

## 求解二重数值积分的混沌布谷鸟优化算法

凌巍高, 欧 旭, 罗德相

(广西医科大学 信息中心, 广西 南宁 530006)

**摘 要:** 引入混沌变量作为布谷鸟优化算法的局部算子, 提出了一种基于混沌布谷鸟 (Chaotic Cuckoo Search, CCS) 优化算法求解二重数值积分的方法. 该方法初始时在矩形积分区域两个方向的区间内各自任意选取一定的节点, 通过 CCS 算法优化这些节点, 并以优化后的节点作为分割点对二重数值积分进行求解. 仿真实例结果表明, 该算法简单, 易实现, 收敛速度快, 得到的积分值精度高, 是一种有效的二重数值积分求解方法.

**关键词:** 混沌; 布谷鸟优化算法; 二重数值积分

中图分类号: TP18

文献标识码: A

文章编号: 1000-7180(2014)08-0148-03

## Two Dimensional Numerical Integration Based on Chaotic Cuckoo Search Optimization Algorithm

LING Wei-gao, OU Xu, LUO De-xiang

(Information Center, Guangxi Medical University, Nanning 530006, China)

**Abstract:** Introducing chaotic factor as a local operator of cuckoo search algorithm, a method was presented to calculate two dimensional numerical integration based on chaotic cuckoo search (CCS) optimization algorithm. CCS algorithm selected some initial partition points randomly on the both direction range in rectangular integral domain and then to optimize these points. Points optimized would be as the segmentation nodes to solve two dimensional numerical integration. The experimental results show that our algorithm is easy to implement and has fast convergence speed and high integral value accuracy. Therefore, CCS is an effective method for solving the two dimensional numerical integration.

**Key words:** chaos; cuckoo search; two dimensional numerical integration

### 1 引言

二重数值积分的传统计算方法是将二重积分转化为二次积分再进行求解. 然而在实际工程应用中, 这种方法常常面临着原函数无法用初等函数表示或者原函数表达过于繁琐而无法计算的问题<sup>[1]</sup>. 随后, 人们采用数值方法比如梯形法、Simpson 方法、Romberg 方法和 Gauss 积分法以及它们的改进方法求解二重数值积分<sup>[2-3]</sup>. 这些数值方法简单易实现, 但是它们大都是基于等距节点分割的积分方法, 不考虑被积函数的形状, 导致积分值的精度不够高.

布谷鸟搜索算法 (cuckoo search, CS)<sup>[4]</sup> 作为一

种新兴的群智能优化算法, 设置的参数少, 易实现, 寻优过程具有较强的开发能力和鲁棒性, 但 CS 算法也存在着易陷入局部最优, 搜索精度较低的不足<sup>[5]</sup>. 针对该不足, 引入混沌变量作为 CS 算法的局部算子, 提出了一种基于混沌布谷鸟 (Chaotic Cuckoo Search, CCS) 的优化算法. 文献<sup>[6]</sup>表明, 利用混沌变量的最优化搜索策略可以摆脱算法陷入局部最优的困境. 因此, 本文将混沌思想引入到 CS 算法中, 可以较好地解决了 CS 算法易陷入局部最优的问题.

基于以上分析, 本文提出一种基于混沌布谷鸟优化算法的二重数值积分方法.

收稿日期: 2013-11-01; 修回日期: 2013-12-27

基金项目: 国家自然科学基金项目 (81060213); 广西教育厅项目 (201010LX057); 广西医科大学青年科学基金项目 (GX-MUYSF201342)

## 2 混沌布谷鸟算法

### 2.1 基本布谷鸟算法

CS的基本算法流程如下:

(1) 在  $D$  维解空间中随机生成  $N$  个鸟窝  $Nest_i(x_1, x_2, \dots, x_D), 1 \leq i \leq N$ , 计算每个鸟窝的适应度值, 记为  $f(Nest_i), 1 \leq i \leq N$ , 保留适应度最优的鸟窝位置以进行迭代。

(2) 设  $x_i^{(t)}$  为第  $i$  个鸟窝在第  $t$  代的鸟窝位置,  $Levy(\lambda)$  为随机搜索路径, 那么布谷鸟寻窝的路径和位置更新公式为

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus Levy(\lambda) \quad (i = 1, 2, \dots, n) \quad (1)$$

式中,  $\alpha$  为步长,  $\oplus$  为点对点乘法。

(3) 设鸟窝主人发现外来鸟蛋的概率为  $P$ , 产生一均匀分布随机数  $r \in [0, 1]$ , 若  $r > P$ , 则对所发现的鸟窝位置进行扰动, 否则不变。

### 2.2 混沌布谷鸟算法

混沌是非线性系统中的一种普遍现象, 是一种具有随机性的运动状态。在一定范围内混沌变量的变化具有随机性、遍历性和规律性, 这些特征使得混沌变量在优化搜索中可以让算法跳出局部最优, 保持群体的多样性, 进而改善算法的全局寻优性能。

与其他群智能优化算法一样, CS 算法在搜索过程中也容易陷入局部最优和搜索精度不够高等不足。为了解决这个问题, 本文把混沌变量引入到 CS 算法中, 利用混沌变量随机性、遍历性和规律性等特性使 CS 可以跳出局部最优, 从而增强 CS 的全局搜索能力, 进而提高 CS 算法解的精度。

一般地, 一个典型的混沌映射系统 Logistic 方程可以表述为

$$y_{n+1} = 4y_n(1 - y_n) \quad (2)$$

式中,  $n = 1, 2, \dots, n$ ,  $y_n$  为混沌变量,  $y_n \in [0, 1]$ 。

在改进算法中, 当算法连续迭代  $k$  次, 解的值都不发生变化时, 我们可以认为算法出现了早熟收敛现象。此时, 利用混沌的遍历性, 对鸟窝当前的最优位置  $x_{best}$  进行混沌优化, 方法如下所示。

将  $x_{best}$  通过方程映射到 Logistic 方程的定义域  $[0, 1]$  上:

$$y_1^k = \frac{x_{best} - x_{min}^k}{x_{max}^k - x_{min}^k} \quad (3)$$

对  $y_1^k$  通过  $y_{n+1}^k = 4y_n^k(1 - y_n^k) (n = 1, 2, \dots, n)$  进行  $T$  次迭代, 得到混沌序列  $y^k = (y_1^k, y_2^k, \dots, y_T^k)$ 。

然后, 将混沌序列通过下面的式子逆映射回原

解空间:

$$x_{best, m}^{*k} = x_{min}^k + (x_{max}^k - x_{min}^k) y_m^k \quad m = 1, 2, \dots, T \quad (4)$$

计算可行解序列中每个可行解矢量的适应值, 并保留适应值最优时对于的可行解矢量, 记为  $x_{best}^{*k}$ 。从当前鸟窝群随机选择一个鸟窝, 并用  $x_{best}^{*k}$  的位置矢量代替选出鸟窝的位置矢量。

## 3 求解二重数值积分的混沌布谷鸟算法

### 3.1 算法原理

在本文中我们提出了基于混沌布谷鸟算法的不等距节点分割的二重数值积分求解方法。该方法的主要思想: 首先, 在矩形积分区间上的两个方向各自随机产生一些分割点, 看作一个鸟窝; 然后通过布谷鸟的自寻优方法来对分割点进行优化, 以达到这些节点是根据函数图像的凹凸形状排列的目的; 最后以优化后的分割点为最优分割点, 进行积分求和。

### 3.2 算法实现

#### 3.2.1 鸟窝个体的定义

对于用混沌布谷鸟算法求解二重数值积分问题, 设定每一种分割方法用一个鸟窝来表示, 即为

$$Nest_i = \begin{pmatrix} x_1, x_2, \dots, x_j, \dots, x_m \\ y_1, y_2, \dots, y_k, \dots, y_n \end{pmatrix} \quad (j \leq m, k \leq n), \text{ 其中}$$

$x_j, y_k$  分别表示  $x$  轴和  $y$  轴上的第  $j$  和  $k$  个分割点,  $m, n$  为  $x$  轴和  $y$  轴上设定的分割点数目。初始鸟窝的生成方法: 分别在  $x$  轴和  $y$  轴上积分范围内按由小到大的顺序随机生成  $m, n$  个分割点, 作为一个鸟窝, 依次方法, 生成  $N$  个鸟窝。

#### 3.2.2 适应度函数的设计

每个鸟窝更新一次后, 依据下面的方法来评价分割方法的好坏: 在  $x$  轴和  $y$  轴上, 分别把要考察的布谷鸟鸟窝置于积分区间的左端点和右端点之间, 按升序排列,  $x$  轴方向上的第一个分割点对应  $y$  轴方向上的第一个分割点, 依次类推。这样  $x$  轴方向上就有  $m+2$  个节点和  $m+1$  个小段,  $y$  轴方向上同样有  $n+2$  个节点和  $n+1$  个小段, 这样矩形积分区域就被分割为  $(m+1) \cdot (n+1)$  个小矩阵。分别计算  $x$  轴和  $y$  轴上  $m+2$  和  $n+2$  个节点相邻节点之间的距离  $d_j$  和  $d_k$ , 其中  $j = 1, 2, \dots, m+1, k = 1, 2, \dots, n+1$  及  $area_{ij} = d_j \cdot d_k$ , 再计算出每个小矩阵的 4 个顶点和矩阵中点对应的函数值, 并找出这 5 个函数值中的最小值  $Min_{jk}$  和最大值  $Max_{jk}$ 。由此, 这样设计鸟窝个体的适应度:

$$f(i) = \frac{1}{2} \sum_{j=1}^{m+1} \sum_{k=1}^{n+1} area_{ij} \cdot |Min_{jk} \cdot Max_{jk}| \quad (5)$$

式中,  $i = 1, 2, \dots, N$  表示第  $i$  个鸟窝. 适应度越趋近于 0, 表明两个方向上对应的鸟窝个体(分割方法)越优良. 算法的终止条件设定为: 当适应度小于  $e$  时程序终止, 其中  $e$  是一个非常接近于 0 的数.

### 3.2.3 积分求和公式的设计

当算法结束后, 最优个体就是最优分割点, 把最优分割点代入积分求和表达式就可以求出数值积分的值. 在本文中, 积分求和表达式设计为

$$I = \frac{1}{5} \sum_{j=1}^{m+1} \sum_{k=1}^{n+1} area_{ij} (g_{jk} + g_{j+1,k} + g_{j,k+1} + g_{j+1,k+1} + g_{midjk}) \quad (6)$$

式中,  $g_{jk}, g_{j+1,k}, g_{j,k+1}, g_{j+1,k+1}, g_{midjk}$  分别为小矩形的 4 个顶点和矩形中点对应的函数值,  $area_{ij}$  为每个小矩形的面积.

### 3.2.4 算法步骤

采用混沌布谷鸟算法求解二重数值积分方法的算法流程图如图 1 所示.

## 4 实验与分析

为了检验本文算法的有效性, 本文以文献[7]中的 2 个算例作为例子进行仿真. 为了结果对比的客观公正, 本文算法和文献[7]一样都采用 Matlab6.5 作为编程软件. 文中涉及到的参数设置如下: 种群规模  $N = 100$ , 步长  $\alpha = 1.5$ , 概率  $P = 0.35$ ,  $k = 5$ , 平均独立运行 15 次.

$$\text{例 1: } \int_0^1 dx \int_0^1 \frac{dy}{(1+x+y)^2} = 2\ln 2 - \ln 3$$

在本例中, 被积函数的  $x, y$  地位是对称的. 已知此积分的准确值为 0.287 682 10, 几种算法的积分结果与相对误差如表 1 所示. 从表 1 我们可以看出, 相对于文献[7]和基本布谷鸟算法, 本文算法可以得到更精确的积分值. 实验结果也可以看出本文算法的收敛速度是比较快的.

表 1 几种算法的积分结果对比

| 积分方法                 | 分割点数目 | 积分值         | 相对误差        |
|----------------------|-------|-------------|-------------|
| 复化梯形法 <sup>[7]</sup> | 16    | 0.288 080 3 | 0.001 384 3 |
| 粒子群算法 <sup>[7]</sup> |       | 0.287 903 1 | 0.000 768 1 |
| 布谷鸟算法                |       | 0.287 812 1 | 0.000 130 0 |
| 本文算法                 |       | 0.287 695 2 | 0.000 013 1 |
| 复化梯形法 <sup>[7]</sup> | 64    | 0.287 708 4 | 0.000 091 7 |
| 粒子群算法 <sup>[7]</sup> |       | 0.287 698 6 | 0.000 057 2 |
| 布谷鸟算法                |       | 0.287 692 2 | 0.000 010 1 |
| 本文算法                 |       | 0.287 685 2 | 0.000 003 1 |

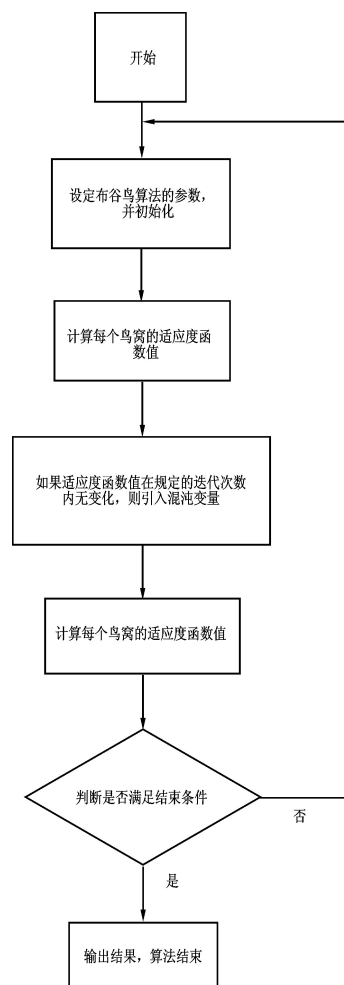


图 1 混沌布谷鸟算法求解二重数值积分的算法流程图

例 2 求由直线  $x + y = c, x + y = d, y = ax, y = bx, (0 < c < d, 0 < a < b)$  所围成闭区域  $D$  的面积  $S$ :

$$S = \iint_D dx dy = \int_a^b du \int_c^d \frac{u}{(1+v)} dv = \frac{(b-a)(d^2 - c^2)}{2(1+a)(1+b)}$$

在本例中, 被积函数的  $u, v$  地位是不对称的, 且函数在区域  $D$  中变化比较平缓. 和文献[7]一样, 在本文中, 设  $a = 1, b = 2; c = 1, d = 2$ . 已知该积分的准确值为 0.250 000 0, 采用几种方法得到的积分结果如表 2 所示, 可以看出, 算法在迭代次数为 16 代左右就收敛了, 收敛速度是很快的.

## 5 结束语

提出了一种基于混沌布谷鸟 (Chaotic Cuckoo Search, CCS) 的优化算法, 并利用其求解二重数值 (下转第 154 页)

实现, CFIR 补偿 CIC 滤波器的衰减并具备 2 倍抽取滤波特性, 且 HBF 频谱具备关于中心频率对称的特性, 使得有一半的系数为 0, 与普通 FIR 滤波器相比, HBF 具备更少的资源消耗<sup>[8]</sup>, 从而从整体上减少整体滤波器组的消耗。

## 5 结束语

本文提出一种降低抽取滤波器组功耗、面积的系统级设计及电路实现方法, 在保证系统性能的前提下大大节省了功耗和面积, 经过对比得出本文的方法相对于 Hogenaur 结构的 CIC 减少了 21% 的功耗和 5% 的面积消耗。对应到整个滤波器组而言, 由于 CIC 工作在高频处, 后续滤波器的工作频率相对较低, CIC 的功耗优化对整个滤波器的功耗减少也有相当的意义。

## 参考文献:

- [1] Koen Cornelissens, Michiel Steyaert, "Design Considerations for Cascade ADC. [J]. IEEE Transactions On Circuits And Systems—II: Express Briefs, 2008, 55 (5):389-393.
- [2] Norsworthy S, Schreier R, Temes G, Delta-sigma data converters: theory, design, and simulation [M]. New York, NY, USA: IEEE Press, 1996.

- [3] Hogenauer E B. An economical class of digital filters for decimation and interpolation [J]. IEEE Tran, 1981, 29(2):155-162.
- [4] 蔡友. Sigma-Delta ADC 中抽取滤波器的研究与实现 [D]. 浙江: 浙江大学, 2007.
- [5] Jang Y, Yang S. Non-recursive cascaded integrator-comb decimation filters with integer multiple factors [C]// Proc 44th IEEE Midwest Symposium on Circuits and Systems. [s. l.]: IEEE, 2001: 130-133.
- [6] A V 奥本海姆, R W 谢弗, J R 巴克. 离散时间信号处理 [M]. 刘树棠, 黄建国, 译. 西安: 西安交通大学出版社, 2001.
- [7] Himan shu Bhatnagar. 高级 ASIC 芯片综合-使用 Synopsys Design Compiler Physical Compiler 和 Primetime [M]. 张文俊, 译. 北京: 清华大学出版社, 2007.
- [8] 马绍宇, 韩雁, 蔡友. ADC 中数字抽取滤波器的多级实现 [J]. 天津大学学报, 2007, 40(12): 1421-1425.

## 作者简介:

- 王龙生 男, (1988-), 硕士. 研究方向为信号处理与系统建模.
- 谢亮 男, (1982-), 博士研究生. 研究方向为 ASIC 设计、ADC、红外传感方向.
- 金湘亮 男, (1974-), 博士, 教授. 研究方向为 SOC 设计、系统设计.

(上接第 150 页)

表 2 几种算法的积分结果对比

| 积分方法                 | 分割点数目 | 积分值         | 相对误差        |
|----------------------|-------|-------------|-------------|
| 复化梯形法 <sup>[7]</sup> | 16    | 0.250 343 3 | 0.001 373 2 |
| 粒子群算法 <sup>[7]</sup> |       | 0.250 178 5 | 0.000 714 0 |
| 布谷鸟算法                |       | 0.250 152 4 | 0.000 152 1 |
| 本文算法                 |       | 0.250 012 7 | 0.000 011 4 |
| 复化梯形法 <sup>[7]</sup> | 64    | 0.250 021 4 | 0.000 085 7 |
| 粒子群算法 <sup>[7]</sup> |       | 0.250 013 8 | 0.000 055 3 |
| 布谷鸟算法                |       | 0.250 011 5 | 0.000 010 8 |
| 本文算法                 |       | 0.250 000 9 | 0.000 000 1 |

积分. 通过 4 个典型实例的数值仿真结果表明, 本文算法能用较少的分割点就可得到精度相对较高的积分值, 并且收敛速度快, 算法稳定, 在数值计算和工程实际应用领域有一定的参考和应用价值。

## 参考文献:

- [1] 周永权, 张明, 赵斌. 基于进化策略方法求任意函数的数值积分 [J]. 计算机学报, 2008, 21(2): 196-206.
- [2] 韦杏琼, 周永权. 基于粒子群算法的数值积分方法研究 [J]. 微电子学与计算机, 2009, 26(7): 117-119.

- [3] 张明. 基于进化计算的数值方法研究 [D]. 南宁: 广西民族大学, 2007.
- [4] YANG Xin-shen, DEB Suash. Cuckoo search via levy flights [J]. Proceedings of World Congress on nature & Biologically Inspired Computing, India: IEEE Publications, 2009: 210-214.
- [5] Zheng Hongqiang, Zhou Yongquan, Luo Qifang. A hybrid cuckoo search algorithm-GRASP for vehicle routing problem [J]. Journal of Convergence Information Technology, 2013, 8(3): 821-829
- [6] 王雷, 凌翔, 胡剑浩. 异构多核协作系统的混沌离散粒子群 NoC 映射算法 [J]. 计算机科学, 2011, 38(9): 287-303.
- [7] 韦杏琼, 周永权. 基于粒子群算法的二重积分方法研究 [J]. 计算机工程与设计, 2009, 30(20): 4705-4707.

## 作者简介:

- 凌巍高 (1984-), 硕士. 研究方向为计算智能及其应用.
- 欧旭 (1979-), 硕士. 研究方向为计算智能和模式识别.
- 罗德相 (1980-), 硕士. 研究方向为计算智能和数据挖掘.