

基于变尺度法和自适应步长的布谷鸟搜索算法

江浩,阮奇

(福州大学石油化工学院,福建福州 350108)

摘要:布谷鸟搜索算法(Cuckoo Search,CS)是一种新型的元启发式算法。针对CS算法局部搜索能力较弱、后期收敛速度偏慢和收敛精度不高等缺点,提出一种基于变尺度法(DFP)和自适应步长(Adaptive Step)的布谷鸟搜索算法(DACS),使Lévy飞行的步长非线性自适应变化来提高算法的收敛速度,同时使经过Lévy飞行机制和淘汰机制进化后的布谷鸟种群再运用DFP快速获取全局最优解。用6种具有各种代表性的测试函数分别测试DACS算法和CS算法的性能。实验结果表明,DACS算法在保持强大的全局搜索能力的同时,比CS算法具有更快的收敛速度、更高的收敛精度和更好的鲁棒性,尤其适合多峰及高维函数的优化。

关键词:布谷鸟搜索;变尺度法;自适应步长;全局寻优

中图分类号:TN911.72

文献标识码:A

文章编号:1673-629X(2015)10-0038-06

doi:10.3969/j.issn.1673-629X.2015.10.008

A Cuckoo Search Algorithm Based on Variable Metric Method and Adaptive Step

JIANG Hao, RUAN Qi

(School of Chemical Engineering, Fuzhou University, Fuzhou 350108, China)

Abstract:Cuckoo Search (CS) is a novel meta-heuristic algorithm. Aiming at the defects of weak local search ability, slow convergence velocity and low convergence accuracy, a modified CS algorithm based on DFP and adaptive step is proposed in this paper. In the improved cuckoo search algorithm, the step of Lévy flight nonlinear dynamic changes improve convergence velocity. After evolved from Lévy flights and elimination mechanism, the cuckoo populations rapidly access to global minima by DFP. Sixth representative benchmark functions are used to test the performance of DACS algorithm and CS algorithm respectively. The conclusions indicate that DACS algorithm has faster convergence speed, higher convergence accuracy and robustness, compared with CS algorithm. Meanwhile, DACS algorithm keeps strong global search capability, which is particularly suitable for the optimization of multimodal function and high dimension function.

Key words:cuckoo search; variable metric method; adaptive step; global optimization

0 引言

在现代科学研究和工程技术的众多领域中,存在许多大型复杂的优化问题迫切需要人们去解决。然而,传统的优化方法对这些问题无能为力,倒是20世纪后期涌现出的一些模拟自然界生物群体行为的元启发式算法,如遗传算法(GA)^[1]、蚁群算法(ACO)^[2]、微粒群算法(PSO)^[3]等越来越成为解决大型复杂优化问题的有效方法^[4-6]。近年来随着仿生技术的不断发展,一些更高效的新型元启发式算法还在不断脱颖而出,如在2009年,Yang和Deb受布谷鸟的繁殖行为和

动物的Lévy飞行模式觅食行为的启发,创立了布谷鸟搜索算法^[7-8](Cuckoo Search,CS)。CS算法具有结构简单、控制参数少和全局寻优能力强等优点。研究表明:在许多优化问题中,CS算法比PSO算法和GA算法更具优势^[8-9]。

在CS算法中,由于布谷鸟每次进行飞行搜索时都是使用随机性很强的Lévy飞行模式,很容易从一个区域飞跃到另一个区域,从而大幅度提高了算法在搜索前期的全局寻优能力,但同时也削弱了布谷鸟在每个鸟巢附近的局部搜索能力,再加上布谷鸟搜索的步

收稿日期:2014-11-19

修回日期:2015-03-10

网络出版时间:2015-09-23

基金项目:国家基础科学人才培养基金资助项目(J1103303)

作者简介:江浩(1990-),男,硕士研究生,研究方向为化工过程模拟与优化、演化算法及其应用;阮奇,教授,硕士生导师,研究方向为化工过程模拟与优化、演化算法及其应用。

网络出版地址:<http://www.cnki.net/kcms/detail/61.1450.TP.20150923.1506.056.html>

长控制因子 α 在整个搜索过程固定不变等原因,导致 CS 算法在后期收敛速度较慢、收敛精度不高。

针对 CS 算法的上述缺点,许多学者分别提出了一些改进的方法:如 Ehsan 等^[10]提出一种淘汰概率 pa 和步长控制因子 α 都随着迭代次数而变化的自适应方法,来平衡算法的全局搜索能力与局部搜索能力,使算法后期的收敛速度得到提高,并指出,当 α 在 $[0.01, 0.5]$ 内变化时对整个搜索过程最为有利;再如:王凡等^[11]提出一种在迭代过程对鸟巢位置加入行高斯扰动的布谷鸟搜索算法,虽然算法的收敛速度有所增加,但是这种在鸟巢位置附近随机的高斯扰动并不具有明确的方向,使得算法的收敛速度并没有显著的提升。因此,文中拟将步长自适应 (Adaptive) 策略与局部搜索能力很强的 DFP 变尺度法进行有机结合,提出一种基于变尺度法和自适应步长的布谷鸟搜索算法 (DACS),来显著提高算法的收敛速度和收敛精度。

1 CS 算法及其机理分析

1.1 CS 算法简介

部分布谷鸟是通过巢寄生的方式来养育后代的,自己从不筑巢而是将自己的蛋产于其他相似鸟类的巢中,并通过伪装或直接移走巢内宿主鸟蛋来提高自己的蛋被孵化的概率。如果宿主鸟发现布谷鸟蛋,则会将布谷鸟蛋推出鸟巢或者放弃这个鸟巢而另觅他处筑巢。一般情况下,布谷鸟蛋孵化略早于宿主鸟蛋,一旦巢寄生的布谷雏鸟孵出,它有将宿主鸟的鸟蛋推出巢外的习性并模仿宿主雏鸟的叫声来获得更多的喂养机会^[7]。基于布谷鸟繁殖行为和动物的 Lévy 飞行觅食行为,Yang 和 Deb 于 2009 年提出了 CS 算法,在导出 CS 算法时他们做了以下三个理想规定^[8]:

(1) 每只布谷鸟一次只产一个蛋,并随机地选择鸟巢进行孵化;

(2) 在随机选择的一组鸟巢中,高质量鸟巢将会被保留至下一代;

(3) 可供布谷鸟选择的宿主鸟巢数量 n 为固定值,宿主鸟巢中的布谷鸟蛋被宿主鸟发现的概率为 $pa \in [0, 1]$ 。

基于上述规定,布谷鸟寻巢的路径和位置更新公式如下:

$$X_i^{k+1} = X_i^k + \alpha \oplus \text{Lévy}(\lambda) \quad (1)$$

式中, X_i^k 和 X_i^{k+1} 分别表示第 i ($i = 1, 2, \dots, n$, n 为鸟巢数量) 个鸟巢在第 k 代和第 $k+1$ 代时的位置向量 ($X_i = x_{i,1}, x_{i,1}, \dots, x_{i,j}, \dots, x_{i,d}$, d 为每个鸟巢的维数, j 为鸟窝的任意第 j 维); \oplus 为点对点乘法; $\text{Lévy}(\lambda)$ 为 Lévy 飞行随机搜索路径; α 为步长控制因子, α 是大于零的常数,对不同的情况 α 可取不同的

值,在文献[8]中 $\alpha = 1$ 。

Lévy 飞行随机搜索路径与时间 t 的关系服从 Lévy 分布^[12]:

$$\text{Lévy} \sim u = t^{-\lambda}, (1 < \lambda \leq 3) \quad (2)$$

式中, λ 为幂次系数。

此外,CS 算法在迭代进化过程中,按式(1)对鸟巢位置进行更新后,用随机数 $\text{rand} \in [0, 1]$ 与 pa 对比,若 $\text{rand} > pa$,则对 X_i^{k+1} 进行随机改变,反之不变。最后保留测试值较好的一组鸟巢位置,此时仍记为 X_i^{k+1} 。具体的 CS 算法流程详见文献[8]。

1.2 CS 算法机理简要分析

CS 算法源自布谷鸟的巢寄生繁育行为,其机理主要涉及 Lévy 飞行随机搜索机制和布谷鸟蛋被淘汰机制两方面。

首先,CS 算法具有强大的全局搜索能力,但局部搜索能力相对偏弱。布谷鸟寻巢产蛋过程是按 Lévy 飞行搜索路径进行的,这体现在式(1)中。式(1)本质上是一个随机行走过程,一般情况下,一个随机行走是一个马尔可夫链,其未来位置 X_i^{k+1} 取决于当前位置 X_i^k (式(1)中的第一项)和转移概率(式(1)中的第二项 $\alpha \oplus \text{Lévy}(\lambda)$)。转移概率中最关键、最核心的技术是采用了 Lévy 飞行随机搜索路径 $\text{Lévy}(\lambda)$,该路径的随机行走步长和方向与时间的关系服从一个带有重尾的概率分布,即式(2)所示的 Lévy 分布,在这种形式的行走中,短距离的探索与偶尔较长距离的行走相间。CS 算法正是采用了 Lévy 飞行的随机搜索策略,使其具有优秀的搜索路径和强大的全局搜索能力等突出优点,但其局部搜索能力仍然较弱,搜索后期的收敛速度和收敛精度仍有进一步提高的空间。

其次,CS 算法的搜索步长极具随机性,但缺乏动态自适应性。对于任何元启发式算法,在全局随机搜索战略和局部精细搜索战略之间保持好的平衡会使算法更有效。换言之,搜索步长应随算法进程自适应变化,即在算法前期搜索步长取大一些,有利于全局搜索,迅速锁定最优值所在的大致区域;随着算法进程的发展,搜索步长应逐渐减小,至算法后期搜索步长应取小一些,有利于局部精细搜索,提高收敛速度和收敛精度。在 CS 算法中,自始至终采用 Lévy 飞行产生步长,这种步长时大时小极具随机性,但缺乏自适应性。虽然 CS 算法在转移概率(式(1)中的第二项)中定义了步长控制因子 α ,本可用于搜索步长的自适应调节,但遗憾的是 CS 算法中取 $\alpha = 1$ 为常数,从而失去了搜索步长的自适应调节功能。

综上所述可知:如果能在保持 CS 算法强大的全局搜索能力的同时设法强化其局部搜索能力,并使算法的搜索步长随进程动态自适应变化,则可望较大幅

度地提升 CS 算法的性能。

2 DACS 算法

2.1 步长自适应策略

根据 1.2 节的分析可知,CS 算法的搜索步长仅由 Lévy 飞行随机决定,不能随算法进程由大到小动态地自适应变化,这会使算法的寻优能力降低。针对上述缺点,进行了大量的 CS 算法性能改进的仿真实验研究,对实验结果进行分析、归纳,从而提出如式(3)所描述的搜索步长控制因子 α 的自适应策略:

$$\alpha^k = \begin{cases} \alpha_{\max} (1 - (\frac{mk}{k_{\max}})^\varphi), & \alpha^k > 0.01 \\ \alpha_{\min}, & \alpha^k \leq 0.01 \end{cases} \quad (3)$$

式中, α_{\max} 、 α_{\min} 和 α^k 分别表示算法的最大步长控制因子、最小步长控制因子和第 k 代的步长控制因子; k 为算法当前搜索代数; k_{\max} 为人为设定的最大搜索代数; m 和 φ 为调节因子。

文献[7]明确指出,当 α 在 $[0.01, 0.5]$ 的区间变化时对整个搜索过程最为有利。当 α 取值相对较大时,搜索步长也大,这样会使算法的收敛性能降低和搜索代数增加;当 α 取值相对较小时,搜索步长也小,这样虽然能加快算法的收敛速度,但不利于找到全局最优,易陷入局部最优。因此,根据仿真实验结果及文献[7]建议取 $\alpha_{\max} = 0.5$ 、 $\alpha_{\min} = 0.01$ 。式(3)中 m 和 n 为调节因子,用于调节 α 的变化幅度。为了获得一个满意的 α 变化趋势,结合算法性能改进的仿真实验结果分析,当取 m 为 200, n 为 1/3 时,由式(3)所决定的 α 自适应策略能够有效地平衡算法的全局搜索能力和局部搜索能力,使算法更高效。

2.2 DFP 变尺度法

DFP 变尺度法是牛顿(Newton)法的一种改进算法,它由 Davidon 于 1959 年提出,并经 Fletcher 和 Powell 在 1963 年进行了改进^[12]。DFP 变尺度法既克服了梯度法在接近极值点时收敛速度慢的缺点,又避开了牛顿法的二阶导数矩阵及其逆矩阵的繁琐计算,所以该法是求解无约束优化问题最有效的算法之一,已经广泛地应用于工程实际问题的求解^[13]。DFP 变尺度法的基本思想是把梯度法与牛顿法相结合,用不包含二阶导数的矩阵 H^k 取代牛顿法中的 Hesse 矩阵的逆矩阵 $[\nabla^2 f(x^k)]^{-1}$,再沿方向 $s^k = -H^k g^k$ 做一维搜索。

计算步骤如下:

(1) 给出初始点 $X_i^k (X_i^k = x_{i,1}^k, x_{i,2}^k, \dots, x_{i,j}^k, \dots, x_{i,d}^k, d$ 为维数,此时 $k=0$) 中的分向量值 $x_{i,j}^0 \in R^n$ 和计算精度 ε 。

(2) 给定变尺度矩阵初始值 $H_{i,j}^0 = I_n (I_n$ 为 n 阶单位矩阵),令 $g_{i,j}^0 = \nabla f(x_{i,j}^0)$,置 $k=0$ 。

(3) 求搜索方向 $s_{i,j}^k = -H_{i,j}^k g_{i,j}^k$ 及最优步长 $\delta_{i,j}^k$ 。

(4) 进行搜索,求下一个迭代点。

$$x_{i,j}^{k+1} = x_{i,j}^k + \lambda_{i,j}^k \delta_{i,j}^k$$

(5) 判断是否满足精度 $\|\nabla f(x_{i,j}^{k+1})\| \leq \varepsilon$,若满足,输出得到的极值,停止迭代。

(6) 当 $k=n$ 时, $x_{i,j}^0 = x_{i,j}^{k+1}$,转步骤(2)。

(7) 当 $k < n$ 时,计算 $g_{i,j}^{k+1}$ 、 $\Delta g_{i,j}^k = g_{i,j}^{k+1} - g_{i,j}^k$ 、 $\Delta x_{i,j}^k = x_{i,j}^{k+1} - x_{i,j}^k$ 、 $H_{i,j}^{k+1} = H_{i,j}^k + \frac{\Delta x_{i,j}^k (\Delta x_{i,j}^k)^T}{(\Delta x_{i,j}^k)^T \Delta g_{i,j}^k} - \frac{H_{i,j}^k \Delta g_{i,j}^k (\Delta g_{i,j}^k)^T H_{i,j}^k}{(\Delta g_{i,j}^k)^T H_{i,j}^k \Delta g_{i,j}^k}$,令 $k = k+1$,转步骤(3)。

由于 DFP 变尺度法强大的局部搜索能力,故拟将其与 CS 算法相结合,来提高 CS 算法的收敛速度和收敛精度。

2.3 变尺度法与自适应步长有机结合的 DACS 算法

DACS 与 CS 算法的关键不同之处在于以下两点:

(1) CS 算法中取步长控制因子 $\alpha = 1$ 不变,而 DACS 算法采用式(3)非线性动态调整步长控制因子 α ,从而使搜索步长随算法进程动态自适应变化,这样有利于消除固定步长的缺陷,提高算法的收敛速度;

(2) CS 算法使经过 Lévy 飞行机制和淘汰机制后的鸟巢位置 X_i^k 直接进入下一轮迭代,而 DACS 算法则是运用 DFP 变尺度法继续进行搜索,快速搜索到新的更优鸟巢位置后,再进入 $k+1$ 次迭代,然后按非线性动态步长的 Lévy 飞行机制进行搜索和位置更新。这样就细化了原来 CS 算法的搜索范围,增大了布谷鸟的搜索活力,从而有效弥补了 CS 算法局部搜索能力不够强的缺点。

综上所述,DACS 算法将局部搜索能力很强的 DFP 变尺度法和具有非线性自适应步长的布谷鸟搜索算法的各自优点有机地结合在一起,可望使算法在保持强大的全局寻优能力的同时,能大幅提升局部搜索能力和收敛速度。

2.4 DACS 算法的基本流程

DACS 算法的主要流程描述如下:

Step1: 初始化种群,包括鸟巢数量、鸟巢的初始位置等,设置算法参数;

Step2: 计算各鸟巢位置相应的适应度值,记录当前的最佳鸟巢位置及其适应度值;

Step3: 根据式(1)至式(3)的 Lévy Flights 机制更新鸟巢位置,得到一组新的鸟巢位置;

Step4: 计算更新后的各鸟巢位置相应的适应度值,经过比较后更新鸟巢的历史最佳位置;

Step5: 用服从均匀分布的随机数 $\text{rand} \in [0, 1]$ 与

pa 比较,保留被发现概率较小的鸟巢位置,同时随机更新被发现概率较大的鸟巢位置,得到一组新的更优的鸟巢位置;

Step6:运用 DFP 变尺度法在每一个鸟巢位置附近进行局部精细搜索,得到新的鸟巢位置;

Step7:计算更新后的各鸟巢位置相应的适应度值,经过比较后更新鸟巢的历史最佳位置;

Step8:判断算法的终止条件,若满足则输出结果,否则返回 Step3。

3 仿真实验

3.1 测试函数

为了评价 DACS 算法的性能,采用如表 1 所示的 6 种常用的测试函数进行算法性能测试,并与 CS 进行比较。这些经典的测试函数^[8-9]被广泛用于评价算法的性能。其中, f_1 为连续的、单峰凸函数,常用于测试

算法的收敛精度; f_2 为典型的高维多峰函数,在求解域内存在 $10d$ 个局部极小值,极具欺骗性,且随着维数的增加,寻找到全局极小值的难度急剧增大,一般算法难以获得该函数的全局极小值; f_3 是一个非凸的病态函数,它唯一的全局极小值存在于一个狭长的、抛物线形的山谷内,但这条山谷内的全局极小值非常难以搜索到,因此该函数通常用来评价优化算法的执行效率^[14]; f_4 是一个比 f_1 更为复杂的单峰函数; f_5 为在求解域内拥有大量局部最小值的高维、多峰函数,一般算法难以获得全局最优值; f_6 在距全局极小值约 3.14 的范围内存在无限多的局部极小值,由于该函数强烈震荡很难全局最优化,常用于检验算法的综合优化能力,即全局寻优能力和收敛能力。

仿真环境为 Windows 2007 操作系统,Intel i3 处理器 2.40 GHz,2.00 GB 内存,仿真软件为 MATLAB 7.5。

表 1 测试函数

函数名称	测试函数	变量定义域	全局最优值
Sphere	$f_1 = \sum_{i=1}^d x_i^2$	$[-5.12, 5.12]$	0
Rastrigin	$f_2 = \sum_{i=1}^d [x_i^2 - 10\cos(2\pi x_i) + 10]$	$[-5.12, 5.12]$	0
Rosenbrock	$f_3 = \sum_{i=1}^{d-1} [100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2]$	$[-2.048, 2.048]$	0
Sum of different power	$f_4 = \sum_{i=1}^d x_i ^{i+1}$	$[-1, 1]$	0
Michalewicz	$f_5 = - \sum_{i=1}^d \sin(x_i) \left[\sin\left(\frac{ix_i^2}{\pi}\right) \right]^{2m} (m = 10)$	$[0, \pi]$	-1.801 ($d = 2$) -4.687 ($d = 5$)
Schaffer	$f_6 = 0.5 + \left(\sin^2 \sqrt{\sum_{i=1}^d x_i^2} - 0.5 \right) / \left[1 + 0.001 \left(\sum_{i=1}^d x_i^2 \right) \right]^2$	$[-5.12, 5.12]$	0

3.2 实验结果与分析

为了公平地比较 DACS 和 CS 算法的性能,对两种算法的参数均取鸟巢数量 n 为 25,淘汰概率 pa 为 0.25。DACs 算法的 α_{\max} 为 0.5, α_{\min} 为 0.01,调节因子 m 为 200, φ 为 1/3;CS 算法的步长控制因子 α 取值为 0.01。两种算法的收敛精度均为 10^{-7} ,若收敛精度低于 10^{-7} 或搜索代数超过设定的最大搜索代数 k_{\max} ,则认为算法未能成功收敛。为了进行有意义的统计分析,对于不同的测试函数、不同的算法分别独立运行 50 次,并记录算法的平均收敛代数、收敛代数的标准偏差及成功收敛率,实验结果见表 2。

图 1 ~ 图 4 为上述测试函数的最佳适应度值随进化代数的变化曲线(6 个函数的最佳适应度值随进化代数的变化曲线基本类似,仅选择其中 4 个具有代表性的变化曲线图列出)。

分析表 2、图 1 以及图 4 发现,两种算法对不同维

数的单峰函数(f_1 和 f_4)进行优化时,在相同的迭代次数下,DACS 算法达到的收敛精度远远高于 CS 算法;在相同的收敛精度下,DACS 算法的收敛速度要快于 CS 算法,且随着维数的增加而愈加明显,这说明 DACS 算法具有较高的收敛精度与较快的收敛速度,对单峰函数的优化几乎瞬间可以完成。同时,由表 2 的 50 次独立仿真实验结果的统计数据可知,对于以上 6 个测试函数,无论是低维的还是高维的,无论是单峰单谷的还是多峰多谷的,DACS 算法的标准偏差均比 CS 算法小得多,上述情况充分说明 DACS 算法比 CS 算法有更好的鲁棒性。

分析表 2、图 2 和图 3 可知,在多峰及高维函数的优化中,DACS 算法性能的优越性体现得尤为明显。如 Rastrigin 函数,随着维数的增加,CS 算法的收敛速度急剧下降,收敛率也明显降低,当维数达到 50 维时,CS 算法无法收敛;而 DACS 算法的收敛速度要快于 CS

表 2 DACS 算法与标准 CS 算法的性能比较

测试函数	维数	最大搜索代数	算法	平均收敛代数	标准偏差	收敛率
Sphere	10	1 000	CS	908	26	1
			DACS	13	6	1
	20	2 000	CS	1 697	62	1
			DACS	30	9	1
	50	5 000	CS	4 328	185	1
			DACS	85	21	1
Rastrigin	10	10 000	CS	7 214	942	1
			DACS	110	56	1
	20	20 000	CS	17 587	1 434	0.72
			DACS	346	103	1
	50	20 000	CS	--	--	0
			DACS	1 222	229	1
Rosenbrock	10	5 000	CS	4 178	299	1
			DACS	116	29	1
Sum of different power	10	400	CS	322	24	1
			DACS	2	0	1
Michalewicz	2	200	CS	143	59	1
			DACS	2	0	1
	5	2 000	CS	1 434	332	0.28
			DACS	32	19	1
Schaffer	2	3 000	CS	2 421	864	1
			DACS	6	4	1

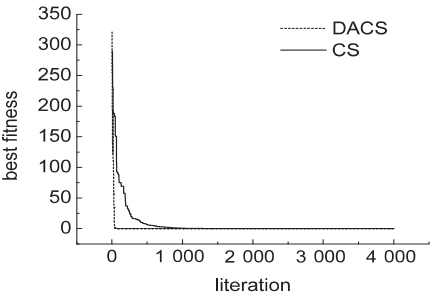


图 1 Sphere 函数($d = 50$)

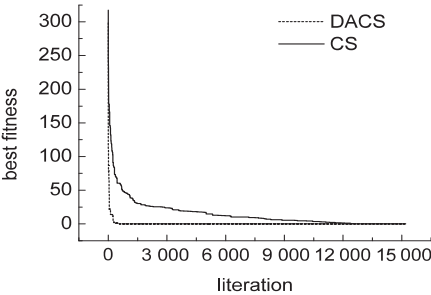


图 2 Rastrigin 函数($d = 20$)

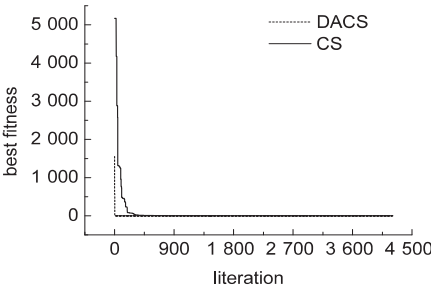


图 3 Rosenbrock 函数($d = 10$)

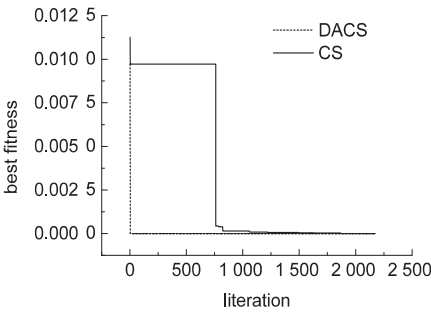


图 4 Schaffer 函数($d = 2$)

算法,同样在 50 维时,DACS 算法仍能快速且稳定地搜索到全局最优值。对于 Rosenbrock 函数,DACS 算法在第 116 代即可搜索到全局最小值,而 CS 算法则需 4 178 代左右。这充分说明 DACS 算法既保持了全局优化能力强的特点,又继承了 DFP 变尺度法局部搜索能力强的优点,同时在优化过程中具有较好的动态自适应能力,从而提高了算法的收敛精度和优化能力,尤其适合多峰及高维函数的优化。

综上所述,DACS 算法比 CS 算法具有更快的收敛速度、更高的收敛精度和更好的鲁棒性。究其原因,在于 DACS 算法采用了 Lévy 飞行搜索步长随算法进程由大到小自适应变化策略,较好地平衡了算法的全局搜索能力和局部搜索能力,提升了算法的综合性能;采用了 DFP 变尺度局部搜索策略,克服了 CS 算法局部搜索能力较弱,算法后期收敛速度较慢,收敛精度不够高的缺点。总而言之,DACS 算法的综合性能明显优

于 CS 算法。

4 结束语

DACS 算法将 CS 算法中的步长改为自适应步长,并将具有很强局部搜索能力的 DFP 变尺度法与 CS 算法有机地结合在一起,不仅保持了 CS 算法强大的全局寻优能力,而且比 CS 算法具有更快的收敛速度、更高的收敛精度和更好的鲁棒性。DACS 算法较大幅度地提升了 CS 算法的综合性能,特别适合于求解高维、多峰多谷复杂函数的优化问题。

参考文献:

[1] Goldberg D E. Genetic algorithm in search, optimization and machine learning [M]. Boston: Addison – Wesley Longman Publishing Co. Inc, 1989.

[2] Dorigo M, Bonabeau E, Theraulaz G. Ant algorithms and stigmergy [J]. Future Generation Computer System, 2000, 16 (8): 851–871.

[3] Kennedy J, Eberhart R. Particle swarm optimization [C] // Proceedings of IEEE international conference on neural networks. Perth: IEEE, 1995: 1942–1948.

[4] 姜 斌, 梁士锋. 化学工程中多目标遗传算法的应用 [J]. 现代化工, 2007, 27 (7): 66–69.

[5] 张志猛, 李九宝, 刘兴高. 一种基于新型蚁群算法的聚丙烯熔融指数预报模型 [J]. 化工学报, 2011, 62 (8): 2270–

2274.

[6] 毕荣山, 杨 霞, 谭心舜, 等. 改进的微粒群优化算法在过程综合中的应用 [J]. 计算机与应用化学, 2004, 21 (4): 565–568.

[7] Payne R B, Sorenson M D, Klitze K. The cuckoos [M]. Oxford: Oxford University Press, 2005.

[8] Yang X S, Deb S. Cuckoo search via Levy flights [C] // Proceedings of world congress on nature & biologically inspired computing. Piscataway: IEEE, 2009: 210–214.

[9] Yang X S, Deb S. Engineering optimization by cuckoo search [J]. International Journal of Mathematical Modeling and Numerical Optimization, 2010, 1 (4): 330–343.

[10] Valian E, Tavakoli S, Mohanna S, et al. Improved cuckoo search for reliability optimization problems [J]. Computers & industrial Engineering, 2013, 64: 456–468.

[11] 王 凡, 贺兴时, 王 燕. 基于高斯扰动的布谷鸟搜索算法 [J]. 西安工程大学学报, 2011, 25 (4): 566–569.

[12] Fletcher R, Powell M J D. A rapidly convergent descent method for minimization [J]. Computer Journal, 1963, 6 (2): 163–168.

[13] 李国旗, 陈新度, 欧阳渺安. DFP 变尺度法在超精密非球面镜误差分析中的应用 [J]. 工具技术, 2007, 41 (8): 85–88.

[14] Ratnaweera A, Halgamuge S K, Watson H C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients [J]. IEEE Trans on Evolutionary Computation, 2004, 8 (3): 240–255.

(上接第 37 页)

提取精度越高,但水印容量相对较小;反之,提取精度降低,水印容量增大。文中算法采用 Logistic 混沌系统,将隐藏的每一位信息散布在原图的多个位置,增加了冗余性,保证水印的有效提取。

参考文献:

[1] 叶 闯, 沈益青, 李 豪, 等. 基于人类视觉特性 (HVS) 的离散小波变换 (DWT) 数字水印算法 [J]. 浙江大学学报: 理学版, 2013, 40 (2): 152–155.

[2] 赵雪章, 席运江. 一种基于多小波域与 DCT 的数字盲水印算法 [J]. 计算机仿真, 2011, 28 (3): 126–129.

[3] Victor H G, Clara C R, Mariko N M. Watermarking algorithm based on the DWT [J]. IEEE Latin America Transactions, 2006, 4 (4): 257–267.

[4] 吕海翠, 傅 鹏, 王 丹. 小波变换的低频数字水印 [J]. 计算机工程与应用, 2009, 45 (11): 99–101.

[5] 刘云如, 易叶青, 羊四清. 基于小波域多元非线性回归的鲁棒性数字水印算法 [J]. 计算技术与自动化, 2008, 27 (1): 57–60.

[6] Hsu C T, Wu J L. Image watermarking by wavelet decomposi-

tion [J]. Academy of Information and Management Sciences Journal, 2000, 3 (1): 70–86.

[7] 甘 甜, 冯少彤, 聂守平, 等. 基于分块离散小波变换的图像信息隐藏与盲提取算法 [J]. 物理学报, 2012, 61 (8): 183–190.

[8] 李 峰, 陈光喜, 丁 勇, 等. 基于混沌和 HVS 的小波域自适应图像水印算法 [J]. 计算机应用研究, 2012, 29 (6): 2224–2227.

[9] 殷 虹, 陈增强, 袁著祉. 基于超混沌和小波变换的鲁棒性数字水印算法 [J]. 控制与决策, 2006, 21 (9): 1024–1027.

[10] Wu C W, Rul’kov N F. Studying chaos via 1–D maps—a tutorial [J]. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 1993, 40 (10): 707–721.

[11] 谢建全, 谢 勃, 阳春华, 等. 基于 Logistic 映射的加密算法的安全性分析与改进 [J]. 小型微型计算机系统, 2010, 31 (6): 1073–1076.

[12] 刘 涛, 曾祥利. 使用小波分析入门 [M]. 北京: 国防工业出版社, 2006.


[13] 杨 榆, 钮心忻. 信息隐藏与数字水印实验教程 [M]. 北京: 国防工业出版社, 2010.

[14] 王相海. 基于小波的图像水印研究进展 [J]. 计算机科学, 2005, 32 (1): 202–207.

基于变尺度法和自适应步长的布谷鸟搜索算法

作者：[江浩](#)，[阮奇](#)，[JIANG Hao](#)，[RUAN Qi](#)

作者单位：[福州大学 石油化工学院, 福建 福州, 350108](#)

刊名：[计算机技术与发展](#)

英文刊名：[Computer Technology and Development](#)

年，卷(期)：2015(10)

引用本文格式：[江浩](#). [阮奇](#). [JIANG Hao](#). [RUAN Qi](#) [基于变尺度法和自适应步长的布谷鸟搜索算法](#)[期刊论文]-[计算机技术与发展](#) 2015(10)