图论与连通性问题选讲

沈吉灏

无向图的点/边连通度

- 若需要至少割掉 >=k 个点/边使得 u,v 不连通，则称 u,v 是 k-点/边连通的
- Menger 定理:
- u, v 是k-边连通的，当且仅当存在k条u->v的两两边不交的路径;
- u, v 是k-点连通的，当且仅当存在k条u->v的除端点外两两点不交的路径。
- 证明可使用最大流最小割定理

双连通性

- 边双连通分量缩点后变成一棵树
- 点双连通分量一般考虑建出圆方树

qoj4809. Maximum Range

- 无向图，每条边有边权。
- 在图中找一个边不重复的环，最大化环上的最大边权-最小边权，需要构造方案。

Solution

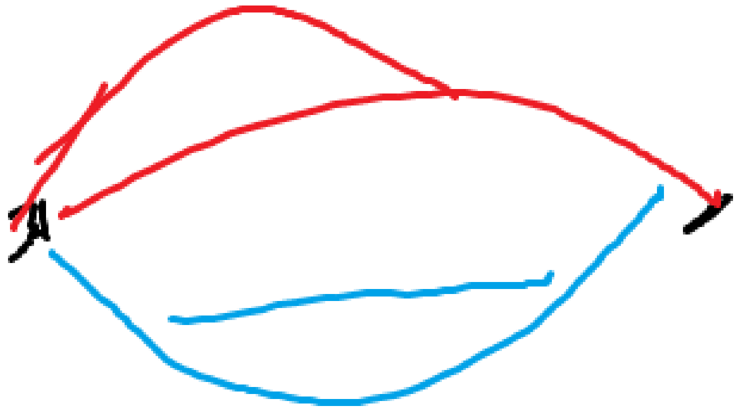
- 对于一个边双，任意选两条边，我们都可以构造一个边不重复的环，同时经过这两条边
- 对于每一个边双，求出最大边权-最小边权，取最大值就是答案。
- 构造方案的一个简洁做法：
- 在两条边中间各自建立一个虚点 s,t，建一张每条边容量为 1 的图
- 求 s 到 t 的最大流，增广到流量为 2 后停止
- 得到有流量的边，跑一个欧拉路（或进行分类讨论），就得到了环
- 时间复杂度线性

SWTR-8 地地铁铁

- 给出无向连通图，每条边标有 D 或 d。
- 计数有多少个点对 (x,y) 满足：x,y 之间存在同时出现 D 和 d 的简单路径。
- https://www.luogu.com.cn/problem/P8456

Solution

- 先考虑每个点双内的点对，显然经过的路径不能到点双外面
- 若一个点双内所有边颜色相同，则所有点对不合法
- 否则，我们可以证明：一个点双内最多有一个点对不合法
- 考察每个点的出边，若存在异色出边的点恰为两个，则它们不合法



Solution

- 再考虑不在同一点双内的点对
- 若它们之间经过的所有点双颜色都相同，它们才可能不合法
- 建立圆方树，对于每个同色连通块统计即可
- 时间复杂度线性

CCPC 2023 Guilin F: Redundant Towers

- 给定平面上 n 个横纵坐标两两都不同的点，两个点之间如果欧几里得距离不超过 R 则连边。
- 每次在线地激活或者屏蔽一个点
- 查询：只考虑被激活的点的子图，全局非割点的数量。
- n<=100000,R<=5
- https://qoj.ac/contest/1404/problem/7682
- 提示：改成所有边保证 |i-j|<=R 也能做

Solution

- 非割点数量 = 圆方树叶子数量
- 考虑维护每个区间的圆方树形态，但每个圆方树大小为 O(n)，无法接受

Solution

- 每个区间只有左边 R 个点和右边 R 个点可能继续连边
- 考虑建出这些关键点的“虚圆方树”
- 虚树上的每条边要记录上面省略了多少个圆点，以及端点是圆/方
- 由于虚树省略了外面的一些子树，对于每个点还要记录它是不是叶子，以及外面已经被省略的叶子数

Solution

- 在线段树每个节点上维护区间的“虚圆方树”
- 记录的信息是 O(R) 的
- 合并时，从虚圆方树反向造一张图，连上区间之间的边跑 tarjan，再压缩出新的虚树
- 每次合并是 O(R^2)，时间复杂度 O((n+qlogn)R^2)

耳分解

- 对于图 G 的一个子图 G’=(V’,E’):
- 若简单路径/简单环 P=x_1->x_2->..->x_k 满足 x_1,x_k 属于 V′, x_2,...,x_k-1 不属于 V′，则称 P 为 G 关于 G′ 的开耳。
- 若无向图 G 能从只包含一个点的子图 G’ 开始，不断向 G’ 中加入 (G,G’) 的开耳，最终得到图 G，则称图 G 是可以被耳分解的。

耳分解

- 一张有向图是可耳分解的，当且仅当它强连通。
- 一张无向图是可耳分解的，当且仅当它边双联通。

[SNOI2013] Quare

- 给出 n 个点 m 条边的无向图，边有边权
- 保留一个边的子集，使得图仍然边双连通，最小化边权之和。
- n<=12, m<=40
- https://www.luogu.com.cn/problem/P5776
- 类似题目：Economic One-way Roads（https://qoj.ac/contest/776/problem/3301）

Solution

- 从空的图开始，每次加入一个耳，直到扩展到全图
- 设 f(S,i,j) 表示：当前加入了 S 这个集合，耳扩展到了 i 这个点，最后要回到 j 这个点
- 设 g(S) 表示 S 的答案
- 每次从 g(S) 枚举一个 j in S，然后转移到 f；f 每次加一个点
- 时间复杂度 O(n^3*2^n)

双极定向

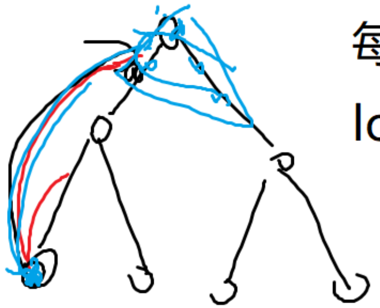
- 给定一张无向图 G 与两个点 s,t
- 1. 构造一个所有点的排列 p_1,..,p_n，使得 p_1=s,p_n=t, 且任意前缀以及后缀的导出子图都是连通的
- 2. 给 G 的所有边定向得到一个有向无环图，使得 s 入度为 0，t 出度为 0，其余点入度、出度均不为 0
- 一个点双连通图一定能求出双极定向

双极定向-方法 1

- 考虑不断染黑点，并且在过程中，黑点、白点的导出子图均连通
- 以 s 为根求出 dfs 树，求出每个点的 fa(u) 和 low(u)（最浅能到达的祖先）。
- 在每个点开一个列表，每次剥掉一个叶子，把该叶子加入 fa(u) 和 low(u) 的列表末尾，表示染黑了 fa(u) 或 low(u) 后就可以染黑 u。这样若一个点染黑，则可以将其列表里的点依次染黑，不断递归下去。
- 提取 s->t 的路径，在剥叶子的过程中，不剥掉这条路径上的点。然后将路径从 s->t 依次染黑，并且递归染黑其列表。

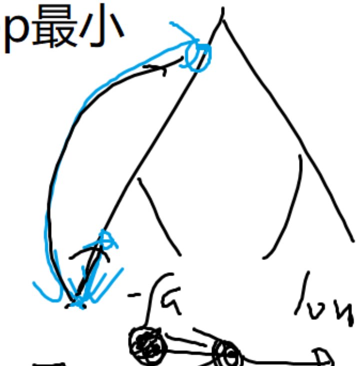
双极定向-方法 1

- 这个做法的本质是:
- 对于叶子节点，只保留了 u-fa(u) 和 u-low(u) 的边，这样并不改变点连通性
- 此时叶子节点度数为 2，进行缩二度点操作
- 若 fa(u) 或 low(u) 中有某一个染黑，则立刻染黑 u
- 这样操作，黑与白连通块仍然连通



每个点 留dep最小

low[i]

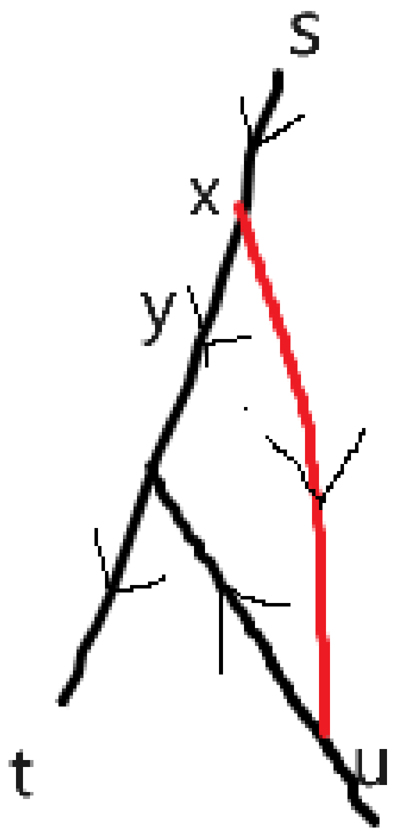


双极定向-方法 2

- 考虑将所有边定向。
- 先以 s 为根求出 dfs 树，然后将 s->t 的路径定向成“向下”。
- 我们先把 t 推进队列里，并把它标记为“向下”（向队列中加入 (t,↓)）。
- 从队列里取出一个点 t，然后不断向上爬祖先，把 t 到某个祖先都标记成相应的方向，直到碰到一个标记过的祖先边结束。

双极定向-方法 2

- 假设我们当前定向了 (x,y) 这条边。这时碰到一条返祖边 (x,u) 且 u 在 y 的子树中（重要细节：不要考虑 u 不在 y 子树中的返祖边）。
- 此时需要给这条返祖边 (x,u) 定向成 (x,y) 相同的方向，然后将 u 向上的一段树边路径定向成 (x,y) 相反的方向。
- 把 u 和这个对应方向推进队列里（比如图中就是加入 (u, 个)），不断 bfs 即可。
- 这也同时刻画了一个耳分解。



IOI2019 景点划分

- 给出一张无向图，需要把点集划分成 a,b,c (a+b+c=n) 大小的三部分，使得至少有两部分连通
- 判断有无解，并构造方案
- https://uoj.ac/problem/535

Solution

- 假设 a<=b<=c，我们想找到大小为 a,b 的连通块，设为两种颜色
- 容易发现，至多有一个点双有两种颜色
- 建立圆方树，枚举一个方点，钦定这个点双中可能有两种颜色

Solution

- 假设以方点为根，得到若干个子树。分类讨论：
- 若最大的子树大小 >n-a，则此方点无解
- 若最大的子树大小 >=b，则在该子树中选一个 b 连通块，其余选一个 a 连通块，构造完毕
- 否则所有子树大小 <b。对所有圆点做双极定向，按照定向的顺序排序，取一个前缀作为 a，一个后缀作为 b，前后缀各自连通
- 一个个加入前缀的子树，某次加入子树从 <a 变成 >=a 的时候，这一个前缀一定 <a+b
- 则剩余后缀的大小 >n-a-b=c>=b。
- 对于前后缀各自取 a,b 大小连通块，构造完毕。

割空间与环空间

- 在图 G=(V,E) 上：
- 将 V 划分成两个子集 V1,V2，定义 V1,V2 间的**割集**为 V1,V2 之间的所有边。
- 将任意边集看作 F_2 上的 m 维向量，所有割集生成的空间称为**割空间**。
- 定义 所有满足每个点的度数都为偶数的子图 的边集构成**环空间** 。
- 同一张无向图的割空间与环空间互为正交补。

切边等价

- 在一个边双连通图中，定义两条边**切边等价**，当且仅当：在任何 G 的简单环中，这两条边要么同时出现要么同时不出现。
- 两条边切边等价等价于，从 G 中删去这两条边后 G 不连通。
- 证明：⇔删去 e1 后 e2 为割边⇔包含 e1 的每个环都包含 e2
- 等价于：任取 G 的一棵生成树 T，假设 e1,e2 都是树边，不存在一条非树边跨过其中一条却不跨过另一条。

切边等价

- 我们不能给每条边设定一个长度为 m 的向量，这样会复杂度过大
- 考虑为每条非树边设定一个随机权值，定义树边的权值为所有跨过它的非树边的权值异或和。
- 权值为 0 的边为割边；将剩余所有边按照权值划分成切边等价类。
- 任意异或和为 0 的边集为图的割集。
- 这个结论的证明：https://rushcheyo.blog.uoj.ac/blog/6704

切边等价的性质

- 在 dfs 树上，权值相同的树边一定形成祖先-后代链。
- 两种不同权值的等价类一定完全包含或完全相离，即不会出现 ABAB 的情况。

WF2015 Tours

- 给定一张 n 个点 m 条边的无向图
- 你需要选择一个颜色种类数 k，然后用这 k 种颜色给每条边染色，
- 要求对于图中任意一个简单环，每种颜色的边的数量都相同。求所有可行的 k。
- https://www.luogu.com.cn/problem/P6914

Koosaga’s Problem

- 给出一张无向图，需要割掉 <=2 条边，使得图成为二分图
- 求出最小需要割掉的边数，以及方案数
- https://qoj.ac/problem/1351

Solution

- 给每条非树边一个随机权值，做 xor hash
- 考虑所有非树边，如果非树边和树边组成奇环，就把它加进集合 S
- 这样做完之后，考虑图上所有边权的 xor 和，它一定等于 S（奇环对所有边权贡献了奇数次）
- 而对于二分图，所有边权的 xor 和 = 0
- 现在想割两条边使得这两条边的 xor = S，这样剩下的边 xor = 0（即为二分图）
- 割1条边：查询有没有 S
- 割2条边：枚举某一条权值 x，查询有没有 S xor x

边三连通分量

- 两个 u,v 在同一个边三连通分量的充要条件是：不能割两条边 e1,e2 使得 u,v 不连通。
- 运用上述切边等价的性质，我们可以得到边三连通分量的求法：
- 把存在异或哈希值相同的边切开，得到的每个连通块就是边三连通分量。
- 将图中每个边三连通分量缩点后，会得到一棵边仙人掌。
- 仙人掌上的每一个环是异或哈希值相同的边。

CF1648F Two Avenues

- 给定任意无向图，n 个点 m 条边，q 对关键点 (x_i,y_i)。
- 你需要把恰好两条边标记为关键边，假设关键边边权为 1、普通边边权为 0，在这个图上求出 dis(x_i,y_i) 的总和。
- 计算在任意选择两条关键边时，上式最大值。输出方案。
- n,m,q<=500000，8s

Solution

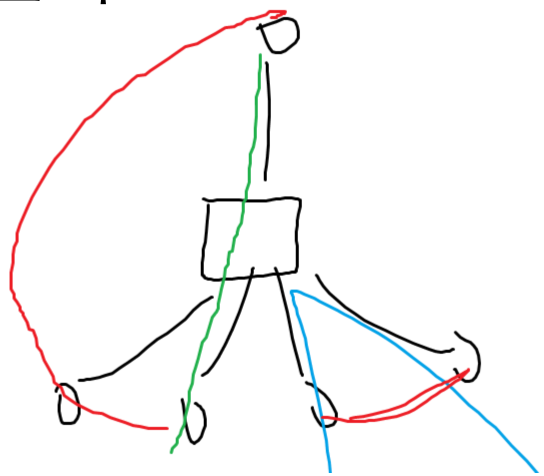
- 这题有很多种做法
- 考虑缩边三后，会形成边仙人掌:
- 割两条树边：贡献容易计算
- 割两条环边：需要在同一个环上
- 在环上枚举某条边删的位置，另一条边的位置有决策单调性
- 使用二位数点可以做到 log^2

Solution

- 每一个点对 (xi,yi) 对应环上的一段区间
- 考虑枚举第一条环边，用线段树维护割另一条边位置会增加多少，那就是一堆区间加，全局单点最大值
- 扫描一下第一条环边的位置即可维护
- 但如果每个环都把 q 个区间进行一下修改，复杂度不能接受

Solution

- 考虑一个经典的"挂在 LCA 处"优化:
- 建立圆方树，把每个 (xi,yi) 挂在它们的 LCA 的方点上
- 此时再考虑每个方点，可能的不同覆盖区间有两种:
- 1. (xi,yi) 的 LCA 是它，区间是任意的，为 q 个
- 2. 区间每个儿子圆点到它的父亲圆点，只有 儿子数量 个
- 这样修改数降到 O(n+q),时间复杂度 O((n+q)logn)



[PA 2020] Trzy drogi

- 给出一张无向图，求有多少种删去3条边的方案，使得图不连通。
- https://www.luogu.com.cn/problem/P9105
- n<=200000,m<=500000

k-edge-connected components

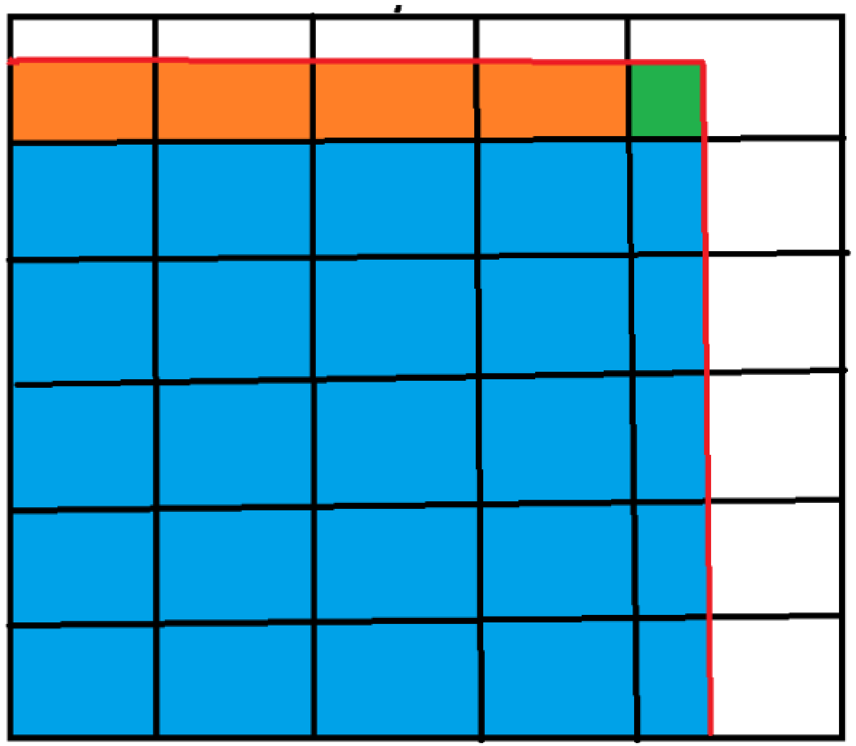
- 上一题 Sol: https://www.cnblogs.com/Rainbowsjy/p/17305030.html
- 上一题中给出的做法可以求出边四连通分量
- Determining 4-edge-connected components in linear time, https://arxiv.org/abs/2105.01699
- 边五连通分量的求法（Computing the 5-Edge-Connected Components in Linear Time, https://arxiv.org/abs/2311.04865）

有向图上可达性问题

- 给出一张 DAG
- 有 q 次询问，每次询问区间 [l,r]，求有多少个点对 (u,v) 满足：u 可达到 v，l<=u,v<=r。
- n<=100000,m<=200000,8s
- https://www.luogu.com.cn/problem/P7349

Solution

- 一看就只能 bitset 统计
- 对于蓝色部分可以直接 bitset，然后前缀和
- 橙色部分是转置，可以建反图跑 bitset
- 绿色部分暴力 $O(w)$ 次 popcount
- 时间复杂度 $O(n^2/w+qw)$



CFgym103119K Candy Ads

- 有 n 块广告牌，每块会覆盖 [x1,x2]-[y1,y2] 的矩形范围
- 你需要选择一些广告牌让它们出现，但广告牌不能互相覆盖
- 有 m 个限制，要求广告牌 a 和 b 至少要有一个出现
- n<=50000，坐标范围 <=2000
- https://codeforces.com/gym/103119/problem/K

Solution

- 2-SAT，对每个广告牌建立两个点 选/不选
- 对于每个广告牌，可以 bitset 处理出它和哪些广告牌相交
- 然后要跑强连通分量
- Tarjan 不能 bitset 优化，怎么办？
- Kosaraju 只用了 dfs 能到哪些点，可以 bitset 优化！

广义串并联图

- 对于图中任意四个不同的点 A,B,C,D，**都不能找到六条路径，使得其中任意两条路径除公共端点外没有公共点**，且这六条路径分别连接四个点中的每一对点——AB,AC,AD,BC,BD,CD。（没有与 K4 同胚的子图）
- 满足上述性质的图为广义串并联图

广义串并联图

- 连通的广义串并联图可以通过以下操作将图变成一个点:
- 删一度点（rake）
- 缩二度点（compress）
- 叠合重边（twist）

[SNOI2020] 生成树

- 给定一张图，已知图是仙人掌上加一条边。
- 求图的生成树个数

UER #3 开学前的涂鸦

- 给定一棵 n 个点的树，以及额外的 k 条边
- 求保留一个边的子集，使得图仍然连通的方案数
- n<=100000,k<=10
- https://uoj.ac/problem/138

Solution 1

- 用随机赋权 xor hash 后，把边分成等价类
- 枚举哪些等价类中的边要割掉，如果有一个 xor 为 0 的子集则不可行
- 等价类可能有很多，但是如果爆搜+用线性基判定有没有 xor=0，就能过

Solution 2

- 来自： https://hehezhou.blog.uoj.ac/blog/7446
- 先对原图做广义串并联图收缩，使得每个点 deg>=3
- 此时 m>=3n/2, m<=n+k-1, n<=2k-2<=18
- 新图的每条边有断开方案数 d[u][v] 和连接方案数 c[u][v]

Solution 2

- 对新图状压 DP
- 设 f[s] 表示 s 子集连通的方案数，g[s] 表示 s 子集任意的方案数
- g[s] = \prod {s 拆分成 T1,T2..Tk} f[Ti] * d[u][v] (u,v 属于不同的 Ti)
- 设 D[s] 表示 \prod d[u][v] (u,v in S)
- 再设 f'[s] = f[s]/D[s], g'[s] = g[s]/D[s]
- g'[s] = \prod {s 拆分成 T1,T2..Tk} f'[Ti]

Solution 2

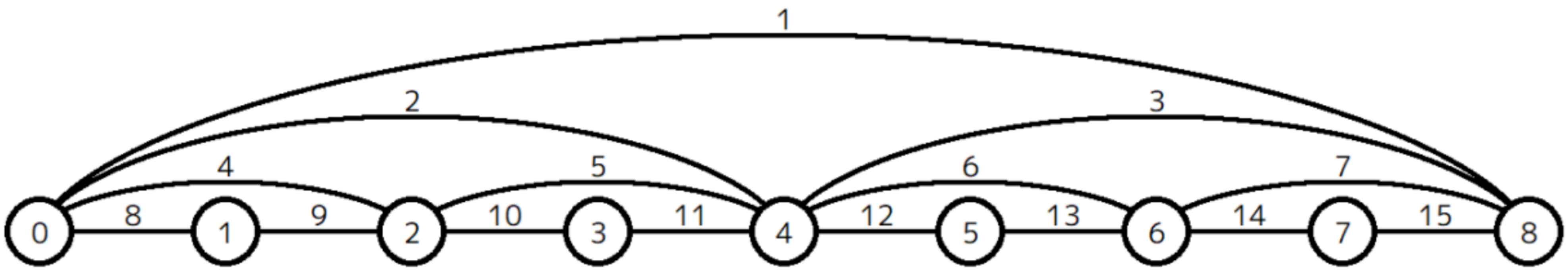
- 从集合幂级数考虑：g’ = exp(f’) , f’ = ln(g’)
- 而 g[s] 是容易求的
- 做集合幂级数 ln，时间复杂度 O(n+k^2 4^k) （新图的点数为 2k-2）

“广义串并联树”

- 上题的 DP 过程可以看作 cluster 的合并
- 这个合并过程可以建树，DP 就是在树上从下到上合并
- 其实是对广义串并联图建出了 Top Tree
- 在这个树的结构上，我们可以做动态 DP 等操作

TTPC2024_1_c Segment Tree

- 给定一棵类似线段树的图
- 支持修改边权，查询两点间最短路
- https://qoj.ac/problem/9877



Solution

- 显然是广义串并联图
- 收缩后建出的广义串并联树高 log
- 暴力枚举祖先的界点即可

还有什么图是广义串并联图？

- 三角剖分图
- 笛卡尔树式建图（一般用于优化建图后）
- 例子：JOISC2017 火车旅行

树分解

- 定义无向图 G=(V,E) 的一个树分解为:
- 构建一棵新树 (V',E')
- 对于树上的每个点，存储一个点集 X_i，X_i 中的每个点对应原图上的点，并且 X_i 的并集为 V（称点集 X_i 为这个点的 bag）
- 对于原图上的任意一条边 (u,v) ∈E，存在一个新树上的 bag X_i，使得 u,v ∈ X_i
- 对于原图上的任意一个点 u，它存在于新树上的若干个 bag 中，这些 bag 在新树上必须是一个连通块

有界树宽

- 一个树分解的宽为 max(|X_i|)-1
- 图 G 的树宽 tw(G) 为所有 G 的树分解中最小的宽。
- k-树是由 k+1 个点的完全图，进行若干次加点，满足每次添加的点恰好与之前的 k 个点之间有边，且这 k 个点之间两两有边，生成的图。
- tw(G)<=k 当且仅当 G 为某一个 k-树的子图。

树分解的应用

- 在树宽 k 较小的情况下，我们将图上的问题转化为树上的问题，从而解决一些图上难解的问题。
- 例如：
- 求图上最大独立集/最大团：
- 从下到上树形 DP，在每个点状压 bag 中的点是否选
- 时间复杂度 O(n*2^k*poly(k))

树分解的应用

- 多次询问，求图上任意两点间最短路：
- 考虑对树进行点分治，对于每一层分治，树形 DP 预处理所有点（的所有 bag 中的点）到分治中心的 bag 中的点的最短距离。
- 查询时，由于 u,v 之间的路径一定经过某个分治中心的 bag，枚举分治中心即可
- 实际上选任意子树中包含 u,v 的一个分治中心即可。
- 时间复杂度 O(n*logn*poly(k)+q*k)
- 一份代码实现：https://uoj.ac/submission/622306

特殊图的树宽

- 广义串并联图的树宽不超过 2。
- Halin graphs，Apollonian networks 的树宽不超过 3。
- 思考题:
- 如何构造广义串并联图的树分解?
- 假设对每个边有一个 bag {u,v}
- 对于每次缩二度点操作，假设缩的是 u-x-v，那么建立一个新的 bag {u,x,v},{u,v}，然后把 {u,x},{x,v},{u,v} 都连到 {u,x,v} 上

特殊图的树宽

- 如何构造 Halin graphs 的树分解（ https://qoj.ac/problem/6103 ）
- Halin graphs：将一棵树的叶子按 dfs 序排成一个序列 u_1,u_2..u_k，连边所有的 (u_i,u_i+1) 以及 (u_1,u_k) 得到的图
- 考虑 dfs 子树，从下到上，每次返回 bag {u,l,r}（l,r 为两端的叶子）
- 合并子树时会产生大小为 4 的 bag
- 可以解决：https://www.luogu.com.cn/problem/P11342

【集训队互测2021】逛公园

- 给出带边权的广义串并联图
- Q次询问，每次询问一个点集，求点集内两两最短路之和
- https://uoj.ac/problem/598

Solution

- 建立树分解，考虑用点分治做最短路的做法
- 在分治中心做掉所有经过它的点对贡献，剩下的递归子树
- 贡献的形式是 min(f[u][i]+f[v][i])，其中 1<=i<=k，k 为 bag 大小
- 广义串并联图中 k<=3，贡献即为若干次二维数点
- 时间复杂度 O(n log n+S log S log n)
- 一份代码实现：https://uoj.ac/submission/622380

任意图的树分解

- 存在 O(n log^2 n) 算法判断对于任意常数k，判断G的树宽是否不超过k，如果是则给出宽不超过k的树分解。
- Better algorithms for the pathwidth and treewidth of graphs. In ICALP, 1991.

Thanks for listening!

欢迎联系:

QQ：2453632155

https://cnblogs.com/Rainbowsjy