H¨older连续多元函数的高效利普希茨全局优化

Kaan Gokcesu、Hakan Gokcesu

arXiv:2303.14293v1 [cs.LG] 24 Mar 2023

摘要——本文提出了一种针对H¨older连续多元函数的有效全局优化技术。不同于构造下界代理函数的传统方法，该算法采用预先确定的查询生成规则，因此在计算上更具优势。我们通过平均遗憾或累积遗憾评估算法的性能，这也意味着简单遗憾存在一个界，并反映了该方法的整体有效性。结果表明，采用适当参数时，该算法能够在给定时间范围 T内，对H¨older指数为 α 、定义于 n维空间的H¨older连续目标函数进行优化，并达到 $O(T^{-\frac{\alpha}{n}})$的平均遗憾界。我们证明该界是极小极大最优的。

I. 引言

在尽可能减少函数评估值的同时寻找函数的全局最优点，是优化理论中的一个经典问题。Pinter [1] 提出了全局优化这一概念，它已成为许多应用中的关键技术，包括复杂学习系统中的超参数调优 [2], [3]。然而，在这类系统中，目标函数可能不具备有助于优化的性质，例如凸性。因此，导数自由优化 [4], 、黑箱优化 [5], 和全局优化 [1], 等术语，均被用于指代对未知且可能非凸函数进行的序贯优化。

全局优化问题在众多研究领域中都受到了广泛关注，包括非凸优化 [6]–[8], 贝叶斯优化 [9], 凸优化 [10]–[12], 赌博机优化 [13], 以及随机优化 [14], [15]。此外，全局优化还可直接应用于各种现实问题，例如分布估计 [16]–[18], 多臂赌博机 [19]–[21], 控制理论 [22], 信号处理 [23], 博弈论[24], 预测 [25], [26], 决策理论 [27], 以及异常检测 [28]–[30]。近年来，全局优化的重要性显著提升，这得益于它在不同研究领域中的广泛适用性，以及它解决传统优化方法难以处理的复杂问题的潜力。

全局优化技术旨在仅通过获取目标函数 $f(\cdot)$ 在特定查询点处的函数值来对其进行优化。实现这一目标可以采用多种启发式方法，但我们着重关注基于正则性的方法 [31], [32], ，尤其是利普希茨正则性（及其推广形式 H¨older 正则性）。Piyavskii 最先提出了利普希茨正则性方法，用于优化单变量利普希茨连续函数。其核心思想是为目标函数构造下界代理函数，并查询它们的最小值 [33]。Shubert 提出了一种类似的方法 [34], ，通常称为 Piyavskii-Shubert 算法。

多年来，针对利普希茨连续函数的 Piyavskii-Shubert 算法已因其在各种应用中的价值而得到广泛研究和改进 [7], [8], [35]–[41]。随着时间推移，该算法通过泰勒展开扩展到了多元函数 [42], ，并在 [43] 中提升了性能。人们还提出了其他方法，例如使用光滑辅助函数 [44] ，以及在 [45] 中提出的一种适用于可微单变量函数的变体。布伦特方法于 [46], 提出，适用于定义在紧区间上且具有有界二阶导数的函数。近期研究还关注了具有广义利普希茨正则性的函数的单变量全局优化 [47]。

全局优化算法根据其收敛到目标函数最优值的能力进行评估，该能力用简单遗憾 [48] 衡量。此指标量化了迄今为止所进行的查询中最佳查询的函数值与最优解的函数值之间的差异。对于 n 维 Lipschitz 连续目标函数，Piyavskii-Shubert 算法的遗憾界为 $O\left(T^{-\frac{1}{n}}\right)$ [37] [7] $\tilde{r}_T =$ ()。对于单变量函数，提出了一种 Piyavskii-Shubert 算法的变体，可在 ϵ 内通过 $O(\int_0^1 (f(x) - f(x^*) + \epsilon)^{-1} dx)$ 次查询达到简单遗憾，这优于先前的结果 [48]。类似地，[45] 改进了关于单变量函数的先前结果 [7], [48] 。Piyavskii-Shubert 算法的一种变体 LIPO 在更强的假设 [49] 下实现了更好的简单遗憾界。最后， [50] 研究了含噪评估条件下 Piyavskii-Shubert 算法的简单遗憾。

Piyavskii-Shubert 算法已针对单变量函数的累积遗憾界进行了效果分析，而累积遗憾界是比简单遗憾[47]更严格的衡量指标。虽然这些方法适用于利普希茨连续函数，但优化下界函数以确定查询点的成本可能相当高。为解决这一问题，近期一项研究 [51] 采用预设采样集，在单变量情形下实现了与 Piyavskii-Shubert 算法相近的累积遗憾界。然而，这些方法仅适用于只有一个自变量的目标函数。另一项工作将这一方法扩展至多变量利普希茨连续函数 [52]。我们进一步扩展该方法，使其适用于一般的 H¨older 连续函数。

第二节介绍一些预备知识，包括问题定义。第三节介绍算法及其实现。第四节对其累积遗憾进行分析，并证明通过适当选择参数可以实现极小极大最优遗憾。

II. 预备知识

我们首先正式定义多变量情形下的全局优化问题。目标是最小化一个包含多个变量的函数 $f(\cdot)$，其中 $f(\cdot) : \Omega \to \Re$，且 Ω 是 $\Re^n$ 的紧子集。在本研究中，我们关注 Ω 为单位 n 维立方体的情形，记作 $\Omega \equiv [0,1]^n$。尽管 $f(\cdot)$ 并非凸函数，但它并非任意函数，通常具有一定的正则性，例如赫尔德连续性。

定义 1. 设 $f(\cdot)$ 满足以下正则性条件：

$$|f(x) - f(y)| \leq C\|x - y\|^\alpha,$$

对于任意 $x, y \in \Omega$，其中 $1 > \alpha > 0$ 为 Hölder 指数，$C > 0$ 为正则性常数，$\|\cdot\|$ 为欧几里得范数。

我们采用迭代方法，通过根据先前已评估查询的函数值反复选择查询，来最小化多元函数 $f(\cdot)$。负责选择下一个查询点的函数记为 $\Gamma(\cdot)$，通常由 Piyavskii-Shubert 变体推导得出。目标是以尽可能少的函数评估值找到最优点。这等价于将目标函数视为损失函数，其中每个函数值 $f(x_t)$ 都表示选择查询点 $x_t \in \Omega$ 所产生的损失。为了分析该方法的性能，我们采用遗憾这一概念，将我们的函数评估值与最优函数值进行比较。具体而言，我们将 x_* 定义为 $f(\cdot)$ 的全局最小值点，并以此比较我们所选查询点对应的函数值。由于 $f(\cdot)$ 的函数值可能任意高，因此我们采用遗憾来比较所选查询点的函数值与最优函数值。

在全局优化问题中，对于时间范围T，有两种遗憾分析方法。第一种是简单遗憾，它只将选定点视为产生损失的点，而不考虑查询点。选定点 $\tilde{x}_T$ 从查询点 $\{x_t\}_{t=1}^T$ 中选出，其函数值与 $f(\cdot)$可能达到的最小函数值进行比较，由此得到简单遗憾 $\tilde{r}_T = f(\tilde{x}_T) - \min_{x\in\Omega} f(x)$。第二种遗憾分析方法是平均遗憾，它认为所有查询点都会产生损失。平均遗憾的计算方法是查询点的平均函数值与 $f(\cdot)$的最小函数值之差，

$$r_T = \frac{1}{T}\sum_{t=1}^{T} f(x_t) - \min_{x\in\Omega} f(x)$$

.

尽管在某些问题场景中，评估算法收敛至最优函数值的情况或简单遗憾可能很有用，但由于全局优化问题本身的难度，这种评估可能并不适用。这一点可以从以下事实看出：虽然基本网格搜索方法的简单遗憾是极小极大最优的，但其平均遗憾仍保持为常数 [52]。此外，平均遗憾的上界意味着简单遗憾也有上界。因此，我们将重点使用平均遗憾来评估算法性能。我们的目标是开发一种算法，使其达到极小极大最优的平均遗憾 $O(T^{-\frac{\alpha}{n}})$。

III. 算法

本节中，我们提出一种算法，用于最小化满足定义 1 中所定义约束的非凸目标函数 $f(\cdot)$。尽管我们的优化是在 n 维单位立方体 $[0,1]^n$ 上进行的，但通过对输入进行适当的缩放和平移，该算法可以扩展到适用于大小和位置各异的任意立方体。与传统的全局优化算法（如 Piyavskii-Shubert 变体）使用下界代理函数不同，我们的算法采用固定的查询构造规则，以提高效率。算法概述如下：

1) 将参数 C_0 设为输入。2) 将定义域 $\Omega = [0,1]^n$ 包裹在一个超矩形$\Theta = [0, \theta^{n-1}] \times [0, \theta^{n-2}] \times \ldots \times [0, \theta^0]$中，其中 $\theta = 2^{\frac{1}{n}}$。通过将任意 $\tilde{x} \in \Theta$ 投影到 Ω 上来计算其函数值，从而解析地扩展 $f(\cdot)$，即 $f(\tilde{x}) = f(\arg\min_{x\in\Omega}\|\tilde{x}-x\|)$，这会将其元素截断为 1。3) 设置中点 $x_a = \{\theta^{-1}, \theta^{-2}, \ldots, \theta^{-n}\}$ 及其边向量 $v_a = \{\theta^{-1}, \theta^{-2}, \ldots, \theta^{-n}\}$。采样 x_a，并将其函数值设为 $f_a = f(\arg\min_{x\in\Omega}\|x - x_a\|)$。4) 使用 x_a、v_a和f_a作为输入，确定两个候选点 x_b、x_c 及其边向量 v_b、v_c 和得分 s_b、s_c。将候选点添加到潜在查询列表中。5) 对得分最低的候选点 s'进行采样，将其从列表中移除，并将其设为查询点 x'。计算 f' 的函数值$f(\arg\min_{x\in\Omega}\|x - x'\|)$，并将其边向量设为 v'。6) 如果查询次数小于 T，则以输入 $x_a = x'$、$v_a = v'$、$f_a = f'$转到步骤 4。否则停止，并返回所有查询点及其函数值。

注 1. 该算法具有以下特点：

- 该算法会保留潜在查询点的参数，直到对其进行评估。
- 每次查询后，算法都会生成两个新的潜在查询。
- 潜在查询的数量会随着查询次数的增加而线性增长。
- 可以根据具体问题和可用的计算资源，考虑采用多种停止准则。例如，算法可以在达到固定的查询次数后终止，也可以在达到一定的准确度后终止。
- 每个潜在查询都可以表示为二进制字符串，这种表示简洁高效，且便于在算法的不同部分之间存储和传递潜在查询。

该算法选择潜在查询点的方式不同于常见的 Piyavskii-Shubert 变体。它并非构造经过采样点的下界代理函数并对其进行最小化，而是将由采样点 x' 及其边向量v' 定义的超矩形沿最长维度等分为两个大小相同的超矩形，以此确定候选查询点。这些新建超矩形的中心点即为潜在的新查询点。下文给出了这些点的具体表达式。

定义 2. 对于目标函数 $f(\cdot)$，给定查询点 x_a 及其边向量 v_a；潜在查询点 x_b、x_c及其边向量 v_b、v_c 定义如下：

$$\begin{aligned} x_b =& x_a + z, & v_b = v_a - z, \\ x_c =& x_a - z, & v_c = v_a - z, \end{aligned}$$

其中，z除在 $I = \arg\min^{i\in\{1,2,\ldots,n\}} v_a(i)$ 处外均为零，其中 $v_a(i)$ 表示 v_a 的第 i^{th} 个元素。因此，

$$z(I) = 0.5v_a(I), \qquad z(i) = 0, i \neq I.$$

注 2. 分数 s_b 和 s_c 按如下方式生成

$$s_b = f_a - C_0\|v_a\|, \qquad s_c = f_a - C_0\|v_a\|.$$

注 3. 我们指出，由于算法的运行结构，即使所有分数都加上某个 ϵ，采样点也不会改变。

引理 1. 对于输入 C_0 以及定义 1 中的正则性及其相应的 C 和 α，有

$$\epsilon_0 \leq C_0^{\frac{\alpha}{\alpha-1}} C^{\frac{-1}{\alpha-1}},$$

其中，ϵ_0 是满足对所有可能的 x，$y \in \Omega$，$\epsilon \geq C\|x-y\|^\alpha - C_0\|x-y\|$. 均成立的最小 ϵ

证明. 关于参数 C_0、C、α，一种等价表示如下：

$$\epsilon_0 = \arg\min_{\epsilon\in\Re} \epsilon: \quad \epsilon \geq C\triangle^\alpha - C_0\triangle, \qquad 0 \leq \triangle \leq \sqrt{n} \quad (1)$$

为求解 ϵ，我们需要考察 $C\triangle^\alpha$ 关于 $\triangle$ 的导数，即 $C\alpha\triangle^{\alpha-1}$。令 $C_0 = C\alpha\triangle_0^{\alpha-1}$，其中 $\triangle_0 \in \Re$ 为某个值，且 $\triangle_0 = \left(\frac{C_0}{C\alpha}\right)^{\frac{1}{\alpha-1}}$.。于是，我们有 $\epsilon_0 + C_0\triangle_0 = C\triangle_0^\alpha$.。因此，

$$\begin{aligned} \epsilon_0 =& C\left(\frac{C_0}{C\alpha}\right)^{\frac{\alpha}{\alpha-1}} - C_0\left(\frac{C_0}{C\alpha}\right)^{\frac{1}{\alpha-1}} & (2) \\ =& C_0^{\frac{\alpha}{\alpha-1}} C^{\frac{-1}{\alpha-1}}\left(\alpha^{\frac{\alpha}{1-\alpha}} - \alpha^{\frac{1}{1-\alpha}}\right) & (3) \\ \leq& C_0^{\frac{\alpha}{\alpha-1}} C^{\frac{-1}{\alpha-1}}, & (4) \end{aligned}$$

由此得证。 □

引理 2. 对于满足定义 1 的目标函数 $f(\cdot)$，当查询点 x_a 及其函数值 f_a 已知，候选点 x_b、x_c 的得分分别为 s_b 和 s_c，且其对应的边向量分别为 v_b、v_c 时，我们有

$$\begin{aligned} s_b - \epsilon_0 \leq& f_a - C\|v_a\|^\alpha, \\ s_c - \epsilon_0 \leq& f_a - C\|v_a\|^\alpha, \end{aligned}$$

即，平移后的得分 $s_b - \epsilon_0$ 和 $s_c - \epsilon_0$ 分别为查询点 x_b 和 x_c 对应区域的函数值提供下界。

证明. 点 x_b 及其边向量 v_b 覆盖超矩形区域 $\mathcal{H}_b$，该区域的中心为 x_b，各边界平面到中心的距离由其边向量 v_b 给出。根据定义 1 中的正则性，对于 $x \in \mathcal{H}_b$，我们有：$f(x) \leq f_a - C\|x - x_a\|^\alpha \leq f_a - C\|v_a\|^\alpha$.。对 x_c、v_c 和 $\mathcal{H}_c$ 也可作类似论证；由此得证。 □

下一节将对我们的算法进行性能分析。

IV. 遗憾分析

我们首先通过界定单次采样点的遗憾来开展遗憾分析。

引理 3. 对于满足定义 1 的给定目标函数 $f(\cdot)$，设我们采样点 x_b，其对应的得分为 s_b；该得分是在查询 x_a 及其函数值 f_a之后生成的。当点 x_b 的函数值为 f_b时，我们有以下结果。

$$f_b - \min_{x\in\Omega} f(x) \leq C_0\theta\|v_b\| + C\|v_b\|^\alpha + \epsilon_0, \qquad (5)$$

其中，v_b 是查询 x_b对应的边向量。

证明。我们知道，查询 x_b的得分，即 s_b，完全构成超矩形 $\mathcal{H}_b$ 内函数评估值的下界；该超矩形对应于中心点 x_b 和边向量 v_b，即 $s_b - \epsilon_0 \leq \min_{x\in\mathcal{H}_b} f(x)$。根据算法的设计，查询列表由一组潜在查询 $\mathcal{X}$ 组成，其关联区域 $\mathcal{H}$ 两两不相交，且其并集覆盖整个搜索域。因此，我们有 $s_b - \epsilon_0 \leq \min_{x\in\Omega} f(x)$。结合引理 2，可得

$$f_a - C_0\|v_a\| - \epsilon_0 \leq \min_{x\in\Omega} f(x). \qquad (6)$$

此外，由于定义 1 中的正则性，我们有

$$f_b \leq f_a + C\|v_b\|^\alpha, \qquad (7)$$

其中，查询 x_b 是在对 x_a 采样并获得其函数值 f_a 后创建的。因此，结合这两个结果，我们得到

$$f_b \leq \min_{x\in\Omega} f(x) + C_0\theta\|v_b\| + C\|v_b\|^\alpha + \epsilon_0, \qquad (8)$$

由于按设计 $\theta\|v_b\| = \|v_a\|$，由此得证。 □

该引理利用采样点的边界值及其对应的函数值，给出了算法中该点的个体遗憾上界，这是一个最坏情况界

引理 4. 该算法的累积遗憾如下

$$\sum_{t=1}^T f_t - \sum_{t=1}^T f(x_*) \leq C_0\theta\sum_{t=1}^T\|v_t\| + C\sum_{t=1}^T\|v_t\|^\alpha + T\epsilon_0.$$

对于查询点 $\{x_t\}_{t=1}^T$，其中 $\{f_t\}_{t=1}^T$ 分别为其函数评估值，$\{v_t\}_{t=1}^T$ 分别为其边向量。

证明。我们运行该算法进行 T 次采样。设这些采样点为 $\{x_t\}_{t=1}^T$，其对应的边向量为 $\{v_t\}_{t=1}^T$，函数评估值为 $\{f_t\}_{t=1}^T$。根据定义 1 中的利普希茨连续性，对于初始采样，我们有

$$f_1 - f(x_*) \leq C\|v_1\|^\alpha. \qquad (9)$$

对于其他采样，即 $t \geq 2$，由引理 3 可得

$$f_t - f(x_*) \leq C_0\theta\|v_b\| + C\|v_b\|^\alpha + \epsilon_0. \qquad (10)$$

因此，累积遗憾有如下界

$$\sum_{t=1}^T f_t - \sum_{t=1}^T f(x_*) \leq C_0\theta\sum_{t=1}^T\|v_t\| + C\sum_{t=1}^T\|v_t\|^\alpha + T\epsilon_0, \qquad (11)$$

由此得证。 □

累积界取决于边向量范数的幂之和。为此，我们得到以下结果。

引理 5。我们有

$$\sum_{t=1}^{T}\|v_t\|^{\beta} \leq \frac{\theta^{n-\beta}}{\theta^{n-\beta}-1}V^{\beta}T^{\frac{n-\beta}{n}},$$

其中 $V=\|v_1\|$ 对于任意 β。

证明。根据算法的构造，每次新增采样时，采样点的边向量中的最大元素都会减半。由于算法开始时，初始边向量为 $v_1=\{\theta^{-1},\theta^{-2},\ldots,\theta^{-n}\}$，其中 $\theta=2^{\frac{1}{n}}$；边向量的范数因此按乘法因子 $2^{\frac{1}{n}}$递减。因此，在最坏情况下，我们有

$$\sum_{t=1}^{T}\|v_t\|^{\beta} \leq \sum_{i=0}^{K-1}2^{i}\frac{V^{\beta}}{2^{\frac{i\beta}{n}}}+M\frac{V^{\beta}}{2^{\frac{K\beta}{n}}}, \tag{12}$$

其中 $2^K-1+M=T$、$2^K\geq M\geq 1$和 $V=\|v_1\|$。利用 $M\leq 2^K$，我们有

$$\begin{aligned}\sum_{t=1}^{T}\|v_t\|^{\beta} \leq&\sum_{i=0}^{K}2^{i}\frac{V^{\beta}}{2^{\frac{i\beta}{n}}}, &(13)\\ \leq&V^{\beta}\sum_{i=0}^{K}2^{i(\frac{n-\beta}{n})} &(14)\\ \leq&V^{\beta}\frac{2^{(K+1)\frac{n-\beta}{n}}}{2^{\frac{n-\beta}{n}}-1} &(15)\\ \leq&\frac{\theta^{n-\beta}}{\theta^{n-\beta}-1}V^{\beta}T^{\frac{n-\beta}{n}}, &(16)\end{aligned}$$

由此即得证。 □

利用引理 5 关于 $\beta\in\{\alpha,1\}$的结果，我们得到以下结论。

定理 1. 当 $C_0=\lambda_0 CV^{\alpha-1}T^{\frac{1-\alpha}{n}}$ for some λ_0 时，我们有以下累积遗憾界

$$\sum_{t=1}^{T}f_t-\sum_{t=1}^{T}f(x_*)\leq O\left((1+\lambda_0+\lambda_0^{\frac{\alpha}{\alpha-1}})CV^{\alpha}T^{\frac{n-\alpha}{n}}\right)$$

证明。由引理 4 和引理 5 可得

$$\begin{aligned}\sum_{t=1}^{T}f_t-\sum_{t=1}^{T}f(x_*)\leq&C_0\theta\sum_{t=1}^{T}\|v_t\|+C\sum_{t=1}^{T}\|v_t\|^{\alpha}+T\epsilon_0, &(17)\\ \leq&C_0\frac{\theta^{n}}{\theta^{n-1}-1}VT^{\frac{n-1}{n}}\\ &+C\frac{\theta^{n-\alpha}}{\theta^{n-\alpha}-1}V^{\alpha}T^{\frac{n-\alpha}{n}}\\ &+C_0^{\frac{\alpha}{\alpha-1}}C^{\frac{-1}{\alpha-1}}T &(18)\\ \leq&O\left((1+\lambda_0+\lambda_0^{\frac{\alpha}{\alpha-1}})CV^{\alpha}T^{\frac{n-\alpha}{n}}\right), &(19)\end{aligned}$$

由此完成证明。 □

接下来，我们证明当 λ_0 为常数时，可以达到极小极大最优累积遗憾。

定理 2. 当 λ_0 为常数时，定理 1 中的结果是极小极大最优的。

证明。假设我们有一个目标函数 $f(\cdot)$，其在某点 x_* 处取得全局最优值，即 $f(x_*)=\min_{x\in\Omega}f(x)$。假设该函数满足：对于某个$\epsilon>0$，全局最优点的 ϵ-邻域按照定义 1 中的条件递减，而在其他所有位置均为常数 0，即，

$$f(x)=\begin{cases}C(\|x-x_*\|^{\alpha}-C\epsilon^{\alpha}), & \|x-x_*\|\leq\epsilon\\ 0, & \|x-x_*\|>\epsilon\end{cases}. \tag{20}$$

令一个算法对点 $\{x_t\}_{t=1}^{T}$进行采样。其累积遗憾为

$$R_T=\sum_{t=1}^{T}(f(x_t)-f(x_*)) \tag{21}$$

假设 $f(\cdot)$ 满足所有函数评估值均为 0，即$f(x_t)=0,\forall t$。因此，$R_T=-Tf(x^*)=CT\epsilon^{\alpha}$。对此，令

$$\epsilon_*=\min_{\tau}\|x_\tau-x_*\|, \tag{22}$$

这将使 ϵ 最大，从而使遗憾最大。因此，极小极大遗憾为

$$\begin{aligned}R_T^*\geq&CT\min_{\{x_t\}_{t=1}^{T}}\max_{x_*\in\Omega}\min_{\tau}\|x_\tau-x_*\|^{\alpha} &(23)\\ \geq&\Omega(Cn^{\frac{\alpha}{2}}T^{\frac{n-\alpha}{n}}), &(24)\end{aligned}$$

通过用小球填满单位立方体，使 C 保持为常数，由此完成证明。 □

注 4. 当 C、α、V 为常数时；我们也可以将遗憾写成更紧凑的形式：

$$R_T\leq O\left((\lambda_0+\lambda_0^{-\frac{\alpha}{1-\alpha}})T^{\frac{n-\alpha}{n}}\right), \tag{25}$$

由于 $\lambda_0+\lambda_0^{-\frac{\alpha}{1-\alpha}}$是凸函数，且在 $\lambda_0=\left(\frac{\alpha}{1-\alpha}\right)^{1-\alpha}$ 处取最小值，因此该表达式有下界 1。

推论 1. 当某个参数 α' 满足 $C_0=T^{\frac{1-\alpha'}{n}}$ 时，我们有 $R_T\leq O(T^{g(\alpha')})$，其中 $g(\cdot)$ 是某个函数。该函数是分段线性函数，在边界 $g(0)=1$ 和 $g(1)=1$ 处取最大值；在 $g(\alpha)=1-\alpha/n$ 处取最小值。

Proof. Let $C_0=T^{\frac{1-\alpha'}{n}}$. Then,

$$\lambda_0=A_{\alpha'}T^{\frac{\alpha-\alpha'}{n}}, \tag{26}$$

对于某个常数 $A_{\alpha'}$。因此，

$$\lambda_0^{-\frac{\alpha}{1-\alpha}}=B_{\alpha'}T^{\frac{\alpha'-\alpha}{1-\alpha}\frac{\alpha}{n}}, \tag{27}$$

对于某个常数 $B_{\alpha'}$。遗憾为

$$R_T\leq O\left(T^{1-\frac{\alpha'}{n}}+T^{1-\frac{1-\alpha'}{1-\alpha}\frac{\alpha}{n}}\right) \tag{28}$$

因此，$R_T\leq O(T^{g(\alpha')})$，其中

$$g(\alpha')=\begin{cases}1-\frac{\alpha'}{n}, & 0<\alpha'\leq\alpha,\\ 1-\frac{1-\alpha'}{1-\alpha}\frac{\alpha'}{n}, & \alpha<\alpha'<1,\end{cases} \tag{29}$$

由此完成证明。 □

参考文献

[1] J. D. Pint´er, 《Global optimization in action》, 《科学美国人》, 第264卷, 第54–63页, 1991年。[2] N. Cesa-Bianchi 和 G. Lugosi, 《预测、学习与博弈》。剑桥大学出版社, 2006年。[3] H. V. Poor, 《信号检测与估计导论》。新泽西州: 施普林格, 1994年。[4] L. M. Rios 和 N. V. Sahinidis, “无导数优化: 算法综述及软件实现比较”, 《全局优化杂志》, 第56卷, 第3期, 第1247–1293页, 2013年。[5] D. R. Jones、M. Schonlau 和 W. J. Welch, “昂贵黑箱函数的高效全局优化”, 《全局优化杂志》, 第13卷, 第4期, 第455–492页, 1998年。[6] P. Jain 和 P. Kar, “机器学习中的非凸优化”, 《机器学习基础与趋势》, 第10卷, 第3–4期, 第142–336页, 2017年。[7] P. Hansen、B. Jaumard 和 S.-H. Lu, “关于皮亚夫斯基全局优化算法的迭代次数”, 《运筹学数学》, 第16卷, 第2期, 第334–350页, 1991年。[8] P. Basso, “全局最大值定位的迭代方法”, 《SIAM数值分析杂志》, 第19卷, 第4期, 第781–792页, 1982年。[9] E. Brochu、V. M. Cora 和 N. De Freitas, “昂贵代价函数贝叶斯优化教程: 在主动用户建模和分层强化学习中的应用”, arXiv预印本 arXiv:1012.2599, 2010年。[10] S. Boyd、S. P. Boyd 和 L. Vandenberghe, 《凸优化》。剑桥大学出版社, 2004年。[11] Y. Nesterov, 《凸优化导论: 基础课程》。施普林格科学与商业媒体, 2003年, 第87卷。[12] S. Bubeck, “凸优化: 算法与复杂性”, 《机器学习基础与趋势》, 第8卷, 357, 2015 3 4, 231号——页码——。[13] R. Munos, “从老虎机到蒙特卡洛树搜索: 将乐观原则应用于优化与规划”, 《机器学习基础与趋势》, 第7卷, 第1期, 第1–129页, 2014年。[14] J. C. Spall 等, “在线学习与在线凸优化”, 《机器学习基础与趋势》, 第4卷, 第2期, 第107–194页, 2012年。[15] J. C. Spall, 《随机搜索与优化导论: 估计、仿真与控制》。约翰·威立父子公司, 2005年, 第65卷。[16] K. Gokcesu 和 S. S. Kozat, “利用指数分布族对非平稳信源进行在线密度估计”, IEEE神经网络汇刊, 学习系统版, 第29卷, 第9期, 第4473–4478页, 2018年。[17] F. M. J. Willems, “对二元独立分段同分布信源进行编码。”《IEEE信息论汇刊》, 第42卷, 第6期, 第2210–2217页, 1996年。[18] G. I. Shamir 和 N. Merhav, “针对分段平稳无记忆信源的低复杂度序贯无损编码”, 《IEEE信息论汇刊》, 第45卷, 第5期, 第1498–1519页, 1999年7月。[19] M. M. Neyshabouri、K. Gokcesu、H. Gokcesu、H. Ozkan 和 S. S. Kozat, “利用层次结构的渐近最优上下文老虎机算法”, 《IEEE神经网络与学习系统汇刊》, 第30卷, 第3期, 第923–937页, 2018年。[20] S. Bubeck 和 N. Cesa-Bianchi, “随机与非随机多臂老虎机问题的遗憾分析”, 《机器学习基础与趋势》, 第5卷, 第1期, 第1–122页, 2012年。[21] K. Gokcesu 和 S. S. Kozat, “针对对抗性多臂老虎机问题的在线极小极大最优算法”, 《IEEE神经网络与学习系统汇刊》, 第29卷, 第11期, 第5565–5580页, 2018年。[22] H. R. Berenji 和 P. Khedkar, “通过强化学习模糊逻辑控制器”, 《IEEE神经网络汇刊》, 第3卷, 第5期, 第724–740页, 1992年9月。[23] H. Ozkan、M. A. Donmez、S. Tunc 和 S. S. Kozat, “在线凸专家混合算法的确定性分析”, 《IEEE神经网络与学习系统汇刊》, 第26卷, 第7期, 第1575–1580页, 2015年7月。[24] R. Song、F. L. Lewis 和 Q. Wei, “用于求解非线性连续时间多人非零和博弈的离策略积分强化学习方法”, 《IEEE神经网络与学习系统汇刊》第PP卷, 第99期, 第1–10页, 2016年。[25] N. D. Vanli、K. Gokcesu、M. O. Sayin、H. Yildiz 和 S. S. Kozat, “层次结构上的序贯预测”, 《IEEE信号处理汇刊》, 第64卷, 第23期, 第6284–6298页, 2016年12月。[26] A. C. Singer 和 M. Feder, “通过模型阶数加权实现通用线性预测”, 《IEEE信号处理汇刊》, 第47卷, 第10期, 第2685–2699页

, 1999年10月。

[27] J. Moody 和 M. Saffell, 《通过直接强化学习交易》, IEEE神经网络汇刊, 第12卷, 第4期, 第875–889页, 2001年7月。[28] K. Gokcesu、M. M. Neyshabouri、H. Gokcesu 和 S. S. Kozat, 《基于增量决策树的序贯离群值检测》, IEEE信号处理汇刊, 第67卷, 第4期, 第993–1005页, 2019年。[29] I. Delibalta、K. Gokcesu、M. Simsek、L. Baruh 和 S. S. Kozat, 《利用嵌套树进行在线异常检测》, IEEE信号处理快报, 第23卷, 第12期, 第1867–1871页, 2016年。[30] K. Gokcesu 和 S. S. Kozat, 《非平稳环境下基于极小极大最优密度估计的在线异常检测》, IEEE信号处理汇刊, 第66卷, 第5期, 第1213–1227页, 2017年。[31] P. L. Bartlett、V. Gabillon 和 M. Valko, 《在最小局部光滑性假设下进行优化的一种简单、无参数且自适应的方法》, 载于《算法学习理论》。PMLR, 2019年, 第184–206页。[32] J.-B. Grill、M. Valko 和 R. Munos, 《未知光滑性条件下含噪函数的黑箱优化》, 《神经信息处理系统进展》第28卷, 第667–675页, 2015年。[33] S. Piyavskii, 《求函数绝对极值的一种算法》, 《苏联计算数学与数学物理》第12卷, 第4期, 第57–67页, 1972年。[34] B. O. Shubert, 《一种寻求函数全局最大值的序贯方法》, 《SIAM数值分析杂志》第9卷, 第3期, 第379–388页, 1972年。[35] F. Schoen, 《关于全局优化问题中的一种序贯搜索策略》, Calcolo, 第19卷, 第3期, 第321–334页, 1982年。[36] D. Q. Mayne 和 E. Polak, 《非可微优化问题的外逼近算法》, 《优化理论与应用杂志》第42卷, 第1期, 第19–30页, 1984年。[37] R. H. Mladineo, 《求多峰多变量函数全局最大值的一种算法》, 《数学规划》第34卷, 第2期, 第188–200页, 1986年。[38] Z. Shen 和 Y. Zhu, 《用于定位全局最大值的 Shubert 迭代方法的区间版本》, Computing, 第38卷, 第3期, 第275–280页, 1987年。[39] R. Horst 和 H. Tuy, 《关于多极值优化中全局方法的收敛性》, 《优化理论与应用杂志》第54卷, 第2期, 第253–271页, 1987年。[40] P. Hansen 和 B. Jaumard, 《Lipschitz优化》, 载于《全局优化手册》。施普林格, 1995年, 第407–493页。[41] R. Horst 和 H. Tuy, 《全局优化: 确定性方法》。施普林格科学与商业媒体, 2013年。[42] L. Breiman 和 A. Cutler, 《一种用于全局优化的确定性算法》, 《数学规划》第58卷, 1993年, 第179–199页。[43] W. Baritompa 和 A. Cutler, 《利用二阶导数加速全局优化覆盖方法》, 《全局优化杂志》第4卷, 第3期, 第329–341页, 1994年。[44] Y. D. Sergeyev, 《利用光滑辅助函数进行全局一维优化》, 《数学规划》第81卷, 第1期, 第127–146页, 1998年。[45] R. Ellaia、M. Z. Es-Sadek 和 H. Kasbioui, 《可微函数的改进 Piyavskii 全局一维优化方法》, 《应用数学》第3卷, 第1306–1320页, 2012年。[46] R. P. Brent, 《无导数最小化算法》。Courier Corporation, 2013年。[47] K. Gokcesu 和 H. Gokcesu, 《具有Lipschitz导数的单变量函数全局优化的遗憾分析》, arXiv预印本 arXiv:2108.10859, 2021年。[48] Y. M. Danilin, 《绝对最小值搜索算法效率的估计》, 《苏联计算数学与数学物理》第11卷, 第4期, 第261–267页, 1971年。[49] C. Malherbe 和 N. Vayatis, 《Lipschitz函数的全局优化》, 载于《国际机器学习会议》。PMLR, 2017年, 第2314–2323页。[50] C. Bouttier、T. Cesari 和 S. Gerchinovitz, 《Piyavskii-Shubert 算法用于全局 Lipschitz优化的遗憾分析》, arXiv预印本 arXiv:2002.02390, 2020年。[51] K. Gokcesu 和 H. Gokcesu, 《用于单变量函数高效全局优化的低遗憾二元采样方法》, arXiv预印本 arXiv:2201.07164, 2022年。[52] ——, 《Lipschitz连续多变量函数的高效极小极大最优全局优化》, arXiv预印本 arXiv:2206.02383, 2022年。