网络流

赵晟昊

浙江金华第一中学

June 30th, 2025

Dinic

相信大家都会。

复杂度 $O(n^2m)$ 。

单位容量复杂度 $O(m\min(m^{\frac{1}{2}}, n^{\frac{2}{3}}))$ 。

单位容量且所有点入度或出度为一，复杂度为 $O(m\sqrt{n})$ 。

更加细致的复杂度分析可以看 OI WIKI ， 在此不再赘述。

EK

相信大家都会。

复杂度 $O(nm^2)$ 。

相比于 Dinic ，优势在于有时可以用 bitset 优化 BFS 。

P2764 最小路径覆盖问题

给定一张 DAG ，把点划分为尽量少的路径，求方案。

$n \leq 150, m \leq 6000$ 。

Solution

把路径划分看成每个点有不超过有一个的前驱和后继，那么把每个点拆成入点和出点，最大化出点和入点匹配即可。

CF1630F Making It Bipartite

给定一张图，每个点有点权，其中 u 和 v 有边当且仅当他们的点权成倍数关系。

求最少删多少点能得到一张二分图。

$n, a_i \leq 5 \times 10^4$ 。

Solution

图是二分图，等价于所有数只有倍数或只有因数。那么把每个点拆成 $(i,0)$ 和 $(i,1)$ ，分别表示只有倍数和只有因数，给矛盾的连一下边。

需要求最大独立集，但是求不了。注意到可以定向成为偏序集，转化为最长反链。

根据 Dilworth 定理，最长反链等于最小链覆盖，再拆一遍点跑匹配即可。

最小割是啥？

有 n 个 01 变量，其中 S 固定为 0 ，T 固定为 1 。

有 m 个限制 (x, y, w) ，含义是如果 x 为 0 且 y 为 1 就会让分数加 w 。

求最小分数。

最小割最大流定理

最小割等于最大流。

AGC038 E Two Permutations

给定两个排列 P, Q ，称一对排列 (A, B) 是合法的，当且仅当所有元素满足 $A_i \in \{i, P_i\}, B_i \in \{i, Q_i\}$ 。

求所有合法排列对中，$A_i \neq B_i$ 个数的最大值。

$n \leq 10^5$ 。

Solution

转化为最小化 $A_i = B_i$ 的个数。

相当于可以把每个环保留或者全部拆成自环。那么只有两个环同时拆，或者两个同时不拆才可能有贡献。自环要特殊处理。

给每个环建一个点，连边最小割即可。为了写成最小割的形式，有一边的状态需要反转一下。

P3227 [HNOI2013] 切糕

给定 P,Q,R ，有一个 $P\times Q$ 的二维数组 $f(x,y)$ ，你需要对每个位置指定一个 $[1,R]$ 的整数作为值。

合法的 f 需要满足，对于所有的 $1\le x,x'\le P$ 和 $1\le y,y'\le Q$，若 $|x-x'|+|y-y'|=1$，则 $|f(x,y)-f(x',y')|\le D$，其中 D 是给定的一个非负整数。

定义一个 f 的权值为 $\sum_{i=1}^{P}\sum_{j=1}^{Q}v(i,j,f(i,j))$ ，其中 v 给定。求所有合法 f 中的最小权值。

$P,Q,R\le 40$ 。

Solution

对于每一个位置 (x, y)，连出一条链，起点为 S，终点为 T，i 向 $i+1$ 连权为 $v(x, y, i)$ 的边，割掉这条边表示 $f(x, y) = i$。

则一个点 i 为 1 表示 $f(x, y) < i$，为 0 表示 $f(x, y) \geq i$。相邻不能相差太远，也可以写成不能同时 $f(x, y) \geq i$ 和 $f(x, y+1) < i - D$，连个 $+\infty$ 边即可。

ARC176 E Max Vector

给定两个长度为 n 的正整数序列 X, Y 。

有 m 个操作，第 i 个操作会给一个长度为 n 的序列 A_i ，你需要选择以下两种操作之一：

- 对于 $1 \le j \le n$ ，令 $X_j \leftarrow \max(X_j, A_{i,j})$ 。
- 对于 $1 \le j \le n$ ，令 $Y_j \leftarrow \max(Y_j, A_{i,j})$ 。

求出操作结束后 $\sum_{j=1}^{n}(X_j + Y_j)$ 的最小值。

$n \le 10, m, X_j, Y_j, A_j \le 500$ 。

Solution

类似切糕，对每个变量建 $O(V)$ 个点。限制形如每个操作必须满足 $\forall 1 \le j \le n, X_j \ge A_{i,j}$ 和 $\forall 1 \le j \le n, Y_j \ge A_{i,j}$ 中的一个。

给每个限制建两个点 p, q ，分别代表两边限制是否满足。把一边的链反过来，相当于要求 p 为 1 或者 q 为 0 ，连个 ∞ 边即可。

CF786E ALT

给定一棵树，以及 m 条路径。把树的边黑白染色，定义一条路径是坏的当且仅当路径上有白色边，最小化黑边数与（给定 m 条路径中）坏路径数的和。输出方案。

$n, m \le 2 \times 10^4$ 。

Solution

每个路径向路径上的边连边，最小割即为答案。直接建图复杂度过高，需要树剖线段树优化建图。

对于线段树上的一条边，它的流量不超过子树 size ，即区间长度，则所有边流量总和为 $O(n\log n + m\log^2 n)$ ，复杂度为 $O(m\sqrt{m}\log^3 n)$ 。

如果使用全局平衡二叉树建图，则流量总和为 $O((n+m)\log n)$ ，复杂度为 $O(((n+m)\log n)^{1.5})$ 。

Hall 定理

对于一个二分图，记 $N(S)$ 为集合 S 内所有邻居集合的并，则二分图有完美匹配（对于左部点而言）当且仅当，$\forall S \subset \{1, 2, \ldots, n\}, |N(S)| \geq |S|$ 。

拓展

二分图的最大匹配等于左部点点数 $+\min_S\{|S|-|N(S)|\}$ 。

QOJ 12015 撸猫

小 B 喜欢撸猫，他家里有 n 只猫。每天，S 这个集合的猫愿意被撸的概率是 A_S 。每天小 B 只能撸一只愿意被撸的猫。

小 B 需要对于每种情况设计一个策略。记 p_i 为第 i 只猫愿意被撸的概率，你需要求出 c 的最大值，使得每只猫被撸概率至少是 $c \times p_i$。

$n \leq 20$ 。

Solution

直接二分 c ，则问题等价于一个要做一个匹配，一边是 n 只猫，每个点容量 $c \times p_i$ ，另一边是 2^n 个状态，容量 A_S。是一个网络流的形式，可以使用 Hall 定理。

或者根据判定的形式，也可以不用二分，可以直接对一些取个最小值。

一个模拟赛题

有一个长度为 n 的序列 $x_0, x_1, \ldots, x_{n-1}$，初始 n 个值均为 0。

你可以进行以下两种操作：

1. 选择 $i(0 \le i < n)$，令 x_i 加一。这种操作最多只能操作 a_i 次。
2. 选择 $i(0 \le i < n-1)$，你可以选择令 x_i 加一，或者令 x_{i+1} 加一。这种操作最多只能操作 b_i 次。

q 次询问，每次给定 l, r，求若干次操作后 $[l, r]$ 的最小值最大可能是多少。询问独立。

$n, q \le 10^5$ 。

Solution

二分答案，当成二分图，使用 Hall 定理，发现只有区间的限制是最严的。

用 P5073 世上最幸福的女孩 的技巧维护即可。

SSP

每次增广最短路即可。

最短路单调不降，所以费用流函数是凸的。

直接做，每次跑 SPFA ，复杂度为 $O(nmf)$ ，其中 f 为最大流。

原始对偶

假设有一个势能 h_i ，每次跑最短路的时候把一条边 (u, v, w) 的权值改为 $w + h_u - h_v$ ，只要最后加上 h_u 就能得到正确的最短路。

尝试选取一个合适的势能，让所有新的边权都是非负的。发现取上一次的最短路即可。

每次跑 Dijkstra ，复杂度为 $O((m \log n)f)$ ，其中 f 为最大流。

“基于 Capacity Scaling 的弱多项式复杂度最小费用流算法”

首先可以给 T 到 S 连一条 $+\infty$ 边，把有源汇转化为无源汇。

注意到把所有边流量乘以二，则最大流和最小费用流乘以二。

从高到低加入每一位，每次会加入若干边，增广一下，然后把所有边流量乘以二。

每次加入一条边的时候，只要找到包含这条边的最小环。跑个最短路即可。

结合原始对偶，复杂度可以做到 $O(m^2 \log V \log n)$ 。

看上去很美好，实际上没什么用（

有负环的费用流

强制负边流满，然后类似上下界操作一下，这样就不会有负边权边了。

AGC031 E Snuke the Phantom Thief

二维平面上有 n 个整点，每个点有一个分数。

有 m 个限制，形式如下：

- 横坐标 $\le a_i$ 的最多选 b_i 个点。
- 横坐标 $\ge a_i$ 的最多选 b_i 个点。
- 纵坐标 $\le a_i$ 的最多选 b_i 个点。
- 纵坐标 $\ge a_i$ 的最多选 b_i 个点。

现在要选一些点，求分数最大值。

$n \le 80, m \le 320, x_i, y_i \le 100$ 。

Solution

枚举总点数，则限制可以写成横坐标/纵坐标第 i 大的应该 $\geq x$ 或 $\leq x$ 。

考虑给每个点分配一个横坐标/纵坐标的排名，则用类似横坐标 $\rightarrow$ 点 $\rightarrow$ 纵坐标的结构，所有限制和贡献都容易表达。

问题在于，可能这个排名不是真的。但是发现如果交换相邻两个，限制会更严，所以无所谓。

AGC034 D Manhattan Max Matching

二维平面上有若干个红球和蓝球，红球和蓝球个数相等。

现在要把红球和蓝球配对，一对球的价值是它们的曼哈顿距离，求价值和的最大值。

$n \le 10^3$ 。

加强版：$n \le 10^5$ 。

Solution

把 $|x|$ 拆成 $\max(x, -x)$ 。由于最终求的是最大值，只要对两种情况分别考虑。

一共两维就是四种情况，每种情况建一个点，让每条路径都是蓝球 $\to$ 中间 $\to$ 红球。

加强版 Solution

考虑一条增广路，发现肯定每两步就会经过中间的一个点。由于不会经过重复点，所以增广路不会太长。

中间点之间移动只能是走到一个红球/篮球再走回去，维护一下这些操作的最小值，每次跑只包含 S, T 和中间点的最短路即可。

上下界网络流

强制流满下界，然后建立超级源汇向流满下界的边提供流量，即可转化为一般图。

GYM 103855 I Marbles

有 n 个弹珠和若干个袋子。每个弹珠是红色或蓝色，初始第 i 弹珠在第 i 个袋子里。

有 m 个操作，形如：

1. 合并两个袋子。
2. 丢弃了一个弹珠。
3. 观察到第 i 个袋子有 $[l, r]$ 个红色弹珠。

判断是否能存在合法弹珠颜色，并构造方案。

$n \leq 2 \times 10^3, m \leq 4 \times 10^4$ 。

Solution

直接对每个袋子建立一个点，我们希望经过这个点的流量是它红色弹珠的个数。

每次删掉一个弹珠的时候，就连出一条边到它出现的地方。

每次操作都新建一个点，则所有操作都是可以表达的。跑个上下界无源汇可行流即可。