

斐波那契树优化算法全局随机性概率收敛分析

董 易[†], 吕丹桔, 王 霞, 王耀民, 李 鹏, 施心陵

(云南大学 信息学院, 昆明 650504)

摘 要: 为改进斐波那契树优化算法的收敛性能, 提出斐波那契树末梢自适应半径参数, 使得算法在最优解邻域附近收敛能力显著提高. 基于斐波那契树结构的全局随机性对斐波那契树优化算法的收敛性进行分析和证明. 通过测试函数的求解精度比较、独立重复求解的收敛达标率比较实验验证了斐波那契树优化算法的收敛性能.

关键词: 斐波那契树优化算法; 末梢自适应半径; 全局随机性; 收敛性

中图分类号: TP301.6

文献标志码: A

A global randomness-based probability convergence analysis of Fibonacci tree optimization algorithm

DONG Yi[†], LV Dan-ju, WANG Xia, WANG Yao-min, LI Peng, SHI Xin-ling

(School of Information Science and Engineering, Yunnan University, Kunming 650504, China)

Abstract: A Fibonacci tree-end self-adaptive radius parameter is proposed to effectively enhance convergence of Fibonacci tree optimization (FTO) algorithm at neighborhood of optima. The convergence of FTO is analyzed and proved on the basis of global randomness of Fibonacci tree. By comparing both the precision in solving benchmark functions and the qualified rate of repeated and independent solutions, the convergence performance of FTO is also examined and confirmed.

Keywords: Fibonacci tree optimization algorithm; Fibonacci tree-end self-adaptive radius; global randomness; convergence

0 引 言

斐波那契法已被证明是最优化分割方法中针对一维单峰函数搜索问题的最优策略. 用试探点不断缩短最优解所在区间长度, 区间缩短比率为斐波那契数列末尾两项比值, 其极限为黄金分割数, 且所使用的试探点或迭代次数最少. 斐波那契法是黄金分割法的极限形式, 由于黄金分割法简单易行, 在实际优化问题中得到了广泛应用, 如风速计故障诊断^[1]、钻井液流变参数优化^[2]、光伏系统能量捕获控制优化^[3]、水稻合理密植^[4]、服装工业设计等领域^[5]. 然而, 斐波那契法只适用于一维单峰区间的优化. 相对于元启发式搜索算法, 斐波那契法的适用范围太窄. 近年来, 针对元启发式搜索算法的理论和应用研究不断深入. 元启发式搜索算法还未形成公认定义, 一些学者给出了较有代表性的提法. Osman 等^[6]认为元启发式搜索算法是一个迭代过程, 利用不同策略, 在

搜索空间中采用探索和开发两类操作有效搜索最优解; Stutzle 等^[7]认为元启发式搜索算法是一种高级策略, 其搜索过程是一种概率决策过程, 以一种智能的形式进行随机搜索, 区别于纯随机搜索的盲目搜索.

目前, 常见的元启发式搜索算法包括遗传算法 (GA)^[8]、粒子群算法 (PSO)^[9]、差分进化算法 (DE)^[10]、人工蜂群算法 (ABC)^[11] 以及列维飞行粒子群算法 (LFPSO)^[12] 等. 元启发式搜索算法的主要灵感来源于对自然界一些规律现象的仿生, 其策略引入随机变量, 不要求目标函数连续, 全局优化能力较好, 具有一系列的参数且需要对不同的问题进行针对性的调整^[13], 但也各自存在待研究和改进的特点. 例如: 遗传算法解空间与搜索空间的量化误差和搜索性能问题^[14], 粒子群算法在当前最优解的引导下容易陷入局部最优解^[15], 差分进化算法个体多样性减弱导致在局部极值附近早熟收敛问题等^[16].

收稿日期: 2017-01-26; 修回日期: 2017-03-30.

基金项目: 国家自然科学基金项目 (61364024); 云南省自然科学基金重点项目 (2013FA008).

责任编委: 陈家伟.

作者简介: 董易 (1984—), 男, 博士生, 从事智能优化算法的研究; 施心陵 (1956—), 男, 教授, 博士生导师, 从事智能优化算法、自适应信号处理与信息系统等研究.

[†]通讯作者. E-mail: 625068010@qq.com

斐波那契树优化算法(FTO)是基于斐波那契法的一种元启发式搜索算法.在斐波那契法的基础上扩展到 n 维欧氏空间,能求解单峰或多峰函数全局最优解,适用范围得到极大拓宽.FTO算法构造了一个斐波那契树结构,全局随机性是该结构的重要特性,使得FTO算法不易陷入局部最优解.然而,FTO收敛到最优解邻域附近时存在收敛能力下降的问题.

本文提出FTO算法的斐波那契树末梢自适应半径参数,使得算法的收敛性能得到显著改进.基于斐波那契树结构的全局随机性,分析并证明FTO算法的收敛性.通过3个实验对FTO算法的收敛性能进行对比分析:1)增加末梢自适应半径参数前后的FTO算法收敛精度对比,2)改进后的FTO算法与其他几个元启发式搜索算法求解精度对比,3)利用PSO算法容易陷入局部最优解的特点,通过独立重复求解实验,统计求解到全局最优解的达标次数和达标率,对比和验证FTO算法基于全局随机性的收敛能力.

1 斐波那契树优化算法

FTO算法基于斐波那契法的思想,在 n 维空间中构造一个斐波那契树结构,该结构具有全局随机性,不易陷入局部最优解.斐波那契树结构在搜索空间中生成的随机点满足均匀分布,能以既定概率命中全局最优解所在邻域,结构内部的迭代过程使得算法能够快速收敛.

1.1 基本结构

斐波那契树结构的生成基于一个基本结构,如图1所示.

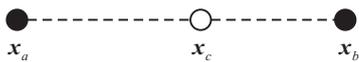


图1 FTO基本结构

其中: $\boldsymbol{x} = [x_1, x_2, \dots, x_n]^T$ 是 n 维欧氏空间中的向量,3向量 $\boldsymbol{x}_a, \boldsymbol{x}_b, \boldsymbol{x}_c$ 的模满足比例关系

$$\frac{\|\boldsymbol{x}_c - \boldsymbol{x}_a\|}{\|\boldsymbol{x}_b - \boldsymbol{x}_a\|} = \frac{\|\boldsymbol{x}_b - \boldsymbol{x}_c\|}{\|\boldsymbol{x}_c - \boldsymbol{x}_a\|} = \frac{F_i}{F_{i+1}}, \quad (1)$$

其中 F_i 为斐波那契数列第 i 项.同时,3个向量满足线性关系

$$\boldsymbol{x}_c = \frac{F_i}{F_{i+1}}(\boldsymbol{x}_b - \boldsymbol{x}_a) + \boldsymbol{x}_a. \quad (2)$$

称 $\boldsymbol{x}_a, \boldsymbol{x}_b$ 为端点, \boldsymbol{x}_c 为分割点.

1.2 斐波那契树

基于FTO基本结构,采用两个迭代规则构造斐波那契树.设FTO算法当前处理的点集为 S ,集合大小满足斐波那契数,即 $|S| = F_i$.

规则1 令基本结构的端点 $\{\boldsymbol{x}_a\} = S, \{\boldsymbol{x}_b\} = \{\boldsymbol{x} \mid \boldsymbol{x} \in \prod_{i=1}^D [x_{lb}^i, x_{ub}^i]\}$, D 为向量维度, x_{lb} 和 x_{ub} 分别为

为向量元素取值范围的下界和上界,向量 $\boldsymbol{x} = [x_1, x_2, \dots, x_d, \dots, x_n]^T$ 的分量 x_d 是满足均匀分布的随机变量 X ,概率分布为 $P(X) = U(x_{lb}, x_{ub})$,根据式(2)求解出分割点集合 $\{\boldsymbol{x}_c\}$.

规则2 令基本结构的端点 $\boldsymbol{x}_a = \boldsymbol{x}_{best} \in S$ 为 S 中当前最优解, $\{\boldsymbol{x}_b\} = \{\boldsymbol{x} \mid \boldsymbol{x} \in S \wedge \boldsymbol{x} \neq \boldsymbol{x}_a\}$,根据式(2)求解出分割点集合 $\{\boldsymbol{x}_c\}$.

按两个规则迭代完成之后,合并 S 、新的端点和分割点,保留前 F_{i+1} 个较优点,丢弃其余点,更新点集 S ,同时集合大小更新为 $|S| = F_{i+1}$.

图2为斐波那契树生成过程示意图.黑色实心点为当前处理的端点 \boldsymbol{x}_a 和 \boldsymbol{x}_b .实线白点为分割点 \boldsymbol{x}_c .每一层虚线白点为上一轮迭代处理的点集,点集 S 最大为 $|S| = F_n$.

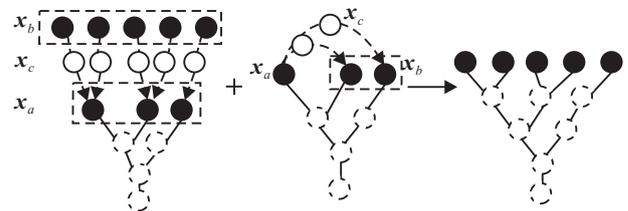


图2 斐波那契树生成过程

1.3 FTO算法流程

Step 1:初始化集合 S ,集合大小为 $|S| = F_i$;

Step 2:按规则1生成全局随机点集 $\{\boldsymbol{x}_b\}$ 和分割点集 $\{\boldsymbol{x}_{c1}\}$;

Step 3:按规则2生成分割点集 $\{\boldsymbol{x}_{c2}\}$;

Step 4:合并集合 $S, \{\boldsymbol{x}_b\}, \{\boldsymbol{x}_{c1}\}, \{\boldsymbol{x}_{c2}\}$,保留前 F_{i+1} 个较优点,更新点集 S ,同时集合大小更新为 $|S| = F_{i+1}$;

Step 5:判断集合大小 $|S|$ 是否为给定斐波那契数 F_n ,如果不满足,则回到Step 2;

Step 6:更新点集 S ,令其大小为给定斐波那契数 F_{nested} ,即 $|S| = F_{nested}, F_{nested} < F_n$;

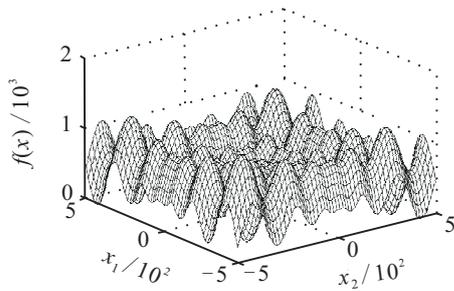
Step 7:判断算法是否达到最大迭代次数,否则回到Step 2.

1.4 全局随机性

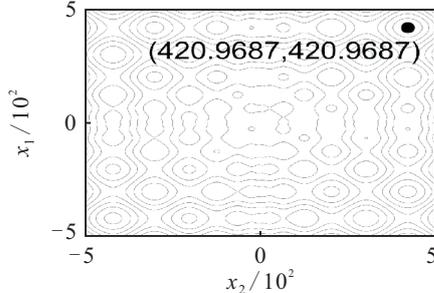
根据斐波那契树生成规则1,斐波那契树当前集合 S 的每次更新都需要生成定义域内满足均匀分布的随机点,概率为 p ,该特性被称为全局随机性.

与PSO算法相比较,PSO的算法机制存在容易陷入局部最优解的特点.FTO的全局随机性至少有概率 p ,使得算法避免陷入局部最优.

图3~图5分别表示采用FTO和PSO求解表1中测试函数 f_5 ,向量维数为2维的解坐标累积分布示意图.实验参数如表2所示,求解结果如表3所示.表1中 U 为单峰函数, M 为多峰函数, R 为坐标变换.

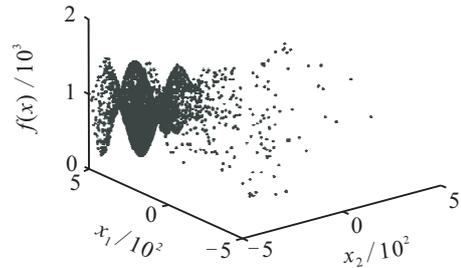


(a) f_5 函数

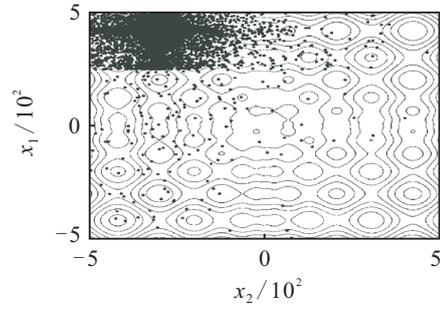


(b) f_5 等高线图及最优解坐标

图 3 f_5 函数等高线图及最优解坐标



(a) PSO 解坐标累积分布 1



(b) PSO 解坐标累积分布 2

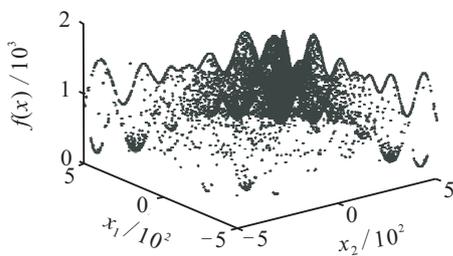
图 4 PSO 解坐标累积分布

表 1 测试函数

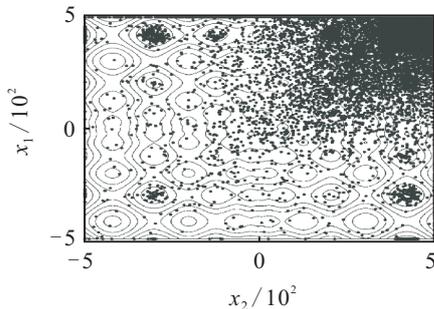
f	名称	定义域	最优解	特性	表达式
f_1	Sphere	$[-100, 100]$	0	U	$f_1 = \sum_{i=1}^n x_i^2$
f_2	Schwefel2.22	$[-10, 10]$	0	U	$f_2 = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $
f_3	Rosenbrock	$[-10, 10]$	0	U	$f_3 = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$
f_4	Noise	$[-1.28, 1.28]$	0	U	$f_4 = \sum_{i=1}^n ix_i^4 + \text{random}[0, 1]$
f_5	Schwefel2.26	$[-500, 500]$	0	M	$f_5 = 418.98288727243369n - \sum_{i=1}^n x_i \sin(\sqrt{ x_i })$
f_6	Rastrigin	$[-5.12, 5.12]$	0	M	$f_6 = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$
f_7	Ackley	$[-32, 32]$	0	M	$f_7 = -20 \exp \left\{ -0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right\} - \exp \left\{ \frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right\} + e + 20$
f_8	Griewank	$[-600, 600]$	0	M	$f_8 = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
f_9	Penalized1	$[-50, 50]$	0	M	$f_9 = \frac{\pi}{n} \left\{ 10 \sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\} + \sum_{i=1}^n u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{1}{4}(x_i + 1)$ $u_{x_i, a, k, m} = \begin{cases} k(x_i - a)^m, & x_i > a; \\ 0, & -a \leq x_i \leq a; \\ k(-x_i - a)^m, & x_i < -a. \end{cases}$
f_{10}	Penalized2	$[-50, 50]$	0	M	$f_{10} = \frac{1}{10} \left\{ \sin^2(\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)^2 [1 + \sin^2(2\pi x_{i+1})] \right\} + \sum_{i=1}^n u(x_i, 10, 100, 4)$
f_{11}	Rotated Schwefel	$[-500, 500]$	0	R	$f_{11} = 418.9828n - \sum_{i=1}^n z_i$ $z_i = \begin{cases} y_i \sin(\sqrt{ y_i }), & y_i \leq 500; \\ 0, & \text{otherwise.} \end{cases}$ $y_i = y'_i + 420.96$ $y' = Mx, \text{ } M \text{ is an orthogonal matrix}$

续表

f	名称	定义域	最优解	特性	表达式
f_{12}	Rotated Rastrigin	$[-5.12, 5.12]$	0	R	$f_{12} = \sum_{i=1}^n [y_i^2 - 10 \cos(2\pi x y_i) + 10]$ $y' = Mx, M \text{ is an orthogonal matrix}$
f_{13}	Rotated Ackley	$[-32, 32]$	0	R	$f_{13} = -20 \exp \left\{ -0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n y_i^2} \right\} - \exp \left\{ \frac{1}{n} \sum_{i=1}^n \cos(2\pi y_i) \right\} + e + 20$ $y = Mx, M \text{ is an orthogonal matrix}$
f_{14}	Rotated Griewank	$[-600, 600]$	0	R	$f_{14} = \frac{1}{4000} \sum_{i=1}^n y_i^2 - \prod_{i=1}^n \cos\left(\frac{y_i}{\sqrt{i}}\right) + 1$ $y = Mx, M \text{ is an orthogonal matrix}$
f_{15}	Sum Square	$[-10, 10]$	0	U	$f_{15} = \sum_{i=1}^n i x_i^2$
f_{16}	Step	$[-100, 100]$	0	U	$f_{16} = \sum_{i=1}^n ([x_i + 0.5])^2$
f_{17}	Quartic	$[-1.28, 1.28]$	0	U	$f_{17} = \sum_{i=1}^n i x_i^4$
f_{18}	Levy	$[-10, 10]$	0	M	$f_{18} = \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + \sin^2(3\pi x_1) + x_n - 1 [1 + \sin^2(3\pi x_n)]$
f_{19}	Schaffer	$[-100, 100]$	0	M	$f_{19} = 0.5 + \sin^2\left(\sqrt{\sum_{i=1}^n x_i^2}\right) - 0.5 / \left(1 + 0.001\left(\sum_{i=1}^n x_i^2\right)\right)^2$
f_{20}	Alpine	$[-10, 10]$	0	M	$f_{20} = \sum_{i=1}^n x_i \sin(x_i) + 0.1 x_i $
f_{21}	Non-continuous Rastrigin	$[-5.12, 5.12]$	0	M	$f_{21} = \sum_{i=1}^n [y_i^2 - 10 \cos(2\pi y_i) + 10]$



(a) FTO解坐标累积分布1



(b) FTO解坐标累积分布2

图5 FTO解坐标累积分布

f_5 为典型的多峰函数,存在多个局部最优解.可以看出,PSO在求解过程中逐渐陷入局部最优解而无法跳出.FTO在PSO陷入的局部最优解区域同样有解坐标的累积分布,如图4(b)和图5(b)所示.反映出FTO在全球随机性的作用下跳出该区域并搜索到全局最优解所在区域.

表2 算法实验参数

迭代次数	2维: 1000, 10维: 2000
重复次数	30
PSO	惯性权重: 0.9~0.5; 加速常数 c_1 : 2; 加速常数 c_2 : 2; 种群个数: 30
FTO	斐波那契数列 $F_i : i = 7$

表3 函数 f_5 求解结果

算法	解值	解坐标
FTO	0	[420.968 7, 420.968 7]
PSO	118.438 3	[-302.524 9, 420.968 7]

图6为FTO和PSO求解 f_5 的收敛曲线示意图.可以看出,PSO陷入局部最优解后收敛曲线变化停滞.FTO的收敛曲线有一个跳出局部最优解的变化过程,并继续收敛到全局最优解.

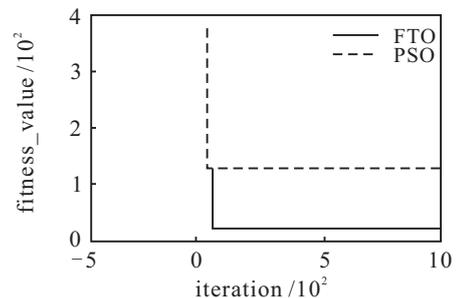


图6 FTO和PSO求解 f_5 的收敛曲线

2 末梢自适应半径参数

斐波那契树的末梢自适应半径 r 定义为当前集合 S 中最佳解 \mathbf{x}_{best} 和最差解 $\mathbf{x}_{\text{worst}}$ 的欧氏距离, 即

$$r = \|\mathbf{x}_{\text{worst}} - \mathbf{x}_{\text{best}}\|. \quad (3)$$

图7为末梢自适应半径参数取值示意图. 实线白点为斐波那契树当前集合 S 中最佳解 \mathbf{x}_{best} , 黑色点为 S 中最差解 $\mathbf{x}_{\text{worst}}$, r 为末梢自适应半径参数.

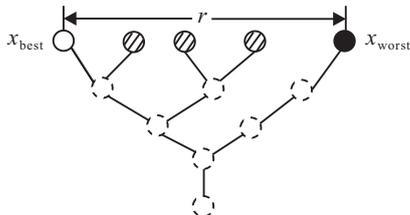


图7 末梢自适应半径取值

在加入参数条件下, 对基本结构的端点 $\{\mathbf{x}_b\}$ 取随机点的范围在当前最佳解 \mathbf{x}_{best} 周围的 $\pm r$ 范围内.

图8为在半径 r 的约束下, 端点 $\{\mathbf{x}_b\}$ 的取值范围示意图. 每个端点 $\forall \mathbf{x} \in \{\mathbf{x}_b\}$ 在 \mathbf{x}_{best} 的半径 r 范围取值并服从均匀分布.

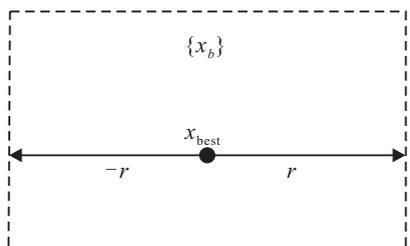


图8 参数 r 条件下的端点 \mathbf{x}_b 取值范围

引入参数 r 后, 斐波那契树的构造引入如下新的规则.

规则3 令基本结构端点 $\{\mathbf{x}_a\} = S, \{\mathbf{x}_b\} = \{\mathbf{x} \mid \mathbf{x} \in \prod_{i=1}^D [x_{lb2}^i, x_{ub2}^i]\}$. 当前最佳解为 $\mathbf{x}_{\text{best}} \in S$, 则端点 $\forall \mathbf{x} \in \{\mathbf{x}_b\}$ 的取值下界和上界分别为 $(x_{lb2})_{D \times 1} = \mathbf{x}_{\text{best}} - r, (x_{ub2})_{D \times 1} = \mathbf{x}_{\text{best}} + r$, 向量 $\mathbf{x} = [x_1, x_2, \dots, x_d, \dots, x_n]^T$ 的分量 x_d 是满足均匀分布的随机变量 X , 概率分布为 $P_r(X) = U(x_{lb2}, x_{ub2})$, 根据式(2)求解出分割点集合 $\{\mathbf{x}_c\}$.

3 FTO算法的收敛性分析

本节基于斐波那契树结构的全局随机性对斐波那契树优化算法的收敛性进行分析并证明.

3.1 FTO算法收敛机制分析

斐波那契树的当前最佳解 $\mathbf{x}_{\text{best}} \in S$ 通过搜索迭代过程进行更新. 对于FTO的第 t 次迭代, 至少存在 $f(\mathbf{x}_{\text{best}}^{t+1}) \leq f(\mathbf{x}_{\text{best}}^t)$, f 为测试函数(只讨论求最小值的情况). 根据斐波那契树的全局随机性特点生成随

机点集 $\{\mathbf{x}_b\}$ 服从均匀分布, FTO搜索到全局最优解的概率为

$$P(X) \sim U(x_{lb}, x_{ub2}) \leq p \leq 1. \quad (4)$$

这一性质使得FTO算法不易陷入局部最优解.

当 \mathbf{x}_{best} 逼近全局最优解 \mathbf{x}^* 一个很小的 ε -邻域时, 根据连续型随机变量的性质可知, p 趋近于0. 引入斐波那契树末梢自适应半径参数 r 后, 规则3的迭代过程

$$P_r(X) \sim U(x_{lb2}, x_{ub2}) \leq p \leq 1, \quad (5)$$

则当 $0 < r < \varepsilon$ 时, FTO将以概率 p 继续逼近 \mathbf{x}^* , 收敛性能得到提高.

3.2 FTO算法收敛性证明

定理1 FTO以概率1收敛到全局最优解, 即有 $P\{T < \infty\} = 1$.

证明 FTO进行 $T < \infty$ 次迭代, 由斐波那契树结构的当前集合 S 得到一个函数值为非升(只讨论求最小值的情况)的序列 $\{\mathbf{x}_T\} = \{\mathbf{x}_t \mid \mathbf{x}_{\text{best}} \in S_t\}, t = 1, 2, \dots, T$. 设 $P(t) = P(|\mathbf{x}^* - \mathbf{x}_t| \geq \varepsilon)$, 根据斐波那契树生成规则, $P(t) = P_X(t) + P_r(t)$, $P_X(t)$ 为规则1在 \mathbf{x} 定义域内生成均匀分布随机点的概率, $P_r(t)$ 为规则3在半径参数 r 定义的区域生成均匀分布随机点的概率, 所以 $0 < P(t) < 1$. t 次迭代后FTO未命中 \mathbf{x}^* 的 ε -邻域的概率为 $\tilde{P}(t) \leq \prod_{i=1}^t (1 - P(i))$, 有 $\lim_{t \rightarrow \infty} \tilde{P}(t) = 0$, 所以 $P\{T < \infty\} \geq P(t) = 1 - \tilde{P}(t), P\{T < \infty\} > 0$. 令 $t \rightarrow \infty$, 则 $P\{T < \infty\} = 1$, 所以FTO以概率1收敛到全局最优解. \square

定理2 FTO算法逼近全局最优解的概率 $p \geq 1/2r$.

证明 根据定理1, 当FTO依概率收敛到全局最优解 \mathbf{x}^* 的 ε -邻域时, 根据斐波那契树生成规则3可知, FTO基本结构中的随机变量 X 满足均匀分布 $P_r(X) \sim U(\mathbf{x}_{\text{best}} - r, \mathbf{x}_{\text{best}} + r) = 1/(\mathbf{x}_{\text{best}} + r - (\mathbf{x}_{\text{best}} - r)) = 1/2r$. \square

4 实验与分析

为测试FTO算法的收敛性, 进行3个对比实验: 1) 引入末梢自适应半径参数后, FTO算法的收敛性能对比; 2) 与其他几个元启发式搜索算法的求解精度进行对比; 3) 验证FTO基于全局随机性收敛到全局最优解的性能. 利用PSO算法易陷入局部最优解的特点进行对比. 通过独立重复求解实验, 在有限的迭代次数范围内, 统计算法收敛到全局最优解的达标率. 实验选用的21个常用无约束测试函数^[12]如表

1所示,算法的实验参数设置如表2所示.

4.1 末梢自适应半径参数对比实验与分析

表4给出了对比实验结果,测试函数维数为2维,求解精度低于 $\varepsilon = 1.0 \times 10^{-100}$ 时取值为0.就最优解的结果对比而言,改进前后的FTO都求解到全局最优解,就均值和标准差结果而言,30次独立重复实验对

于多峰函数的求解都没有陷入局部最优解,验证了全局随机性避免算法陷入局部最优解的能力. f_6 、 f_{11} 、 f_{16} 等3个函数最优解结果相同, f_{16} 均值相同.改进后的FTO算法最优解收敛精度远高于改进前的FTO,均值全部高于改进前的FTO.由此说明末梢自适应半径参数的引入显著提高了FTO算法的收敛性能.

表4 末梢自适应半径参数对比实验结果

f	改进前FTO			改进后FTO		
	最优解	均值	标准差	最优解	均值	标准差
f_1	8.6132e-37	3.9803e-05	1.7516e-04	0.0000e+00	0.0000e+00	0.0000e+00
f_2	4.8572e-17	7.6396e-04	1.7995e-03	0.0000e+00	0.0000e+00	0.0000e+00
f_3	4.4475e-09	1.3457e-03	3.0235e-03	1.2326e-32	1.0888e-04	5.2814e-04
f_4	6.6187e-84	4.7167e-14	1.9197e-13	0.0000e+00	0.0000e+00	0.0000e+00
f_5	1.8645e-11	3.0740e-02	7.0185e-02	0.0000e+00	0.0000e+00	0.0000e+00
f_6	0.0000e+00	1.2794e-02	1.8697e-02	0.0000e+00	0.0000e+00	0.0000e+00
f_7	9.1305e-13	1.5926e-02	3.1896e-02	0.0000e+00	3.5527e-16	1.0840e-15
f_8	8.2920e-03	3.4220e-02	1.8439e-02	0.0000e+00	5.4288e-03	3.3297e-03
f_9	9.4754e-24	5.1493e-05	2.1291e-04	1.5705e-32	1.5705e-32	8.3511e-48
f_{10}	7.7896e-18	5.5070e-05	1.6466e-04	1.4998e-33	1.4998e-33	0.0000e+00
f_{11}	0.0000e+00	6.0899e-02	1.4024e-01	0.0000e+00	0.0000e+00	0.0000e+00
f_{12}	4.0236e-10	8.5406e-03	1.4265e-02	0.0000e+00	0.0000e+00	0.0000e+00
f_{13}	1.0516e-12	1.2885e-02	5.0879e-02	0.0000e+00	1.1842e-16	6.4863e-16
f_{14}	8.8478e-05	1.7861e-02	1.0566e-02	0.0000e+00	2.7125e-03	5.9122e-03
f_{15}	1.9683e-33	1.8571e-05	9.9193e-05	0.0000e+00	0.0000e+00	0.0000e+00
f_{16}	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00	0.0000e+00
f_{17}	9.1910e-69	1.6335e-14	8.9429e-14	0.0000e+00	0.0000e+00	0.0000e+00
f_{18}	6.9714e-12	4.6285e-03	1.0273e-02	1.3498e-31	1.9097e-11	4.4798e-11
f_{19}	1.8652e-13	1.7744e-03	2.2639e-03	0.0000e+00	3.1266e-04	9.5402e-04
f_{20}	6.9198e-15	1.7230e-04	2.7963e-04	0.0000e+00	3.6787e-05	1.4921e-04
f_{21}	7.4039e-12	2.0152e-02	3.6983e-02	0.0000e+00	0.0000e+00	0.0000e+00

4.2 收敛精度对比实验与分析

采用FTO求解表1中 $f_1 \sim f_8$ 、 f_{10} 、 f_{16} 这10个函数,并与其他几个元启发式搜索算法进行比较,结果如表5所示.该结果来自于文献[12,17],与遗传算法(GA)、差分进化算法(DE)、人工蜂群算法(ABC)、

粒子群算法(PSO)、列维飞行粒子群算法(LFPSO)进行收敛精度比较.各算法实验参数均为文献给出的最优设置.10个测试函数维数取30维,测试函数评价次数(FEs)取500000次,记录重复实验30次结果的均值.求解精度低于 $\varepsilon = 1.0 \times 10^{-12}$ 时记为0.

表5 求解精度对比实验结果(Opt.为全局最优解)

f	Opt.	GA	DE	ABC	LFPSO	PSO	FTO
f_1	0.00e+00	1.11e+03	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
f_2	0.00e+00	1.10e+01	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
f_3	0.00e+00	1.96e+05	1.82e+01	8.88e-02	2.31e+01	2.44e+01	1.56e+01
f_4	0.00e+00	1.81e-01	1.36e-03	3.00e-02	1.27e-03	0.00e+00	0.00e+00
f_5	0.00e+00	9.76e+02	2.30e+03	1.30e-02	1.18e+03	1.23e+03	5.92e+03
f_6	0.00e+00	5.29e+01	1.17e+01	0.00e+00	0.00e+00	2.37e+01	0.00e+00
f_7	0.00e+00	1.47e+01	0.00e+00	0.00e+00	0.00e+00	0.00e+00	0.00e+00
f_8	0.00e+00	1.06e+01	1.48e-03	0.00e+00	0.00e+00	1.94e-02	0.00e+00
f_{10}	0.00e+00	1.25e+02	2.20e-03	0.00e+00	0.00e+00	0.00e+00	0.00e+00
f_{16}	0.00e+00	1.17e+03	0.00e+00	0.00e+00	0.00e+00	3.33e-02	0.00e+00

由实验条件可知:FTO有8个函数达标,表现最优.ABC算法和LFPSO算法都有7个函数达标,其他算法达标数量都较差.ABC算法对于 f_3 和 f_5 的求解精度是最好的.总体而言,GA、DE和PSO表现一般,

ABC、LFPSO和FTO各有优势.

4.3 收敛达标率对比实验与分析

用FTO和PSO求解表1中的21个测试函数,其自变量分别为2维和10维,独立重复进行30次并对达

标率进行统计,验证FTO基于全局随机性收敛到全局最优解的性能. 设定求解精度低于 $\varepsilon = 1.0e-10$ 为达标,最优解精度低于 $\varepsilon = 1.0e-300$ 记为0.

表6给出了自变量为2维的对比结果. 由达标率可知,FTO算法达标率全部为100%,对于多峰函数没有陷入局部最优解. PSO求解 f_5 和 f_8 两个多峰函数都有3次陷入局部最优解,达标率未达100%. f_1 、 f_2 、 f_{15} 、 f_{17} 的求解精度FTO高于PSO, f_{18} 的求解精度PSO高于FTO,结果在表6突出标出. 对于其余函数结果而言,两种算法相同,都收敛到全局最优解.

表7给出了自变量为10维的对比结果,表8给出了对表7中达标率相关指标的函数个数统计结果. FTO有14个函数未陷入局部最优解,PSO有12个函数未陷入局部最优解. FTO有7个函数达标率高于PSO,PSO有两个函数达标率高于FTO,结果在表7中突出标出. 达标率为0是由实验条件限制导致的,例如迭代次数不够,斐波那契树规模太小,PSO种群个数太少等因素. 通过达标率的统计结果可以看出,FTO由于具有全局随机性不易陷入局部最优解,收敛性能表现较好.

表6 收敛达标率对比实验结果(2维)

f	FTO					PSO				
	最优解	均值	标准差	达标次数	达标率/%	最优解	均值	标准差	达标次数	达标率/%
f_1	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	1.3735 e-156	1.100 4 e-144	4.432 2 e-144	30	100
f_2	0.000 0 e+00	1.500 0 e-323	0.000 0 e+00	30	100	3.342 0 e-78	2.981 6 e-74	8.817 6 e-74	30	100
f_3	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	6.573 8 e-33	3.600 6 e-32	30	100
f_4	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	2.589 8 e-293	0.000 0 e+00	30	100
f_5	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	1.184 4 e+01	3.613 9 e+01	27	90
f_6	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_7	0.000 0 e+00	3.079 0 e-15	1.114 7 e-14	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_8	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	7.396 0 e-04	2.256 7 e-03	27	90
f_9	1.570 5 e-32	1.570 5 e-32	5.567 4 e-48	30	100	1.570 5 e-32	1.570 5 e-32	5.567 4 e-48	30	100
f_{10}	1.499 8 e-33	1.173 8 e-32	3.978 8 e-32	30	100	1.499 8 e-33	1.499 8 e-33	5.219 4 e-49	30	100
f_{11}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_{12}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_{13}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_{14}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_{15}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	3.382 6 e-155	1.403 4 e-145	6.587 3 e-145	30	100
f_{16}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_{17}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	1.324 0 e-317	7.520 6 e-292	0.000 0 e+00	30	100
f_{18}	3.990 0 e-15	9.574 0 e-12	1.254 4 e-11	30	100	1.349 8 e-31	1.349 8 e-31	2.227 0 e-47	30	100
f_{19}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_{20}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	3.700 7 e-17	2.027 0 e-16	30	100
f_{21}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100

表7 收敛达标率对比实验结果(10维)

f	FTO					PSO				
	最优解	均值	标准差	达标次数	达标率/%	最优解	均值	标准差	达标次数	达标率/%
f_1	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	2.740 0 e-53	4.819 1 e-47	1.622 3 e-46	30	100
f_2	6.940 0 e-301	8.666 7 e-20	4.746 9 e-19	0	100	4.790 0 e-31	9.828 8 e-29	1.583 1 e-28	30	100
f_3	6.457 0 e-01	1.075 1 e+01	1.570 4 e+01	0	0	1.399 9 e-02	2.703 8 e+00	1.881 1 e+00	0	0
f_4	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	2.930 0 e-94	4.327 5 e-84	2.123 7 e-83	30	100
f_5	0.000 0 e+00	2.137 5 e+03	7.906 6 e+02	3	10	0.000 0 e+00	2.724 0 e+02	9.407 4 e+01	1	3
f_6	0.000 0 e+00	9.008 0 e+00	7.096 5 e+00	10	33	0.000 0 e+00	2.719 6 e+00	1.381 0 e+00	3	10
f_7	0.000 0 e+00	1.656 7 e-15	1.801 3 e-15	30	100	3.550 0 e-15	5.448 7 e-15	1.806 4 e-15	30	100
f_8	0.000 0 e+00	2.955 5 e-01	1.732 8 e-01	3	10	2.710 0 e-02	7.078 8 e-02	2.908 6 e-02	0	0
f_9	1.570 0 e-32	1.570 0 e-32	2.783 7 e-48	30	100	1.570 0 e-32	1.570 0 e-32	2.783 7 e-48	30	100
f_{10}	1.500 0 e-33	1.500 0 e-33	6.959 2 e-49	30	100	1.500 0 e-33	1.541 0 e-33	2.245 7 e-34	30	100
f_{11}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	1.397 9 e+03	9.507 7 e+02	9	30
f_{12}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	1.446 1 e+01	3.005 0 e+01	6.018 5 e+00	0	0
f_{13}	0.000 0 e+00	9.466 7 e-16	1.596 7 e-15	30	100	3.550 0 e-15	5.211 3 e-15	1.806 4 e-15	30	100
f_{14}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	3.541 3 e-01	5.817 2 e-01	9.464 4 e-02	0	0
f_{15}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	4.700 0 e-54	2.091 0 e-49	5.446 6 e-49	30	100
f_{16}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100
f_{17}	0.000 0 e+00	0.000 0 e+00	0.000 0 e+00	30	100	1.320 0 e-93	4.927 1 e-83	2.683 3 e-82	30	100
f_{18}	2.190 0 e-09	1.744 4 e-07	1.956 7 e-07	30	100	1.350 0 e-31	1.350 0 e-31	2.227 0 e-47	30	100
f_{19}	0.000 0 e+00	2.918 2 e-03	7.932 4 e-04	2	7	3.126 6 e-03	3.229 9 e-03	5.655 4 e-04	0	0
f_{20}	0.000 0 e+00	2.116 1 e-01	6.769 9 e-01	27	90	1.200 0 e-40	2.297 9 e-15	1.560 4 e-15	30	100
f_{21}	3.469 7 e+00	6.346 5 e+00	1.254 1 e+00	0	0	0.000 0 e+00	2.509 6 e+00	1.429 9 e+00	2	7

表8 达标率指标相关的函数个数统计表

达标性能	FTO	PSO
100%达标	14	12
达标率较优	7	2
0%达标	2	5

从最优解的对比结果可以看出,FTO有12个函数求解精度要高于PSO,而PSO有两个函数求解精度高于FTO,结果在表7中突出标出.该结果说明FTO引入末梢半径自适应参数后收敛性能表现良好.

5 结论

为提高FTO算法收敛性能,引入斐波那契树末梢自适应半径参数,FTO在最优解邻域附近收敛能力显著提升.基于斐波那契树的全局随机性特点,对FTO算法的收敛性进行了分析并证明.通过末梢自适应半径参数对比实验对改进前后的FTO算法收敛性能进行了验证和分析.通过与5个元启发式搜索算法的求解精度对比实验验证了FTO算法的收敛性能.全局随机性特点令FTO不易陷入局部最优解,利用PSO算法容易陷入局部最优解的特点进行对比实验.对21个测试函数维数分别取2维和10维的求解实验独立进行30次,对结果进行对比分析,从收敛达标率的统计结果和收敛性能对比结果可以看出,FTO算法均表现良好.

后续工作将结合实际问题,将算法结合网络流量控制、工业设计等方面,进一步应用并改进FTO算法.

参考文献(References)

- [1] 禹华军. 基于斐波那契法的风速计故障诊断研究[J]. 机械制造, 2016, 54(7): 75-77.
(Yu H J. Investigation on fault diagnosis of anemometer based on Fibonacci method[J]. Machinery, 2016, 54(7): 75-77.)
- [2] 李琪, 王再兴, 沈黎阳, 等. 基于改进黄金分割法的钻井液流变模式优选[J]. 钻井液与完井液, 2016, 33(1): 57-62.
(Li Q, Wang Z X, Shen L Y, et al. Optimization of drilling fluid rheology pattern using improved golden section method[J]. Drilling Fluid & Completion Fluid, 2016, 33(1): 57-62.)
- [3] 马睿, 皇甫宜耿, 李玉忍, 等. 一种迭代黄金分割法的独立光伏系统[J]. 电机与控制学报, 2016, 20(1): 81-87.
(Ma R, Huangfu Y G, Li Y R, et al. A stand-alone PV system based on golden section search method[J]. Electric Machines and Control, 2016, 20(1): 81-87.)
- [4] 谢小兵, 蒋鹏, 黄敏, 等. 基于黄金分割法的双季稻合理密植研究[J]. 核农学报, 2016, 30(12): 2467-2476.
(Xie X B, Jiang P, Huang M, et al. Study on rational close planting based on golden section method for double cropping rice[J]. J of Nuclear Agricultural Sciences, 2016, 30(12): 2467-2476.)
- [5] 许平山, 丁同文, 李宁. 黄金分割法在男西服中的应用[J]. 纺织学报, 2015, 36(1): 134-140.
(Xu P S, Ding T W, Li N. Application of golden ratio in men's suits[J]. J of Textile Research, 2015, 36(1): 134-140.)
- [6] Ibrahim H Osman, Gilbert Laporte. Metaheuristics: A bibliography[J]. Annals of Operations Research, 1996, 63(5): 511-623.
- [7] Cordón O, Herrera F, Stützle T. A review on the ant colony optimization metaheuristic: Basis, models and new trends[J]. North American J of Sports Physical Therapy Najspt, 2006, 1(3): 62-72.
- [8] Tarek A Elmihoub, Adrian A Hopgood, Lars Nolle, et al. Hybrid genetic algorithms: A review[J]. Engineering Letters, 2006, 3(2): 124-137.
- [9] Kennedy J, Eberhart R. Particle swarm optimization[C]. IEEE Int Conf on Neural Networks. Perth: The University of Western Australia, 1995: 1942-1948.
- [10] Rainer Storn, Kenneth Price. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces[J]. J of Global Optimization, 1997, 11(4): 341-359.
- [11] Dervis Karaboga, Bahriye Basturk. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony(abc) algorithm[J]. J of Global Optimization, 2007, 39(3): 459-471.
- [12] Hüseyin Haki, Harun Uuz. A novel particle swarm optimization algorithm with levy flight[J]. Applied Soft Computing, 2014, 23(5): 333-345.
- [13] Ilhem Boussaïd, Julien Lepagnot, Atrick Siarry. A survey on optimization metaheuristics[J]. Information Sciences, 2013, 237(237): 82-117.
- [14] Ting Yee Lim. Structured population genetic algorithms: A literature survey[J]. Artificial Intelligence Review, 2014, 41(3): 385-399.
- [15] 王皓, 欧阳海滨, 高立群. 一种改进的全局粒子群优化算法[J]. 控制与决策, 2016, 31(7): 1161-1168.
(Wang H, Ouyang H B, Gao L Q. An improved global particle swarm optimization[J]. Control and Decision, 2016, 31(7): 1161-1168.)
- [16] Elena-Niculina Dragoi, Vlad Dafinescu. Parameter control and hybridization techniques in differential evolution: A survey[J]. Artificial Intelligence Review, 2016, 45(4): 447-470.
- [17] Dervis Karaboga, Bahriye Akay. A comparative study of artificial bee colony algorithm[J]. Applied Mathematics and Computation, 2009, 214(1): 108-132.

(责任编辑: 闫妍)