

融合正弦余弦算法的花授粉算法

刘 升,赵齐辉,陈思静

(上海工程技术大学 管理学院,上海 201620)

摘 要: 针对基本花授粉算法存在容易陷入局部最优、求解精度粗糙等问题,提出一种融合正弦余弦算法的花授粉算法。将正弦余弦算法作为一种局部算法嵌入到花授粉中,对花粉个体分别进行正弦和余弦优化,并用六种标准函数测试算法的性能,结果表明 SCA-FPA 寻优性能优于基本花授粉算法和基于差分进化的花授粉算法。

关键词: 花授粉算法;正弦余弦算法;函数优化

中图分类号: TP3

文献标识码: A

文章编号: 1000-7180(2018)06-0084-04

Flower Pollination Algorithm Based on Sine Cosine Algorithm

LIU Sheng, ZHAO Qi-hui, CHEN Si-jing

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

Abstract: To solve the problems of basic flower pollination algorithm easily falling into local optimum and the low computational accuracy, presenting a flower pollination algorithm based on sine cosine algorithm, which means designing a sine cosine algorithm as a local optimization operator to optimizing flower with sine and cosine function. Six standard functions are tested, and the results show that SCA-FPA is better than basic flower pollination algorithm and DEFPA.

Key words: flower pollination algorithm; sine cosine algorithm; function optimization

1 引言

花授粉算法 (Flewer Pollination Algorithm, FPA) 的优点是结构简单、容易实现、通用性强,具有较好的稳定性,FPA 算法优秀的应用性能现已成为人工智能一个新的热点。但是 FPA 也存在易陷入局部最优、求解精度粗糙、收敛性证明缺乏等问题^[1]。

针对花授粉算法易陷入局部最优、求解精度粗糙的缺点,国内外学者做了很多改进,包括:参入局部优化算子和其他算法混合。文献^[2]将混沌优化引入到花授粉算法中,利用混沌算法的遍历性、随机性,进行花粉个体的优化,较好的平衡了局部搜索与全局搜索的关系,并用于大整数规划问题的求解。肖辉辉^[3]等人将差分进化的机制参入花朵授粉算法之中,将花授粉的求解结果作为差分进化的初始点进行变异、交叉、选择,其先对花授粉的寻优性能进行

深刻的分析,并指出其融入了蝙蝠算法和布谷鸟算法的优点,寻优性能是优于这两种算法的,并选择典型的标准测试函数与基本 FPA,BA,PSO 进行对比,取得较好的优化效果。

正弦余弦算法 (Sine Cosine Algorithm, SCA)^[4]是 2016 年出现的一种新颖的群智能算法,该算法仅仅是采用正弦公式和余弦公式进行迭代计算,实现简单、通俗易懂,是一种很有前景的算法。

本文针对 FPA 的不足,提出融合正弦余弦算法的花授粉算法(SCA-FPA)。

2 基本花授粉算法

花授粉算法(FPA)是受到自然界植物花授粉这一行为启发而得出灵感,自然界大约有 90%的植物进行生物传粉(Biotic pollination),而存在 10%进行非生物传粉(Abiotic pollination),非生物传粉需要

通过一些介质进行传播,包括:蜜蜂、蝙蝠、鸟类,而这些动物的传播授粉行为服从莱维分布。

基于植物花授粉的一些特征,在提出该算法之前应当遵循以下理想化规则^[1]:

(1)进行异花授粉是通过蜜蜂或者鸟类来进行的,而这些异花授粉的‘媒介’是进行莱维飞行来传播花粉的,异花授粉我们看出一个全局授粉的过程;

(2)自花授粉的过程可以视为一个局部授粉的过程;

(3)花恒常性可以认为是授粉行为涉及到两朵花的繁殖概率与相似度成比例;

(4)全局授粉和局部授粉由转换概率 $p \in [0,1]$ 控制,受物理位置邻近性的影响以及其他自然因素(风、雨雪等),局部授粉和全局授粉中 p 代表着重要的意义。

在 FPA 算法中,有两个关键的步骤:全局授粉和局部授粉。

在全局授粉的过程中,花粉是由‘媒介’(例如昆虫)进行飞行而传播,鉴于此,我们把 g_* 定义为当前种群中最优花粉所处的位置,则其位置 x_i^t 更新公式如下:

$$x_i^{t+1} = x_i^t + L(x_i^t - g_*) \quad (1)$$

式中, x_i^{t+1} 代表着花粉 i 在 $t+1$ 循环时的位置, L 代表着授粉的强弱程度,但其本质是一个服从莱维分布的步长因子,服从以下公式(2)分布:

$$L \sim \frac{\lambda \Gamma(\lambda) \sin\left(\frac{\pi\lambda}{2}\right)}{\pi} \frac{1}{s^{1+\lambda}}, (s \gg s_0 > 0) \quad (2)$$

式中, $\Gamma(\lambda)$ 是标准伽马函数,并且这一分布较为合理当 $s(s > 0)$ 为一个较大值的情况,一般地,取 $\lambda = 3/2$ 。

当花粉进行自花授粉的过程时,其位置 x_i^t 更新的公式如下:

$$x_i^{t+1} = x_i^t + \epsilon(x_j^t - x_k^t) \quad (3)$$

x_j^t 和 x_k^t 是不同于花粉 i 的两个花粉的位置, $\epsilon \in U[0,1]$ 。

基于以上分析,我们可以将 FPA 算法描述如下:

Randomly initialize the position of N flowers in the searching place

Define the objective function(Max or Min)

Evaluate the position of each flower

Initialize the best agent of each flower(g_*)

While $t < \max t$

For $i = 1:N$

Define P(switch probability)

```

    If rand  $\leq P$ 
       $x_i^{t+1} = x_i^t + L(x_i^t - g_*)$       else
       $x_i^{t+1} = x_i^t + \epsilon(x_j^t - x_k^t)$ 
    End if
  End for
  Check whether all conditions are met
  Evaluate the new position of each flower
  Update the best agent of flowers

```

3 基于正弦余弦算法的花授粉算法

3.1 正弦余弦算法(SCA)^[4]

SCA 不是模拟自然界中某些生物现象而产生,故其不需要假设条件,SCA 算法解的更新过程是靠随机数与转换概率的比较来决定是进行余弦操作还是正弦操作,其公式更新如下:

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 * \sin(r_2) * |r_3 * P_i^t - X_i^t|, & r_4 < 0.5 \\ X_i^t + r_1 * \cos(r_2) * |r_3 * P_i^t - X_i^t|, & r_4 > 0.5 \end{cases} \quad (4)$$

式中, $X_i^{t+1} = (X_{i1}, X_{i2}, \dots, X_{in})^T$ 表示 n 维个体 i 的空间位置, $P_i^t = (P_{i1}, P_{i2}, \dots, P_{in})^T$ 表示每一代的最优个体位置。

式中存在四个参数 r_1, r_2, r_3, r_4 . $r_1 = a - t * \frac{a}{\max t}$, 其决定着下个位置的区域(或者移动方向); a 是一个常数; r_2 取 $[0, 2\pi]$ 之间的一个随机数,其决定移动的距离; r_3 是一个随机权重,取值为 $[0, 2]$ 之间的一个随机数, $r_3 > 1$ 则对下一次迭代影响效果明显, $r_3 < 1$ 则不明显; r_4 取 $[0, 1]$ 之间的一个随机数。

基于以上分析,我们可以将 SCA 算法描述如下:

```

  Initialize a set of search agents(solution)(X)
  Do
    Evaluate each of the search agents by the
    objective function( $f(X)$ )
    Update the best solution obtained so far( $\text{best} = X *$ )
    Define  $r_1, r_2, r_3$  and  $r_4$ 
    Update the position of each agents via sine and
    cosine formula
    While( $t < \text{maximum number of iterations}$ )
    Return the best solution

```

3.2 SCA-FPA

本文提出的一种 SCA-FPA 并不完全等同于混合算法,而是可以看成是正弦余弦优化作为局部优化算子嵌入到 FPA 中. 正弦或余弦的函数值是在 $[-1, 1]$ 这一区间内的,SCA 的优化过程可以看成正弦(余弦)波的传播,波传播时能量是逐渐消化的,正弦(余弦)波最后会固定在某个值,故 SCA 是收敛

的,因此某种程度来说 SCA 可以加强 FPA 的收敛性能.

SCA 算法中,无论是正弦还是余弦优化每一代个体会与最优个体进行信息交流,再加上 SCA 两个重要参数(r_2, r_3)分别控制搜索的方向和距离. 因此 SCA 算法的引入会指引着花粉个体不断地往最优值方向靠近,个体的寻优不具有盲目性,每一个花粉个体会充分吸收自身与最优个体的位置差信息,因此混合 SCA 算法会获得较为理想的寻优结果.

一般来说,混合其他算法会导致新的算法时间复杂度增加,但是 FPA 和 SCA 的结构较为简单,一般说来并不会带来过高的时间复杂度,因此相对提高的精度来说,增加的时间复杂度是可以接受的.

基于以上分析,将 SCA-FPA 描述如下:

Step 1:定义目标函数 $f(x)$;

$$x=(x_1,x_2,\cdots,x_d)^T$$

Step 2:花粉种群初始化,随机产生初始最优解;

Step 3:定义转换概率 p ;

Step 4:设定迭代终止的条件(一个固定的迭代次数或者需要达到的目标精度);

Step 5:如果 $\text{rand}<p$,进行异花授粉;

Step 6:如果 $\text{rand}>p$,进行自花授粉;

Step 7:评价产生的新解,如果优于产生的随机解,则替代掉;否则转到 Step 4-Step 6;

Step 8:更新找到的最优解.

Step9:把 Step 8 得到的解作为正弦余弦的初始点进行正弦、余弦操作,然后得到一组新解,具体如下:

$$x_i^{t+1} =$$

$$\begin{cases} x_i^t + r_1 * \sin(r_2) * |r_3 * g_* - x_i^t|, r_4 < 0.5 \\ x_i^t + r_1 * \cos(r_2) * |r_3 * g_* - x_i^t|, r_4 > 0.5 \end{cases}$$

Step10:计算 Step 9 的新解对应的全局最优值,更新全局最优解.

4 数值实验

4.1 参数设置与测试函数

本文选取了基本 FPA 和基于差分进化算法的 FPA(DEFPA)^[3]进行对比,由于花授粉算法只涉及一个参数,即转换概率 p ,Yang 在文献[1]中阐明了 p 在选取 0.8 时实验效果较好,因此 FPA 算法参数选取转换概率 $p=0.8$;DEFPA 算法参数选取转换概率 $p=0.8$, $CR=0.1$, $F=0.5$;本文算法选取 $p=0.8,a=2$;种群为 20,迭代 2 000 次,选取相一

致的测试函数,独立运行 30 次,分别进行固定迭代次数的寻优的精度对比和固定精度的迭代次数对比.

表 1 测试函数

特点	函数	名称	维数	范围	最优值
高维单峰	f_1	Sphere	30	$[-100,100]$	0
	f_2	Schwefel	30	$[-10,10]$	0
	f_3	Rastrigin	30	$[-5.12,5.12]$	0
高维多峰	f_4	Ackley	30	$[-32.768,32.768]$	0
	f_5	Griewank	30	$[-600,600]$	0
低维	f_6	Schaffer	2	$[-100,100]$	-1

4.2 结果分析

(1)固定迭代次数的寻优精度对比

对比三种算法的求解结果的最优值、平均值与最差值. f_1 是一个高维单峰函数,其最优值较容易寻得,通常是用来测试算法的收敛速度,三种算法的求解精度 SCA-FPA 是最优的,达到了 10^{-220} 这一数量级,并且从收敛曲线来看收敛速度也是快于其他两种算法;Schwefel 函数亦是个单峰函数,三种算法均没有找到理论的最优值,但是 SCA-FPA 的求解精确度远高于其他两种算法,并且在 900 代左右已开始收敛; f_3 属于典型的高维多峰函数,特点是强烈震荡,DEFPA 和 SCA-FPA 均可以搜索到其理论最优值,但是曲线图中 SCA-FPA 的收敛速度快于 DEFPA; f_4 是一个指数叠加并且加上余弦函数适当放大的连续型函数,其特点是:曲面起伏不平,很多算法由于在“爬山”搜索的过程中不免落入了局部最小的陷进,SCA-FPA 只是收敛到了 $8.8818\text{e-}16$,但是这一结果优于 FPA 和 DEFPA 的寻优结果,并且收敛较为迅速;Griewank 函数是非线性多模态函数,它的特点是强烈震荡,并且趋于单峰函数,SCA-FPA 是直接可以搜索到其理论最优值的,搜索的效率也较为高效,DEFPA 的结果的寻优精度仅仅是 10^{-8} ; f_6 函数的全局最优值被许多局部最小值包围,因此较难求解,SCA-FPA 在 30 次迭代过程均可以快速的找到其最小值,其余算法或多或少存在一定的误差.

(2)固定精度的寻优迭代次数对比

为了综合验证算法的有效性,本文选取了固定精度进行迭代次数的对比,包括最小迭代次数和平均迭代次数.

FPA 只有在测试 f_6 时迭代次数让人满意,其他的都超过了 1 000 次的平均迭代,有的甚至找不

到结果;虽然 DEFPA 相对 FPA 已经有了较大的改进,但是在 $f_1 - f_5$ 的求解结果来看无论是最小迭代次数还是平均迭代次数 SCA-FPA 均要领先于 DEFPA,个别的是较大幅度领先;特别地, f_6 的求解最小迭代次数上 DEFPA 表现的最好,但是其平均结果却不如 SCA-FPA,这说明了 DEFPA 稳定性

不如 SCA-FPA。
综和(1)和(2),由仿真实验可得,SCA-FPA 在处理单峰及复杂多峰函数时能够表现出一定的优势,无论是从求解的精确程度来看还是算法的收敛的速度,因此可以说明本文对 FPA 的改进在函数优化方面是有效可行的。

表 2 固定精度的迭代次数对比

函数	维数	固定精度	最小迭代次数			平均迭代次数		
			FPA	DEFPA	SCA-FPA	FPA	DEFPA	SCA-FPA
f_1	30	1.0E-05	1 081	119	15	1 311.9	138.2	43.4
f_2	30	1.0E-01	1 020	80	12	1.272 7E+03	95.66	21.8
f_3	30	1.0E-07	—	267	40	—	291.04	40.5
f_4	30	1.0E-03	—	186	30	—	209.08	47
f_5	30	1.0E-06	1 548	467	60	1 986.48	542.9	70.3
f_6	2	1.0E-02	59	8	20	212.48	30.42	25

5 结束语

本文介绍了一种融合正弦余弦算法的花授粉算法,将正弦余弦算法作为局部优化算子在迭代后期对花粉个体进行正弦优化和余弦优化,增强了普通个体与最优个体的交流,并且引导花粉个体进行高效的搜索,有效的避免算法陷入局部最优.选取了六个常用的基准函数进行测试,并与基本花授粉算法和基于差分进化策略的花授粉算法进行对比,证明其函数寻优结果较优,尤其是针对高维多峰函数的优化存在一定的优势。

本文提出的改进版本是对花授粉的一种有力补充而非替代,下一步的工作应该是完善其理论基础,还包括进一步运用到求解组合优化问题如:旅行商问题、二维排样优化问题。

参考文献:

[1] Yang X S. Flower pollination algorithm for global optimization [M]//Unconventional computation and natural computation. Springer Berlin Heidelberg, 2012: 240-249.

[2] Kaur G, Singh D, Kaur M. Robust and Efficient RGB based Fractal Image Compression: Flower Pollination based Optimization [J]. Proc. of International Journal of Computer Applications, 2013, 78(10): 11-15.

[3] Abdel-Raouf O, El-Henawy I, Abdel-Baset M. A Novel Hybrid Flower Pollination Algorithm with Chaotic Harmony Search for Solving Sudoku Puzzles [J]. International Journal of Modern Education and Computer Science (IJMECS), 2014, 6(3): 38-44.

[5] El-henawy I, Ismail M. An improved chaotic flower pollination algorithm for solving large integer programming problems [J]. International Journal of Digital Content Technology and its Applications, 2014, 8(3): 72.

[6] 肖辉辉,万常选,段艳明. 一种改进的新型元启发式花朵授粉算法 [J]. 计算机应用研究, 2016, (01): 126-131.

[7] Mirjalili S (2016) SCA: a sine cosine algorithm for solving optimization problems. Knowl Based Syst , 2016, (96):120-133.

[8] Mohamed A. Tawhid Vimal Savsani. Multi-objective sine-cosine algorithm (MO-SCA) for multi-objective engineering design problems [J]. Neural Computing and Applications , 2017(4):1-15.

[9] Sujin Bureerat Nantiwat Pholdee. Adaptive Sine Cosine Algorithm Integrated with Differential Evolution for Structural Damage Detection [C]. International Conference on Computational Science and Its Applications ICCSA 2017: Computational Science and Its Applications - ICCSA ,2017 ,(10404): 71-86.

[10] Sujoy Das, Aniruddha Bhattacharya, Ajoy Kumar Chakraborty. Solution of short-term hydrothermal scheduling using sine cosine algorithm[J]. Soft Computing,2017,(06):1-19.

作者简介:

刘升男,(1966-),博士,教授.研究方向为智能计算、机器学习. E-mail:845714323@qq.com.

赵齐辉男,(1993-),硕士研究生.研究方向为群智能算法的改进与应用.

陈思静女,(1990-),硕士研究生.研究方向为信息管理与信息系统,渠道优化.