多策略集成的樽海鞘群算法的机器人路径规划

王秋萍,王彦军,戴 芳
(西安理工大学理学院,陕西西安 710054)

摘 要: 针对求解机器人路径规划问题,本文提出了一种多策略集成的樽海鞘群算法.在该算法中,提出了新的自适应领导者结构,以平衡算法的探索和开发能力;引入可以提高 Lyapunov 指数的 Logistic-Cubic 级联混沌映射作为食物源的扰动算子,来避免算法陷入局部最优;采用基于自适应参数的分散觅食策略使部分追随者探索有前景的区域.在 CEC 2014 测试集的多种函数上,本文算法与 3 种改进的樽海鞘群算法和 5 种先进的群智能算法进行比较,结果表明本文算法综合优化性能更好.本文算法 2 将其用于求解机器人路径规划问题,其中用三次样条插值对路径进行平滑.在障碍是 8,9,13 的环境下分别进行仿真实验,仿真结果表明,本文算法在给定的仿真场景下与给定的对比算法相比获得了最好的结果.

关键词: 樽海鞘群算法; 路径规划; Lyapunov 指数; 级联混沌; 三次样条
中图分类号: TP242 **文献标识码:** A **文章编号:** 0372-2112 (2020)11-2101-13
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2020.11.003

Multi-Strategy Ensemble Salp Swarm Algorithm for Robot Path Planning

WANG Qiu-ping, WANG Yan-jun, Dai Fang
(*Faculty of Sciences, Xi'an University of Technology, Xi'an, Shaanxi* 710054, *China*)

Abstract: A multi-strategy ensemble salp swarm algorithm is proposed for solving problem of robot path planning. In the algorithm, a new adaptive leader structure is proposed to balance the exploration and exploitation ability of the algorithm. The chaotic map of Logistic-Cubic cascade which can improve the Lyapunov exponent of the cascade chaotic system is introduced as the disturbance operator of the food source to avoid the algorithm falling into the local optimum. A disperse foraging strategy based on adaptive parameters is adopted to force a part of followers to explore promising areas. The algorithm in this paper is compared with three improved SSA algorithms and five state-of-the-art swarm intelligence algorithms on IEEE CEC 2014 functions. The results show that the comprehensive optimization performance of the algorithm in this paper is better. The proposed algorithm is applied to solve the robot path planning problem, in which the path is smoothed by cubic spline interpolation. Simulation experiments are implemented on computer in the environments where the obstacles are 8, 9, 13, respectively. The simulation results demonstrate that the proposed algorithm can achieve the best results compared with the given contrast algorithms in given simulation scenarios.

Key words: salp swarm algorithm; path planning; Lyapunov exponent; cascade chaotic; cubic spline

1 引言

目前,移动机器人广泛应用于军事、空间、医疗卫生等诸多领域.路径规划是移动机器人技术研究中最基本、关键的问题之一.按照规划目标可将机器人路径规划分为全局路径规划和局部路径规划[1].根据工作环境中障碍物的性质的不同,路径规划可分为静态路径规划和动态路径规划[2].本文研究在工作环境信息已知的情景移动机器人全局路径规划问题.

常用的路径规划技术有 Dijkstra 算法、A* 算法、人工势场法和 RRT 算法等. Dijkstra 算法是遍历算法,能搜索到网络中任意两点之间最短路径,然而随着网络图中节点数目的增多,Dijkstra 算法的搜索效率显著降低. A* 算法存在着空间复杂度和时间复杂度过高等不足,而人工势场法易出现引力和斥力相互抵消的情况,从而无法到达终点[2]. RRT 算法搜索连通路径后,由于

收稿日期:2020-01-20;修回日期:2020-08-05;责任编辑:马兰英
基金项目:国家自然科学基金 (No. 61976176)

其算法本身的随机性[3],产生的路径会存在绕远或者出现明显的拐角,使生成的路径不平滑. 所以随着环境系统复杂性及任务难度的增加,传统的路径规划方法难以取得理想的效果[4]. 为了克服这些困难,研究者们提出了基于元启发式算法来解决路径规划问题,如蚁群算法[2]、蝙蝠算法[4],粒子群算法[5]等,为解决复杂环境下的路径规划问题提供了新思路.

樽海鞘群算法(Salp Swarm Algorithm, SSA)[6]是Mirjalili等人通过模拟樽海鞘在海洋中航行和觅食时的群体行为于2017年提出的一种新的元启发式算法. 该算法采用链式结构,链的前端是领导者,其余部分是追随者,通过调节唯一的控制参数c_1使得领导者在食物源附近进行探索和开发,追随者的渐进运动有效地降低了陷入局部极值的可能性. 因此,该算法利用樽海鞘链追踪移动的食物源来逼近全局最优. 自SSA提出以来,已成功应用于诸多领域如参数识别、图像分割、数据挖掘. 文献[7]将具有交叉算子的二进制SSA用于特征选择问题,交叉算子在一定程度上增强种群探索能力. 文献[8]利用混沌映射调节SSA中的参数c_2,改善了算法的性能,同时结果验证了logistic映射是所使用的10个映射中最优的. 文献[9]在SSA算法中,引入了文化基因算法,基于种群独立寻优和基于群落的信息交流的框架下,提高算法全局搜索、局部探索的能力和收敛稳定性. 文献[10]通过改进参数c_1,增强了算法的开发能力,提出了一种增强型樽海鞘群算法,算法的探索能力还可以进一步提高. 然而,SSA在解决一些复杂优化问题时仍存在着搜索能力差、求解精度低、收敛速度较慢等问题.

SSA算法保留了到目前为止得到的最好的解,将其标记为食物源变量,即使整个种群恶化,最好的解也不会丢失. 算法只有一个主要控制参数,SSA算法简单易实现. 为进一步提高SSA算法的性能,拓展其应用领域,本文提出一种多策略集成的樽海鞘群算法(Multi-strategy Ensemble Salps Swarm Algorithm, MESSA),用于求解路径规划问题. 在MESSA中:(1)为了平衡算法探索和开发能力,提出一种自适应领导者结构;(2)引入随机性比Logistic映射更强的Logistic-Cubic级联混沌映射作为食物源的扰动算子,以避免算法陷入局部最优;(3)采用分散觅食策略来迫使追随者个体探索有前景的区域,以增强算法搜索能力. 通过求解CEC2014基准测试函数的最优解,验证了MESSA的良好性能,并用该算法成功解决了移动机器人路径规划问题.

2 樽海鞘群算法[6]

在SSA中,樽海鞘链由领导者和追随者组成,其中领导者位于樽海鞘链前端的第1个,而其他樽海鞘则是追随者. 所有樽海鞘的位置都存储在一个$N\times D$二维矩阵x中(N是种群规模,D是给定问题的变量数). 假设在搜索空间中有一个食物源F作为樽海鞘链的目标. 领导者更新位置如式(1):

$$x_j^1=\begin{cases}F_j+c_1((ub_j-lb_j)c_2+lb_j), & c_3\geqslant 0.5\\ F_j-c_1((ub_j-lb_j)c_2+lb_j), & c_3<0.5\end{cases} \tag{1}$$

其中,x_j^1表示第一个樽海鞘(领导者)位置的第j维;ub_j、lb_j是第j维的上、下界;F_j是食物位置的第j维;c_2、c_3是在[0,1]区间内均匀生成的随机数;c_1是控制参数,平衡了算法在迭代过程中的勘探与开发能力,如式(2):

$$c_1=2e^{-(\frac{4t}{T})^2} \tag{2}$$

其中t是当前迭代次数;T是最大的迭代数目.

使用式(3)更新追随者位置:

$$x_j^i=\frac{1}{2}(x_j^i+x_j^{i-1}) \tag{3}$$

其中$i\geqslant 2$,x_j^i表示第i个追随者位置的第j维.

3 多策略集成的樽海鞘群算法

3.1 自适应领导者结构

在基本的SSA中,只有一个领导者,这种机制在解决具有大量局部最优解的高维问题时,算法有较大的概率陷入局部最优. 对此文献[11]提出了一种领导者为$N/2$的樽海鞘群算法,以增强算法的局部开发能力. 本文在此基础上提出一种自适应领导者结构,在迭代中领导者个数从1自适应增加到$N/2$,以更好地平衡算法探索和开发能力.

在迭代前期,算法应具有较强的探索能力,以便找到更加有希望的解空间,在迭代后期,为更好地在当前最好解的附近空间中搜索,算法应具有较强的开发能力. 因此,自适应领导者结构策略能够在迭代前期具有较多的追随者个体,从而增强算法的探索能力,随着迭代的进行,领导者数量逐渐增加,从而算法的开发能力逐渐增强. 用N_1和N_2分别表示领导者个数和追随者个数,则:

$$N_1=\lceil (t/T)\times(N/2)\rceil \tag{4}$$

其中$\lceil\ \rceil$是向上取整. 于是,$N_2=N-N_1$.

3.2 Logistic-Cubic级联混沌扰动

初值敏感性是影响混沌随机性的重要因素,而Lyapunov指数是初值敏感性的一种度量[12]. 混沌系统的级联是一种提高系统初值敏感性的一种简单有用的方法.

定理[12] 设混沌子系统$f_1(x_n)$和$f_2(x_n)$构成的级联混沌系统为$f_2(f_1(x_n))$,则混沌系统$f_2(f_1(x_n))$的

Lyapunov 指数等于混沌系统 $f_1(x_n)$、$f_2(x_n)$ 的 Lyapunov 指数之和.

Logistic-Cubic 级联混沌映射[12]为:

$$r_{n+1}=\left|\frac{[\mu r_n(1-r_n)]^3}{a^2}-3\mu r_n(1-r_n)\right| \tag{5}$$

其中映射参数区间 $\mu\in[0,4]$,$a=0.5$,$r_0\in(0,1)$.

Logistic-Cubic 级联混沌的最大 Lyapunov 指数为 1.7011[12]. 因此,本文为避免算法陷入局部最优,利用 Logistic-Cubic 级联混沌映射对当前食物源位置进行扰动,以提高算法的搜索性能. 其公式如下:

$$F^*(t+1)=F(t)+r_{n+1}\cdot F(t) \tag{6}$$

$fit(x)$ 为 x 的适应度值,采用贪心保留策略,

$$F(t+1)=\begin{cases}F^*(t+1), & fit(F^*)<fit(F)\\ F(t), & \text{否则}\end{cases} \tag{7}$$

3.3　分散觅食策略(Disperse foraging)

进化后期,种群将会聚集在当前食物源附近,导致种群多样性降低,算法出现搜索停滞现象. 因此,本文将分散觅食策略[13]引入追随者位置更新,以增加算法全局搜索能力,避免陷入局部最优.

在优化过程,利用分散率(DR)[13]的自适应行为来控制分散到新区域的追随者数量.

$$DR=DR_{\max}-(DR_{\max}-DR_{\min})\cdot t/T \tag{8}$$

其中 $DR_{\max}=0.4$[13],$DR_{\min}=0$[13],T 为最大的迭代次数.

追随者位置更新公式如下:

$$x_j^i(t+1)=x_j^i(t)+\rho\cdot\Delta_j^i(t)\cdot B_j^i(t) \tag{9}$$

$$\Delta_j^i(t)=x_j^m(t)-x_j^n(t) \tag{10}$$

其中,ρ 是缩放因子,$\rho\sim N(0.5,0.1^2)$;$\Delta_j^i(t)$ 表示随机移动距离;$x_j^m(t)$ 和 $x_j^n(t)$ 为在种群中随机选择的两个个体;$B_j^i(t)$ 是用来确定追随者是否被分散的逻辑值,其公式如下:

$$B_j^i(t)=\begin{cases}1, & rand>DR\\ 0, & \text{其他}\end{cases} \tag{11}$$

3.4　MESSA 算法

算法 1　MESSA 算法,步骤如下:

初始化阶段

Step 1:初始化参数,种群规模 N,维数 D,最大迭代次数 T,变量的上界 ub、下界 lb;

Step 2:在搜索空间中随机产生 N 个樽海鞘;

Step 3:计算适应度值,并进行排序;

Step 4:根据排序的适应度值对樽海鞘位置进行排序;对食物赋值 F(最好的樽海鞘);

主循环阶段

Step 5:While $t <= T$

　执行自适应领导者结构策略,式(4);

　For $i=1,2,\cdots,N$

　If $i <= N_1$

　　更新领导者位置,式(1);

　Else

　　更新追随者位置,式(3);

　　执行分散觅食策略,式(8)~式(11);

　End

Step 6:适应度评估;

Step7:食物(F)更新:

　　执行 Logistic-Cubic 级联混沌扰动,式(6)~式(7);

　　食物位置更新;

　End For

　$t=t+1$

输出阶段

Step 8:输出解 F.

4　路径规划数学模型

假设机器人工作环境为二维空间,在空间中分布着有限数量的静态障碍物(凸多边形),机器人路径规划的任务是在起始点和终止点之间找到一条最短的、较平滑的、且避开所有障碍物的路径. 本文采用导航点模型[5]来构建机器人工作环境,如图 1 所示.

4.1　路径编码

起点 S 与终点 G 的连线为 X'轴构建坐标系 S-$X'Y'$,如图 1 所示. 然后将坐标系 O-XY 中的点变换到 S-$X'Y'$ 中,变换公式如下:

$$\begin{bmatrix}x'\\ y'\end{bmatrix}=\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix}+\begin{bmatrix}x_s\\ y_s\end{bmatrix} \tag{12}$$

其中,(x_s,y_s) 是坐标系 O-XY 中起点 S 的坐标,(x',y') 是点 (x,y) 在坐标系 S-$X'Y'$ 中对应的点,θ 是 X 轴与直线 SG 的夹角.

如图 1 所示,用 m 条平行线簇 $l_1,l_2,\cdots,l_m$ 将 SG 平均分为 $m+1$ 段,则相邻两条平行线间的距离为 $\Delta l=\|SG\|/(m+1)$. 在每条平行线上随机产生一个点,可以构造一条完整的路径 $(S,P_1,P_2,\cdots,P_m,G)$,则一条路径上的所有结点作为一个樽海鞘个体的编码. 通过变换式(12),可以将移动机器人路径规划问题转化为对这一集合点的变量进行优化[1].

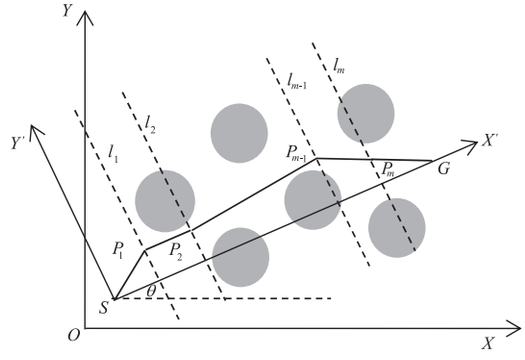

图1　环境中的路径表示

4.2 适应度函数

路径规划主要考虑路径长度、路径安全度和路径平滑度3个指标.

(1)路径长度

设起点 S 和终点 G 的坐标为 $S(x_0,y_0)$、$G(x_{m+1},y_{m+1})$,任意路径节点坐标表示为 $P_i(x_i,y_i)$,$1\leqslant i\leqslant m$. 路径长度去量纲之后如下:

$$f_{\text{length}}=\sum_{i=0}^{m}\frac{\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2}}{\sqrt{(x_{m+1}-x_0)^2+(y_{m+1}-y_0)^2}} \tag{13}$$

(2)路径安全度

为避免机器人与障碍物发生碰撞且路径更加平滑,采用准则是在三次样条插值节点处与障碍物无碰撞. 假设已知 m 个路径结点的坐标(x_1,y_1),(x_2,y_2),…,(x_m,y_m)以及起点和终点的坐标(x_0,y_0),(x_{m+1},y_{m+1}). 在横坐标和纵坐标上,分别通过三次样条插值得到 d 个插值点的横坐标$(x_1,x_2,\cdots,x_d)$和纵坐标$(y_1,y_2,\cdots,y_d)$. 此时,得到了 d 个插值点,需判断 d 个插值点是否与障碍物发生碰撞,本文将 d 设置为100[4]. 因此,路径安全度指标计算如下:

$$f_{\text{safe}}=w\cdot v=w\sum_{k=1}^{K}\sum_{j=1}^{d}\max(1-D_{j,k}/R(k),0) \tag{14}$$

其中,w 为100是安全因子[4,5];$D_{j,k}$为第 j 个插值点到第 k 障碍物中心的距离;$R(k)$是第 k 个障碍物的半径;K 表示障碍物的个数.

(3)路径平滑度[1]

设路径节点为$(S,P_1,P_2,\cdots,P_m,G)$,ϕ_i 表示相邻路段的夹角,则路径光滑度指标计算如下:

$$f_{\text{smooth}}=\sum_{i=1}^{m}\phi_i=\sum_{i=1}^{m}\arccos\left(\frac{(P_i-P_{i-1})\cdot(P_{i+1}-P_i)}{|P_i-P_{i-1}||P_{i+1}-P_i|}\right) \tag{15}$$

ϕ_i 越小表示路径光滑程度越优.

综上所述,本文构建的目标函数如下:

$$J=f_{\text{length}}+f_{\text{safe}}+f_{\text{smooth}} \tag{16}$$

4.3 基于 MESSA 的路径规划算法

算法2 运用 MESSA 算法求解机器人路径规划问题,算法的具体步骤如下:

Step1:进行坐标系变换,且利用式(12)对起点、终点、障碍物位置的坐标进行变换.

Step2:用 m 条平行线簇 $l_1,l_2,\cdots,lm$ 将 SG 平均分为 $m+1$ 段,则间距为 $\Delta l=\|SG\|/(m+1)$.

Step3:初始化参数,如 N、m、T、ub 和 lb 等.

Step4:在每条平行线 $l_j(0<j\leqslant m)$ 上随机生成一个点,则樽海鞘 $x_i=(x_1^i,x_2^i,\cdots,x_m^i)$,$0<i\leqslant N$,初始化 N 个樽海鞘.

Step5:利用适应度函数式(16),对樽海鞘进行评估,当前群体最佳个体,记为食物 F.

Step6:While t <= T then

利用式(2)计算控制因子 c_1.

执行主循环.

Step7:end While

Step8:return F.

5 CEC 2014 基准测试集上的数值实验

为了分析 MESSA 的性能,在 CEC 2014[14] 基准测试函数上比较了 MESSA 与 SSA,其他三种改进的 SSA 算法和五种先进的群智能算法.

5.1 基准测试函数和参数设置

CEC 2014 基准测试集包含了单峰、多峰、混合和复合类型的30个测试函数. 按照 CEC 2014 提供的准则,每个变量的搜索空间范围是[-100,100],种群规模 $N=50$,维数 $D=50$,算法终止准则为函数的最大评估次数$(10^4\times D)$,$DR_{\max}=0.4$[13],$DR_{\min}=0$[13]. 为实验的公平,初始化位置是相同的. 对于每个测试函数,每个算法独立运行51次,记录其均值和标准差.

5.2 MESSA 与改进的 SSA 的比较

将 MESSA 算法与 SSA 和三种改进的 SSA(CSSA[8]、ESSA[10]、CASSA[15])进行了实验比较,结果如表1所示,其中最好的结果以粗体显示.

表1 改进算法在 CEC2014 上的实验结果

函数	指标	算法				
		SSA	CSSA	ESSA	CASSA	MESSA
F_1	Mean	7.3264E+07	7.6410E+09	4.2150E+08	6.8620E+06	**5.0071E+06**
	Std	3.4581E+07	3.5056E+09	1.5335E+08	1.5591E+06	**2.3840E+06**
F_2	Mean	1.0766E+07	1.3826E+07	1.5918E+10	3.7254E+06	**9.4250E+03**
	Std	1.4668E+07	2.3752E+07	4.6751E+09	5.3636E+05	**9.2830E+03**
F_3	Mean	4.3453E+04	4.1172E+04	6.2777E+04	7.2609E+04	**2.2929E+03**
	Std	9.5885E+03	1.0150E+04	9.2800E+03	6.4383E+03	**1.0049E+04**

续表

函数	指标	算法				
		SSA	CSSA	ESSA	CASSA	MESSA
F_4	Mean	5.9259E+02	5.9529E+04	2.8586E+03	5.2034E+03	**4.9848E+02**
	Std	4.5434E+01	5.4615E+01	7.8774E+02	1.1273E+02	**3.1190E+01**
F_5	Mean	5.2004E+02	5.2004E+02	5.2079E+02	5.2104E+02	**5.2004E+02**
	Std	4.4113E−02	5.3386E−02	1.2318E−01	4.5981E−02	**1.2502E−02**
F_6	Mean	6.2917E+02	6.2909E+02	6.3711E+02	**6.2581E+02**	6.2992E+02
	Std	3.4337E+00	3.4862E+00	2.5534E+00	**1.8265E+00**	5.4087E+00
F_7	Mean	**7.2117E+02**	7.2589E+02	7.3451E+02	7.3021E+02	7.2452E+02
	Std	**2.8674E−01**	2.1546E−01	5.2867E+01	5.7139E+01	1.1386E−02
F_8	Mean	9.1274E+02	9.4128E+02	1.0145E+03	1.0959E+03	**9.1158E+02**
	Std	2.5749E+01	2.8569E+01	2.4832E+01	1.5919E+01	**2.7939E+01**
F_9	Mean	1.0253E+03	1.0528E+03	1.1482E+03	1.2158E+03	**1.0228E+03**
	Std	2.0703E+01	3.0445E+01	2.9280E+01	1.8763E+01	**3.6076E+01**
F_{10}	Mean	4.5657E+03	**4.5579E+03**	6.6219E+03	8.1196E+03	4.6738E+03
	Std	5.2149E+02	**5.9513E+02**	6.8671E+02	3.4275E+02	7.7403E+02
F_{11}	Mean	5.1969E+03	**4.9453E+03**	7.5338E+03	8.9709E+03	4.9608E+03
	Std	6.4069E+02	**6.6471E+02**	5.9472E+02	3.1645E+02	6.7970E+02
F_{12}	Mean	1.2011E+03	1.2011E+03	1.2021E+03	1.2032E+03	**1.2002E+03**
	Std	4.1513E−01	4.3726E−01	4.9089E−01	4.1714E−01	**1.4169E−01**
F_{13}	Mean	1.3006E+03	1.3006E+03	1.3036E+03	1.3049E+03	**1.3005E+03**
	Std	1.6198E−01	1.1785E−01	5.4402E−01	4.6480E−01	**1.2774E−01**
F_{14}	Mean	1.4004E+03	1.4004E+03	1.4726E+03	1.5250E+03	**1.4004E+03**
	Std	1.6479E−01	1.6947E−01	2.1285E+01	2.1763E+01	**1.0081E−01**
F_{15}	Mean	1.5399E+03	1.5435E+03	8.3387E+03	4.1905E+04	**1.5067E+03**
	Std	1.6936E+01	1.5405E+01	5.7896E+03	2.0738E+04	**2.3740E+00**
F_{16}	Mean	1.6119E+03	1.6122E+03	1.6130E+03	1.6133E+03	**1.6112E+03**
	Std	5.3775E−01	4.2786E−01	3.4382E−01	2.1064E−01	**1.8508E−01**
F_{17}	Mean	2.1103E+06	2.8516E+06	2.2565E+07	3.2438E+07	**2.6818E+05**
	Std	1.4556E+06	1.9729E+06	1.7573E+07	1.2975E+07	**2.1693E+05**
F_{18}	Mean	4.8930E+03	**4.6925E+03**	9.3349E+07	7.2814E+05	4.7028E+05
	Std	3.8091E+03	**3.9682E+03**	8.5294E+07	3.2092E+05	2.3651E+06
F_{19}	Mean	1.9293E+03	1.9356E+03	2.1005E+03	2.1333E+03	**1.9169E+03**
	Std	2.1227E+01	2.9404E+01	5.1573E+01	5.3416E+01	**1.6738E+01**
F_{20}	Mean	1.7948E+04	1.9136E+04	7.2324E+04	8.6153E+04	**5.1666E+03**
	Std	8.3130E+03	1.0184E+04	4.1213E+04	4.4968E+04	**3.3015E+03**
F_{21}	Mean	5.1597E+05	6.6956E+05	4.9905E+06	1.0559E+07	**1.2702E+05**
	Std	3.6738E+05	6.6808E+05	4.4864E+06	4.8708E+06	**9.2090E+04**
F_{22}	Mean	2.8044E+03	2.8467E+03	3.2125E+03	3.7544E+03	**2.6804E+03**
	Std	1.7734E+02	2.6049E+02	2.7497E+02	2.5226E+02	**1.9362E+02**

续表

函数	指标	算法				
		SSA	CSSA	ESSA	CASSA	MESSA
F_{23}	Mean	2.6437E+03	2.6210E+03	**2.5000E+03**	**2.5000E+03**	**2.5000E+03**
	Std	2.2556E+01	6.1893E+01	**0.0000E+00**	**0.0000E+00**	**0.0000E+00**
F_{24}	Mean	2.6359E+03	2.6324E+03	**2.6000E+03**	**2.6000E+03**	**2.6000E+03**
	Std	4.7535E+00	5.7302E+00	**0.0000E+00**	**0.0000E+00**	**0.0000E+00**
F_{25}	Mean	2.7071E+03	2.7080E+03	**2.7000E+03**	**2.7000E+03**	**2.7000E+03**
	Std	2.1439E+00	2.6828E+00	**0.0000E+00**	**0.0000E+00**	**0.0000E+00**
F_{26}	Mean	2.7006E+03	2.7006E+03	2.7118E+03	2.7044E+03	**2.7005E+03**
	Std	1.7894E−01	1.5713E−01	2.6118E+01	5.7406E−01	**1.2882E−01**
F_{27}	Mean	3.5390E+03	3.5444E+03	**2.9000E+03**	**2.9000E+03**	**2.9000E+03**
	Std	3.1074E+02	3.1502E+02	**0.0000E+00**	**0.0000E+00**	**0.0000E+00**
F_{28}	Mean	4.6683E+03	4.6625E+03	**3.0000E+03**	**3.0000E+03**	**3.0000E+03**
	Std	5.3623E+02	4.9232E+02	**0.0000E+00**	**0.0000E+00**	**0.0000E+00**
F_{29}	Mean	1.7033E+07	1.3623E+07	**3.1000E+03**	**3.1000E+03**	4.4118E+04
	Std	2.9908E+07	1.7444E+07	**0.0000E+00**	**0.0000E+00**	1.1698E+05
F_{30}	Mean	9.2179E+04	1.1552E+05	**3.2000E+03**	**3.2000E+03**	**3.2000E+03**
	Std	5.2031E+04	7.5780E+04	**0.0000E+00**	**0.0000E+00**	**0.0000E+00**

由表1可知,依均值结果MESSA在绝大多数测试函数(24个函数)上都要优于SSA. 在F_{23}、F_{24}、F_{25}、F_{27}、F_{28}、F_{30}上,MESSA、CASSA和ESSA获得了相同的均值和标准差. CSSA在其中3个测试函数上获得了最好的结果. 总体来讲,MESSA在5个算法中具有最好的性能.

性能曲线[13]曲线可以清楚地反映由n_g个算法组成的算法集G中一个算法在函数集F上的性能. 选择平均适应度值作为性能度量. $\mu_{f,g}$为算法g在函数f上的平均适应度值,性能比$r_{f,g}$计算如式(17):

$$r_{f,g}=\frac{\mu_{f,g}}{\min\{\mu_{f,g}:g\in G\}} \tag{17}$$

为了评估算法g在所有测试函数上总体性能,定义:

$$\rho_g(\tau)=\frac{1}{n_f}\text{size}\{f\in F:r_{f,g}\leqslant\tau\} \tag{18}$$

其中,n_f是函数集F中测试函数的个数,$\tau\in R$.

由图2可知,按照性能比指标来评价算法,MESSA在4个改进的SSA中是最好的.

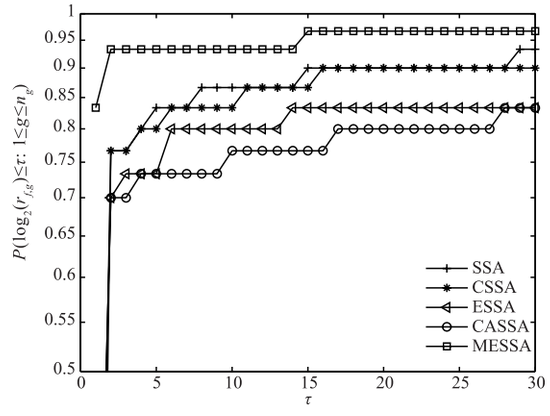

图2 5种算法在CEC 2014基准函数上的性能概况

5.3 MESSA与其他群智能算法的比较

将MESSA算法与PSO[5]、GWO[16]、KH[17]、MVO[18]和HHO[19]进行仿真对比实验,其对比算法的参数设置如表2所示. 实验结果如表3所示,其中最好的结果用粗体表示.

表2 算法参数设置

算法	参数设置
PSO	$\varpi_{\max}=0.95,\varpi_{\min}=0.4,c_1=2.0,c_2=2.0$
GWO	$a_{\max}=2.0,a_{\min}=0$
KH	$N_{\max}=0.01,v_f=0.02,D_{\max}=0.005$
MVO	$\varpi_{\max}=1.0,\varpi_{\min}=0.2,\ p=6.0$
HHO	$E_0=2rand()-1$

表 3　MESSA 与其他群智能算法对比实验结果

函数	指标	算法					
		PSO	KH	MVO	GWO	HHO	MESSA
F_1	Mean	8.5983E+08	2.3245E+07	9.3902E+07	5.7481E+07	1.5364E+08	**8.6416E+06**
	Std	2.2215E+09	7.3700E+06	5.8002E+07	3.4867E+07	7.1432E+07	**3.4057E+06**
F_2	Mean	1.0134E+10	2.2124E+04	2.3560E+05	4.0890E+09	4.1447E+09	**9.6117E+03**
	Std	4.9219E+10	1.2333E+04	4.6078E+04	2.2423E+09	1.7241E+09	**9.0449E+03**
F_3	Mean	**1.7582E+03**	1.0501E+05	4.5786E+02	4.7190E+04	9.3802E+04	3.0925E+04
	Std	**4.0104E+02**	1.2874E+04	4.4814E+01	1.2415E+04	1.7622E+04	1.0172E+04
F_4	Mean	5.3163E+02	8.9144E+02	9.0340E+02	8.3452E+02	1.2776E+03	**5.1655E+02**
	Std	5.3282E+01	4.1048E+01	2.9140E+01	1.7241E+02	2.2690E+02	**2.8597E+01**
F_5	Mean	5.2105E+02	**5.2001E+02**	5.2024E+02	5.2115E+02	5.2101E+02	5.2008E+02
	Std	7.4045E−02	**3.9042E−03**	5.1583E−02	3.2664E−02	9.1492E−02	1.7619E−01
F_6	Mean	6.2759E+02	6.4045E+02	**6.2106E+02**	6.2647E+02	6.6282E+02	6.3299E+02
	Std	1.3927E+01	5.9162E+00	**4.3527E+00**	3.2630E+00	4.7307E+00	5.2319E+00
F_7	Mean	7.0525E+02	7.0008E+02	7.0042E+02	7.3357E+02	7.3328E+02	**7.0001E+02**
	Std	8.2084E+00	2.1500E−02	7.6869E−02	1.6503E+01	1.1904E+01	**7.6480E−03**
F_8	Mean	1.0307E+03	1.0380E+03	**9.6240E+02**	9.6643E+02	1.1047E+03	1.0456E+03
	Std	8.4940E+01	2.4897E+01	**3.8838E+01**	2.1525E+01	2.1205E+01	4.7015E+01
F_9	Mean	1.1030E+03	1.1540E+03	1.0848E+03	5.0759E+03	1.3264E+03	**1.0054E+03**
	Std	3.5887E+01	3.9398E+01	4.1607E+01	3.3125E+01	3.4359E+01	**2.9008E+01**
F_{10}	Mean	**5.6806E+03**	7.9536E+03	6.8108E+03	6.2896E+03	7.7621E+03	7.9350E+03
	Std	**6.4063E+02**	7.8298E+02	7.4080E+02	7.8343E+02	1.1280E+03	1.4178E+03
F_{11}	Mean	6.0417E+03	8.1696E+03	6.5372E+03	6.5797E+03	1.0737E+04	**5.5073E+03**
	Std	7.0206E+02	9.4315E+02	9.1822E+02	8.0842E+02	1.1594E+03	**1.4116E+02**
F_{12}	Mean	1.2003E+03	1.2001E+03	1.2004E+03	1.2024E+03	1.2030E+03	**1.2001E+03**
	Std	2.0925E−01	5.5708E−02	2.3273E−01	1.5755E+00	5.7535E−01	**4.1605E−01**
F_{13}	Mean	1.3005E+03	1.3005E+03	1.3006E+03	1.3006E+03	1.3005E+03	**1.3001E+03**
	Std	8.9378E−02	7.2572E−02	1.1496E−01	9.4713E−02	1.0967E−01	**1.2982E−01**
F_{14}	Mean	1.4004E+03	1.4002E+03	1.4006E+03	1.4034E+03	1.4013E+03	**1.4002E+03**
	Std	2.1736E−01	8.3527E−02	3.9667E−01	5.1602E+00	2.3774E+00	**3.0099E−01**
F_{15}	Mean	**1.5153E+03**	1.5285E+03	1.5176E+03	1.7090E+03	2.1518E+03	2.5149E+03
	Std	**6.9950E+00**	6.1712E+00	3.8794E+00	3.0180E+02	6.1720E+02	3.9891E+00
F_{16}	Mean	1.6205E+03	1.6219E+03	1.6209E+03	**1.6200E+03**	1.6224E+03	1.6215E+03
	Std	7.1728E−01	3.4786E−01	6.4552E−01	**9.8137E−01**	4.1726E−01	6.9763E−01
F_{17}	Mean	8.5571E+07	6.2249E+05	5.3316E+05	3.5781E+06	3.4456E+07	**3.5453E+05**
	Std	2.2610E+08	9.5980E+04	2.8895E+05	3.5548E+06	2.1504E+07	**4.1300E+05**
F_{18}	Mean	4.9670E+09	4.3568E+03	4.9872E+03	3.5545E+07	1.7169E+07	**4.6153E+03**
	Std	9.7603E+09	1.3024E+03	1.9129E+03	8.1241E+07	4.4433E+07	**2.0253E+03**
F_{19}	Mean	2.2908E+03	1.9525E+03	**1.9210E+03**	1.9676E+03	2.0091E+03	1.9464E+03
	Std	9.3211E+02	2.4424E+01	**3.1957E+00**	1.7812E+01	4.0167E+01	2.3866E+01

续表

函数	指标	算法					
		PSO	KH	MVO	GWO	HHO	MESSA
F_{20}	Mean	9.8135E+04	3.0984E+04	**2.7249E+03**	1.5942E+04	6.3252E+04	1.5018E+04
	Std	6.8181E+05	9.9569E+03	**1.2991E+02**	5.4861E+03	2.2133E+04	8.3168E+03
F_{21}	Mean	1.9745E+07	2.2043E+06	9.5438E+06	2.0236E+06	7.3915E+06	**5.9326E+05**
	Std	5.5422E+07	8.4679E+06	2.0409E+06	1.6473E+06	6.2560E+06	**3.8180E+05**
F_{22}	Mean	2.6000E+04	5.4875E+03	5.8970E+03	4.9488E+03	4.2225E+03	**3.2219E+03**
	Std	6.2415E+04	3.2606E+02	2.4401E+02	2.8430E+02	4.4215E+02	**3.1869E+02**
F_{23}	Mean	2.9424E+03	2.5013E+03	2.6522E+03	2.7095E+03	2.5000E+03	**2.5000E+03**
	Std	8.4419E+02	9.8545E−01	3.5458E+00	2.0443E+01	0.0000E+00	**0.0000E+00**
F_{24}	Mean	2.6609E+03	2.6017E+03	2.6783E+03	2.6000E+03	2.6000E+03	**2.6000E+03**
	Std	6.6449E+00	6.8354E+00	3.7258E+00	1.0191E−03	3.0389E−04	**0.0000E+00**
F_{25}	Mean	2.7198E+03	2.7000E+03	2.7122E+03	2.7254E+03	2.7000E+03	**2.7000E+03**
	Std	8.9622E+00	2.0038E−02	2.0806E+00	5.3464E+00	0.0000E+00	**0.0000E+00**
F_{26}	Mean	2.8035E+03	2.7933E+03	2.7570E+03	2.7749E+03	2.7884E+03	**2.7007E+03**
	Std	5.1949E+01	2.3407E+01	7.2460E+01	5.8496E+01	3.2215E+01	**1.3947E−01**
F_{27}	Mean	4.7522E+03	4.4040E+03	3.5739E+03	3.6622E+03	2.9000E+03	**2.9000E+03**
	Std	9.4951E+02	1.6139E+02	1.1289E+02	9.7798E+01	1.8190E−13	**0.0000E+00**
F_{28}	Mean	1.6016E+04	7.3119E+03	4.4599E+03	4.6764E+03	3.0000E+03	**3.0000E+03**
	Std	4.8573E+03	1.2852E+03	4.4607E+02	4.7474E+02	6.5269E−13	**9.1854E−13**
F_{29}	Mean	1.7903E+09	6.9442E+07	1.2678E+06	7.0109E+05	3.1000E+03	**6.2979E+04**
	Std	1.0575E+09	1.1804E+08	8.5423E+06	1.4783E+06	0.0000E+00	**4.2762E+05**
F_{30}	Mean	2.5510E+07	2.9939E+04	3.5309E+04	1.1147E+05	8.0127E+04	**3.2000E+03**
	Std	2.2613E+07	7.2619E+03	9.8133E+03	5.2365E+04	2.8748E+05	**0.0000E+00**

由表3可知,MESSA在21个测试函数上得到了更好的结果,尤其是在函数F_{21}~F_{30}上具有显著的优势. 在F_3、F_{10}和F_{15}上PSO获得了最好的结果,在F_5上KH获得了最好的结果,在F_{16}上GWO获得了最好的结果,MVO在4个测试函数上具有最好的结果. 综上所述,相较于对比算法,本文算法在CEC 2014测试函数上具有显著优势.

5.4 策略有效性分析

将仅采用“自适应领导者结构”,“Logistic-Cubic级联混沌扰动”,“分散觅食策略”的SSA分别记为ASSA,LCSSA,DFSSA. 将基本SSA与上述三种改进的SSA进行实验比较. 如表4所示,其中最好的结果以粗体显示. 由表4可知,总体来讲,本文所提策略是有效的.

表4 单策略改进算法在CEC2014上的实验结果

函数	指标	算法			
		ASSA	LCSSA	DFSSA	SSA
F_1	Mean	1.2117E+07	**9.5394E+05**	1.0585E+06	6.7342E+07
	Std	1.9246E+06	**5.5326E+05**	3.7073E+07	3.1985E+07
F_3	Mean	2.2966E+04	1.2905E+04	**1.2703E+04**	4.7513E+04
	Std	4.2988E+04	3.2968E+04	**1.9043E+04**	9.0055E+03
F_5	Mean	**5.2004E+02**	5.2005E+02	5.2005E+02	5.2004E+02
	Std	**2.2515E−02**	4.6728E−01	6.7046E−02	4.4002E−02
F_7	Mean	7.0167E+02	**7.0055E+02**	7.0176E+02	7.2254E+02
	Std	5.6156E−01	**2.5517E−01**	8.4180E−01	2.8784E−01

续表

函数	指标	算法			
		ASSA	LCSSA	DFSSA	SSA
F_9	Mean	**9.2864E+02**	9.2950E+02	1.2139E+03	1.0265E+03
	Std	**1.8865E+01**	7.0767E+01	5.9073E+01	2.0253E+01
F_{11}	Mean	4.6392E+03	**3.6446E+03**	8.1386E+03	5.8952E+03
	Std	9.7721E+02	**1.3041E+02**	8.9497E+02	6.4520E+02
F_{13}	Mean	1.3006E+03	1.3006E+03	**1.3006E+03**	1.3006E+03
	Std	1.2055E−01	1.2245E−01	**1.0783E−01**	1.5610E−01
F_{15}	Mean	1.6776E+03	1.5574E+02	**1.5520E+02**	1.9854E+03
	Std	1.6434E+02	2.4204E+01	3.9183E+01	**1.7958E+01**
F_{17}	Mean	**7.0071E+05**	5.7704E+06	7.3964E+05	2.1021E+06
	Std	**4.2977E+05**	3.5843E+06	5.4822E+05	1.5462E+06
F_{19}	Mean	1.9672E+03	1.9832E+03	1.9638E+03	**1.5489E+03**
	Std	**2.0584E+01**	3.3110E+01	2.1975E+01	2.2546E+01
F_{21}	Mean	4.8414E+06	3.8067E+06	3.3582E+06	**5.5410E+05**
	Std	3.7150E+06	2.5087E+06	2.4070E+06	**3.8541E+05**
F_{23}	Mean	2.7491E+03	**2.7213E+03**	2.7365E+03	2.7845E+03
	Std	2.8815E+01	**2.4125E+01**	3.3235E+01	2.5641E+01
F_{25}	Mean	2.7509E+03	**2.7420E+03**	2.7448E+03	2.8952E+03
	Std	**1.0019E+01**	1.1362E+01	8.5205E+00	2.9654E+00
F_{27}	Mean	4.3251E+03	4.9872E+03	**4.2864E+03**	4.5561E+03
	Std	1.6358E+02	3.9224E+02	**1.3952E+02**	9.6321E+02
F_{29}	Mean	7.0152E+07	**6.7950E+06**	8.5196E+07	9.6235E+07
	Std	7.9861E+07	**5.2234E+06**	7.6038E+07	6.3215E+07

6　机器人路径规划仿真实验

6.1　实验设置

为了测试本文算法在解决机器人路径规划问题上的有效性,在三个测试环境上进行实验,且起点均设置为(0,0),终点均设置为(10,10),并与 SSA、CSSA 以及非启发式算法:人工势场法[20]、快速行进算法[21](Fast Marching Method,FMM)进行对比实验.

6.2　与群智能算法的比较

为了实验的公平性,均采用相同的初始化位置,维数均设置为 5,种群大小均设置为 30. 为了使实验结果可靠,所有实验均独立运行 30 次,记录其平均值、标准差、最优值、最劣值,其中最好结果我们用粗体标记,实验结果见表 5.

对上述实验结果在显著性水平 0.05 进行了 Wilcoxon 秩和检验,Wilcoxon 检验的 p 值记录在表 6 中. 当 MESSA 优于其对比算法时,使用“+”,使用“−”,“≈”分别表示劣于对比算法,无差异.

表 5　群智能算法的适应度值

测试环境	适应度值	SSA	CSSA	MESSA
1 (8 障碍物)	最好值	1.1832	4.5641	**1.0545**
	均值	2.2830	5.5412	**1.1637**
	标准差	0.8274	1.1542	**0.0120**
	最劣值	3.9686	6.3214	**1.1838**
2 (9 障碍物)	最好值	1.5689	4.9632	**1.2575**
	均值	1.9796	5.6542	**1.2851**
	标准差	1.5295	1.0154	**0.0134**
	最劣值	2.3245	7.3564	**1.3512**
3 (13 障碍物)	最好值	2.4569	5.6231	**1.6542**
	均值	2.8202	7.3245	**2.6546**
	标准差	1.5508	1.8907	**0.9698**
	最劣值	4.7506	9.6235	**2.9856**

从表 5 中看出,MESSA 在三个测试环境上得到适应度值都优于其两种对比算法. 从表 6 的非参数检验

结果可以看到,MESSA 得到的实验结果在统计学上具有显著性意义. 图 3 是获得的最优路径. 从 3 图中可以直观地看出 MESSA 在三个环境中都能以最小的代价找到最优路径. 综上,MESSA 在求解移动机器人路径规划问题中是有效的.

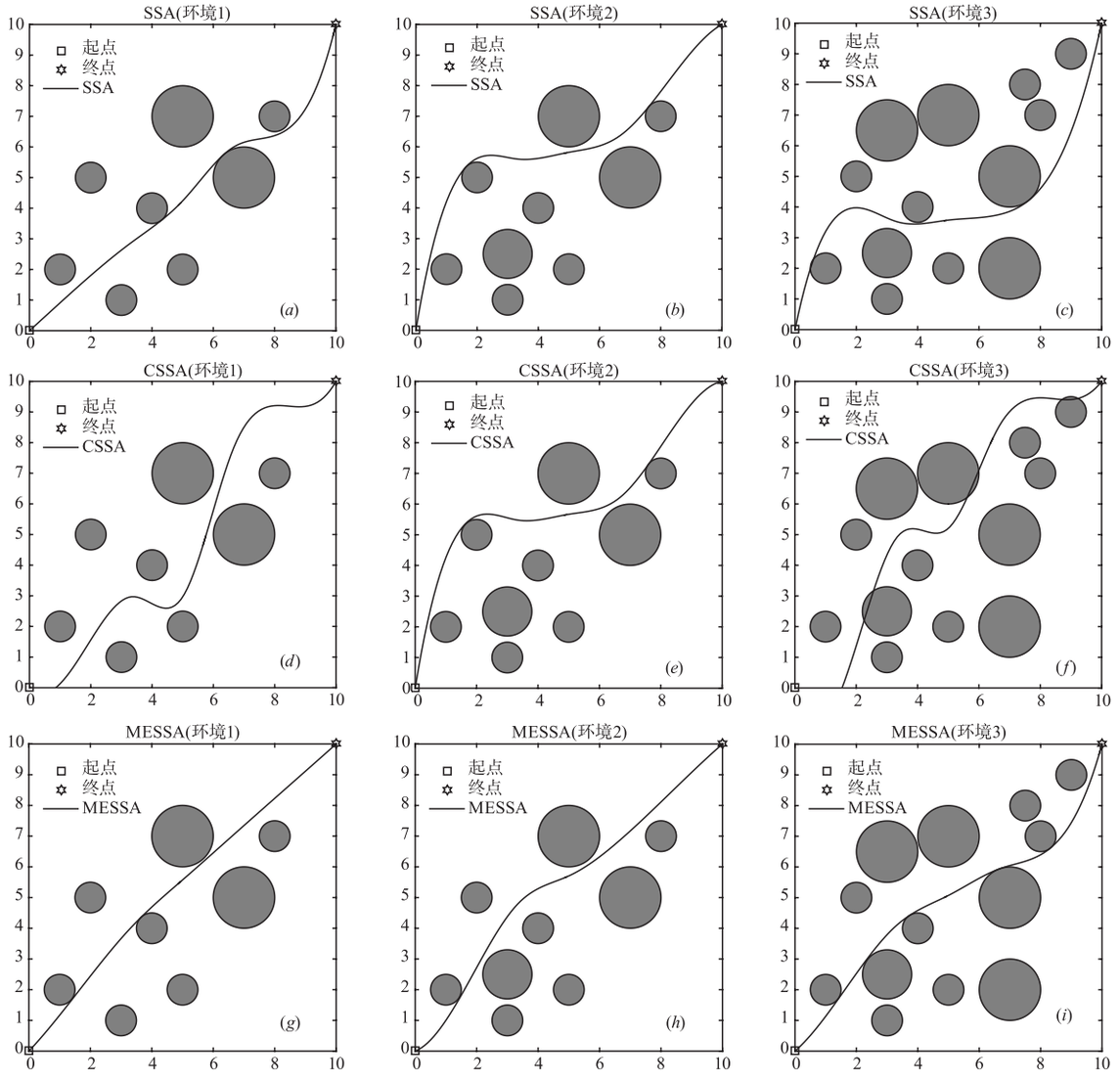

图3 群智能算法路径图

表 6 Wilcoxon 检验的 p 值

测试环境	MESSA vs.	SSA	CSSA
1 (8 障碍物)	P 值	4.0771e-11	1.2117e-12
	结果	+	+
2 (9 障碍物)	P 值	9.3645e-09	1.21178e-12
	结果	+	+
3 (13 障碍物)	P 值	3.4571e-10	1.2117e-12
	结果	+	+

6.3 与非启发式算法的比较

将 MESSA 与人工势场法和 FMM 进行对比实验. 人工势场法参数设置:引力势场常数为 14,斥力势场常数为 11,障碍物能够影响的最大距离为障碍物的半径,斥力影响系数为 0.5,局部极小值的检测半径为 0.2,移动的步长为 0.1. FMM 路径规划算法:将规划空间(M)进行栅格化(步长为 0.1,规划空间用二进制网格地图表示,其中无碰撞空间中的每个网格是 1,障碍物区域中的网格是 0);根据 M 和起点计算时间矩阵 T;在时间矩阵 T 上,应用梯度下降法从终点到起点搜索最优路径.

图 4 是两种非启发式算法在三个仿真环境上得到的最优路径,可以直观地看出,在前两个环境中,人工势场法和 FMM 都能找到安全路径,但是在障碍物较多

的环境 3,FMM 亦然能找到安全路径,而人工势场法没有找到安全路径. 由于场景 3 较为复杂,因此容易出现引力和斥力的合力为 0 的情况,所以该算法陷入了局部极值,出现了不能到达目标点的情况.

最后,表 7 给出了几种算法在三个仿真环境下获得最优路径的长度和时间情况,其中"—"表示算法没有找到安全路径. 从起点到终点理论上最短的无障碍路径长度是 14. 1421.

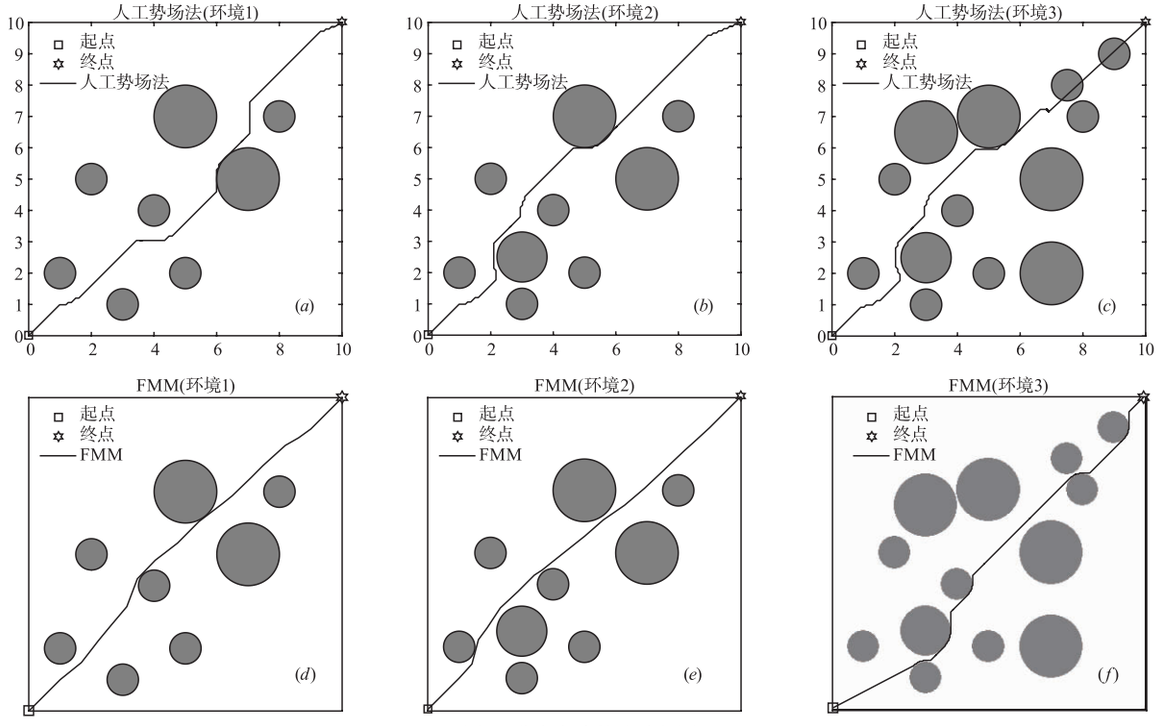

图4　非启发式算法路径图

表 7　路径长度和时间

环境	指标	SSA	CSSA	人工势场法	FMM	MESSA
1 (8 障碍物)	长度	15. 6548	19. 0125	16. 8644	14. 8525	**14. 6512**
	时间(s)	1. 043	1. 116	**0. 956**	3. 893	1. 325
2 (9 障碍物)	长度	16. 6352	20. 9065	16. 9855	**14. 8945**	15. 0032
	时间(s)	1. 564	1. 506	**1. 025**	4. 645	1. 894
3 (13 障碍物)	长度	17. 7585	—	—	**15. 0254**	15. 1522
	时间(s)	**2. 046**	—	—	5. 056	2. 652

由表 7 可见,在时间消耗方面,FMM 最多,人工势场法耗时最少. 但是,人工势场法在环境 1 和 2 中与 MESSA 和 FMM 相比获得的路径最长,且在环境 3 中没有找到安全路径. 在路径长度方面,MESSA 在环境 1 上获得了最短的安全路径,在环境 2 和 3 上比 FMM 算法略逊. 但是,在 3 个环境上,MESSA 在时间方面明显较 FMM 有优势.

综合上述两个指标,相较于对比算法,MESSA 性能更佳.

7　结论

本文提出了一种多策略集成的改进樽海鞘群算法. 在 MESSA 中,自适应领导者结构增强了算法的开发能力;Logistic-Cubic 级联混沌扰动在很大程度上保持了种群多样性;分散觅食策略提高了算法的搜索效率. MESSA 不仅保留了 SSA 的局部搜索能力强、算法简单、易于实现的优点,而且充分利用了各搜索策略的特点,具有更好的全局和局部搜索能力. MESSA 与 SSA,三种改进的 SSA 算法和其他 5 种群智能算法在 CEC 2014 基准测试函数上进行了仿真对比实验,实验结果和性能比分布曲线表明:MESSA 显著提高了解的精度和稳定性. 此外,利用移动机器人路径规划问题来证明 MESSA 对现实世界应用的有效性. 在简单和复杂障碍环境下与 SSA、CSSA 以及非启发式算法:人工势场法、FMM

进行对比实验,结果表明,相比于对比算法,本文算法性能更佳.

参考文献

[1] 薛裕颖,张祥银,张国梁,等. 基于量子行为烟花算法的移动机器人路径规划及平滑[J]. 控制理论与应用,2019,36(9):1398 - 1408.
XUE Yu-ying, ZHANG Xiang-yin, ZHANG Guo-liang, et al. Path planning and smoothing based on quantum-behaved fireworks algorithm for mobile robot[J]. Control Theory & Applications, 2019, 36(9): 1398 - 1408. (in Chinese)

[2] 许凯波,鲁海燕,黄洋,等. 基于双层蚁群算法和动态环境的机器人路径规划方法[J]. 电子学报,2019,47(10):2166 - 2176.
XU Kai-bo, LU Hai-yan, HUANG Yang, et al. Robot path planning based on double-layer ant colony optimization algorithm and dynamic environment[J]. Acta Electronica Sinica, 2019, 47(10): 2166 - 2176. (in Chinese)

[3] 尹高扬,周绍磊,吴青坡. 基于改进 RRT 算法的无人机航迹规划[J]. 电子学报,2017,45(7):1764 - 1769.
YIN Gao-yang, ZHOU Shao-lei, WU Qing-po. An improved RRT algorithm for UAV path planning[J]. Acta Electronica Sinica, 2017, 45(7): 1764 - 1769. (in Chinese)

[4] 刘景森,吉宏远,李煜. 基于改进蝙蝠算法和三次样条插值的机器人路径规划[J/OL]. 自动化学报, http://kns.cnki.net/kcms/detail/11.2109.TP.20190325.1534.003.html. 2019-03-25.
LIU Jing-Sen, JI Hong-Yuan, LI Yu. Robot path planning based on improved bat algorithm and cubic spline interpolation[J/OL]. Acta Automatica Sinica, http://kns.cnki.net/kcms/detail/11.2109.TP.20190325.1534.003.html. 2019-03-25. (in Chinese)

[5] 贾会群,魏仲慧,何昕,等. 基于改进粒子群算法的路径规划[J]. 农业机械学报,2018,49(12):371 - 377.
JIA Huiqun, WEI Zhonghui, HE Xin, et al. Path planning based on improved particle swarm optimization algorithm [J]. Transactions of the Chinese Society for Agricultural Machinery, 2018, 49(12): 371 - 377. (in Chinese)

[6] Mirjalili S, Gandomi A H, Mirjalili S Z, et al. Salp swarm algorithm: A bio-inspired optimizer for engineering design problems[J]. Advances in Engineering Software, 2017, 114(6): 163 - 191.

[7] Faris H, Mafarja M M, Heidari A A, et al. An efficient binary salp swarm algorithm with crossover scheme for feature selection problems [J]. Knowledge-Based Systems, 2018, 154: 43 - 67.

[8] Sayed G I, Khoriba G, Haggag M H. A novel chaotic salp swarm algorithm for global optimization and feature selection [J]. Applied Intelligence, 2018, 48(10): 3462 - 3481.

[9] 杨博,钟林恩,朱德娜,等. 部分遮蔽下改进樽海鞘群算法的光伏系统最大功率跟踪[J]. 控制理论与应用,2019,36(3):339 - 352.
YANG Bo, ZHONG Lin-en, ZHU De-na, et al. Modified salp swarm algorithm based maximum power point tracking of power-voltage system under partial shading condition [J]. Control Theory & Applications, 2019, 36(3): 339 - 352. (in Chinese)

[10] Mohammed H Qais, Hany M Hasanien, Saad Alghuwainem. Enhanced salp swarm algorithm: Application to variable speed wind generators[J]. Engineering Applications of Artificial Intelligence, 2018, 80: 82 - 96.

[11] Ibrahim Aljarah, Majdi Mafarja, Ali Asghar Heidari, et al. Asynchroous accelerating multi-leader salp chains for feature selection[J]. Applied Soft Computing, 2018, 71: 964 - 979.

[12] 王光义,袁方. 级联混沌及其动力学特性研究[J]. 物理学报,2013,62(2):020506.
Wang Guang-Yi, Yuan Fang. Cascade chaos and its dynamic characteristics [J]. Acta Physica Sinica, 2013, 62(2): 020506. (in Chinese)

[13] Tu Q, Chen X, Liu X. Multi-strategy ensemble grey wolf optimizer and its application to feature selection [J]. Applied Soft Computing, 2019, 76: 16 - 30.

[14] Liang J J, Qu B Y, Suganthan P N. Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization[R]. Zhengzhou: Zhengzhou University, 2013.

[15] 张达敏,陈忠云,辛梓芸,等. 基于疯狂自适应的樽海鞘群算法[J]. 控制与决策,2020,35(9):2112 - 2120.
ZHANG Da-min, CHEN Zhong-yun, XIN Zi-yun, et al. Salp swarm algorithm based on craziness and adaptive [J]. Control and Decision, 2020, 35(9): 2112 - 2120. (in Chinese)

[16] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer [J]. Advances in Engineering Software, 2014, 69: 46 - 61.

[17] Gandomi A H, Alavi A H. Krill herd: A new bio-inspired optimization algorithm[J]. Communications in Nonlinear Science and Numerical Simulation, 2012, 17(12): 4831 - 4845.

[18] Mirjalili S, Abdolreza H. Multi-verse optimizer: A nature-inspired algorithm for global optimization [J]. Neural Computing and Applications, 2016, 27(2): 495 - 513.

[19] Heidari A A, Mirjalili S, Faris H, et al. Harris hawks optimization: Algorithm and applications[J]. Future Generation Computer Systems, 2019, 97: 849 - 872.

[20] Zhang Q, Chen D, Chen T. An obstacle avoidance method of soccer robot based on evolutionary artificial potential

field[J]. Energy Procedia,2012,16:1792 – 1798.

[21] Liu Yuanchang, Bucknall Richard. The angle guidance path planning algorithms for unmanned surface vehicle formations by using the fast marching method[J]. Applied Ocean Research,2016,59:327 – 344.

作者简介

王秋萍(通讯作者)　女,1964 年出生于陕西白水,教授,博士,硕士生导师,主要研究方向为:智能计算及其应用,灰色系统理论,预测技术与决策分析.

E-mail:wqp566@ sina. com

王彦军　男,1993 年 3 月出生于甘肃临洮,硕士研究生,研究方向为智能计算及在机器人路径规划中的应用.

E-mail:1725917163@ qq. com

戴　芳　女,1966 年出生,陕西蓝田人,教授,博士,硕士生导师,主要研究方向为图像处理与计算机视觉.

E-mail:daifang@ xaut. edu. cn