计算机算法设计与分析

083500M01001H

Chap 9&10&11 课程作业

2022 年 12 月 1 号

Professor: 刘玉贵

学生: 周胤昌

学号: 202228018670052

学院: 网络空间安全学院

所属专业: 网络空间安全

方向: 安全协议理论与技术

Problem 1

判断正误:

- Las Vegas 算法不会得到不正确的解. ( )
- Monte Carlo 算法不会得到不正确的解. ( )
- Las Vegas 算法总能求得一个解. ( )
- Monte Carlo 算法总能求得一个解. ( )

Solution:

- 正确, 拉斯维加斯算法不会得到不正确的解. 一旦用拉斯维加斯算法找到一个解, 这个解就一定是正确解. 但有时用拉斯维加斯算法找不到解.
- 错误, Monte Carlo 算法每次都能得到问题的解, 但不保证所得解的正确性. 请注意, 可以在 Monte Carlo 算法给出的解上加一个验证算法, 如果正确就得到解, 如果错误就不能生成问题的解, 这样 Monte Carlo 算法便转化为了 Las Vegas 算法.
- 错误, Las Vegas 算法并不能保证每次都能得到一个解, 但是如果一旦某一次得到解, 那么就一定是正确的.
- 正确, Monte Carlo 算法每次运行都能给出一个解, 但正确性就不能保证了.

Problem 2

判断正误:

- 一般情况下, 无法有效判定 Las Vegas 算法所得解是否正确. ( )
- 一般情况下, 无法有效判定 Monte Carlo 算法所得解是否正确. ( )
- 虽然在某些步骤引入随机选择, 但 Sherwood 算法总能求得问题的一个解, 且所求得的解总是正确的. ( )
- 虽然在某些步骤引入随机选择, 但 Sherwood 算法总能求得问题的一个解, 但一般情况下, 无法有效判定所求得的解是否正确. ( )

Solution:

- 错误, Las Vegas 算法并不能保证每次都能得到解, 但是如果一旦某一次得到解, 那么就一定是正确的.
- 错误, 虽然 Monte Carlo 算法每次运行都能给出一个解, 可能是错的也可能是对的, 但是可以通过检验解来有效判定其正确性. 判定解的正确性跟算法本身没有多大关系, 只要代进去验证即可. 特殊点在于, 只要 Las Vegas 算法求得解了, 那么就一定是正确的, 就不用再浪费时间来判定了; 但是对于 Monte Carlo 算法的所得解, 必须要进行正确性检验.
- 正确, Sherwood 算法总能求得问题的一个解, 且所求得的解总是正确的.
- 错误.

Problem 3

判断正误:

- 旅行商问题存在多项式时间近似方案. (　)
- 0/1 背包问题存在多项式时间近似方案. (　)
- 0/1 背包问题的贪心算法 (单位价值高优先装入) 是绝对近似算法. (　)
- 多机调度问题的贪心近似算法 (按输入顺序将作业分配给当前最小负载机器) 是 ϵ-近似算法. (　)

Solution:

- 错误. 根据教材可知, 旅行商问题不存在多项式时间近似算法, 除非 $\mathcal{P} = \mathcal{NP}$. 如果存在的话, 那么就可以证得 $\mathcal{P} = \mathcal{NP}$, 即可以拿图灵奖了.
- 正确, **PTAS 算法**就是 0/1 背包问题的多项式时间近似方案.
- 错误, 0/1 背包问题的贪心算法**不是**绝对近似算法.
- 正确, 多机调度问题的贪心近似算法有 GMPS 和 DGMPS 分别是 2-近似和 3/2-近似算法.

Problem 4

设 Las Vegas 算法获得解的概率为 $p(x) \geq \delta, 0 < \delta < 1$, 则调用 k 次算法后, 获得解的概率为:_____.
Solution: 不妨求一下调用 k 次算法后, 求解失败 (即 k 次调用都求解失败) 的概率:

$$P\left(\text{失败}\right) = (1 - p\left(x\right))^k \leq (1 - \delta)^k \Rightarrow P\left(\text{成功}\right) = 1 - P\left(\text{失败}\right) \geq 1 - (1 - \delta)^k$$

即获得解的概率至少为 $1 - (1 - \delta)^k \to 1(\text{当}k \to \infty)$.

Problem 5

对于判定问题 Π 的 Monte Carlo 算法, 当返回 false(true) 时解总是正确的, 但当返回 true(false) 时解可能有错误, 该算法是 ______.

(A) .偏真的*Monte Carlo*算法　　(B) .偏假的*Monte Carlo*算法

(C) .一致的*Monte Carlo*算法　　(D) .不一致的*Monte Carlo*算法

Solution: 答案选 B, 只要将偏真的 Monte Carlo 算法的定义中的 true/false 互换即可得到偏真的 Monte Carlo 算法的定义.

Problem 6

写出禁忌搜索算法的主要步骤.

Solution: 禁忌搜索算法的主要步骤如下算法1中所示:

Algorithm 1 禁忌搜索算法步骤

选定一个初始可行解 $\boldsymbol{x}^{\mathrm{cb}}$ 并初始化禁忌表 $H \leftarrow \{\}$;
while 不满足停止规则 **do**
    在 $\boldsymbol{x}^{\mathrm{cb}}$ 的邻域中选出满足禁忌要求的候选集 Can-$N\left(\boldsymbol{x}^{\mathrm{cb}}\right)$;
    从该候选集中选出一个评价值最佳的解 $\boldsymbol{x}^{\mathrm{lb}}$;
    令 $\boldsymbol{x}^{\mathrm{cb}} \leftarrow \boldsymbol{x}^{\mathrm{lb}}$ 并更新记录 H;
end while

Problem 7

禁忌对象特赦可以基于影响力规则: 即特赦影响力大的禁忌对象. 影响力大什么含义? 举例说明该规则的好处.

Solution: 影响力大意味着有些对象变化对目标值影响很大. 如 0/1 背包问题, 当包中无法装入新物品时, 特赦体积大的分量来避开局部最优解.

Problem 8

判断正误:

- 禁忌搜索中, 禁忌某些对象是为了避免领域中的不可行解. ( )
- 禁忌长度越大越好. ( )
- 禁忌长度越小越好. ( )

Solution:

- 错误, 选取禁忌对象是为了引起解的变化, 根本目的在于避开邻域内的局部最优解而不是不可行解.
- 错误, 禁忌长度短了则可能陷入局部最优解.
- 错误, 禁忌长度长了则导致计算时间长.

Problem 9

写出模拟退火算法的主要步骤.

Solution: 模拟退火算法的主要步骤如下算法2中所示:

Algorithm 2 模拟退火算法步骤

任选初始解 x_0 并初始化 $x_i \leftarrow x_0, k \leftarrow 0, t_0 \leftarrow t_{\max}$(初始温度);
while $k \le k_{\max}$ && $t_k \ge T_f$ **do**
    从邻域 $N(x_i)$ 中随机选择 x_j, 即 $x_j \leftarrow_R N(x_i)$;
    计算 $\Delta f_{ij} = f(x_j) - f(x_i)$;
    if $\Delta f_{ij} \le 0 \,||\, \exp\left(-\Delta f_{ij}/t_k\right) > \text{RANDOM}(0,1)$ **then**
        $x_i \leftarrow x_j$;
    end if
    $t_{k+1} \leftarrow d(t_k)$;
    $k \leftarrow k + 1$;
end while

Problem 10

写出遗传算法的主要步骤.

Solution: 遗传算法的主要步骤如下算法3中所示:

Algorithm 3 遗传算法步骤

选择问题的一个编码并初始化种群 (N 个染色体)pop (1) $:= \left\{\text{pop}_j\,(1)\,|\, j = 1, 2, \cdots, N\right\}, t := 1$;
对种群 pop(1) 的每个染色体 $\text{pop}_i(1)$ 计算其适应性函数 $f_i = \text{fitness}(\text{pop}_i(1))$;
while 停止规则不满足 **do**
    计算得出概率分布 $p_i = \dfrac{f_i}{\sum_{1 \le j \le N} f_j}$ $(*)$;
    根据概率分布 $(*)$ 从 pop(t) 中随机选取 N 个染色体并形成种群

$$\text{newpop}(t+1) := \left\{\text{pop}_j\,(t)\,|\, j = 1, 2, \cdots, N\right\}$$

    通过交叉 (交叉概率为 P_c) 得到一个有 N 个染色体的种群 crosspop$(t+1)$;
    以较小的概率 p, 使得染色体的基因发生变异, 形成种群 mutpop$(t+1)$;
    $t := t + 1$, 诞生新种群 pop(t) := mutpop(t);
    对种群 pop(t) 的每个染色体 $\text{pop}_i(t)$ 计算其适应性函数 $f_i = \text{fitness}(\text{pop}_i(t))$;
end while

Problem 11

为避免陷入局部最优 (小), 模拟退火算法以概率 $\exp\left(-\Delta f_{ij}/t_k\right)$ 接受一个退步 (比当前最优解差) 的解, 以跳出局部最优. 试说明参数 $t_k, \Delta f_{ij}$ 对是否接受退步解的影响.

Solution: 很明显, 当 t_k 较大时, 接受退步解的概率越大; 当 Δf_{ij} 较大时, 接受退步解的概率越小.

Problem 12

下面属于模拟退火算法实现的关键技术问题的有 ______.

(A) .初始温度 (B) .温度下降控制 (C) .邻域定义 (D) .目标函数

Solution: 模拟退火算法实现的关键技术问题有**邻域的定义 (构造)、起始温度的选择、温度下降方法、每一温度的迭代长度以及算法终止规则**. 因此选择 (A), (B), (C).

Problem 13

用遗传算法解某些问题, fitnees $= f(x)$ 可能导致适应函数难以区分这些染色体. 请给出一种解决办法.

Solution: 采用线性加速适应函数: $\text{fitness}\,(x) = \alpha f(x) + \beta f(x)$

Problem 14

用非常规编码染色体实现的遗传算法, 如 TSP 问题使用 $1, 2, \cdots, n$ 的排列编码, 简单交配会产生什么问题? 如何解决?

Solution: 后代可能会出现非可行解, 因此需要通过罚值和交叉新规则来解决.

Problem 15

下面属于遗传算法实现的关键技术问题的有 ______.

(A) .解的编码　　(B) .初始种群的选择　　(C) .邻域定义　　(D) .适应函数

Solution: 遗传算法实现的关键技术问题有**解的编码、适应函数、初始种群的选取、交叉规则以及终止规则**. 因此选择 (A), (B), (D).

Problem 16

设旅行商问题的解表示为 $D = F = \left\{S|S = (i_1, i_2, \cdots, i_n), i_1, i_2, \cdots, i_n\text{是}1, 2, \cdots, n\text{的一个排列}\right\}$, 邻域定义为 2-OPT(即 S 中的两个元素对换), 求 $S = (3, 1, 2, 4)$ 的邻域 $N(S)$.

Solution: 将 S 中的两个元素对换即可得到 $N(S)$:

$$N(S) = \{(1,3,2,4), (2,1,3,4), (4,1,2,3), (3,2,1,4), (3,4,2,1), (3,1,4,2)\}$$

Problem 17

0/1 背包问题的解记作 $X = (x_1, x_2, \cdots, x_n), x_i \in \{0, 1\}, i = 1, 2, \cdots, n$. 邻域定义为

$$N(X) = \left\{ Y \,\middle|\, \sum_{i=1}^{n} |y_i - x_i| \leq 1 \right\}, X = (1,1,0,0,1)$$

求邻域 $N(X)$.

Solution: 每次只允许一个分量变化即可求出邻域 $N(X)$:

$$N(X) = \{(0,1,0,0,1), (1,0,0,0,1), (1,1,1,0,1), (1,1,0,1,1), (1,1,0,0,0)\}$$

Problem 18

顶点覆盖问题: 任给一个图 $G = \langle V, E \rangle$, 求 G 的顶点数最少的顶点覆盖. 复习顶点覆盖问题的近似算法及其证明.

Solution: MVC算法如下所示:

Algorithm 4 算法 **MVC**(G)

Input: 图 $G = \langle V, E \rangle$
Output: 最小顶点覆盖 V'

$V' \leftarrow \emptyset, e_1 \leftarrow E$;
while $e_1 \neq \emptyset$ **do**
  从 e_1 中任选一条边 (u, v);
  $V' \leftarrow V' \cup \{u, v\}$;
  从 e_1 中删去与 u 和 v 相关联的所有边;
end while
return V';
end {MVC};

显然算法 MVC 的时间复杂度为 $O(m), m = |E|$. 记 $|V'| = 2k$, V' 中的顶点是 k 条边的端点, 这 k 条边互不关联. 为了覆盖这 k 条边则需要 k 个顶点, 从而 $\mathrm{OPT}(I) \geq k$. 于是有

$$\frac{\mathrm{MVC}(I)}{\mathrm{OPT}(I)} \leq \frac{2k}{k} = 2$$

故 MVC 是最小顶点覆盖问题的 2-近似算法,□.

Problem 19

1.下面说法，正确的是：____________.

(1)P 类问题是存在多项式时间算法的问题。

(2)NP 类问题是不存在多项式时间算法的问题。

(3)P 类问题一定也是 NP 类问题。

(4)NP 类问题比 P 类问题难解。

2.下面说法，正确的是：__________.

(1)$P \subset NP$ (2)$P \subseteq NP$ (3)$P=NP$ (4)$P \neq NP$

3. 下面说法，正确的是：_________.

(1)NP-难问题是 NP 中最难的问题

(2)NP-完全问题是 NP 中最难的问题

(3)NP-难不比任何 NP 问题容易

(4)NP-完全问题也是 NP-难问题。

参考答案

1.(1)(3) 2.(2) 3.(2)(3)(4)