

引用格式:张伟康,刘升. 自适应 t 分布与黄金正弦改进的麻雀搜索算法及其应用[J]. 微电子学与计算机, 2022, 39(3): 17-24.
[ZHANG W K, LIU S. Improved sparrow search algorithm based on adaptive t-distribution and golden sine and its application
[J]. Microelectronics & Computer, 2022, 39(3): 17-24.]DOI: 10. 19304/J. ISSN1000-7180. 2021. 0026

自适应 t 分布与黄金正弦改进的麻雀搜索算法及其应用

张伟康, 刘 升

(上海工程技术大学 管理学院, 上海 201620)

摘 要: 针对麻雀搜索算法存在的容易陷入局部最优、收敛速度慢等问题, 提出一种基于自适应 t 分布与黄金正弦改进的麻雀搜索算法(t-GSSA). 首先, 通过黄金正弦算法改进发现者的位置更新方式, 增强算法局部开发和全局探索能力, 并且提高算法的收敛能力; 其次, 利用自适应 t 分布变异方法, 对个体位置进行扰动, 提升算法跳出局部最优的能力; 然后, 在仿真实验中与 4 种基本算法和 3 种改进算法基于 10 个基准测试函数进行比较, 结果表明所提算法具有更好的收敛性和求解精度; 最后, 将 t-GSSA 算法应用到比例-积分-微分(PID)参数整定中. 仿真结果表明, 所提算法优化后的控制器具有更快的响应速度的更稳定的精度.

关键词: 麻雀搜索算法; 黄金正弦算法; 自适应 t 分布; PID 参数整定

中图分类号: TP301

文献标识码: A

文章编号: 1000-7180(2022)03-0017-08

Improved sparrow search algorithm based on adaptive t-distribution and golden sine and its application

ZHANG Weikang, LIU Sheng

(School of Management, Shanghai University of Engineering Science, Shanghai 201620, China)

Abstract: Aiming at the shortcomings such as easy to fall into local optimization and low convergence, an improved Sparrow Search Algorithm based on adaptive t-distribution and gold sine improvement is proposed. Firstly, the golden sine algorithm is used to improve the location update mode of the finder, enhance the local development and global exploration ability of the algorithm, and improve the convergence ability of the algorithm. Secondly, the t-GSSA algorithm is used to disturb the individual position to improve the algorithm's ability to jump out of local optimality. And then, compared with 4 basic algorithms and 3 improved algorithms based on 10 benchmark functions, the simulation results show that the proposed algorithm has better convergence and solution accuracy. Finally, the t-GSSA algorithm is applied to the proportional-integral-score (PID) parameter integration. The simulation results show that the controller that optimized by the proposed algorithm has more stable accuracy with faster response speed.

Key words: sparrow search algorithm; golden sine algorithm; adaptive t distribution; PID parameter setting

1 引言

麻雀搜索算法(Sparrow Search Algorithm, SSA)是由薛建凯^[1]等人提出的一种模拟麻雀觅食

行为和反捕食行为的新型群智能算法. 研究发现该算法的寻优精度、收敛速度以及鲁棒性等方面均优于灰狼优化算法(GWO)、引力搜索算法(GSA)和粒子群算法(PSO). 然而基本的麻雀搜索

算法中在搜索后期也存在种群多样性减少,容易陷入局部最优等缺陷^[2]. 针对这个缺陷,吕鑫^[2-3]等人提出了两种改进后的麻雀搜索算法——其一,通过 Tent 混沌序列提高麻雀种群初始解的质量并对陷入局部最优的个体进行扰动,然后引入高斯变异策略,加强局部搜索能力,形成混沌麻雀搜索算法;其二,融合鸟群算法中个体飞行的特点和思想,改善麻雀个体的飞行方式,并通过测试函数及在多阈值图像分割中的应用验证了改进算法的优秀性能. 毛清华^[4]等人以正余弦算法的更新方式替代麻雀搜索算法中发现者的更新方式,然后在跟随者更新部分采用 Lévy 飞行策略对最优个体进行扰动,并采用了一种非线性的动态学习因子对探索和开发能力进行平衡.

以上学者从多种角度及多种策略探究并验证了如何提高 SSA 的寻优能力,但目前而言 SSA 的性能仍有较大提升空间,探索其他有效的改进策略,拓展其应用范围仍需更加深入的研究. 本文受到 Erkan Tanyildizi^[5]提出的黄金正弦算法 (Golden-SA) 以及肖子雅^[6]等人将 Golden-SA 应用到改进鲸鱼优化算法的启发,利用 Golden-SA 位置更新方式改进麻雀搜索算法发现者的位置更新方式,提升算法全局开发能力. 然后引入自适应 t 分布和动态选择概率,进一步协调局部搜索和全局搜索的比重. 通过与 3 种基本算法和 3 种改进算法基于 10 个基准测试函数进行比较以验证其有效性,并将其用于 PID 参数的整定. 结果表明所提算法在求解函数优化问题上表现优秀,并能有效提升控制器的响应速度的精度.

2 麻雀搜索算法

麻雀搜索算法的灵感来源于麻雀在群居生活中的分工与协作,根据不同麻雀在觅食和反捕食过程中的分工,将其具体分为发现者、加入者和侦察者. 其中发现者负责为整个种群寻找食物,指引种群的觅食方向,相较于加入者其具有更大的搜索范围;其数量一般占到种群的 10%~20%,主要由每次迭代中位置最好的部分麻雀组成,其具体位置更新公式如下:

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

式中, $j = 1, 2, 3, \dots, d$, $\text{iter}_{\max}$ 表示算法的最大迭代次

数, t 表示目前迭代次数, α 是 $(0, 1]$ 之间的随机数, 预警值和安全值的范围分别为 $R_2 \in [0, 1]$ 和 $ST \in [0.5, 1]$, Q 是服从正态分布的随机数, L 为 $1 \times d$ 的矩阵, 其元素值全为 1.

发现者确定以后, 其余个体即为加入者, 其与发现者的数量比重是不变的, 且其寻优方向受发现者的影响, 具体位置更新公式如下:

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

式中, X_{worst} 表示第 t 次迭代时适应度值最差的个体所处的位置, X_p^{t+1} 表示第 $t+1$ 次迭代适应度值最优的发现者所处的位置, A 表示 $1 \times d$ 的矩阵, 其中元素随机赋值为 1 或 -1 , 且 $A^+ = A^T (AA^T)^{-1}$.

侦察者是麻雀搜索算法中较为特殊的一部分, 表示种群中意识到危险的个体, 一般占种群总数的 10%~20%, 但这一部分由发现者和加入者共同承担,

其位置更新公式如下:

$$X_{i,j}^{t+1} = \begin{cases} X_{\text{best}}^t + \beta \cdot |X_{t,i,j} - X_{\text{best}}^{t+1}|, & f_i > f_g \\ X_{t,i,j}^t + K \cdot \left(\frac{|X_{t,i,j}^t - X_{\text{worst}}^t|}{(f_i - f_w) + \epsilon}\right), & f_i = f_g \end{cases} \quad (3)$$

式中, X_{best}^t 表示当前具有最游走适应度值的个体所处的位置, β 是均值为 0、方差为 1 的正态分布随机数, 表示步长控制系数, $K \in [-1, 1]$ 是一个随机数, f_i 为当前个体适应度值, f_g 和 f_w 分别表示当前最优个体适应度值和最差个体适应度值, ϵ 为最小常数, 是为了防止分母为零的情况出现.

3 麻雀搜索算法的改进

3.1 黄金正弦策略

Golden-SA 是受到正弦函数的启发, 利用数学中的正弦函数进行迭代寻优, 具有较强的全局搜索能力. 同时在位置更新过程中引入黄金分割系数来使算法能够对局部区域进行充分搜索, 使全局搜索和局部开发能力达到良好的平衡. Golden-SA 的核心是其位置更新公式, 首先随机产生 s 个个体, 第 i ($i = 1, 2, 3, \dots, s$) 个个体第 t 次迭代中在 d 维解空间的位置为 $X_i^t = (X_{i1}, X_{i2}, \dots, X_{id})$, $P_i^t = (P_{i1}, P_{i2}, \dots, P_{id})$ 表示第 t 代个体 i 的最优位置. 然后利用公式(4)对每个个体进行更新:

$$X_i^{t+1} = X_i^t \times |\sin(R_1)| + R_2 \times \sin(R_1) \times |x_1 \times P_i^t - x_2 \times X_i^t| \quad (4)$$

式中, $R_1 \in [0, 2\pi]$ 是决定下次迭代中个体的移动距离的一个随机数, $R_2 \in [0, \pi]$ 是决定下次移动方向的随机数. $x_1 = -\pi + (1 - \tau) * 2\pi$, $x_2 = -\pi + \tau * 2\pi$ 是根据黄金分割系数 $\tau (\tau = (\sqrt{5} - 1)/2)$ 得到的系数, 这些系数缩小了算法的搜索空间, 引领个体逐渐去向最优值.

根据 SSA 算法发现者位置更新公式可知, 当 $R_2 < ST$ 时, 发现者的每一维都在变小, 当 $R_2 \geq ST$ 时, 发现者会按照正态分布随机移动到当前位置. 这样一来算法在迭代初就有向全局最优解靠近的趋势, 容易导致算法出现早熟收敛, 陷入局部最优. 因此, 本文引入 Golden-SA 改进 SSA 中发现者的位置更新公式, 改进后的位置更新方式如公式(5)所示:

$$X_{i,j}^{t+1} = \begin{cases} X_{ij}^t \times |\sin(R_1)| + R_2 \times \sin(R_1) \times |x_1 \times X_{best}^t - x_2 \times X_{ij}^t|, & R_2 < ST \\ X_{i,j}^t + Q \cdot L, & R_2 \geq ST \end{cases} \quad (5)$$

改进后的位置更新公式中, 一方面, 在每一次迭代时麻雀个体都会与最优个体进行信息交流, 以便充分利用当前最优解的信息, 改善了原算法中缺乏个体间信息交流的缺陷; 另一方面, 针对原算法容易陷入局部最优的问题, Golden-SA 的引入可以在一定程度上扩大搜索空间, 从而提升算法的全局寻优能力; 同时, 黄金分割系数逐步缩小了搜索空间, 并通过 R_1 和 R_2 控制麻雀的搜索距离和方向, 优化了 SSA 算法的寻优方式, 进一步协调了算法的全局探索和局部开发能力.

3.2 自适应 t 分布和动态选择策略

t 标准高斯分布和柯西分布是 t 分布的两个边界特例分布^[7-8], 三者的函数图像如图 1 所示. 本文采用以迭代次数 $iter$ 为 t 分布的自由度参数的 t 分布变异算子对麻雀位置进行扰动, 使得算法在迭代前期具有较好的全局开发能力, 在迭代后期具有良好的局部探索能力, 并提高算法的收敛速度, 具体的位置更新方式如下:

$$X_i^{t+1} = X_i^t + X_i^t \cdot t(iter) \quad (6)$$

式中, X_i^{t+1} 为扰动后麻雀的位置, X_i^t 为麻雀 i 第 t 次迭代时位置. 所提更新公式在 X_i^t 的基础上, 增加了随机干扰项 $X_i^t \cdot t(iter)$, 既充分利用了当前位置信息, 又增加了随机干扰信息, 有利于算法跳出局部最优. 而随着迭代次数 $iter$ 增加, t 分布逐渐向高斯分

布靠拢, 有利于增强算法的收敛速度.

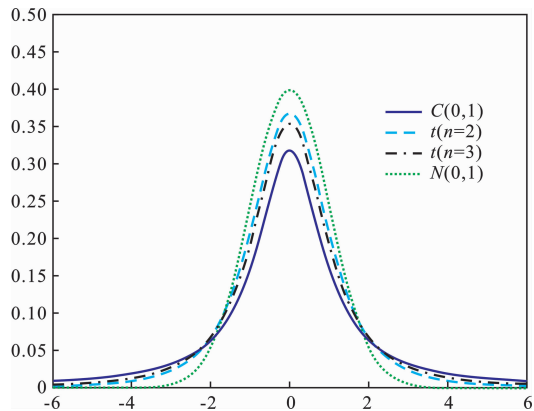


图1 柯西分布、 t 分布和高斯分布概率密度图

Fig. 1 Probability density maps of Cauchy distribution, t distribution and Gaussian distribution

自适应 t 分布变异算子的引入, 虽然能够很大程度上提升算法的寻优性能, 但是若无差别的对每次迭代中的所有个体使用, 一方面会增加算法的计算时间, 另一方面不利于发挥原算法本身的特点. 因此, 本文采用动态选择概率 p 来调节自适应 t 分布变异算子的使用. 其计算公式如下:

$$p = w_1 - w_2 \times (\maxiter - iter) / \maxiter \quad (7)$$

式中, $\maxiter$ 为最大迭代次数, $iter$ 为当前迭代次数, w_1 决定了动态选择概率的上限, w_2 决定了动态选择概率的变化幅度, 本文经过反复试验发现当 $w_1 = 0.5$, $w_2 = 0.1$ 时, 调节作用最优. 动态选择概率 p 使得算法在迭代前期有较大概率于利用自适应 t 分布变异算子对麻雀的位置进行扰动, 改善原算法在迭代初期就存在收敛于最优解得倾向; 同时在迭代后期, 充分发挥原算法良好的局部开发能力, 并以较小概率的 t 分布变异作为补充, 提升算法的收敛速度.

3.3 改进算法的流程图

t -GSSA 的具体执行流程如图 2 所示.

4 仿真实验与结果分析

4.1 仿真实验环境

为保证实验的严谨和公平, 本文的仿真实验均是基于同一实验环境, 运用 MATLAB R2018b 作为仿真软件, 操作系统为 Microsoft Win10, 硬件配置为 Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90GHz, 12 GB 内存.

4.2 比较对象和参数设置

本文选取 7 种算法作为对比对象, 包括经典和

表 2 基准测试函数

Tab. 2 Benchmark function

类型	函数	维数	范围	最优值
f_1	Sphere	高维 单峰	$[-100,100]^{30}$	0
f_2	Schwefel2.22		$[-10,10]^{30}$	0
f_3	Schwefel1.2		$[-100,100]^{30}$	0
f_4	Schwefel2.21		$[-100,100]^{30}$	0
f_5	Quartic	高维 多峰	$[-1.28,1.28]^{30}$	0
f_6	Rastrigin		$[-5.12,5.12]^{30}$	0
f_7	Ackley		$[-32,32]^{30}$	0
f_8	Griewank		$[-600,600]^{30}$	0
f_9	Kowalik	低维	$[-5,5]^4$	0.000 3
f_{10}	Shekel	多峰	$[0,10]^4$	-10.153 2

4.4 实验结果与分析

本文将选取的 7 个对比算法和所提的 t-GSSA，分别在 10 个基本测试函数上独立运行 30 次，具体结果如表 3 所示。

由表 3 可知，在高维单峰函数的测试结果中，本文所提出的 t-GSSA 的寻优效果较 7 个对比算法具有明显的优势；且 t-GSSA 的标准差远远小于其他算法，说明其具有较高的稳定性；在高维多峰函数的实验中，t-GSSA 与 SSA、ISSA、CWBOA、以及 EGolden-SWOA 的寻优性能不相上下，均可以搜索到最优值，且寻优效果达到了 100%，相比之下基本的 PSO、BOA 和 WOA 算法的寻优效果则略微逊色；在低维多峰函数的实验中，t-GSSA 在 f_9 函数上的寻优性能和稳定性相比于其他算法有一定程度的提高；在 f_{10} 函数上 t-GSSA 和 ISSA 都可以直接搜索到最优解，但 t-GSSA 的标准差远小于 ISSA，也即 t-GSSA 具有更强的稳定性，且明显优于其他算法。

收敛曲线能够直观的对比较算法的收敛精度和收敛速度，以及算法跳出局部最优的能力。本文根据迭代次数和适应度值绘制了测试函数的收敛曲线图，具体如图 3 所示。根据收敛曲线可知，本文所提算法 t-GSSA 在高维单峰函数 f_1 和 f_4 上的收敛曲线位于其他算法收敛曲线的下方，表明其收敛精度优于其他对比算法；此外，t-GSSA 收敛曲线的坡度明显大于其他所有算法，表明其具有更快的收敛速度。对于高维多峰函数 f_8 而言，t-GSSA 与 SSA、WOA、ISSA、CWBOA、EGolden-SWOA 均在一定的迭代次数后即找到了全局最优值，但值得注意的是，t-GSSA 在 20 次迭代以内便达到了全局最优，因此其具有更快的收敛速度；这表明，SSA 本身具有较高的全部探索能力，甚至与改进后的其他算法保持在同一水平，但较强的全部探索能力使得 SSA 丧失了

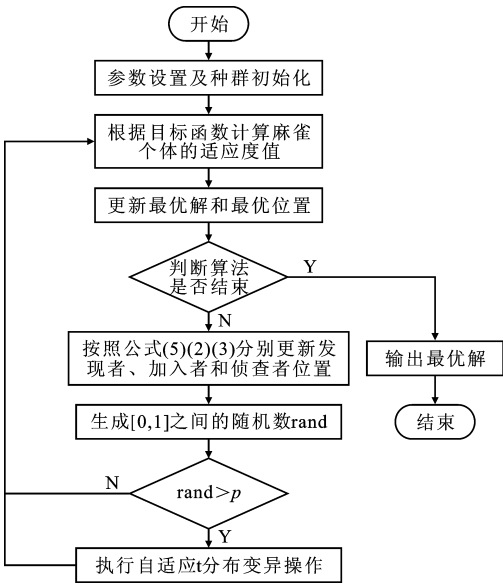


图 2 t-GSSA 的算法流程图

Fig. 2 Flow chart of t-GSSA

新型的基本算法——基本粒子群算法 (PSO)、基本麻雀算法 (SSA)，基本蝴蝶算法 (BOA) 以及基本鲸鱼算法 (WOA)；改进算法——文献 [3] 基于鸟群算法改进的麻雀搜索算法 (ISSA)，文献 [9] 柯西变异和自适应权重优化的蝴蝶算法 (CWBOA) 以及文献 [6] 精英反向黄金正弦鲸鱼算法 (EGolden-SWOA)。各算法的共同参数包括群体规模和最大迭代次数，分别设置为 30 和 200，其他特殊参数设置遵照原文献，具体如表 1 所示。

表 1 各算法实验参数设置

Tab. 1 Experimental parameter setting of each algorithm

算法	参数
SSA	$ST=0.8, PD=0.2, SD=0.2$
ISSA	$ST=0.8, PD=0.2, SD=0.2, \theta=0.05$
t-GSSA	$ST=0.8, PD=0.2, SD=0.2$
PSO	$w=1, w_{damp}=0.99, c_1=1.5, c_2=2, c=0.01, p=0.8$
BOA	a 从 0.1 迭代到 0.3
CWBOA	$c=0.01, a \in [0.10, 3]$
WOA	$b=1$
EGolden-SWOA	$b=1$

4.3 测试函数

经典测试函数是目前检测算法性能应用最为广泛的方式之一，主要包括高维单峰、高维多峰以及低维多峰三种类型，本文选取了 10 个不同特点的基准测试函数，用于检验改进算法的寻优性能，具体函数信息如表 2。

表 3 函数测试结果
Tab. 3 Test results of function

统计量	算法	f_1	f_2	f_3	f_4	f_5
平均值	SSA	1.123E-44	2.021E-30	1.012E-41	5.281E-30	1.086E-3
	ISSA	3.138E-119	1.242E-59	4.441E-109	5.215E-58	6.091E-4
	t-GSSA	2.935E-222	2.191E-102	8.031E-158	1.250E-109	4.389E-4
	PSO	1.897E-2	1.403E-1	6.706E+2	4.788E+00	4.605E-2
	BOA	5.431E-08	6.660E-06	5.350E-08	1.127E-05	3.478E-3
	CWBOA	4.388E-116	2.103E-48	9.459E-118	4.245E-51	3.857E-4
	WOA	3.159E-27	1.602E-19	7.945E+4	6.061E+1	9.195E-3
	EGolden-SWOA	3.258E-131	1.625E-65	7.782E-126	1.320E-68	2.029E-4
标准差	SSA	6.151E-44	1.069E-29	5.460E-41	2.886E-29	1.000E-3
	ISSA	1.507E-118	4.179E-59	2.427E-108	2.077E-57	4.925E-4
	t-GSSA	0	1.176E-101	4.399E-157	6.848E-109	3.601E-46
	PSO	6.263E-2	1.272E-1	4.379E+2	1.374	1.766E-2
	BOA	3.723E-09	1.318E-06	4.134E-09	9.293E-07	1.323E-3
	CWBOA	2.368E-115	7.430E-48	2.940E-117	8.493E-51	3.241E-4
	WOA	1.376E-26	3.437E-19	1.668E+4	2.698E+1	7.496E-3
	EGolden-SWOA	1.784E-130	8.890E-65	4.262E-125	7.228E-68	1.921E-4
平均值		f_6	f_7	f_8	f_9	f_{10}
	SSA	0	8.882E-16	0	3.127E-4	-7.904
	ISSA	0	8.882E-16	0	3.253E-4	-10.1532
	t-GSSA	0	8.882E-16	0	2.124E-4	-10.1532
	PSO	5.443E+1	1.666	0.13	5.340E-4	-6.071
	BOA	1.751E+1	9.912E-06	7.784E-08	7.059E-4	-3.807
	CWBOA	0	8.882E-16	0	6.029E-4	-4.869
	WOA	1.686E-13	2.564E-14	3.053E-2	7.952E-4	-6.283
标准差	EGolden-SWOA	0	8.882E-16	0	3.695E-4	-10.153
	SSA	0	0	0	1.552E-05	2.543
	ISSA	0	0	0	8.322E-05	1.960E-4
	t-GSSA	0	0	0	1.020E-05	5.560E-07
	PSO	1.269E+1	7.105E-1	3.267E-2	3.069E-4	3.681
	BOA	5.247E+1	6.887E-07	1.280E-08	5.679E-4	6.277E-1
	CWBOA	0	0	0	2.751E-4	9.225E-1
	WOA	8.172E-13	1.869E-14	1.172E-1	5.239E-4	2.401
	EGolden-SWOA	0	0	0	7.699E-05	9.323E-05

一定的收敛速度,而改进后的 t-GSSA 弥补了这一缺陷.对于函数 f_{10} ,所有算法的收敛曲线最终都趋于平稳,但是 t-GSSA、ISSA 和 EGolden-SWOA 具有更高的收敛精度;t-GSSA 的收敛精度和收敛速度与 ISSA 和 EGolden-SWOA 维持在同一数量级,但相较 SSA 有一定程度的提高.

5 改进算法在 PID 参数整定中的应用

5.1 PID 控制的基本原理

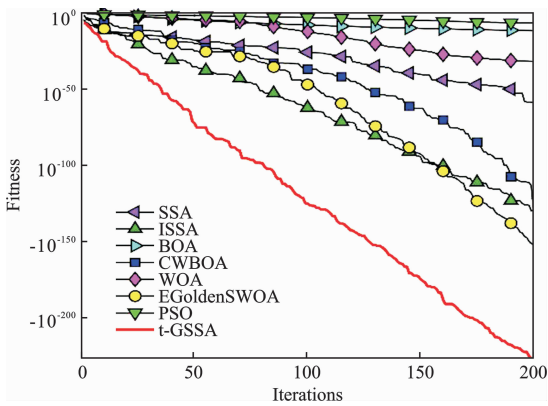
PID 控制器(比例-积分-微分控制器)是工业过程控制领域常用的控制系统.其具体系统结构如图

4 所示.

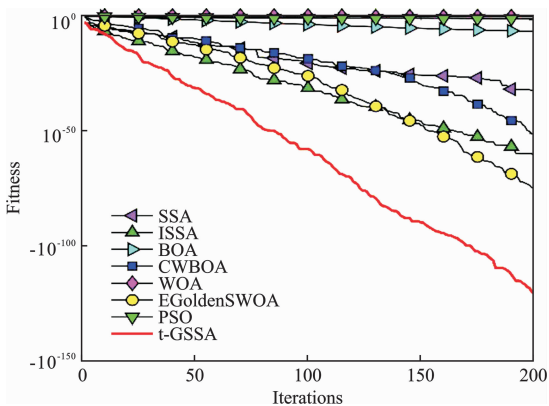
第一部分为 $r(t)$,代表控制系统的目标信号;第二部分控制变量公式部分,在整个系统中起到关键作用,具体形式如公式(8)所示;第三部分为具体问题中的被控对象,由 $u(t)$ 表示;第四部分为最终控制系统得到的响应结果.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (8)$$

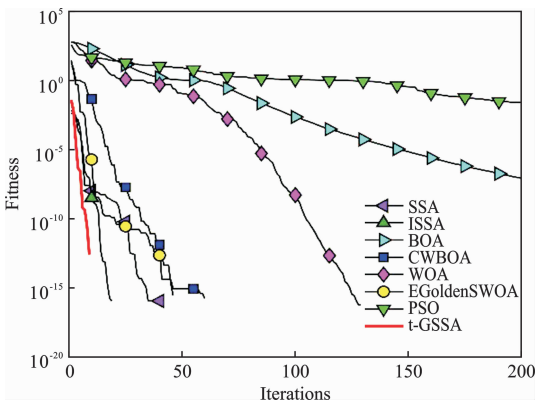
式中, $e(t)$ 是系统误差, K_p 、 K_i 和 K_d 分别为比例增益系数、积分增益系数和微分增益系数.



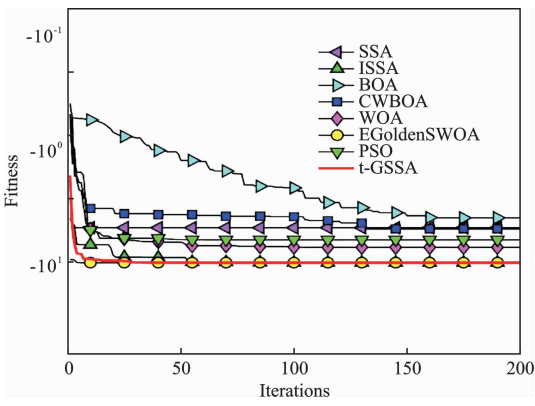
f_1 收敛曲线



f_4 收敛曲线



f_8 收敛曲线



f_{10} 收敛曲线

图 3 测试函数收敛曲线
Fig. 3 Convergence curve of test function

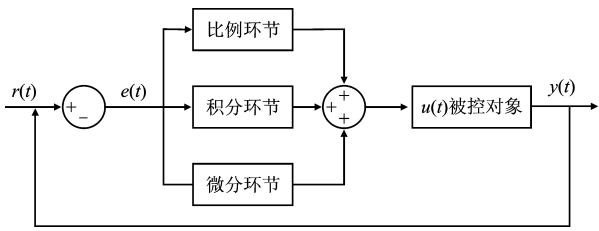


图 4 PID 控制器原理
Fig. 4 Principle of PID

5.2 被控对象和性能指标

本文选取的被控对象为以下不稳定系统:

$$G(s) = \frac{s + 2}{s^4 + 8s^3 + 4s^2 - s + 0.4} \quad (9)$$

为了评价系统优化过程中的性能,为群智能算法提供寻优方向,本文选取时间绝对偏差积分(Integral Time Absolute Error, ITAE)作为评价指标,其表达式如公式(10)所示.

$$ITAE = \int_0^t |e(t)| dt \quad (10)$$

5.3 仿真条件设置

本文将 t -GSSA 与 SSA 以及目前 PID 参数整定中最常用的 PSO 算法进行仿真实验,以验证所提算法解决实际问题的能力. 为保证实验的公平性,本文将以上三个算法的种群均设置为 50,迭代次数为 100 次;其中 SSA 和 t -GSSA 中的相关参数分别设置为 $ST = 0.8, PD = 0.2, SD = 0.2$;PSO 算法中的相关参数分别设置为 $w = 1, w_{damp} = 0.99, c_1 = 1.5, c_2 = 2$. 对于 PID 系统,以 0.1s 为采样间隔,单位阶跃信号为仿真的输入信号,整体仿真时间为 10 s.

5.4 仿真结果与分析

利用 MATLAB 分别对 t -GSSA、SSA 以及 PSO 算法优化 PID 参数的过程进行仿真,得到的具体结果如表 4、图 5 及图 6 所示.

表 4 三种算法的动态响应性能

Tab. 4 Dynamic response performance of three algorithms

算法	上升时间/s	调节时间/s	超调量/%	ITAE
PSO	2.347	4.161	7.613	1.205
SSA	2.309	3.822	6.336	1.035
t-GSSA	2.205	3.241	5.354	1.027

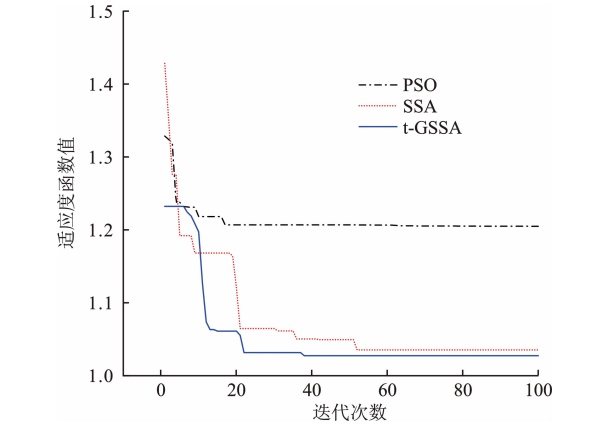


图 5 适应度值变化曲线
Fig. 5 Fitness value change curve

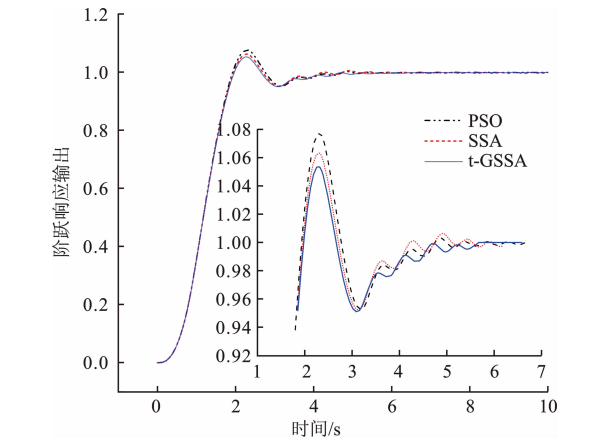


图 6 阶跃响应曲线
Fig. 6 Step response curve

由图 5 可以明显的看出三种算法均在迭代 10 次左右便开始收敛,但本文提出的 t-GSSA 算法的收敛速度较其他两个算法更快;而且所提算法的收敛精度远大于 PSO 算法,较 SSA 也有一定程度的提升;此外,根据几次转折点可以看出,三种算法均陷入过局部最优,但明显可以看出,SSA 算法更容易陷入局部最优,而改进后的 t-SSA 跳出局部最优的能力有所提升,达到和 PSO 算法相似的程度. 由表 4 和图 6 可知三种算法的响应时间相差不大,但改进后 t-GSSA 算法调节时间最短,超调量和 ITAE 也有较大幅度下降,相对于 PSO 算法而言更是有较

大改善.
综上所述,t-GSSA 算法在 PID 参数整定中,其阶跃响应的超调量相对较小、收敛速度和精度有所提升,系统的控制品质得到了一定的改善.

6 结束语

为了增强麻雀搜索算法跳出局部最优和全局探索能力,本文提出了一种基于自适应 t 分布和黄金正弦改进的麻雀搜索算法,将黄金正弦算法和自适应 t 分布变异算子引入麻雀搜索算法中,通过与其他 7 种算法在 10 个基准测试函数上的比较,以及在 PID 参数整定中的应用,检验了 t-GSSA 良好的寻优性能和稳定性.

参考文献:

[1] XUE J K, SHEN B. A novel swarm intelligence optimization approach: sparrow search algorithm[J]. Systems Science & Control Engineering, 2020, 8(1): 22-34. DOI: 10.1080/21642583.2019.1708830.

[2] 吕鑫,慕晓冬,张钧,等. 混沌麻雀搜索优化算法[J]. 北京航空航天大学学报, 2021, 47(8): 1712-1720. DOI: 10.13700/j. bh. 1001-5965. 2020.0298

LV C, MU X D, ZHANG J, et al. Chaos sparrow search optimization algorithm[J]. Journal of Beijing University of Aeronautics and Astronautics, 2021, 47(8): 1712-1720. DOI: 10.13700/j. bh. 1001-5965. 2020.0298

[3] 吕鑫,慕晓冬,张钧. 基于改进麻雀搜索算法的多阈值图像分割[J]. 系统工程与电子技术, 2021, 43(2): 318-327. DOI: 10.12305/j. issn. 1001-506X. 2021.02.05

LV C, MU X D, ZHANG J. Multi-threshold image segmentation based on improved sparrow search algorithm[J]. Systems Engineering and Electronics, 2021, 43(2): 318-327. DOI: 10.12305/j. issn. 1001-506X. 2021.02.05

[4] 毛清华,张强. 融合柯西变异和反向学习的改进麻雀算法[J]. 计算机科学与探索, 2021, 15(6): 1155-1164. DOI: 10.3778/j. issn. 1673-9418. 2010032.

MAO Q H, ZHANG Q. Improved sparrow algorithm combining Cauchy mutation and Opposition-based learning[J]. Journal of Frontiers of Computer Science & Technology, 2021, 15(6): 1155-1164. DOI: 10.3778/j. issn. 1673-9418. 2010032.

[5] TANYILDIZI E, DEMIR G. Golden sine algorithm: a novel math-inspired algorithm[J]. Advances in Electrical and Computer Engineering, 2017, 17(2): 71-78. DOI: 10.4316/AECE.2017.02010.

- [6] 肖子雅, 刘升. 精英反向黄金正弦鲸鱼算法及其工程优化研究[J]. 电子学报, 2019, 47(10): 2177-2186. DOI: 10.3969/j.issn.0372-2112.2019.10.020.
- XIAO Z Y, LIU S. Study on elite opposition-based golden-sine whale optimization algorithm and its application of project optimization[J]. Acta Electronica Sinica, 2019, 47(10): 2177-2186. DOI: 10.3969/j.issn.0372-2112.2019.10.020.
- [7] 周方俊, 王向军, 张民. 基于 t 分布变异的进化规划[J]. 电子学报, 2008, 36(4): 667-671. DOI: 10.3321/j.issn:0372-2112.2008.04.011.
- ZHOU F J, WANG X J, ZHANG M. Evolutionary programming using mutations based on the t probability distribution[J]. Acta Electronica Sinica, 2008, 36(4): 667-671. DOI: 10.3321/j.issn:0372-2112.2008.04.011.
- [8] YAO X, LIU Y, LIN G M. Evolutionary programming made faster[J]. IEEE Transactions on Evolutionary Computation, 1999, 3(2): 82-102. DOI: 10.1109/4235.771163.
- [9] 高文欣, 刘升, 肖子雅, 等. 柯西变异和自适应权重优化的蝴蝶算法[J]. 计算机工程与应用, 2020, 56(15): 43-50. DOI: 10.3778/j.issn.1002-8331.1907-0048.
- GAO W X, LIU S, XIAO Z Y, et al. Butterfly optimization algorithm based on Cauchy variation and adaptive weight[J]. Computer Engineering and Applications, 2020, 56(15): 43-50. DOI: 10.3778/j.issn.1002-8331.1907-0048.
- [10] 葛育晓, 赵荣珍. 基于改进 SOA 算法自整定 PID 系统优化研究[J]. 仪表技术与传感器, 2020(10): 108-113. DOI: 10.3969/j.issn.1002-1841.2020.10.023.
- GE Y X, ZHAO R Z. Research on self-tuning PID system optimization based on improved seeker optimization algorithm[J]. Instrument Technique and Sensor, 2020(10): 108-113. DOI: 10.3969/j.issn.1002-1841.2020.10.023.
- [11] 凤丽洲, 王友卫, 韩琳琳, 等. 双重驱动的果蝇优化算法及其在 PID 控制器中的应用[J]. 控制与决策, 2021, 36(9): 2225-2233. DOI: 10.13195/j.kzyjc.2020.0046.
- FENG L Z, WANG Y W, HAN L L, et al. Double drive fruit fly optimization algorithm and its application in PID controller[J]. Control and Decision, 2021, 36(9): 2225-2233. DOI: 10.13195/j.kzyjc.2020.0046.

作者简介:

张伟康 男, (1996-), 硕士研究生. 研究方向为商务统计、智能计算.

刘升(通讯作者) 男, (1966-), 博士, 教授. 研究方向为人工智能、智能计. E-mail: ls6601@163.com