# NOI 2025 Day1 A. 机器人

VegChicken

2025 年 7 月 14 日

## 问题（机器人）

- 给定一个 $n$ 个点 $m$ 条边的有向图，每个点的出边已经按照某种方式排序。
- 有一个全局变量 $z$，初始为 $1$，上限为 $k$，每次默认从当前编号为 $z$ 的出边走下去，如果没有对应出边则非法。
- 你可以随时花费 $v_z$ 的代价使 $z$ 加一，也可以花费 $w_z$ 的代价使 $z$ 减一。
- 请对于每一个点，计算从起点开始到达它经过的边权与修改 $z$ 花费的代价之和的最小值。

数据范围：$n, m \leq 3 \times 10^5$，$k \leq 2.5 \times 10^5$，所有的权值非负。

- 本题作为整个比赛的第一题，理论上有很多做法。
- 出题人这里提供两种思路，第一种是基于暴力优化的，第二种相对第一种更优美，但是更难一步想到。

$n, m \leq 10^3$

$n, m \leq 10^3$

- 我们把当前的点以及 $z$ 的值看作一个状态，容易发现 $z$ 不可能超过 $m$。

$n, m \le 10^3$

- 我们把当前的点以及 $z$ 的值看作一个状态，容易发现 $z$ 不可能超过 $m$。
- 将状态之间的转移看作连边，我们事实上得到了一个最短路模型。

$n, m \le 10^3$

- 我们把当前的点以及 $z$ 的值看作一个状态，容易发现 $z$ 不可能超过 $m$。
- 将状态之间的转移看作连边，我们事实上得到了一个最短路模型。
- 跑 Dijistra 算法即可，时间复杂度 $O(nm \log nm)$，期望得分：$20$。

$n \le 5 \times 10^4, \ k \le 10^2$

$n \le 5 \times 10^4$，$k \le 10^2$

- 同理显然 $z$ 的大小不可能超过 $k$，所以当 $k$ 很小的时候同样可以这么做。

$n \le 5 \times 10^4$，$k \le 10^2$

- 同理显然 $z$ 的大小不可能超过 $k$，所以当 $k$ 很小的时候同样可以这么做。
- 时间复杂度 $O(nk \log nk)$，期望得分：$40$。

特殊性质 A： $v_z = w_z = 0$

特殊性质 A：$v_z = w_z = 0$

- 此时仅有边权有用，直接使用最短路模板即可。

特殊性质 A：$v_z = w_z = 0$

- 此时仅有边权有用，直接使用最短路模板即可。
- 时间复杂度 $O(m\log m)$，期望得分：$20$。

特殊性质 C：出度大于等于 $10$ 的点至多只有 $10$ 个

特殊性质 C：出度大于等于 $10$ 的点至多只有 $10$ 个

- 一个简单的想法是每一个点有用的 $z$ 可以继续压缩。

特殊性质 C：出度大于等于 $10$ 的点至多只有 $10$ 个

- 一个简单的想法是每一个点有用的 $z$ 可以继续压缩。
- 注意到这个范围其实可以压缩到这个点的出度与入边编号的最大值。

特殊性质 C：出度大于等于 $10$ 的点至多只有 $10$ 个

- 一个简单的想法是每一个点有用的 $z$ 可以继续压缩。
- 注意到这个范围其实可以压缩到这个点的出度与入边编号的最大值。
- 对于每一个大点我们暴力把所有状态先建立出来，总共不超过 $10n$。

特殊性质 C：出度大于等于 $10$ 的点至多只有 $10$ 个

- 一个简单的想法是每一个点有用的 $z$ 可以继续压缩。
- 注意到这个范围其实可以压缩到这个点的出度与入边编号的最大值。
- 对于每一个大点我们暴力把所有状态先建立出来，总共不超过 $10n$。
- 对于每一个小点，出度不超过 $10$，如果当前的 $z$ 大于 $10$，那么一定会一直减小直到 $10$。所以对于小点也只用至多 $10$ 个状态。

特殊性质 C：出度大于等于 10 的点至多只有 10 个

- 一个简单的想法是每一个点有用的 $z$ 可以继续压缩。
- 注意到这个范围其实可以压缩到这个点的出度与入边编号的最大值。
- 对于每一个大点我们暴力把所有状态先建立出来，总共不超过 $10n$。
- 对于每一个小点，出度不超过 10，如果当前的 $z$ 大于 10，那么一定会一直减小直到 10。所以对于小点也只用至多 10 个状态。
- 在新建的图上求最短路即可，时间复杂度 $O(m\log m)$，期望得分：75。

- 类似特殊性质 C 的做法，我们可以继续缩小 $z$ 的范围。

- 类似特殊性质 C 的做法，我们可以继续缩小 $z$ 的范围。
- 假设当前点度数为 $d$，我们首先建立 $d$ 个状态。

- 类似特殊性质 C 的做法，我们可以继续缩小 $z$ 的范围。
- 假设当前点度数为 $d$，我们首先建立 $d$ 个状态。
- 对于一条编号大于 $d$ 的入边，要么直接停止，要么将 $z$ 减到 $d$ 或更小。

- 类似特殊性质 C 的做法，我们可以继续缩小 $z$ 的范围。
- 假设当前点度数为 $d$，我们首先建立 $d$ 个状态。
- 对于一条编号大于 $d$ 的入边，要么直接停止，要么将 $z$ 减到 $d$ 或更小。
- 也即对于当前点来说，只有不超过 $d$ 的 $z$ 对应的状态是有用的。

- 类似特殊性质 C 的做法，我们可以继续缩小 $z$ 的范围。
- 假设当前点度数为 $d$，我们首先建立 $d$ 个状态。
- 对于一条编号大于 $d$ 的入边，要么直接停止，要么将 $z$ 减到 $d$ 或更小。
- 也即对于当前点来说，只有不超过 $d$ 的 $z$ 对应的状态是有用的。
- 因此对于每一个点，状态数与出边个数相同，状态数总和为 $m$。

- 类似特殊性质 C 的做法，我们可以继续缩小 $z$ 的范围。
- 假设当前点度数为 $d$，我们首先建立 $d$ 个状态。
- 对于一条编号大于 $d$ 的入边，要么直接停止，要么将 $z$ 减到 $d$ 或更小。
- 也即对于当前点来说，只有不超过 $d$ 的 $z$ 对应的状态是有用的。
- 因此对于每一个点，状态数与出边个数相同，状态数总和为 $m$。
- 在新建的图上求最短路即可，时间复杂度 $O(m \log m)$，期望得分：$100$。

- 看到 $z$ 的数值决定着图的走向，不难想到分层图。

- 看到 $z$ 的数值决定着图的走向，不难想到分层图。
- 对于每一个 $z$，我们将原图上所有编号为 $z$ 的边的导出子图抽出来进行建图，对于每一个点分裂出来的所有点排序后连边。

- 看到 $z$ 的数值决定着图的走向，不难想到分层图。
- 对于每一个 $z$，我们将原图上所有编号为 $z$ 的边的导出子图抽出来进行建图，对于每一个点分裂出来的所有点排序后连边。
- 容易发现这样子的总点数为 $2m$，总边数为 $4m$，进行最短路即可。

- 看到 $z$ 的数值决定着图的走向，不难想到分层图。
- 对于每一个 $z$，我们将原图上所有编号为 $z$ 的边的导出子图抽出来进行建图，对于每一个点分裂出来的所有点排序后连边。
- 容易发现这样子的总点数为 $2m$，总边数为 $4m$，进行最短路即可。
- 时间复杂度 $O(m\log m)$，期望得分：$100$。