《魔法波》解题报告

南京外国语学校 乔明达

1 题目大意

有一个$N \times N$的方格，其中某些方格是障碍物，其余每个方格中有一盏灯。初始时一部分灯是亮的，另一部分是灭的。一次“操作”定义为：选择一个有灯的方格，从这个方格出发，分别朝上下左右四个方向拓展，一旦遇到障碍物或者超出边界就停止。对于向四个方向拓展时经过的方格以及一开始选择的方格，其中亮的灯会熄灭，灭的灯会变亮。规定从每个方格出发，最多只能进行一次操作。求一种操作的方案，使得所有灯都变亮，数据保证这种方案存在。

2 数据规模和约定

10%的数据满足$N \leq 5$；

30%的数据满足$N \leq 30$；

100%的数据满足$N \leq 800$，且方格中障碍物不超过200个。

3 算法一

所有可能的操作不会超过N^2种，因此我们可以枚举每种操作是否进行，再暴力模拟操作过程判断正确性。

算法一时间复杂度为$\Theta(2^{N^2} \times N^3)$，期望得分为10分。

4 算法二

不难发现，对一盏灯操作，影响到的灯是固定的。从另一个角度来看，一盏灯最终的状态取决于其初始状态，以及对某些灯进行的操作。开关灯实质上是一种异或运算，因此我们可以联想到异或方程组。

用S_i表示第i盏灯的初始状态（0表示灭，1表示亮），用X_i表示是否对第i盏灯进行操作（0表示不操作，1表示操作）。那么以下方程成立（这里的求和符号解释为求异或和）：

S_i xor $\sum X_j = 1$，其中对第j盏灯操作会改变第i盏灯的状态

未知数的数目和方程数相等，都等于灯的数目，在最坏情况下为$\Theta(N^2)$。使用高斯消元解异或方程组，则算法二的时间复杂度为$\Theta(N^6)$。若高斯消元实现得较好，则期望得分为30分。

5 算法三

上述的算法二显然无法应对全部数据，因为当$N = 800$时，未知数个数将达到640000个，这是我们无法处理的。

注意到有一个条件我们仍未使用：方格中障碍物数目（记为M）不超过200。200个障碍物与640000个方格相比，障碍物在整个方格中是非常稀疏的，我们如何使用这个条件呢？

联想到某些图论题中将方格图建成二分图的处理方法，我们定义“水平块”和“垂直块”两个概念。如果某一行上若干个（至少1个）连续的方格都不包含障碍物（也就是说只包含灯），并且假如向左或向右再拓展一个方格，就会遇到障碍或者超出边界，那么将其称为一个水平块。类似地，我们可以定义垂直块。

由于我们定义的水平块和垂直块具有连续性和极大性，我们可以推出很多利于解题的性质。

首先，每盏灯恰好属于一个水平块和一个垂直块。

其次，考虑水平块的个数。每个水平块的左侧，要么是方格图的左边界，要么是某一个障碍物。并且不同水平块不可能对应相同的边界或者相同的障碍物。因此水平块的个数不会超过$N + M \le 800 + 200 = 1000$。类似地，垂直块的数目也不超过1000。

对于某一个特定的水平块而言，如果对其中的一个方格操作，那么块中所有灯的状态都会改变，并且水平方向上仅有这些灯的状态会改变。类似地，垂直块也能得到同样的结论。

需要注意的是，如果对某个格子进行了操作，那么以上讨论中，这个操作被计算了两遍（水平块和垂直块），恰好抵消掉了。因此如果X_i为1，还需要额

外地改变一次第i盏灯的状态。

我们将水平块和垂直块统称为“块”，进行统一的标号。用Y_i表示第i个块中所有X的异或和，我们用L_i和R_i分别表示第i盏灯所属的水平块和垂直块的编号。那么我们可以得到以下方程组（这里的求和符号解释为求异或和）：

$Y_i = \sum X_j$，其中第j盏灯属于第i个块

$S_i \text{ xor } Y_{L_i} \text{ xor } Y_{R_i} \text{ xor } X_i = 1$

以上方程组包含的未知数的数目和方程数相等，均为灯的数目加块的数目，看上去反而比算法二差不少，但是我们可以手工地进行一部分消元工作。

观察第二种方程，可以变形为：

$X_i = S_i \text{ xor } Y_{L_i} \text{ xor } Y_{R_i} \text{ xor } 1$

将所有变形后的方程代入所有第一类方程，这显然不会影响方程的解。

代入之后，未知数的数目和方程数都降为块的数目。使用高斯消元解出所有Y_i之后，代入上式就能得到所有的X_i，进而得到本题的解。以上算法三的时间复杂度为$\Theta((N+M)^3)$，其中M表示障碍物的数目。尽管未知数的数目在最大测试数据中将达到$2(M+N)=2000$，但是我们可以使用压位或者bitset优化高斯消元，这样时间复杂度中隐含的常数因子是很小的。算法三在实际测试中可以极快地计算出结果，期望得分100分。