Ciel and password cracking 解题报告

绍兴市第一中学 任之洲

1 试题来源

Codechef OCT 12 CIELHACK

2 试题大意

Ciel有一个保险箱，这个保险箱有K个密码，然而她只记得第i个密码是一个不超过N_i的正整数，她打算进行暴力破解。

由于Ciel的电脑太慢了，所以她打算借用市里的电脑中心。市里共有C个电脑中心，Ciel打算借用其中恰好K个来分别破解K个密码。

这C个电脑中心和Ciel的餐馆在同一条街上，以Ciel的餐馆为坐标轴零点，第i个电脑中心的坐标为X_i，坐标的正负表示所在的方向。

Ciel将会选择K个不同的电脑中心去破解她的K个密码，她会以任意顺序去遍历它们，从她的餐馆出发，最后回到她的餐馆。

Ciel的行走速度为V，即从坐标A走到坐标B需要消耗$\frac{|A-B|}{V}$单位时间。

在第j个电脑中心，Ciel可以使用最多P_j台电脑，每台电脑需要消耗T_j单位时间去枚举验证一种密码。这些电脑被一种奇怪的网络连接着，于是Ciel的计划将以下面的步骤进行：

- 选择一部分电脑（不超过P_j台），这一步不消耗时间。
- 在一台电脑中安装破解程序，这一步不消耗时间。
- 把破解程序通过网络传输到其他被选择的电脑中，这一步的消耗时间在后文会详述。

- 在所有被传输的电脑上同时运行破解程序，这一步的消耗时间在后文会详述。
- 一旦其中一台电脑成功破解了密码，那么这里的工作将会全部停止，然后她将会奔赴下一个电脑中心。

接下来将描述这个问题的更多细节。

2.1 破解程序细节

破解程序每次会随机验证一个没有被验证过的密码，对于第j个电脑中心的电脑，验证一次将消耗T_j单位的时间。

假如某一次验证的密码成功了，程序将通知Ciel并终止。

电脑和电脑之间并没有共享信息，即可能会验证在其他电脑上已经验证过的密码。

2.2 网络连接细节

通过网络连接，已经装有破解程序的电脑，可以将破解程序备份传输到没有安装的电脑上。

连接所需要的时间是随机的，这个随机值呈指数分布，对于第j个电脑中心的电脑，均值为S_j。

只有已经装有破解程序的电脑才可以与其他电脑连接，一旦连接完成，备份传输可视为不消耗时间。

在每个时刻，一台电脑只能尝试向一台电脑连接，但一台电脑可以同时接受多台电脑的连接，一旦其中一个连接成功确立，那么其他连接将会中断停止，那些电脑将会去尝试与其他电脑连接。

2.3 连接方案细节

可以为每台电脑预先制定一个连接列表，电脑每次会选择连接列表上的第一个没有装有破解程序的电脑进行连接。一旦连接确立或中断，它将立刻选择下一个连接目标。

在所有被选择的电脑都装有破解程序后，所有程序将同时开始运行。

Ciel想要通过分配电脑中心，选择电脑，调整连接方案，来最小化期望所需时间。

答案的相对误差要求小于10^{-6}。

数据范围：$1 \le C \le 1000$，$1 \le K \le min(5, C)$，$1 \le V, S_j, T_j \le 10^{20}$，$1 \le N_i, P_i \le 10^{18}$，$|X_i| \le 10^{20}$

3 算法介绍

3.1 问题转化

设$Time[i][j]$为第i个密码在第j个电脑中心破解的最小期望所需时间，假设这个已经完成了计算，那么只需要一个简单的状压DP就可以解决选择电脑中心的问题，所以主要问题是计算$Time[i][j]$。

计算$Time[i][j]$涉及到密码范围N_i，电脑数量P_j，连接时间S_j，以及运算速度T_j，设$Time[i][j] = F(N_i, P_j, S_j, T_j)$，现在需要快速计算函数$F(N, P, S, T)$。

设$A(k)$为假设$S = 1$的前提下，连接k台电脑的最小期望时间，$B(N, k)$为假设$T = 1$的前提下，用k台电脑破解范围为N的密码的期望时间。那么

$$F(N, P, S, T) = \min_{k=1}^{P} (S * A(k) + T * B(N, k))$$

现在问题转化为计算$A(k)$和$B(N, k)$，再求得最小值来计算$F(N, P, S, T)$。

3.2 计算A(k)

由于连接所需的时间是指数分布的，所以具有无记忆性，即设函数返回值为R，对于两个数$a, b \ge 0$，$P(R > a + b \mid R > b) = P(R > a)$。

根据无记忆性，连接列表中的顺序并不会影响连接完成所需的期望时间，所以可以假定所有电脑的连接列表都为$1 \sim k$。

根据指数分布函数的性质，m个参数为$\lambda_1, \lambda_2, ..., \lambda_m$的指数分布函数取$min$后是一个参数为$\lambda = \sum_{i=1}^{m} \lambda_i$的指数分布函数。所以$m$台电脑同时连接一台电脑的期

望连接时间为原先的$\frac{1}{m}$倍，这个结论通过无记忆性也可以直观地得到。

由此可以得到

$$A(k) = \sum_{i=1}^{k-1} \frac{1}{i}$$

可以预处理$k \le 10^6$的情况，对于$k > 10^6$的情况

$$\sum_{i=1}^{k-1} \frac{1}{i} \approx \ln(k-1) + \gamma$$

其中γ为欧拉常数。

3.3 计算B(N,k)

考虑需要进行第i次验证的概率为$\left(\frac{N-i+1}{N}\right)^k$，根据期望的线性贡献易得

$$B(N,k) = \sum_{i=1}^{N} \left(\frac{i}{N}\right)^k$$

考虑设定阈值，设$t = \frac{k+1}{N}$，当$t < 2$时，使用欧拉-麦克劳林公式。

$$\sum_{i=m+1}^{n} f(i) - \int_m^n f(x)dx \approx \sum_{i=1} \frac{B_i}{i!} \left(f^{(i-1)}(n) - f^{(i-1)}(m)\right)$$

其中B_i为第i项伯努利数，由于伯努利数除第1项外的奇数项都为0，所以这个式子可以写为

$$\sum_{i=m+1}^{n} f(i) - \int_m^n f(x)dx \approx B_1(f(n) - f(m)) + \sum_{i=1} \frac{B_{2i}}{(2i)!} \left(f^{(2i-1)}(n) - f^{(2i-1)}(m)\right)$$

考虑对于函数$f(x) = \left(\frac{x}{N}\right)^k$套用这个公式，并取阈值$p = min(10, \frac{k}{2})$，计算

$$B(N,k) \approx \frac{N}{k+1} + \frac{1}{2} + \sum_{i=1}^{p} \frac{B_{2i}}{(2i)!} \prod_{j=0}^{2i-2} \frac{k-j}{N}$$

这样计算的相对误差不超过$3\left(\frac{t}{2\pi}\right)^{2p}$，所以可以取$p = 10$。

当$t \ge 2$时，大多数项都比较小，取$p = min(n-1, 10)$。

$$B(N,k) \approx \sum_{i=0}^{p} \left(\frac{N-i}{N}\right)^k$$

这样计算的相对误差不超过$\frac{e^{-pt}}{t}$，所以可以取$p = 10$。

3.4 计算F(N,P,S,T)

设$G(k) = S * A(k) + T * B(N,k)$，可以发现这是一个单峰函数，考虑计算

$$G(k+1) - G(k) = \frac{S}{k} - T(B(N,k) - B(N,k+1))$$

通过二分并判断差值可以快速找到极值点。

3.5 误差处理

虽然只要求相对误差小于10^{-6}，但是还是存在精度误差的问题。

考虑在计算$B(N,k)$的过程中，$t \geq 2$时，需要计算$\left(\frac{N-p}{N}\right)^k$。由于$k$的取值范围会达到$10^{18}$，通常情况下可以计算$\exp(k \ln \frac{N-p}{N})$来完成，但当$\frac{p}{N}$非常小时会有较大误差。

考虑将$ln(1-x)$泰勒展开

$$ln(1-x) \approx -x - \frac{x^2}{2} - \frac{x^3}{3} - ...$$

可以在$\frac{p}{N} < 10^{-9}$时，计算$\exp\left(-k(\frac{p}{N} + \frac{(\frac{p}{N})^2}{2})\right)$来保证精度。

在计算$F(N,P,S,T)$过程中，二分时需要计算$G(k+1) - G(k)$，也就是要计算$B(N,k) - B(N,k+1)$，这时分开计算$B(N,k)$和$B(N,k+1)$是不行的。

同样设阈值$t = \frac{k+1}{N}$，当$t < 2$时

$$B(N,k) - B(N,k+1) \approx \frac{N}{(k+1)(k+2)} + \sum_{i=1}^{p} \frac{B_{2i}}{(2i)!} \left(\prod_{j=0}^{2i-2} \frac{k-j}{N} - \prod_{j=0}^{2i-2} \frac{k+1-j}{N} \right)$$

其中，对于$i > 1$的情况

$$\begin{aligned}
\prod_{j=0}^{2i-2} \frac{k-j}{N} - \prod_{j=0}^{2i-2} \frac{k+1-j}{N} &= \prod_{j=0}^{2i-2} \frac{k-j}{N} - \prod_{j=-1}^{2i-3} \frac{k-j}{N} \\
&= \frac{(k-2i+2) - (k+1)}{N} \prod_{j=0}^{2i-3} \frac{k-j}{N} \\
&= -\frac{2i-1}{N} \prod_{j=0}^{2i-3} \frac{k-j}{N}
\end{aligned}$$

当$t \geq 2$时

$$B(N,k) - B(N,k+1) \approx \sum_{i=1}^{p} \frac{i}{N} \left(\frac{N-i}{N}\right)^k$$

这样计算即可保证在二分时所需的精度。

时间复杂度$O(CK \log P + 2^K CK)$，空间复杂度$O(CK + 2^K C)$。