随机启发式搜索算法的若干理论问题

何军

School of Science and Technology, Nottingham Trent University, UK
jun.he@ieee.org

2019.07.05

内容概要

- 目的

  简要地介绍随机启发式搜索算法的几个基本问题及其相关研究方法

- 内容
  - 随机启发式搜索算法的含义
  - 数学模型
  - 收敛性分析
  - 计算时间分析
  - 解的质量分析
  - 收敛速度分析
  - 理论研究中的两个困难

随机启发式搜索算法

- 受自然界的启发，人们设计了许多智能优化算法，包括
  - 遗传算法
  - 模拟退火算法
  - 粒子群优化算法
- 算法直观易懂，容易实现
- 算法的一些共性
  - 随机性：随机产生新解
  - 启发式：启发式设计搜索策略
  - 迭代：一代一代地改进解
- 理论研究中就顾名思义称之为随机启发式搜索算法(randomised search heuristics)
- 相关算法名称
  - 生物启发式优化算法
  - 自然启发式优化算法
  - 演化算法等等

最优化问题和随机启发式搜索算法

- 最优化(极大化)问题

$$\max\{f(x); x \in \mathcal{S}\}$$

随机启发式搜索算法的抽象描述

$X_0 \leftarrow$ 按照某种初始概率分布$\Pr(X_0)$，产生一组初始解;
迭代次数$t \leftarrow 0$;
while 最优解尚未找到 **do**
  $X_{t+1} \leftarrow$ 按照某种条件转移概率$\Pr(X_{t+1} \mid X_0, \cdots, X_t)$产生一组新解;
  $t \leftarrow t+1$;
end while

个体: 一个解$x \in \mathcal{S}$。个体的适应值：$f(x)$
种群: ：一组解$X \subset \mathcal{S}$。种群的适应值：种群中最好个体的适应值$f(X) = \max\{f(x) \mid X \in X\}$.

- 不同的随机启发式搜索算法$\Rightarrow$ 不同的条件转移概率

随机序列

- 种群序列
$$X_0 \to X_1 \to X_2 \to \cdots$$
- 对应的适应值序列
$$f(X_0) \to f(X_1) \to f(X_2) \to \cdots$$
期望值$f_t = \mathbb{E}[f(X_t)]$
- 对应的误差序列
$$e(X_0) \to e(X_1) \to e(X_2) \to \cdots$$
误差$e(X_t) = |f(X_t) - f_{opt}|$，期望值$e_t = \mathbb{E}[e(X_t)]$
- 注： f_t，e_t依赖于初始解X_0，尽管我们没有显示表示出这一点。

几个理论问题

- 几个基本的理论问题

收敛性	算法能够找到最优解吗?
计算时间	算法需要多长时间可以找到最优解?
解的质量	给定有限的迭代次数，算法找到的解的质量如何?
收敛速度	算法每次迭代解的质量改善程度多大?

Definition

- 适应值序列$\{f(X_t); t = 0, 1, \cdots\}$是下鞅，如果对于任意$t$, $\mathbb{E}[f(X_{t+1}] \geq f(X_t)$
- 误差序列$\{e(X_t); t = 0, 1, \cdots\}$是上鞅，如果对于任意$t$, $\mathbb{E}[e(X_{t+1})] \leq e(X_t)$

Definition

种群序列$\{X_t; t=0,1,\cdots\}$是一个马尔可夫链，如果对于任意t，条件转移概率$\Pr(X_{t+1} \mid X^{(0)},\cdots,X_t) = \Pr(X_{t+1} \mid X_t)$

假设种群序列是一个齐次马尔可夫链，搜索空间的状态是有限的，最优解是吸收的，那么转移矩阵的标准形式是

$$\mathbf{P} = \begin{array}{c c} & \begin{array}{cc} S_{\text{opt}} & S_{\text{non}} \end{array} \\ \begin{array}{c} S_{\text{opt}} \\ S_{\text{non}} \end{array} & \begin{pmatrix} \mathbf{I} & \mathbf{O} \\ * & \mathbf{Q} \end{pmatrix} \end{array} \tag{1}$$

- $\mathbf{I}$：单位矩阵，表示最优解之间的转移概率
- $\mathbf{Q}$：表示非最优解之间的转移概率
- $\mathbf{O}$：零矩阵
- $*$：表示非最优解到最优解的转移概率

马尔可夫链转移矩阵

- X_t在所有解上的概率分布表示为向量$\mathbf{p}_t$
- p_t在非最优解上的部分记为向量$\mathbf{q}_t$
- 随机启发式搜索算法可以表示为一个矩阵迭代

$$\mathbf{p}_t^T = \mathbf{p}_{t-1}^T \mathbf{P} = \mathbf{p}_0^T \mathbf{P}^t. \tag{2}$$

$$\mathbf{q}_t^T = \mathbf{q}_{t-1}^T \mathbf{Q} = \mathbf{q}_0^T \mathbf{Q}^t. \tag{3}$$

收敛性：定义

Definition

适应值序列$\{f(X_0), f(X_1), \cdots\}$ 依概率1收敛于f_{opt}，如果$\Pr(\lim_{t\to+\infty}|f(X_t) - f_{opt}| = 0) = 1$

Definition

适应值序列$\{f(X_0), f(X_1), \cdots\}$ 平均收敛于f_{opt} 如果$\lim_{t\to+\infty}\mathbb{E}[e(X_t)] = 0$

- 对于误差序列$\{e(X_0), e(X_1), \cdots\}$ 可以类似定义

收敛性分析1：上鞅

Theorem ([1])

- 如果误差序列$\{e(X_0), e(X_1), \cdots\}$ 是上鞅，那么该序列总是平均收敛于一个常数，但是该常数不一定等于0
- 如果误差序列$\{e(X_0), e(X_1), \cdots\}$ 是上鞅，并且对于任意非最优解X_t，$\mathbf{E}[e(X_t) - e(X_{t+1})] > 0$, 那么该序列收敛于0

- 证明方法：应用Doob鞅的收敛性定理

Theorem

如果种群序列$\{X_t; t = 0, 1, \cdots\}$ 是一个马尔可夫链，搜索空间的状态是有限的，最优解是吸收的，

$$\mathbf{P} = \begin{pmatrix} \mathbf{I} & \mathbf{O} \\ * & \mathbf{Q} \end{pmatrix} \tag{4}$$

那么误差序列依概率*1*或平均收敛于0 的充分必要条件是矩阵$\mathbf{Q}$的谱半径$\rho(\mathbf{Q}) < 1$

证明方法：马尔可夫链的收敛性

计算时间：定义

- 两种不同但是相关的定义

  Hitting 时间：算法首次找到最优解时所花费的迭代次数

  Running 时间：算法首次找到最优解所花费的适应值评价次数

  关系 Running 时间$=$ Hitting 时间$\times$ 每代所需要的适应值评价次数

Definition

给定一个种群序列$\{X_0, X_1, \cdots\}$ 和最优解集S_{opt}，hitting 时间 是

$$T(X_0) = \sum_{t=0}^{+\infty} t\mathrm{P}(X_t \in S_{\mathrm{opt}} \mid X_{t-1} \in S_{\mathrm{non}}, \cdots, X_1 \in S_{\mathrm{non}}, X_0 \in S_{\mathrm{non}}). \quad (5)$$

- $T(X_0)$ 依赖于初始解X_0

Theorem ([2])

假设种群序列是一个齐次马尔可夫链，搜索空间的状态是有限的，最优解是吸收的，令向量$\mathbf{m}$表示从不同非最优解状态出发的*hitting*时间，那么

$$\mathbf{m} = (\mathbf{I} - \mathbf{Q})^{-1}\mathbf{1} \tag{6}$$

证明方法：吸收马尔可夫链的基本矩阵$(\mathbf{I} - \mathbf{Q})^{-1}$

- 直观上
$$\text{时间} = \frac{\text{距离}}{\text{速度}}$$
- 对于搜索空间里的一个点，构造一个该点离最优集的距离$d(X)$，满足两个条件
  1. $d(X) \geq 0$ 是一个非负函数
  2. 如果X是最优解，$d(X) = 0$
- 对于一个种群序列$\{X_0, X_1, \cdots\}$, 定义其速度(drift) $\Delta_t(X)$为
$$\Delta_t(X) := \mathbb{E}[d(X_t) - d(X_{t+1}) \mid X_0, \cdots X_{t-1}].$$

$$\text{Hitting 时间} = \frac{\text{距离}}{\text{速度}}$$
- 注意：距离$d(X)$ 根据问题而构造

Additive Drift

Definition

- 对于搜索空间中的每点X, pointwise drift:

$$\Delta_t(X) = \mathbb{E}[d(X_t) - d(X_{t+1}) \mid X_t = X]. \tag{7}$$

- 对于时刻t, average drift

$$\bar{\Delta}_t = \mathbb{E}[\mathbb{E}[d(X_t) - d(X_{t+1}) \mid X_t]]. \tag{8}$$

Theorem ([3])

- 对于任意t，如果*average drift* $\bar{\Delta}_t \leq c$, 那么*hitting*时间的期望$\mathbb{E}[T(X_0)] \geq \mathbb{E}[d(X_0)]/c$
- 对于任意t，如果*average drift* $\bar{\Delta}_t \geq c$, 那么*hitting* 时间的期望$\mathbb{E}[T(X_0)] \leq \mathbb{E}[d(X_0)]/c$

证明方法：鞅的停时定理的应用

Theorem ([4])

令$d_{\min} = \min\{d(X) \mid X \in \mathcal{S}_{\mathrm{non}}\}$。如果对于所有的非最优解$X$，所有的$t \geq 0$,

$$\frac{\mathbb{E}[d(X_{t+1})]}{d(X_t = X)} \leq (1 - \delta)$$

那么

$$\mathbb{E}[T] \leq \frac{1}{\delta}\left(1 + \mathbb{E}\left[\ln \frac{d(X_0)}{d_{\min}}\right]\right). \tag{9}$$

其他形式的Drift Analysis

- 最近的综述：Lengler，Drift analysis. arXiv:1712.00964, 2018 [5]
  - Multiplicative drift
  - Variable Drift
  $$\Delta_t(X) \geq (\leq) h(X),$$
  - Negative Drift:
  $$\Delta_t(X) < 0$$
  算法找不到最优解？Hitting 时间为无穷大？
  - Population-Based Drift
  - General Drift with Tail Bounds

解的质量

- 除了计算时间，实际应用中人们更关心的是在有限的迭代次数内算法所找到解的质量
  - 适应值f_t
  - 误差值e_t
- 有两种研究解的质量的思路：

  Fixed Budget Performance 在预先规定的迭代次数t内，估计算法所产生解的适应值f_t[6]

  近似误差估计 研究误差e_t如何随着t变化[?]
- 两种方法本质上等价，因为fixed budget performance = 近似误差估计
  - 极大化问题：$f_t = f_{opt} - e_t$
  - 极小化问题：$f_t = e_t - f_{opt}$
- 但是fixed budget performance 比近似误差估计更复杂
  - f_t 随函数$f(x)$ 变化

- 假如随机启发式搜索算法是一个有限状态上的齐次马尔可夫链

$$\mathbf{q}_t^T = \mathbf{q}_0^T \mathbf{Q}^t \tag{10}$$

Theorem

令向量f表示在不同非最优解状态上的适应值，那么在时刻t

$$f_t = \mathbf{q}_0^T \mathbf{Q}^t \mathbf{f} \tag{11}$$

解的质量分析2：误差估计/上鞅方法

Theorem

对于误差序列$\{e_t; t = 0, 1, \dots\}$,

1. 如果存在一个$\delta > 0$，使得对于任意的t，$e_{t+1}/e_t \leq \delta$，那么误差的上界是$e_t \leq e_0\delta^t$
2. 如果存在一个$\delta > 0$，使得对于任意的t，$e_{t+1}/e_t \geq \delta$, 那么误差的上界是$e_t \geq e_0\delta^t$。

收敛速度：定义

Definition

收敛速度 定义成误差e_t每次迭代减少的比值：

$$\frac{e_t}{e_{t-1}}. \tag{12}$$

- 对于确定性迭代算法，计算e_t/e_{t-1}没有问题
- 对于随机启发式算法，计算e_t/e_{t-1}不太稳定，需要大量的采样

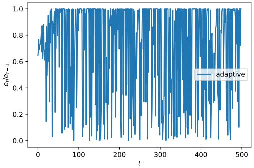

Figure: 计算实验显示的比值e_t/e_{t-1}

- 收敛速度e_t/e_{t-1}不适合随机启发式搜索算法

平均收敛速度

Definition

e_t/e_{t-1}的t代几何平均值

$$R_t := 1 - \left(\frac{e_1}{e_0}\cdots\frac{e_t}{e_{t-1}}\right)^{1/t} = 1 - \left(\frac{e_t}{e_0}\right)^{1/t}. \tag{13}$$

当$e_k = 0$时，$R_t = 1$.

正则化 $1-(e_t/e_0)^{1/t}$ 将$(e_t/e_0)^{1/t}$正则化在区间$(-\infty, 1]$.

- R_t值越大，算法收敛越快

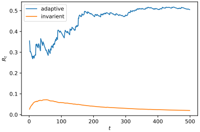

Figure: 计算实验显示的两个算法的R_t值。蓝色线对应的算法比橙色线对应的算法收敛速度快

收敛速度分析1：误差估计/上鞅

Theorem

对于误差序列$\{e_t; t = 0, 1, \ldots\}$,

1. 如果存在一个$\delta > 0$ 使得对于任意的t，$e_{t+1}/e_t \leq \delta$，那么平均收敛速度$1 - (e_t/e_0)^{1/t} \geq \delta$
2. 如果存在一个$\delta > 0$ 使得对于任意的t，$e_{t+1}/e_t \geq \delta$，那么平均收敛速度$1 - (e_t/e_0)^{1/} \leq \delta$

收敛速度分析2：马尔可夫链

- 假如随机启发式搜索算法是一个马尔可夫链

$$\mathbf{q}_t^T = \mathbf{q}_0^T \mathbf{Q}^t. \tag{14}$$

Theorem ([7])

令$\mathbf{Q}$ 是一个随机启发式算法的转移矩阵，$\rho(\mathbf{Q})$为矩阵$\mathbf{Q}$最大特征值，

1. R_t的下界为

$$R_t \geq 1 - \| \mathbf{Q}^t \|^{1/t}. \tag{15}$$

2. 如果初始种群随机选取，那么R_t的极限值是

$$\lim_{t \to +\infty} R_t = 1 - \rho(\mathbf{Q}). \tag{16}$$

证明方法：Gelfand's 谱半径公式和Perron-Frobenius 定理在非负矩阵上的推广

小结

- 研究方法的比较
    - 马尔可夫链方法需要知道转移矩阵。对于复杂的问题，转移矩阵很难得到
    - 上/下鞅方法不需要精确的转移概率，但是不一定能够得到准确的估计
- 收敛性研究相对简单，可以直接用随机序列的收敛性理论
- 计算时间是理论研究的主流。目前的主要工具是drift analysis,但是还是不能很好分析基于种群的算法。因此和实践有点脱节
- 解的质量在实践中广泛用于评价随机启发式算法的性能。理论上不太重视
- 收敛速度过去有一些研究，主要研究算法的局部收敛速度。但是和实践有点脱节，因为随机启发式搜索算法的主要目的是整体优化问题

理论分析中的两个困难

比较算法 受自然的启发，从模拟蚂蚁(Ant Colony Optimisation)到模拟文化基因（Memetic Algorithm），人们提出了许许多多的随机启发式搜索算法，犹如算法动物园。
是否可以类似生物分类学，从理论上比较这些算法的异同，进行算法分类，评价不同算法的好坏?

设计算法 理论分析不仅仅是用来理解算法，更重要的是指导算法的设计。
现有的理论很少能够指导算法的设计。如何发展适当的理论来指导算法的设计?

欢迎提问

参考文献

Günter Rudolph.
Convergence rates of evolutionary algorithms for a class of convex objective functions.
Control and Cybernetics, 26:375–390, 1997.

J. He and X. Yao.
Towards an analytic framework for analysing the computation time of evolutionary algorithms.
Artificial Intelligence, 145(1-2):59–97, 2003.

J. He and X. Yao.
Average drift analysis and population scalability.
IEEE Transactions on Evolutionary Computation, 21(3):426–439, 2017.

Benjamin Doerr, Daniel Johannsen, and Carola Winzen.
Multiplicative drift analysis.
Algorithmica, 64(4):673–697, 2012.

Johannes Lengler.
Drift analysis.
arXiv:1712.00964, 2018.

Thomas Jansen and Christine Zarges.
Performance analysis of randomised search heuristics operating with a fixed budget.
Theoretical Computer Science, 545:39–58, 2014.

J. He and G. Lin.
Average convergence rate of evolutionary algorithms.
IEEE Transactions on Evolutionary Computation, 20(2):316–321, 2016.