论 文

基于平均增益模型的连续型 (1+1) 进化算法计算时间复杂性分析

黄翰^{①*}, 徐威迪^①, 张宇山^{②③}, 林智勇^④, 郝志峰^⑤^① 华南理工大学软件学院, 广州 510006^② 广东财经大学数学与统计学院, 广州 510320^③ 南京大学计算机软件新技术国家重点实验室, 南京 210093^④ 广东技术师范学院, 计算机科学学院, 广州 510665^⑤ 广东工业大学计算机学院, 广州 510006

* 通信作者. E-mail: hhan@scut.edu.cn

收稿日期: 2013–10–03; 接受日期: 2014–03–23

国家自然科学基金 (批准号: 61370102, 61170193, 61202453, 61203310)、中央高校基本科研业务费 (批准号: 2012ZZ0087, 2014ZG0043)、广东省自然科学基金项目 (批准号: S2011040002890, S2012010010613) 和珠江科技新星项目 (批准号: 2012-J2200007) 资助项目

摘要 连续型进化算法的计算时间复杂性分析是进化计算理论研究的一项公开难题, 目前相关研究成果较少. 针对连续型 (1+1)EA, 基于适应值差函数提出了平均增益模型及其分析方法, 给出了平均计算时间的计算理论, 为算法的计算时间复杂性分析提供了依据. 在此基础上, 研究还选取了学术界关注的球形函数作为研究对象, 分别推导了变步长满足标准正态分布和均匀分布的连续型 (1+1)EA 在优化球形函数时的平均增益, 并估算出了它们的平均计算时间. 理论分析说明: 1) 两种算法的计算时间复杂性都是指数级的; 2) 在给定相同精度和初始适应值差的前提下, 采用均匀分布变异算子的算法其寻优速度优于采用标准正态分布变异算子的算法. 进一步地, 通过数值实验对理论分析结果进行了验证, 结果表明平均增益模型分析是有效的.

关键词 进化算法 进化计算理论基础 计算时间复杂性 平均增益模型 Sphere 函数

1 引言

进化算法 (EA) 是一类模拟生物物种进化的随机优化算法, 包括遗传算法 (GA)、遗传规划 (GP)、进化策略 (ES) 和进化规划 (EP) 等^[1]. 它们利用一些启发式规则进行优化, 其思想根源来自于进化论. 其中, 遗传算法广泛应用于离散优化问题, 而进化策略和进化规划多应用于连续函数优化、模式识别、机器学习、神经网络训练、系统辨识和智能控制等技术领域^[2].

计算时间分析是进化计算研究领域的一项公开难题^[1]. 早在 1970 年代, 进化算法就开始被广泛应用于工程设计与工程优化等领域, 但其计算时间复杂性的研究却相对滞后. 直至 20 世纪 90 年代, 人们才逐渐开始研究离散型 (1+1) 进化算法 ((1+1)EA) 用以求解伪布尔函数 (比如 ONEMAX 和 Trap 函数) 的时间复杂性^[3,4]. 这些研究主要是比较了 (1+1)EA 在优化不同函数时的性能优劣并获得了一

引用格式: 黄翰, 徐威迪, 张宇山, 等. 基于平均增益模型的连续型 (1+1) 进化算法计算时间复杂性分析. 中国科学: 信息科学, 2014, 44: 811–824, doi: 10.1360/112013-3

些进化算法个案的计算时间分析结论^[5~7]. 随着越来越多个案被研究, 针对计算时间复杂性分析的一般理论与方法也逐渐涌现. 其中, He 等^[8~10]提出的 Drift Analysis 方法是典型的代表, Oliveto 等^[11]在 He 工作的基础上提出了一种简化而易用的新 Drift 理论, 并应用于分析求解最大匹配的 (1+1)EA 算法计算时间复杂性, 得到了一个指数式的时间下界. Chen 等^[12]提出了一种基于边际概率分类的方法, 用于研究分布估计算法 (EDA) 的计算时间, 并证明了该方法在 EDA 算法分析上的有效性. 此后 Oliveto 等^[13]详细讨论了 RLS 和 (1+1)EA 在一些 NP-hard 问题上的性能差异. 他们还指出随机进化算法应用于图论问题时在最坏的情况下通常需要指数时间; 然而, 当图的结点度数小于等于二时, 计算时间复杂性将降至多项式时间.

上述关于进化算法计算时间分析理论的众多工作^[8~13]是针对离散空间的简单优化问题而进行的. 相比之下, 对于进化算法在求解连续空间优化问题时的计算时间分析, 相关研究工作则比较少见, 是学术界公认的关键难题. Agapie A 和 Agapie M^[14]应用转移函数分析了连续型 EA 的搜索性能, 获得了计算时间分析的初步结论. Jägersküpper^[15]则用几何概率的方法证明了源自实践经验的 1/5 定律的正确性. 即, 对于球形 (Sphere) 函数来说, 当步长的变化满足 1/5 定律时, 减小一半误差需要计算 $\Theta(n)$ 次适应值函数. 在此基础上, Jägersküpper^[16]还研究了 (1+1)ES 求解 Sphere 函数的计算时间分析框架. Agapie 等^[17]完整描述了进化算法求解连续型优化问题的数学模型, 为计算时间分析作了铺垫.

尽管有以上提及的连续型进化算法计算时间分析结论出现, 但是, 绝大多数结论^[14~17]并没有明确给出进化算法求解具体连续优化问题实例的计算时间分析结果. 其中的一个原因在于连续型进化算法的算子描述复杂多变, 通常难以进行统一分析. 在本文的工作中, 我们针对连续型进化算法的特性建立了平均增益模型. 在此基础上, 以当前学术界较为关注的 (1+1)EA 和 Sphere 函数为实例进行分析, 获得了采用正态分布变异算子的 (1+1)EA 以及采用均匀分布变异算子的 (1+1)EA 的计算时间复杂性结果. 同样以 Sphere 函数为案例, 与文献^[16]的工作相比, 本文研究分析的正态分布和均匀分布变异的算子在连续型进化算法中更有代表性^[1]. 同样以 drift analysis 为理论框架, 与文献^[8~10]的工作相比, 本文提出的平均增益模型理论更适于连续型进化算法的分析.

2 连续型 (1+1)EA 的平均增益建模

本研究针对一个连续型的进化算法框架 (1+1)EA 和 Sphere 函数问题. 本节将完成问题描述、算法介绍以及对算法的数学建模.

2.1 问题描述与算法简介

定义 1 (极小化问题) 令 $S \subseteq \mathbb{R}^n$ 为 n 维实数域 $\mathbb{R}^n$ 的子空间, $f: S \rightarrow \mathbb{R}$ 为 n 维实值函数, 则以 (S, f) 形式表示的极小化问题是找一个 n 维实向量 $x_{\min} \in S$ 使得: $\forall x \in S, f(x_{\min}) \leq f(x)$.

本文重点研究的 Sphere 函数问题^[16]就是一个定义在 n 维连续空间 $\mathbb{R}^n$ 上的单峰极小化问题, 相应的函数表达式为

$$f(x) = \sqrt{\sum_{i=1}^n x_i^2},$$

其中 $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$, $f(x)$ 函数值是 x 点到原点的欧氏距离.

1. Generate an initial individual x_0 ;
2. Generate a random n -dimension vector Δ_t as a mutation variant. Every dimension of Δ_t is generated independently following a uniform distribution or Gaussian distribution;
3. Produce an n -dimension offspring vector with $\tilde{x}_t = x_t + \Delta_t$;
4. If $d(\tilde{x}_t) < d(x_t)$, $x_{t+1} = \tilde{x}_t$; otherwise, $x_{t+1} = x_t$.

图 1 连续型 (1+1)EA 算法流程

Figure 1 Procedure of (1+1)EA for continuous optimization

$f(x)$ 可以视为研究对象进化算法优化的目标函数. 假设进化算法求解 Sphere 函数问题的渐进过程就是建立在 n 维实向量 x 的基础上. 每次按照某种随机规律生成一个 n 维向量 Δ 作为 x 的更新步长, 得到一个新的 $\tilde{x} = x + \Delta$. 如果目标函数值 $f(x)$ 因为更新而减少 (最小化问题为例) 则接受新向量 $\tilde{x}$ 作为当前最优解, 否则不接受. 显然, 该函数的全局最优解为原点 $(0, 0, \dots, 0)$. 不失一般性, 我们研究算法从点 $(1, 1, \dots, 1)$ 开始到全局最优解原点的渐进过程.

进化算法的种类很多, 为不失一般性, 这里以 Generated-and-test 框架下连续型 (1+1)EA 算法为例. 具体如下图 1 所示.

连续型 (1+1)EA 保留了经典 (1+1)EA 的算法框架, 除了个体 $x_t (t = 0, 1, 2, \dots)$ 是实向量之外, 还相对离散型^[3,4] 作了两点修改, 以满足连续问题的分析需要. 这两点修改分别是, 以步骤 3 的加法公式作为变异更新个体, 以及将步骤 5 的停机条件改为误差判定方式.

选择连续型 (1+1)EA 作为分析对象的原因有三: 首先, 选择 (1+1)EA 算法框架, 简化了种群规模、交叉算子和选择算子的影响, 主要研究点在于不同概率分布的变异步长对计算时间的影响; 其次, (1+1)EA 是离散型进化算法计算时间分析的经典对象^[3,4,8~11], 这里沿用其框架将使研究具有代表性; 再次, 流程中步骤 2 中 Δ_t 的随机性与步骤 3 的更新公式设计源于进化规划算法^[1,17] 的变异算子设计, 所以本分析结论将对进化规划算法的计算时间研究有积极意义.

2.2 连续型 (1+1)EA 的平均增益模型

与已有连续型进化算法的研究^[14~17] 不同, 研究将连续型 (1+1)EA 视为随机状态从初始个体 x_0 开始到全局最优解原点的渐进过程. 我们将为这个过程建立平均增益模型, 首先, 需要给出如下定义和定理.

定义 2 (适应值差函数) 设函数 $d: S \rightarrow \mathbb{R}$ 满足 $d(x) = f(x) - f(x^*)$, $\forall x \in S$, x^* 为问题的最优解, 称 d 为适应值差函数.

显然, $d(x^*) = 0$. 例如, 在 Sphere 函数中, S 是 n 维实空间 $\mathbb{R}^n$; 而且因为 $f(x^*) = 0$, 有 $d(x) = f(x)$. 与 drift analysis 上界定理^[8~10] 不同, 定理 1 的适应值差函数 d 可以为负数, 以适合连续型 (1+1)EA 算法的分析.

定义 3 (平均计算时间) 在连续型 (1+1)EA 中, 对于任意 $0 \leq a < b$, 若初始解的适应值差 $d(x_0) = b$, 那么, 记 $T|_a^b = \min\{t | d(x_t) \leq a\}$ 为算法经过区间 $[a, b]$ 的计算时间, 记 $E(T|_a^b)$ 为经过区间 $[a, b]$ 的平均计算时间.

$E(T|_a^b)$ 表示了连续型 (1+1)EA 从适应值差为 a 的起点到适应值差为 b 的终点所需的平均计算

时间. 因为图 1 中 $d(x_0) = \sqrt{n}$, 所以连续型 (1+1)EA 的平均计算时间可以表示为 $E(T|_0^{\sqrt{n}})$, 在满足 $\varepsilon > 0$ 的精度下, 平均计算时间可以表示为 $E(T|_\varepsilon^{\sqrt{n}})$.

引理 1 在连续型 (1+1)EA 中, 当 $V := \min\{E(d(x_t) - d(x_{t+1})) | t \in \mathbb{N}\} > 0$ 时, 经过区间 $[0, d(x_0)]$ 的计算时间 $T|_0^{d(x_0)}$ 及其期望 $E(T|_0^{d(x_0)})$ 满足

$$E(T|_0^{d(x_0)}) \leq \frac{d(x_0)}{V}, \quad (1)$$

其中, x_t 是第 t 代的随机状态来自状态空间 S , x_0 为初始个体. 证明见附录 A.

因为连续型 (1+1)EA 只有一个个体 x_t , 所以 x_t 既代表了进化个体也代表了算法的随机状态; 同理, S 代表了问题的解空间, 同时也代表了 x_t 来自的状态空间. 除此以外, 引理 1 的证明并没有利用到连续型 (1+1)EA 的性质, 即引理 1 的结论对满足个体规模为 1 的更一般算法也是适合的.

在连续状态空间 S 中, x_t 到达最优解的概率几乎为零, 连续型 (1+1)EA 算法到达最优解的概率的计算应与其初始位置 x_0 和求解精度 $\varepsilon > 0$ 有关. 基于此观点, 我们用平均增益 (定义 4) 作为模型, 即 x_t 和 x_{t+1} 的期望适应值差, 作为平均计算时间的估算指标. 平均增益的计算过程中需要用极限去逼近积分, 因此这里引用黎曼积分^[18]作为工具. 基于黎曼积分, 定义 4 给出了平均增益的定义及计算方法.

定义 4 (平均增益) 在连续型 (1+1)EA 中, $G(r, t) = E(d(x_t) - d(x_{t+1}) | d(x_t) = r)$ 称为算法在 t 时刻关于适应值差 r 的平均增益. 由于 (1+1)EA 中的 Δ_t 的每一维数值的产生都相互独立, 并服从均匀分布或者正态分布, Δ_t 与迭代时间 t 无关. 因此, $G(r, t)$ 在本文中迭代时间 t 无关, 简单记为 $G(r)$.

在当前个体与最优解的适应值差为 r 前提下, $G(r)$ 表示连续型 (1+1)EA 迭代中个体 x_t 和下一代新个体 x_{t+1} 的期望适应值差. 直观而言, $G(r)$ 体现了在适应值差为 r 的前提下, (1+1)EA 的进化个体逼近全局最优解的平均速度. 基于定义 4 和引理 1, 我们可以得到一个连续型 (1+1)EA 平均计算时间的计算方法, 定理 1、定理 2 和推论 1 阐述了这一结论.

定理 1 设 $d(x_0) = L$, 若连续型 (1+1)EA 的平均增益 $G(r)$ 关于 $r \geq 0$ 可积, 那么其达到精度 ε 的平均计算时间满足: $E(T|_\varepsilon^L) \leq \int_\varepsilon^L 1/G(r)dr$. 证明见附录 B.

在定理 1 中, 我们得到平均计算时间的上界. 这个上界是平均增量函数倒数 $1/G(r)$ 在区间 $[\varepsilon, L]$ 上的定积分. 另一方面, 定理 2 给出了平均计算时间的下界.

定理 2 设 $d(x_0) = L$, 若连续型 (1+1)EA 的平均增益 $G(r)$ 关于 $r \geq 0$ 可积, 那么其达到精度 ε 的平均计算时间满足: $E(T|_\varepsilon^L) \geq \int_\varepsilon^L 1/G(r)dr$. 证明见附录 C.

有了定理 1 和定理 2, 可以得到平均计算时间的计算公式, 如推论 1 所示.

推论 1 对于连续型 (1+1)EA 算法, 给定一个初始解 x_0 不妨设 $d(x_0) = L$, 那么其达到精度 ε 的平均计算时间 $E(T|_\varepsilon^L) = \int_\varepsilon^L 1/G(r)dr$.

证明 根据引理 1 和引理 2, 可以直接得到 $E(T|_\varepsilon^L) = \int_\varepsilon^L 1/G(r)dr$.

因此, 连续型 (1+1)EA 的平均计算时间等于 $1/G(r)$ 在区间 $[\varepsilon, L]$ 上的定积分. 例如, 如果图 1 中初始解 $x_0 = (\frac{1}{2\sqrt{n}}, \frac{1}{2\sqrt{n}}, \dots, \frac{1}{2\sqrt{n}})$, 所以, $d(x_0) = \frac{1}{2}$. 此时, 推论 1 结论可以表示为 $E(T|_\varepsilon^{1/2}) = \int_\varepsilon^{1/2} 1/G(r)dr$, 即, 当平均增益函数 $G(r)$ 和精度 ε 确定时, 我们可以直接计算出连续型 (1+1)EA 的平均计算时间. 根据定义 4, $G(r)$ 由 2 个因素决定:

1) $G(r)$ 与适应值差函数 $d(x)$ 有关, 而 $d(x)$ 与目标函数 $f(x)$ 有关; 因为 $f(x)$ 与优化问题有关, 所以 $G(r)$ 与优化问题直接相关;

2) $G(r)$ 与算法状态从 x_t 到 x_{t+1} 的转移概率有关, 而这个转移概率由变异步长 Δ_t 决定, Δ_t 则是取决于算法采用的变异算子, 所以 $G(r)$ 由变异算子决定.

综上所述, 当确定了变异算子 (Δ_t) 和优化问题 ($d(x)$), 我们就可以确定 $G(r)$, 并估算出连续型 (1+1)EA 在精度 ε 下的平均计算时间. 第 3 节将以标准正态分布与均匀分布两种变异算子为例, 展示平均计算时间的分析过程.

3 连续型 (1+1)EA 个案的平均计算时间分析

基于平均增益模型 (定义 4 和推论 1), 本节将完成 2 个连续型 (1+1)EA 个案的平均计算时间分析. 首先, 需要给出个体和变异步长服从的随机分布类型. 随机算法中的步长 Δ_t 是一个 n 维随机变量, 而且它的每一维都独立同分布, 即满足标准正态分布或者均匀分布, 分布密度函数分别见式 (2) 和 (3).

$$f_1(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}, \quad (2)$$

$$f_2(x) = \begin{cases} \frac{1}{2}, & x \in (-1, 1), \\ 0, & x \notin (-1, 1). \end{cases} \quad (3)$$

在下面的文章中, 我们记步长满足标准正态分布的连续型 (1+1)EA 算法为 EA-I; 记步长满足 $(-1, 1)$ 均匀分布的连续型 (1+1)EA 算法为 EA-II. EA-I 和 EA-II 的初始解均为 $x_0 = (\frac{1}{2\sqrt{n}}, \frac{1}{2\sqrt{n}}, \dots, \frac{1}{2\sqrt{n}})$, 即 $d(x_0) = 0.5$. 选择这个起始点作分析的原因是, 式 (3) 代表的均匀分布在 $0 \leq d(x_0) \leq 0.5$ 区域平均增量的积分计算比较简单, 便于对比分析结果.

标准正态分布是连续型进化算法中变异算子设计的常用工具^[1], 但关于标准正态分布的进化算子对计算时间影响的研究结果较少, 多数研究工作还是针对二进制随机变异^[8~11]和均匀分布变异^[14~17]. 因此, 这里选择这 2 种分布作为研究个案是较有针对性的. 不仅如此, 本文还将对比两种变异分布在求解 Sphere 函数的计算时间复杂性差异.

3.1 标准正态分布的 EA-I 算法计算时间分析

由于 EA-I 的初始解 $x_0 = (\frac{1}{2\sqrt{n}}, \frac{1}{2\sqrt{n}}, \dots, \frac{1}{2\sqrt{n}})$, 由式 (1) 中 Sphere 函数的计算方式, $d(x_0) = 0.5$. 因此, EA-I 的平均计算时间可以记为 $E(T|_{\varepsilon}^{0.5})$.

根据推论 1, 必须先求平均增益 $G(r)$, 才可以计算 EI-I 的平均计算时间. 引理 2 给出了变异步长满足标准正态分布 $G(r)$ 的计算方法.

引理 2 在变异步长 Δ_t 满足标准正态分布的 EA-I 中, 平均增益满足

$$\frac{e^{-\frac{r^2}{2}} \cdot r^{n+1}}{\Gamma(n/2+1) \cdot (\sqrt{2})^n \cdot (n+1)} \leq G(r) \leq \frac{r^{n+1}}{\Gamma(n/2+1) \cdot (\sqrt{2})^n \cdot (n+1)}. \quad (4)$$

证明见附录 D.

考虑在连续型 (1+1)EA 中, 初始个体为 $x_0 = (\frac{1}{2\sqrt{n}}, \frac{1}{2\sqrt{n}}, \dots, \frac{1}{2\sqrt{n}})$, 则根据图 1 和定义 4, $0 < \varepsilon \leq r \leq 0.5$. 因此, $\frac{e^{-\frac{r^2}{2}} \cdot \varepsilon^{n+1}}{\Gamma(n/2+1) \cdot (\sqrt{2})^n \cdot (n+1)} \leq G(r) \leq \frac{(1/2)^{n+1}}{\Gamma(n/2+1) \cdot (\sqrt{2})^n \cdot (n+1)}$. 根据推论 1 和引理 2, 可以得到定理 3.

图 2 球形区域和立方体区域相交得到不规则区域

Figure 2 Irregular-region intersection of sphere region and cubic region

定理 3 在 EA-I 算法中, 若其变异步长 Δ_t 服从标准正态分布, 那么它达到精度 $\varepsilon(d(x_t) < \varepsilon)$ 的平均计算时间 $E(T|_\varepsilon^{0.5})$ 满足

$$E(T|_\varepsilon^{0.5}) > \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \cdot \frac{1}{n} \cdot (\varepsilon^{-n} - (0.5)^{-n}). \quad (5)$$

证明见附录 E.

因为 $0 < \varepsilon \leq r \leq 0.5$, 由式 (5) 可得 EA-I 的计算时间复杂性为 $\omega(\Gamma(n/2 + 1) \cdot \left(\frac{\sqrt{2}}{\varepsilon}\right)^n)$.

定理 4 在连续型 (1+1)EA 算法中, 若其变异步长 Δ_t 服从标准正态分布, 那么它经过迭代达到精度 $\varepsilon(d(x_t) < \varepsilon)$ 的平均计算时间 $E(T|_\varepsilon^{0.5})$ 满足

$$E(T|_\varepsilon^{0.5}) < \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \frac{1}{n \cdot \varepsilon^n}. \quad (6)$$

证明见附录 F.

由式 (5) 可得 EA-I 的计算时间复杂性为 $o(\Gamma(n/2 + 1) \cdot \left(\frac{\sqrt{2}}{\varepsilon}\right)^n)$. 因此, 综合定理 3 和定理 4 的结论, EA-I 的计算时间复杂性 $\Theta(\Gamma(n/2 + 1) \cdot \left(\frac{\sqrt{2}}{\varepsilon}\right)^n)$.

3.2 均匀分布的 EA-II 算法计算时间分析

由于 EA-II 的初始解 $x_0 = (\frac{1}{2\sqrt{n}}, \frac{1}{2\sqrt{n}}, \dots, \frac{1}{2\sqrt{n}})$, 由式 (1) 中 Sphere 函数的计算方式 $d(x_0) = 0.5$. 因此, EA-II 的平均计算时间可以记为 $E(T|_\varepsilon^{0.5})$.

首先, 介绍 $E(T|_\varepsilon^{0.5})$ 计算的几何意义. 当步长 Δ_t 满足均匀分布 (式 2) 时, 对于任意坐标的解 x_t , $\tilde{x}_{t+1}$ 的可达范围是以该解为中点, 边长为 2 的 n 维立方体. 这将导致 x_{t+1} 的可达范围的不规则. 例如, 当 $r > 0.5$ 时, 3 维情形如图 2 所示, x_t 为白点所在位置, 那么 x_{t+1} 的可达范围即为球形和立方体的相交区域. 特别地, 当 EA-II 初始解 $d(x_0) = 0.5$ 时, 立方体将包围球体 (证明详见附录 G), 他们的交集即为一个球体.

表 1 EA-I 和 EA-II 求解 Sphere 问题的对比
Table 1 Comparison of EA-I and EA-II for Sphere function

Dimension	$n = 5$		$n = 10$		$n = 30$		$n = 50$	
Algorithm	EA-I	EA-II	EA-I	EA-II	EA-I	EA-II	EA-I	EA-II
$\varepsilon = 10^{-1}$	$\approx 10^6$	$\approx 10^5$	$\approx 10^{13}$	$\approx 10^{10}$	$\approx 10^{43}$	$\approx 10^{36}$	$\approx 10^{76}$	$\approx 10^{65}$
$\varepsilon = 10^{-3}$	$\approx 10^{16}$	$\approx 10^{15}$	$\approx 10^{33}$	$\approx 10^{30}$	$\approx 10^{103}$	$\approx 10^{96}$	$\approx 10^{176}$	$\approx 10^{165}$
$\varepsilon = 10^{-5}$	$\approx 10^{26}$	$\approx 10^{25}$	$\approx 10^{53}$	$\approx 10^{50}$	$\approx 10^{163}$	$\approx 10^{106}$	$\approx 10^{276}$	$\approx 10^{265}$

引理 3 当 $r \leq 0.5$ 时, EA-II 的平均增益满足 $G(r) = \frac{r^{n+1} \cdot \sqrt{\pi}^n}{2^n \cdot \Gamma(n/2+1) \cdot (n+1)}$.

证明见附录 G.

由引理 3 的结论可以看出, EA-II 的平均增益 $G(r)$ 是一个在实数域可积的函数, 所以可以直接根据推论 1 得到定理 5 的结论.

定理 5 给定初始解 x_0 满足 $d(x_0) = 0.5$, 那么 EA-II 达到精度 $d(x_t) < \varepsilon$ 的平均计算时间满足 $E(T|_{\varepsilon}^{0.5}) = \Gamma(n/2+1) \cdot \frac{(n+1)}{n} \cdot \frac{2^n}{(\sqrt{\pi})^n} \cdot (\frac{1}{\varepsilon^n} - \frac{1}{0.5^n})$. 证明见附录 I.

根据图 1 中连续型 (1+1)EA 算法的设置, ε^{-n} 大于 0.5^{-n} , 则根据计算复杂性 Θ 符号的运算规则, EA-II 的计算时间复杂性为 $\Theta(\Gamma(n/2+1) \cdot (\frac{2}{\varepsilon \cdot \sqrt{\pi}})^n)$.

4 计算时间复杂性对比分析

虽然 EA-I 和 EA-II 的计算时间复杂性都比较高; 当给定相同精度 ε 和初始适应值差 $d(x_0) = 0.5$ 时, 可以由定理 4 和定理 5 得出, EA-II 计算时间复杂性低于 EA-I, 即, 均匀分布变异算子在求解 Sphere 函数时性能优于标准正态分布变异算子. 假设 t_1 是 EA-I 的平均计算时间, t_2 是 EA-II 的平均计算时间, 则有

$$\lim_{n \rightarrow +\infty} \frac{t_1}{t_2} = \lim_{n \rightarrow +\infty} \frac{\Gamma(n/2+1) \cdot \left(\frac{\sqrt{2}}{\varepsilon}\right)^n}{\Gamma(n/2+1) \cdot \left(\frac{2}{\varepsilon \cdot \sqrt{\pi}}\right)^n} = \left(\sqrt{\frac{\pi}{2}}\right)^n = +\infty, \quad \lim_{n \rightarrow +\infty} \frac{t_2}{t_1} = 0. \quad (7)$$

更具体地, 根据定理 3, 4 和 5 的估算结果如表 1 所示.

为了验证理论分析的正确性, 本研究还对 EA-I 和 EA-II 求解小规模 Sphere 问题进行了实验观察. 2 个算法初始解均满足 $d(x_0) = 0.5$, 观察问题的规模为 $n = 5$ 和 $n = 10$, 结果如图 3 和图 4 所示.

图 3 和图 4 表明, 针对 Sphere 问题, 给定初始点距离为 0.5 时, EA-II 比 EA-I 有更快的计算速度. 这与式 (7) 结论一致. 两个算法的计算复杂性极高造成了实验困难, 图 3 和图 4 仅仅给出了 $n = 5$ 和 $n = 10$ 的对比结果, 实验结果验证了定理 3, 4 和 5 的正确性与有效性.

5 结论

本研究建立了连续型 (1+1)EA 算法的平均增益模型, 是研究连续型进化算法计算时间分析的突破点. 平均增益模型刻画了算法在连续状态空间迭代收敛的过程, 直接决定了连续型进化算法的计算时间复杂性. 基于平均增益模型, 我们完成了 (1+1)EA 求解 Sphere 问题的计算时间复杂性分析, 并从

图 3 (网络版彩图) $n=5$ 的 Sphere 问题计算时间对比

Figure 3 (Color online) Running time comparison for Sphere function when $n = 10$

图 4 (网络版彩图) $n=10$ 的 Sphere 问题计算时间对比

Figure 4 (Color online) Running time comparison for Sphere function when $n = 5$

理论上对比了标准正态分布变异算子与均匀变异分布算子在给定初始解条件下的平均计算时间差异, 并通过数值实验验证了该研究结论. 未来的研究工作将建立一般化连续型进化算法的平均增益模型, 并将研究结果扩展到其他变异算子的计算性能分析, 最后完成其他未有计算时间复杂性分析结论的连续函数优化问题.

参考文献

- 1 Yao X, Xu Y. Recent advances in evolutionary computation. J Comput Sci Technol, 2006, 21: 1–18
- 2 Mallipeddi R, Mallipeddi S, Suganthan P N. Ensemble strategies with adaptive evolutionary programming. J Inform Sci, 2010, 180: 1571–1581

- 3 Droste S, Jansen T, Wegener I. A rigorous complexity analysis of the (1+1) evolutionary algorithm for linear functions with Boolean inputs. *Evol Comput.* 1998, 6: 185–196
- 4 Droste S, Jansen T, Wegener I. On the analysis of the (1+1) evolutionary algorithms. *Theor Comput Sci*, 2002, 276: 51–81
- 5 He J, Reeves C, Witt C, et al. A note on problem difficulty measures in black-box optimization: classification, realizations and predictability. *Evol Comput*, 2007, 15: 435–443
- 6 Oliveto P S, He J, Yao X. Time complexity of evolutionary algorithms for combinatorial optimization: a decade of results. *Int J Automat Comput*, 2007, 4: 281–293
- 7 Eiben A E, Aarts E H, Van Hee K M. *Global Convergence of Genetic Algorithms: An Infinite Markov Chain Analysis*. Berlin: Springer-Verlag, 1991. 4–12
- 8 He J, Yao X. Drift analysis and average time complexity of evolutionary algorithms. *Artifi Intel*, 2001, 127: 57–85
- 9 He J, Yao X. From an individual to a population: an analysis of the first hitting time of population-based evolutionary algorithms. *IEEE Trans Evol Comput*, 2002, 6: 495–511
- 10 He J, Yao X. Erratum to: drift analysis and average time complexity of evolutionary algorithms: [Artificial Intelligence 127 (2001) 57–85]. *Artifi Intel*, 2002, 140: 245–248
- 11 Oliveto P S, Witt C. Simplified drift analysis for proving lower bounds in evolutionary computation. In: *Proceedings of the 10th International Conference on Parallel Problem Solving from Nature: PPSN X*, Berlin, 2008. 82–91
- 12 Chen T S, Tang K, Chen G L, et al. Analysis of computational time of simple estimation of distribution algorithms. *IEEE Trans Evol Comput*, 2010, 14: 1–22
- 13 Oliveto P S, He J, Yao X. Analysis of the (1+1)-EA for finding approximate solutions to vertex cover problems. *IEEE Trans Evol Comput*, 2009, 13: 1006–1029
- 14 Agapie A, Agapie M. Transition functions for evolutionary algorithms on continuous state-space. *J Math Model Algorithm*, 2007, 6: 297–315
- 15 Jägersküpper J. Algorithmic analysis of a basic evolutionary algorithm for continuous optimization. *Theor Comput Sci*, 2007, 379: 329–347
- 16 Jägersküpper J. How the (1+1) ES using isotropic mutations minimises positive definite quadratic forms. *Theor Comput Sci*, 2006, 361: 38–56
- 17 Agapie A, Agapie M, Zbaganu G. Evolutionary algorithms for continuous space optimization. *Int J Sys Sci*, 2013, 44: 502–512
- 18 Walter R. *Principles of Mathematical Analysis*. New York: McGraw-Hill Companies, Inc, 1976

附录 A 引理 1 的证明

$T|_0^{d(x_0)}$ 简单记为 T . 记 $D_t = d(x_t) - d(x_{t+1})$ 和 $P(T = t)$ 为计算时间是 t 的概率有以下结论:

$$\begin{aligned}
 d(x_0) &\geq E\left(\sum_{t=0}^{T-1} D_t\right) = \sum_{t=0}^{+\infty} P(T = t) \cdot E\left(\sum_{i=0}^{T-1} D_i | T = t\right) \\
 &= \sum_{t=0}^{+\infty} P(T = t) \cdot \sum_{i=0}^{T-1} E(D_i | T = t) \\
 &= \sum_{t=0}^{+\infty} \sum_{i=0}^t P(T = t) \cdot E(D_i | T = t) \\
 &= \sum_{i=0}^{+\infty} \sum_{t=i}^{+\infty} P(T = t) \cdot E(D_i | T = t) \\
 &\geq \sum_{i=0}^{+\infty} \sum_{t=i}^{+\infty} P(T \geq i) \cdot P(T = t | T \geq i) \cdot E(D_i | T = t)
 \end{aligned}$$

$$\begin{aligned}
 &= \sum_{i=0}^{+\infty} P(T \geq i) \cdot \sum_{t=i}^{+\infty} P(T=t|T \geq i) \cdot E(D_i | T=t \wedge T \geq i) \\
 &= \sum_{i=0}^{+\infty} P(T \geq i) \cdot E(D_i | T=t) \geq V \cdot \sum_{i=0}^{+\infty} P(T \geq i) \\
 &= V \cdot \sum_{i=0}^{+\infty} \sum_{j=i}^{+\infty} P(T=j) = V \cdot \sum_{i=0}^{+\infty} i \cdot P(T=i) \geq V \cdot E(T).
 \end{aligned}$$

因此, 当 $V \neq 0$ 时 $E(T) \leq \frac{d(x_0)}{V}$.

附录 B 定理 1 的证明

给定 l (满足 $\varepsilon < l < L$), 考虑子区间 $Z' = \{x | l < d(x) \leq l + \Delta l\}$, 并且定义一个适应值差函数 $d'(x) = d(x) - l$, 我们在这个子区间上运行连续型 (1+1)EA 算法, 得到该区间上的平均计算时间 $E(T|_l^{l+\Delta l})$.

我们让算法从区间右端点开始 ($d(x_0) = l + \Delta l$), 经过时间 $T|_l^{l+\Delta l}$ 到达区间左端点 l . 令 $d'(x) = d(x) - l$ 表示适应值差函数 d 和评价函数 f , 根据引理 1 可得 $V' := \min\{E(d'(x_t) - d'(x_{t+1})) | t \in \mathbb{N}\}$, 那么有

$$\begin{aligned}
 E(T|_l^{l+\Delta l}) &\leq \frac{d'(x_0)}{V} = \frac{\Delta l}{\min\{E(d'(x_t) - d'(x_{t+1})) | 0 \leq d'(x_t) \leq \Delta l\}} \\
 &= \frac{\Delta l}{\min\{E(d(x_t) - d(x_{t+1})) | l \leq d(x_t) \leq l + \Delta l\}} \\
 &= \frac{\Delta l}{\min\{G(r) | l \leq r \leq l + \Delta l\}}.
 \end{aligned}$$

因为 $G(r)$ 是关于 r 的递增函数, 所以 $\min\{G(r) | l \leq r \leq l + \Delta l\} = G(l)$. 因此, $E(T|_l^{l+\Delta l}) \leq \Delta l / G(r)$.

根据黎曼积分的性质, 我们可以得到连续型 (1+1)EA 算法的平均计算时间

$$E(T|_\varepsilon^L) = \lim_{n \rightarrow +\infty} \sum_{i=1}^n E(T|_{y_{i-1}}^{y_i}) \leq \lim_{\gamma \rightarrow 0} \sum_{i=1}^n 1/G(y_{i-1}) \Delta y_i = \int_\varepsilon^L 1/G(r) dr,$$

其中我们将区间分成 n 段: $\varepsilon = y_0 < y_1 < \dots < y_{n-1} < y_n = L$.

附录 C 定理 2 证明

下界的证明不像上界那么直接, 具体分析如下:

根据黎曼积分的定义, 我们将区间分成 n 段 $\varepsilon = y_0 < y_1 < \dots < y_{n-1} < y_n = L$, $E(T|_\varepsilon^L) = \lim_{n \rightarrow +\infty} \sum_{i=1}^n E(T|_{y_{i-1}}^{y_i})$, 其中, 对 $i = 1, 2, \dots, n$ 有

$$E(T|_{y_{i-1}}^{y_i}) = \int_0^{+\infty} t \cdot g_T(t) dt = \int_0^{+\infty} g_{Dt}(r) \cdot \frac{\Delta y_i}{r} dr,$$

其中 $g_T(t)$ 表示经过区间 $[y_{i-1}, y_i]$ 运行时间 T 的概率密度函数. $g_{Dt}(r)$ 表示当前步长的概率密度函数, 且有 $t = \frac{\Delta y_i}{r}$.

由柯西不等式: $\int_0^{+\infty} g_{Dt}(r) \cdot \frac{1}{r} dr \cdot \int_0^{+\infty} g_{Dt}(r) \cdot r dr \geq \left(\int_0^{+\infty} g_{Dt}(r) dr \right)^2 = 1$,

则有: $\int_0^{+\infty} g_{Dt}(r) \cdot \frac{\Delta y_i}{r} dr \geq \frac{\Delta y_i}{\int_0^{+\infty} g_{Dt}(r) \cdot r dr} = \frac{\Delta y_i}{G(x_i)}$.

因此, $E(T|_\varepsilon^L) = \lim_{n \rightarrow +\infty} \sum_{i=1}^n E(T|_{y_{i-1}}^{y_i}) \geq \lim_{n \rightarrow +\infty} \sum_{i=1}^n \frac{\Delta y_i}{G(y_i)} = \int_\varepsilon^L 1/G(r) dr$ 成立.

附录 D 引理 2 证明

首先根据 $G(r)$ 的定义

$$G(r) = E[d(x_t) - d(x_{t+1}) | d(x_t) = r] = \int_\delta f_{x_{t+1}}(x | d(x_t) = r) (r - d(x)) dx, \quad (D1)$$

其中, $\delta = \{x | d(x) < r\}$ $f_{x_{t+1}}(x | d(x_t) = r)$ 表示当 $d(x_t) = r$ 时, x_{t+1} 的概率密度函数.

另一方面: 根据算法定义, 当 $d(x_{t+1}) < d(x_t) = r$ (即 x_{t+1} 在 δ 区域内) 时, $x_{t+1} = x_t + \Delta_t$, 所以, $f_{x_{t+1}}(x | d(x_t) = r)$ 等价于 $f_\Delta(x)$, 其中 $\Delta = x_t - x_{t+1}$, $f_\Delta(x)$ 为 Δ 的概率密度函数. 易知

$$f_\Delta(x) = \prod_{i=1}^n f_2(x_i) = \frac{1}{(\sqrt{2\pi})^n} \cdot e^{-\frac{d(x)^2}{2}}, \quad (D2)$$

那么, 显然 $f_{\Delta}(x)$ 随着 Δ 的长度的减少而减少, 又因为在连续型 EA-I 算法中, $0 \leq d(\Delta) \leq 2d(x_t) = 2r$, 所以代入 (D2) 式得,

$$\frac{1}{(\sqrt{2\pi})^n} \cdot e^{-\frac{d(x_t)^2}{2}} \leq f_{\Delta}(x) \leq \frac{1}{(\sqrt{2\pi})^n},$$

所以

$$\frac{1}{(\sqrt{2\pi})^n} \cdot e^{-\frac{d(x_t)^2}{2}} \leq f_{x_{t+1}}(x|d(x_t) = r) \leq \frac{1}{(\sqrt{2\pi})^n}, \quad (\text{D3})$$

(D3) 式代入 (D1) 式, 可得

$$\frac{1}{(\sqrt{2\pi})^n} \cdot e^{-\frac{d(x_t)^2}{2}} \cdot \int_{\delta} (r - d(x)) dx \leq G(r) \leq \frac{1}{(\sqrt{2\pi})^n} \cdot \int_{\delta} (r - d(x)) dx,$$

结合附录 I 可得

$$\frac{e^{-\frac{r^2}{2}} \cdot r^{n+1}}{\Gamma(n/2 + 1) \cdot (\sqrt{2})^n \cdot (n + 1)} \leq G(r) \leq \frac{r^{n+1}}{\Gamma(n/2 + 1) \cdot (\sqrt{2})^n \cdot (n + 1)}.$$

附录 E 定理 3 证明

将引理 2 的结论代入推论 1, 可得

$$\begin{aligned} & \int_{\varepsilon}^{0.5} 1/G(r) dr \\ & \geq \int_{\varepsilon}^{0.5} \Gamma(n/2 + 1) \cdot (n + 1) \cdot r^{-(n+1)} \cdot (\sqrt{2})^n dr \\ & = \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \int_{\varepsilon}^{0.5} r^{-(n+1)} dr \\ & = \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \cdot \left(-\frac{1}{n} \cdot r^{-n} \Big|_{r=\varepsilon}^{0.5} \right) \\ & > \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \cdot \frac{1}{n} \cdot (\varepsilon^{-n} - (0.5)^{-n}). \end{aligned}$$

附录 F 定理 4 证明

将引理 2 的结论代入推论 1, 可得

$$\begin{aligned} & \int_{\varepsilon}^{0.5} 1/G_l(r) dr \\ & \leq \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \int_{\varepsilon}^{0.5} r^{-(n+1)} \cdot e^{2r^2} dr \\ & < \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \cdot e^2 \cdot \int_{\varepsilon}^{0.5} r^{-(n+1)} dr \\ & = \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \cdot e^2 \cdot \left(\frac{-1}{n \cdot (0.5)^n} + \frac{1}{n \cdot \varepsilon^n} \right) \\ & < \Gamma(n/2 + 1) \cdot (n + 1) \cdot (\sqrt{2})^n \frac{1}{n \cdot \varepsilon^n}. \end{aligned}$$

附录 G 引理 3 证明

与引理 2 的证明类似. 根据 $G(r)$ 的定义

$$G(r) = E[d(x_t) - d(x_{t+1}) | d(x_t) = r] = \int_{\delta_1} f_{x_{t+1}}(x | d(x_t) = r) (r - d(x)) dx, \quad (\text{G1})$$

其中, $\delta_1 = \{x | d(x) < r\}$, 当 $d(x_t) = r$ 时, x_{t+1} 的概率密度函数是 $f_{x_{t+1}}(x | d(x_t) = r)$.

然而因为算法规定可知: 当步长满足均匀分布时, 给定一个解 x_t , 那么下一个解 x_{t+1} 的范围是有限的. 具体的, 令 $\delta_2 = \{x | |(x - x_t)_i| \leq 1, i = 1, 2, \dots, n\}$, 因为在 $(-1, 1)$ 均匀分布的随机过程中, 当 $|x_i| > 1 (i = 1, 2, \dots, n)$ 时, $f_{\Delta}(x) = 0 (f_{\Delta}(x)$ 表示步长 Δ 的概率密度函数); 所以在空间 $\mathbb{R}^n - \delta_2 = \{x | |(x - x_t)_i| > 1, i = 1, 2, \dots, n\}$ 上, $f_{x_{t+1}}(x | d(x_t) = r) = 0$, 所以根据 (G1) 式有

$$G(r) = \int_{\delta} f_{x_{t+1}}(x | d(x_t) = r) (r - d(x)) dx,$$

其中, $\delta = \delta_1 \cap \delta_2$, 即集合 δ_1 与集合 δ_2 的交集. 因为当 $|x| \neq 0$ 和 $r \leq 0.5$ 时, 有 $|d(x)| < |x|$, 即 $\delta_1 \cap \delta_2 = \delta_1$; 图 G1 所示, 立方体区域包围球体区域. 因此, $\delta = \delta_1$.

图 G1 (网络版彩图) 当 $r \leq 0.5$, 立方体区域包围球体区域
 Figure G1 (Color online) Sphere region in cubic region when $r \leq 0.5$

因为 $\delta_1 = \{x | d(x) < r\}$ 是一个高维球体, 所以其积分为

$$\int_{\delta} f_{x_{t+1}}(x | d(x_t) = r)(r - d(x)) dx = \left(\frac{1}{2^n}\right) \int_{\delta} (r - d(x)) dx,$$

然后根据附录 H 得到: 当 $r \leq 0.5$, $G(r) = 2^{-n} \cdot r^{n+1} \cdot \sqrt{\pi}^n / (\Gamma(n/2 + 1) \cdot (n + 1))$.

附录 H 积分 $\int_{\delta} (r - d(x)) dx$ 计算

根据 B 函数的定义: $B(m + 1, n + 1) = 2 \int_0^{\pi/2} (\cos x)^{2m+1} (\sin x)^{2n+1} dx$,
 所以, 对于 $k \in \mathbb{N}$ 有 $\int_0^{\pi/2} (\sin x)^k dx = \frac{1}{2} \cdot B\left(\frac{1}{2}, \frac{k+1}{2}\right)$.

由于 $B(m, n) = \Gamma(m) \cdot \Gamma(n) / \Gamma(m + n)$, $\Gamma(1/2) = \sqrt{\pi}$, 所以

$$\int_0^{\pi} (\sin x)^k dx = 2 \int_0^{\pi/2} (\sin x)^k dx = B\left(\frac{1}{2}, \frac{k+1}{2}\right) = \sqrt{\pi} \cdot \frac{\Gamma(k/2 + 1/2)}{\Gamma(k/2 + 1)}.$$

为了方便计算 $\int_{\delta} (r - d(x)) dx$, 我们用符号 l 代替定量 r , 用 r 表 $d(x)$ 这样得到下式

$$\begin{aligned} & \int_{r=0}^l \int_{\beta_n=0}^{\pi} \int_{\beta_{n-1}=0}^{\pi} \cdots \int_{\beta_3=0}^{\pi} \int_{\alpha=0}^{2\pi} (l - r) dr (r \sin \beta_n \cdots \sin \beta_3 d\alpha) \cdots (r d\beta_n) \\ &= \int_{r=0}^l \int_{\beta_n=0}^{\pi} \int_{\beta_{n-1}=0}^{\pi} \cdots \int_{\beta_3=0}^{\pi} \int_{\alpha=0}^{2\pi} (l - r) dr (r \sin \beta_n \cdots \sin \beta_3 d\alpha) (r \sin \beta_n \cdots \sin \beta_4 d\beta_3) \cdots (r d\beta_n) \\ &= l \int_{r=0}^l \int_{\beta_n=0}^{\pi} \int_{\beta_{n-1}=0}^{\pi} \cdots \int_{\beta_3=0}^{\pi} \int_{\alpha=0}^{2\pi} dr (r \sin \beta_n \cdots \sin \beta_3 d\alpha) (r \sin \beta_n \cdots \sin \beta_4 d\beta_3) \cdots (r d\beta_n) \\ &\quad - \int_{r=0}^l \int_{\beta_n=0}^{\pi} \int_{\beta_{n-1}=0}^{\pi} \cdots \int_{\beta_3=0}^{\pi} \int_{\alpha=0}^{2\pi} r dr (r \sin \beta_n \cdots \sin \beta_3 d\alpha) (r \sin \beta_n \cdots \sin \beta_4 d\beta_3) \cdots (r d\beta_n) \\ &= l \cdot \int_{r=0}^l dr \cdot r^{n-1} \cdot 2\pi \cdot \prod_{i=1}^{n-2} \int_{\beta=0}^{\pi} (\sin \beta)^i d\beta - \int_{r=0}^l r dr \cdot r^{n-1} \cdot 2\pi \cdot \prod_{i=1}^{n-2} \int_{\beta=0}^{\pi} (\sin \beta)^i d\beta \\ &= l \cdot \int_{r=0}^l dr \cdot r^{n-1} \cdot 2\pi \cdot \prod_{i=1}^{n-2} B(1/2, (i+1)/2) - \int_{r=0}^l r dr \cdot r^{n-1} \cdot 2\pi \cdot \prod_{i=1}^{n-2} B(1/2, (i+1)/2) \\ &= l \cdot \int_{r=0}^l dr \cdot r^{n-1} \cdot 2\sqrt{\pi}^n / \Gamma(n/2) - \int_{r=0}^l r dr \cdot r^{n-1} \cdot 2\sqrt{\pi}^n / \Gamma(n/2) \\ &= l \cdot \sqrt{\pi}^n \cdot l^n / \Gamma(n/2 + 1) - 1/(n+1) \cdot l^{n+1} \cdot 2\sqrt{\pi}^n / \Gamma(n/2) \\ &= l^{n+1} \cdot \sqrt{\pi}^n / \Gamma(n/2) \cdot 2/n - l^{n+1} \cdot 2\sqrt{\pi}^n / \Gamma(n/2) \cdot 1/(n+1) \\ &= l^{n+1} \cdot \sqrt{\pi}^n / (\Gamma(n/2 + 1) \cdot (n + 1)). \end{aligned}$$

附录 I 定理 5 证明

将引理 3 的结论代入推论 1, 可得

$$\begin{aligned}
 \int_{\varepsilon}^{0.5} 1/G_l(r)dr &= \int_{\varepsilon}^{0.5} \Gamma(n/2+1) \cdot (n+1) \cdot r^{-(n+1)} \cdot \sqrt{\pi}^{-n} / 2^n dr \\
 &= \Gamma(n/2+1) \cdot (n+1) \cdot \int_{\varepsilon}^{0.5} r^{-(n+1)} \cdot \sqrt{\pi}^{-n} \cdot 2^n dr \\
 &= \Gamma(n/2+1) \cdot (n+1) \cdot \sqrt{\pi}^{-n} \cdot 2^n \cdot \int_{\varepsilon}^{0.5} r^{-(n+1)} dr \\
 &= \Gamma(n/2+1) \cdot (n+1) \cdot \sqrt{\pi}^{-n} \cdot 2^n \cdot \left(-\frac{1}{n} \cdot r^{-n}\right) \Big|_{r=\varepsilon}^{0.5} \\
 &= \Gamma(n/2+1) \cdot \frac{(n+1)}{n} \cdot \sqrt{\pi}^{-n} \cdot 2^n \cdot (\varepsilon^{-n} - 0.5^{-n}) \\
 &= \Gamma(n/2+1) \cdot \frac{(n+1)}{n} \cdot \frac{2^n}{(\sqrt{\pi})^n} \cdot \left(\frac{1}{\varepsilon^n} - \frac{1}{0.5^n}\right).
 \end{aligned}$$

Runtime analysis for continuous (1+1) evolutionary algorithm based on average gain model

HUANG Han^{1*}, XU WeiDi¹, ZHANG YuShan^{2,3}, LIN ZhiYong⁴ & HAO ZhiFeng⁵

1 School of software Engineering, South China University of Technology, Guangzhou 510006, China;

2 School of Mathematics & Statistics, Guangdong University of Finance & Economics, Guangzhou 510320, China;

3 State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China;

4 School of Computer Science, Guangdong Polytechnic Normal University, Guangzhou 510665, China;

5 Faculty of Computer Science, Guangdong University of Technology, Guangzhou 510006, China

*E-mail: hhan@scut.edu.cn

Abstract Runtime analysis of continuous evolutionary algorithm (EA) is an open problem in theoretical foundation of evolutionary computation. There are fewer results about it than the runtime studies of discrete EA. For an example of (1+1)EA, an average gain model and its calculating method were proposed to produce a theory of runtime analysis as an index of computational time complexity. The average gain was computed to estimate the average runtime of two (1+1)EAs based on the mutation of standard normal distribution and uniform distribution, for Sphere function which is focused on by many researchers. The analysis result indicates that computational time complexity of the (1+1)EAs is exponential order. Furthermore, the solution speed of uniform-distribution mutation is faster than standard normal distribution with the same error accuracy and initial distance. Numerical results also verify the correctness of the proposed theory and the usefulness of the average gain model.

Keywords evolutionary algorithm, theoretical foundation of evolutionary computation, runtime analysis, average gain model, Sphere function

HUANG Han was born in 1980. He received the Ph.D. degree in Computer Science from the South China University of Technology (SCUT), Guangzhou, in 2008. Currently, he is an associated professor at School of Software Engineering in SCUT. His research interests include evolutionary computation, swarm intelligence and their application. Dr. Huang is a senior member of CCF.

XU WeiDi was born in 1992. He is now an undergraduate student in the South China University of Technology, Guangzhou. His current research interests include machine learning, computational intelligence, computer vision, and data mining.

ZHANG YuShan was born in 1975. He received the Ph.D. degree in computer science from South China University of Technology, Guangzhou, in 2013. His main research interests include theoretical foundation of evolutionary algorithms, convergence and time complexity analysis of bio-inspired algorithms.

LIN ZhiYong was born in 1977. He received the Ph.D. degree in computer application technology from South China University of Technology, Guangzhou, in 2009. Currently, he is an associate professor at the School of Computer Science, Guangdong Polytechnic Normal University. His research interests include algorithm design and analysis, machine learning, and computational intelligence.