1038/s41598-024-81075-w>

[3] Hyperparameter optimization of deep neural network using univariate dynamic encoding algorithm for searches [J]. Knowledge-Based Systems, 2019, 178: 74-83.

[4] Chai X, Zheng Z, Xiao J, et al. Multi-strategy fusion differential evolution algorithm for UAV path planning in complex environment [J]. Aerospace Science and Technology, 2022, 121: 107287. <https://doi.org/10.1016/j.ast.2021.107287>

[5] Kennedy J, Eberhart R. Particle swarm optimization [C] // Proceedings of ICNN'95-international conference on neural networks. iee, 1995, 4: 1942-1948. <https://doi.org/10.1109/ICNN.1995.488968>

[6] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer [J]. Advances in engineering software, 2014, 69: 46-61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>

[7] Mirjalili S, Mirjalili S M, Hatamlou A. Multi-verse optimizer: a nature-inspired algorithm for global optimization [J]. Neural computing and applications, 2016, 27(2): 495-513. <https://doi.org/10.1007/s00521-015-1870-7>

[8] Arora S, Singh S. Butterfly optimization algorithm: a novel approach for global optimization [J]. Soft computing, 2019, 23(3): 715-734. <https://doi.org/10.1007/s00500-018-3102-4>

[9] Mahdavi S, Rahnamayan S, Deb K. Opposition based learning: A literature review [J]. Swarm and evolutionary computation, 2018, 39: 1-23. <https://doi.org/10.1016/j.swevo.2017.09.010>

[10] Huang D, Yang J, Yu J. A particle swarm optimization with fitness-distance balance strategy [C] // 2022 18th International Conference on Computational Intelligence and Security (CIS). IEEE, 2022: 336-340. <https://doi.org/10.1109/CIS58238.2022.00077>

[11] Davies T K, Mees C C, Milner-Gulland E J. The past, present and future use of drifting fish aggregating devices (FADs) in the Indian Ocean [J]. Marine policy, 2014, 45: 163-170.

- [12] Kent R M, Barbosa W A S, Gauthier D J. Controlling chaotic maps using next-generation reservoir computing [J]. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 2024, 34(2).
- [13] Rubinstein R Y, Kroese D P. *Simulation and the Monte Carlo method* [M]. John Wiley & Sons, 2016.
- [14] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [J]. *Advances in neural information processing systems*, 2017, 30.
- [15] Keles F D, Wijewardena P M, Hegde C. On the computational complexity of self-attention [C] // *International conference on algorithmic learning theory*. PMLR, 2023: 597-619.
- [16] Tzanetos A, Blondin M. A qualitative systematic review of metaheuristics applied to tension/compression spring design problem: Current situation, recommendations, and research direction [J]. *Engineering Applications of Artificial Intelligence*, 2023, 118: 105521.
<https://doi.org/10.1016/j.engappai.2022.105521>
- [17] Moyya S, Thejasree P, Abraham B C, et al. Design and analysis of single and multi-layer pressure vessel [J]. *Materials Today: Proceedings*, 2023.
- [18] Alawida M. Enhancing logistic chaotic map for improved cryptographic security in random number generation [J]. *Journal of Information Security and Applications*, 2024, 80: 103685.
<https://doi.org/10.1016/j.jisa.2023.103685>
- [19] Akraam M, Rashid T, Zafar S. An image encryption scheme proposed by modifying chaotic tent map using fuzzy numbers [J]. *Multimedia Tools and Applications*, 2023, 82(11): 16861-16879. <https://doi.org/10.1007/s11042-022-13941-6>
- [20] Szegedy C, Ioffe S, Vanhoucke V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning [C]//*Proceedings of the AAAI conference on artificial intelligence*. 2017, 31(1).