

有源汇可行流

对于一个简单有向图 $\langle V, E \rangle$, 我们钦定两个点 S, T 分别为源点和汇点, 每条边 (u, v) 有一个非负整数的限制 $c(u, v)$ 称为该边的容量。记 (u, v, c) 为一条 u 到 v 的容量为 c 的边。

我们称该图的一个可行流为, 一个给该图每条边都附上一个非负整数权值 $f(u, v)$ 的方案, 其中 $f(u, v)$ 称为该边的流量。令 $in(u) = \sum_{(v,u) \in E} f(v, u), out(u) = \sum_{(u,v) \in E} f(u, v)$, 赋值的 f 需要保证如下限制:

- $\forall (u, v) \in E, f(u, v) \leq c(u, v)$ 。
- $\forall u \in V, u \neq S, u \neq T, in(u) = out(u)$ 。
- $in(S) = out(T) = 0$ 。

显然, 一定有 $out(S) = in(T)$, 我们称 $out(S)$ 为这个可行流的流量。

有源汇最大流

我们称一个图的最大流量为其所有可行流的流量最大值。称流量和该图最大流量相同的可行流为该图的最大流。显然, 最大流不一定唯一。

求解最大流

其中一种贪心想法就是每次找一条 S 到 T 的路径, 然后令这条路径上每条边的容量减一, 显然这是错的。我们加一点修正策略: 给每条有向边建立一条初始容量为 0 的反向边, 每次我们找到一条 S 到 T 的路径 (下称增广路), 就给这条路径上的每条边容量减一, 给它们的反向边容量加一。走一条原图的边的反向边相当于给这条边的流量减一, 反向边记录的就是这条边可以走反向边多少次。正确性考虑一个最大流方案为 F , 当前流方案为 F^* , 考虑 $F - F^*$ 是一个当前残余网络的流方案, 所以如果 $|F^*| < |F|$, 则一定有 S 到 T 的增广路。

然而这还是太慢了, 我们可以加入分层图优化: 每次跑一个 S 到 T 的 bfs 树, 然后再不断地找 S 到 T 的路径, 要求路径上的点深度严格递增, 如果找不到了就重建 bfs 树, 直到 S 到 T 没有路径。注意到我们每次重建 bfs 树, T 所在深度至少加一, 所以重建次数只有 $O(n)$ 次, 然后如果我们每次找到了一条增广路, 则我们可以一直选定这条路径直到这条路径中某条边容量被减到 0 了, 所以每次找到一条增广路至少一条边容量变成 0, 至多找到 m 条, 每条长度至多为 $O(n)$, 如果我们暴力修改增广路上的边容量, 则此时复杂度为 $O(n^2m)$ 。

然而我们找增广路的过程太慢了, 如果暴力 dfs 的话复杂度将是 $O(nm^2)$ 。我们加入当前弧优化, 如果顺着一条边走找不到最短路, 那么在重建之前这条边都没有用处, 删去, 则找增广路的复杂度为 $O(nm + m) = O(nm)$, 重建 bfs 树次数为 $O(n)$, 总复杂度 $O(n^2m)$ 。

特殊图的分析

如果图中的边权完全相同, 那么暴力修改增广路的复杂度会变为 $O(m)$, 最大流总复杂度会变为 $O(nm)$ 。注意到每次增广如果流了某条边, 那么这条边得流量会被流完, 所以可以当作边权为 1 的图。

如果图中边权全都为 1, 可以证明:

Lemma 1: 假设一个边权全为 1 的图残余流量为 M , 则在残余网络构建分层图时汇点 T 所在层数至多为 $\frac{2n}{\sqrt{M}}$ 。

Proof: 设 S 到 T 此时最短路为 l , 由于残余流为 M , 令 V_i 为距离 S 最短路为 i 的点的数量, 那么有 $V_i \times V_{i+1} \geq M$, 那么 V_i, V_{i+1} 至少一个 $\geq \sqrt{M}$ 。由于 $\sum V_i \leq n$, 那么 $\lfloor \frac{l+1}{2} \rfloor \sqrt{M} \leq n$, 那么有 $l \leq \frac{2n}{\sqrt{M}}$ 。

记最大流为 f^* , 假设现在我们获得的流量为 f , 当 $f > f^* - n^{2/3}$, 至多再进行 $n^{2/3}$ 次增广。当 $f \leq f^* - n^{2/3}$, 那么 $M \geq f^* - f \geq n^{2/3}$, 增广路长度至多为 $\frac{2n}{\sqrt{M}} \leq 2n^{2/3}$, 至多增广 $O(n^{2/3})$ 次。所以总增广次数为 $O(n^{2/3})$, 每次增广复杂度为 $O(m)$, 最大流复杂度为 $O(n^{2/3}m)$ 。

然后还是考虑 $F - F^*$, 如果增广了 B 次, 那么此时每条增广长度至少为 B 。由于每条边边权为 1, 所以 $F - F^*$ 至多有 $\frac{m}{B}$ 条增广路, 至多再增广 $\frac{m}{B}$ 次, 所以总增广次数为 $O(\sqrt{m})$ 。综上, 最大流复杂度为 $O(\min(n^{2/3}m, m^{3/2}))$ 。

如果是二分图, 注意到 $F - F^*$ 的边数为 $O(n)$ 的, 所以总增广次数为 $O(\sqrt{n})$, 最大流复杂度为 $O(m\sqrt{n})$ 。

最小割

对于一个有源汇的有向图 $\langle V, E \rangle$, 我们选定一个包含 S 且不包含 T 的点集 U , 记 $U' = V \setminus U$, 则记 $E' = \{(u, v) \mid (u, v) \in E, u \in U, v \in U'\}$ 为该图的一个割集。若该割集里所有边的容量和是该图所有割集中最小的, 则称该割集为该图的最小割。

最大流最小割定理

考虑选定一个割集, S 到 T 的每一条路径一定经过这个割集里某一条边, 所以任意一个流的流量小于等于任意一个割的权值和, 即最大流小于等于最小割。

考虑一个最大流, 考虑令残余网络上可以被 S 到达的点集为 U , 则这个点集生成的割的权值和最大流量相同, 所以最大流大于等于最小割。

所以最大流等于最小割。

luogu P3227

给定一个 P, Q, R, D 和一个 $P \times Q \times R$ 的数组 $v_{i,j,k}$, 你需要构造一个 $P \times Q$ 的数组 $f_{i,j}$, 满足 $1 \leq f_{i,j} \leq R$, 且 $\forall i, j, u, v, |i - u| + |j - v| = 1, |f_{i,j} - f_{u,v}| \leq D$ 。你的目的是让 $\sum_{i,j} v_{i,j,f_{i,j}}$ 最小。

$1 \leq P, Q, R \leq 40, 0 \leq D \leq R, 0 \leq v_{i,j,k} \leq 1000$ 。

费用流

给每条边加上参数 $p(u, v)$ 称为该边的费用, 记一个可行流的费用为 $\sum f(u, v)p(u, v)$ 。

最小费用最大流即求所有最大流中费用的最小值。记 (u, v, c, w) 为一条 u 到 v 的容量为 c , 费用为 w 的边。

求解费用流

把求最大流时的 bfs 改为 spfa 即可。

最大流常用trick

最大权闭合子图

有 n 个物品, 每个物品有价值 V_i (可能为负), 有若干个限制形如 (i, j) 的限制, 表示如果选择了物品 i 就必须选择物品 j 。你需要选择若干个物品使价值和最大。

我们考虑使用最小割解决这个问题:

- 当 $V_i > 0$, 连边 $(S, i, V_i), (i, T, 0)$, 割掉前者说明不选择这个物品, 否则选择该物品。
- 当 $V_i < 0$, 连边 $(S, i, 0), (i, T, -V_i)$, 割掉前者说明不选择这个物品, 否则选择该物品。

对于限制 (i, j) , 连边 (i, j, inf) , 如果 j 割掉了与 S 的连边, 则 i 必须割掉与 S 的连边, 否则有路径 $S \rightarrow i \rightarrow j \rightarrow T$, 否则没有限制, 这就相当于选择 i 必须选择 j 。

用 $\sum [V_i > 0] V_i$ 减去最小割即为答案。

luogu P3749

有一个 $n \times n$ 的矩阵, 每个点 $(i, j) (i \leq j)$ 上有数 $w_{i,j}$, 和一个长度为 n 的序列 a_i 。你可以选择若干个区间 $[l_i, r_i]$, 你的收益是 $\sum_{i \leq j} [\exists k, l_k \leq i \leq j \leq r_k] w_{i,j}$ 。考虑 $\text{cnt}_x = \sum_{i=1}^n [a_i = x \wedge \exists k, l_k \leq i \leq r_k]$, 那么你的花费为 $\sum_x \text{cnt}_x x + [\text{cnt}_x > 0] m x^2$ 。

求你的收益减花费最大值。

$1 \leq n \leq 100, 1 \leq a_i \leq 1000, 0 \leq m \leq 1, -500 \leq w_{i,j} \leq 500$ 。

平面图最小割

考虑一个平面图 (把点放在平面上, 任意两条边除了端点没有交点)。

考虑源点和汇点分别向无限远处引出一条射线, 这两条射线互不相交且和所有边除了端点没有交点, 那么这两条射线与这些边把平面分成了若干个部分, 我们构建一个新图, 新图里的每个点代表一个部分, 如果两个部分有共用的边, 那么给它们对应的点连边, 边权与共用边边权一致。那么平面图最小割等于其中一个面积无限大的部分对应的点到另一个面积无限大的部分对应的点的最短路。

QOJ#7737

给定一个 $n \times m$ 的网格图以及正整数 k , 即 $(x, y), (x', y')$ 之间有边当且仅当 $|x - x'| + |y - y'| = 1$, 每条边有正边权。你可以进行任意次如下操作: 选择一条边, 该边权加一。记

$d = \min_{1 \leq p, q \leq n} \{\text{dis}((p, 1), (q, m))\}$, 其中 dis 为网格图上两点最短路, 求最小操作次数使得 d 至少增加 k 。

$2 \leq n, m \leq 500, 1 \leq n \times m \leq 500, 1 \leq k \leq 100$, 边权范围 $[1, 10^9]$ 。

二分图最小点覆盖、最大独立集

点覆盖: 选定一个点集, 使每条边都有至少一个端点在点集里。

独立集: 选定一个点集, 使得点集里任意两个点没有连边。

考虑点覆盖的补集一定是独立集, 所以最小点覆盖+最大独立集大于等于总点数。

考虑独立集的补集一定是点覆盖, 所以最大独立集+最小点覆盖小于等于总点数。

所以最小点覆盖+最大独立集等于总点数。

考虑如何求二分图最小点覆盖。考虑最小割, 对于左部点连边 $(S, i, 1)$, 对于右部点连边 $(i, T, 1)$, 如果割掉了说明对应的点在点覆盖内。那么对于原图的一条边 (u, v) , 连边 (u, v, inf) , 代表 u, v 至少一个点在点覆盖内。所以最小割即最小点覆盖。

考虑最小割等于最大流, 注意到每个点与源汇连边的边权为 1, 把 (u, v, inf) 改为 $(u, v, 1)$ 最大流不变, 所以最小点覆盖等于最大匹配。

最小不交链覆盖

给定一个有向无环图, 你需要选择这个有向无环图上若干条路径, 使得每个点出现在某条路径上恰好一次。你需要使得路径数最小。

考虑把这个问题改变为匹配问题:

对于每个点拆成左部点 u 和右部点 u' ，对于原图的有向边 (u, v) ，在二分图上连边 (u, v') 。如果 u 匹配了 v' ，说明在我们选择的路径中 u 是 v 的前驱。每个点只能是一个点的前驱，一个点的后继，对应了其左部点的匹配和右部点的匹配。

所以最小不交链覆盖就是 n 减去匹配数。

最小可交链覆盖

给定一个有向无环图，你需要选择这个有向无环图上若干条路径，使得每个点至少出现在某条路径一次。你需要使得路径数最小。

我们对这个 DAG 进行传递闭包，即令 u 向所有它能到达的点连边，则原图的最小可交链覆盖相当于新图的最小不交链覆盖。

最长反链

给定一个有向无环图，你需要选择这个有向无环图上若干个点，使得这些点两两不能互相到达。你需要使得选的点最多。

考虑原图最小可交链覆盖为 k 。则上述构造的二分图独立集大小为 $2n - (n - k) = n + k$ ，那么反链长度大于等于 $n + k - n = k$ 。故最长反链大于等于最小可交链覆盖。

构造最小割方案

考虑跑完 dinic 后找到 S 能到达的点集 U ，那么从 U 连向 U' 的边集就是一个最小割。

二分图匹配，但某些点必须在匹配内

做一个新图 G 。考虑拎出必须匹配的左部点和所有右部点，跑二分图匹配，如果没有完备匹配说明无解，否则视为在 G 上，每个左部点向它匹配的右部点连一条边。然后拎出必须所有左部点和必须匹配的右边点，类似地做，在 G 上，每个右部点向它匹配的左部点连一条边。此时发现在 G 上，每个点的度数至多为 2，所以只有环和链。对于环，发现全都是偶环，隔一条边取一条边作为匹配边，显然环上的每个点都在匹配里了。对于链，从链开头那条边开始，隔一条边取一条边作为匹配边（比如 $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$ ，就取 $(1, 2), (3, 4)$ ，发现除了链最后一个点，其他点一定都在匹配内，仔细思考发现最后一个点没有限制（不然它就有出边），所以不用管。这样跑两次二分图匹配就可以得到一个可行的匹配了，复杂度 $O(m\sqrt{n})$ 。

Hall 定理

假设有二分图，考虑左部点的某个子集 S ，即 $N(S) = \{v \mid \exists u \in S, s.t. (u, v) \in E\}$ ，即 S 的邻居点集。

那么 S 有完美匹配，当且仅当 $\forall T \subseteq S, |N(T)| \geq |T|$ 。

\implies :

显然，令 $p(u)$ 为 u 的匹配点，则 $\{p(u) \mid u \in T\} \subseteq N(T)$ ，所以 $|N(T)| \geq |T|$ 。

\impliedby :

首先归纳证明，在满足 $\forall T \subseteq S, |N(T)| \geq |T|$ 时，假如存在 $|N(T)| = |T|$ ，则 T 和 $N(T)$ 有完美匹配：

当 $|T| = 1$ 时，命题显然成立。

当 $|T| > 1$ 时, 选取 T 中的一个点 u 以及它的一个邻居 v , 假如删去 u, v 两个点后, T 的每个子集都还满足 Hall 定理的条件, 则让 u, v 匹配。否则, 假如 $P \subset T$ 不满足条件, 显然 $P \neq T$, 且 $|N(P)| = |P|$, 则由归纳假设让 P 和 $N(P)$ 有完美匹配, 直接让它们匹配后删去它们, 剩下的点继续做即可。注意到我们让它们匹配后删去它们一定不会让其他集合不满足 Hall 定理的条件, 因为假如删去后有 $|Q| > |N(Q)|$, 则一开始就有 $|Q \cup P| = |Q| + |P| > |N(Q)| + |N(P)| \geq |N(Q) \cup N(P)|$ 。

那么如果有 $|T| = |N(T)|$, 那么它们互相匹配后就删掉, 对剩下的图继续做。那么现在一定满足 $\forall T \subseteq S, |N(T)| \geq |T|$, 那么我们显然可以随意挑选一个 $u \in S$, 令其和随便一个邻居匹配后一起删掉, 之后因为每个 $N(T)$ 至多去掉一个点, 所以还是满足 Hall 定理的条件, 归纳下去即可。

gym101173B

给定左部点数为 n_1 , 右部点数为 n_2 的二分图。

设这 $n_1 + n_2$ 个点构成的点集为 V 。求有多少个 V 的子集 S , 使得存在一种二分图的匹配方式, S 内所有点都在匹配内。

$1 \leq n_1, n_2 \leq 20$ 。

有上下界网络流

给每条边加上参数 $l(u, v)$ 称为该边的下界, 即要求 $l(u, v) \leq f(u, v) \leq c(u, v)$ 。

记 $(u, v, [L, R], w)$ 为一条 u 到 v 的流量上下界为 $[L, R]$, 费用为 w 的边。

无源汇上下界可行流

这个问题要求每条边的流量分别在自身容量上下界内, 且每个点流入流出平衡。

那么我们可以建立超级源点 SS 和超级汇点 TT 。假如有一条 $(u, v, [L, R], c)$, 那么我们将其改为 $(u, TT, L, 0), (SS, v, L, 0), (u, v, R - L, c)$, 让费用加上 $L \times c$ 。建完边后跑 SS 到 TT 的最大流, 如果没有流满 (这里指超级汇点的入边流量都达到容量上界), 说明无解, 否则每条边现在的流量加上自身的下界就是真正的流量。

正确性: 发现这样建边后, 假设不考虑 SS 和 TT , 相当于钦定 u 点入流量减出流量为 L , v 点出流量减入流量为 L , 显然必须跑满才是一组解。

有源汇上下界最大流

先连接 $(T, S, \inf, 0)$, 使原图变为无源汇图, 然后跑无源汇上下界可行流。然后删去那条边, 在残余网络上跑 S 到 T 的最大流即可。

正确性: 发现 `dinic` 的过程中, 除了源汇之外的点的出流量减入流量的数值不变, 显然原来的流可行, 跑完 S 到 T 的最大流之后也可行。

有源汇上下界最小流

先连接 $(T, S, \inf, 0)$, 使原图变为无源汇图, 然后跑无源汇上下界可行流。然后删去那条边, 在残余网络上跑 T 到 S 的最大流即可。

正确性: 可行性证明和上文一致, 发现 T 到 S 的最大流相当于退尽可能多的流量。

CF704D

有 n 个平面上的点，你需要给他们染上黑白色，需要满足 m 个形如在某条与坐标轴平行的直线上白点和黑点数之差小于某个数的限制，给一个点染黑的花费为 b ，染白的花费为 w 。

求最小花费。

$n, m \leq 10^5$ 。

CF843E

给定一个带源汇的有向图，但边权不定，有一个隐藏的最大流方案 F 。每条边有一个参数 g_i ， $g_i = 0$ 表示在 F 中这条边没有流量， $g_i = 1$ 表示在 F 中这条边有流量。

你需要给每条边定一个容量，使得源点到汇点的最小割包含**边数**最小，且存在一个最大流方案满足所有 g_i 的限制。

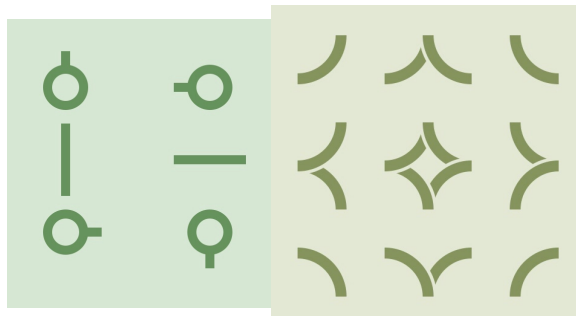
$2 \leq n \leq 100, 1 \leq m \leq 1000$ 。

杂题

uoj#336

曾经有一款流行的游戏，叫做 *Infinity Loop*，先来简单的介绍一下这个游戏：

游戏在一个 $n \times m$ 的网格状棋盘上进行，其中有些小方格中会有水管，水管可能在方格某些方向的边界的中点有接口，所有水管的粗细都相同，所以如果两个相邻方格的公共边界的中点都有接头，那么可以看作这两个接头互相连接。水管有以下 15 种形状：



游戏开始时，棋盘水管可能存在漏水的地方。

形式化地：如果存在某个接头，没有和其它接头相连接，那么它就是一个漏水的地方。

玩家可以进行一种操作：选定一个含有**非直线型**水管的方格，将其中的水管绕方格中心顺时针或逆时针旋转 90 度。

直线型水管是指左图里中间一行的两种水管。

现给出一个初始局面，请问最少进行多少次操作可以使棋盘上不存在漏水的地方。

$1 \leq nm \leq 2000$ 。

loj#2226

宅男 JYY 非常喜欢玩 RPG 游戏，比如仙剑，轩辕剑等等。不过 JYY 喜欢的并不是战斗场景，而是类似电视剧一般的充满恩怨情仇的剧情。这些游戏往往都有很多的支线剧情，现在 JYY 想花费最少的时间看完所有的支线剧情。

JYY 现在所玩的 RPG 游戏中，一共有 N 个剧情点，由 1 到 N 编号，第 i 个剧情点可以根据 JYY 的不同选择，而经过不同的支线剧情，前往 K_i 种不同的新的剧情点。当然 K_i 如果为 0，则说明 i 号剧情点是游戏的一个结局了。

JYY 观看一个支线剧情需要一定的时间。JYY 一开始处在 1 号剧情点，也就是游戏的开始。显然任何一个剧情点都是从 1 号剧情点可达的。此外，随着游戏的进行，剧情是不可逆的。所以游戏保证从任意剧情点出发，都不能再回到这个剧情点。

由于 JYY 过度使用修改器，导致游戏的「存档」和「读档」功能损坏了，所以 JYY 要想回到之前的剧情点，唯一的方法就是退出当前游戏，并开始新的游戏，也就是回到 1 号剧情点。JYY 可以在任何时刻退出游戏并重新开始。

不断开始新的游戏重复观看已经看过的剧情很是痛苦，JYY 希望花费最少的时间，看完所有不同的支线剧情。

CF1416F

对于大小为 $n \cdot m$ 的矩阵 A 和 B ，其中 A 的每个元素为一个权值 $w(i, j)$ ， B 的每个元素为一个方向 L/R/D/U

初始你在 (i, j) ，若 $B_{i,j} = L$ ，你可以走到 $(i, j - 1)$ 处，依次类推。

定义 $S_{i,j}$ 表示从 (i, j) 出发能够到达的点的 $A_{i,j}$ 的和。

给定矩阵 S ，构造 A 和 B 使得其生成的矩阵为 S

A 的每个元素均为正整数。

$$1 \leq n \cdot m \leq 10^5, S_{i,j} \in [2, 10^9]$$

PTZ winter 2020 Day8 A

给定一个左部点有 n_1 个，右部点有 n_2 个，边有 m 条的二分图，求其的一个匹配，使得你求的匹配大小至少是最大匹配的 0.95 倍。

$$1 \leq n_1, n_2, m \leq 10^6。$$

CF1284G

有一个 $n \times m$ 的网格图，一些格子上有障碍，保证 $(1, 1)$ 没有障碍。将网格黑白染色，使得 $(1, 1)$ 为黑色，有公共边的格子不同色。你可以在一些格子的公共边上放墙，两个格子直接连接当且仅当他们有公共边且公共边上没有放墙。你的目标是使得任意两个没有障碍的格子之间有且仅有一条简单路径，且除了 $(1, 1)$ 之外的每个黑色格子的度数至少为 2。

求一种放墙的方案。

$$2 \leq n, m \leq 20。$$

思考：假如范围为 $1 \leq nm \leq 2 \times 10^5$ 呢？