组队作业1报告：支配树

515030910627_方博慧，515030910631_秋闻达，515030910636_徐值天

2016年4月5日

1 功能介绍

Flow Graph：有向图$G=(V,E,r)$，且r可以到达图中所有点。如果r到w的任何路径都包含点v，则称点v支配点w $(w \neq v)$。如果点v支配点w，且其他支配w的点都支配v，那么v即为w的最近支配点，记作直接支配点$idom[w]$。每个点仅有一个直接支配点。显然，如果每个点和他的直接支配点连边，就构成了一个树结构，称为支配树(Dominator Tree)。

Tarjan算法的主要功能就是对于给定的Flow Graph G，求出它的支配树。

2 原理说明

Tarjan算法主要基于dfs思想。

下面是大致的原理：先进行dfs，按dfs序求出dfn数组，并找到一棵搜索树T，然后利用半必经点的性质求出每个点的半必经点，利用半必经点与直接必经点的联系，计算出每个点的直接必经点。

为了简单起见，我们定义点的大小关系就是点dfn值的大小关系。

给出半必经点的定义：点w的半必经点记为$sdom[w]$，$sdom[w]=min\{v|$有一条路径$v=v_0,v_1,v_2,...,v_k=w$且对$1\leq i<k$有$v_i>w\}$

3 正确性分析

主要证明半必经点定理和必经点定理。“$u\rightarrow v$”表示u是v在搜索树上的祖先。

3.1 半必经点定理

任意点$w \neq r$，$min(\{v|(v,w) \in E\, and\, v < w\} \bigcup \{sdom[u]|u > w, u$为$v$祖先且存在$(v,w) \in E\})$

设右式为x，则只需证明$sdom(w) \geq x$且$sdom(w) \leq x$即可。

3.1.1 sdom(w) $\leq$ x

若x属于第一个集合$\{v|(v,w) \in E\, and\, v < w\}$，则由定义可知，$sdom(w) \leq x$。

若x属于第二个集合$\{sdom[u]|u > w, u$为v祖先且存在$(v,w) \in E\}$，此时$x = sdom(u)$，则会有一条路径从$x = sdom(u)$到u，其中经过的非端点的点$v_i > u > w$。而从u到v的树边必然满足，从u到v的父亲节点$v_j \geq u > w$。所以我们得到了一条从x到w的路径，路径上每个点都大于w，则$sdom(w) \leq x$。

3.1.2 sdom(w) $\geq$ x

考虑从$sdom(w)$到w的路径，设路径为v_0到v_k。

如果中间不经过其他的点，即$k = 1$，则$(sdom(w), w) \in E$，就有$sdom(w) \geq$"第一个集合的x"。如果期间至少经过一个点不是端点，即$k > 1$，则考虑从$sdom(w)$到w的路径上，某个编号最小的v_j，满足$v_j \to v_{k-1}$（j永远存在，因为$v_{k-1} \to v_{k-1}$）。此时有"从v_1到v_{j-1}，每个点都大于v_j"，不然可选那一个点来代替v_j。此时我们找到了中间点v_j，则$sdom(w) \geq sdom(v_j) \geq x$。

$sdom(v_j) \leq sdom(w)$由$v_1..v_{j-1} > v_j$可知。

$x \leq sdom(v_j)$由v_j已经是最小值可得。

3.2 必经点定理

令$path$为$sdom(w)$到w的路径除$sdom(w)$外的点集，u为$path$中$sdom$最小的那一个点。那么

$$idom(w) = \begin{cases} sdom(w) & sdom(w) = sdom(u) \\ idom(u) & sdom(w) > sdom(u) \end{cases}$$

3.2.1 $sdom(w) = sdom(u)$

先证明$sdom(w)$一定能支配w：对于所有r到w的路径，找到最后一个点x使得$x < sdom(w)$，如果找不到则说明$sdom(w) = r$，显然能支配w。再找到路径上x之后的第一个点y，使得树边上$sdom(w)$可以走到y再走到w。截取r到w的这条路径上x到y的一段q，q上除端点x，y外其余点都大于y，否则，x，y中至少有一个不满足选取的条件。所以$sdom(y) \le x < sdom(w)$，所以$y = sdom(w)$（否则的话，y在p中且$sdom(y) < sdom(w)$，与假设不符）。所以任意r到w的路径都能找到y，也就是$sdom(w)$，即，$sdom(w)$支配w。

再证明$sdom(w) = idom(w)$：因为$idom(w) \le sdom(w)$，且$sdom(w)$已经支配w了，根据$idom$的定义，$idom(w) = sdom(w)$。

3.2.2 $sdom(w) > sdom(u)$

先证明$idom(u)$一定能支配w：对于所有r到w的路径，找到最后一个点x使得$x < idom(u)$（如果找不到则说明$idom(u) = r$，显然能支配w），再找到路径上x之后的第一个点y，使得树边上$idom(u)$可以走到y再走到w。类似上一种情况，x到y的一段q，q上除端点x，y外其余点都大于y，所以$sdom(y) \le x < idom(u)$，结合$idom$一定在$sdom$之上，所以$sdom(y) \le x < idom(u) \le sdom(u)$。根据$sdom(y) < sdom(u)$，$u$又是拥有最小$sdom$值的点，所以$y$不在$(sdom(w), w]$这一段中。又有$y$不在$(idom(u), u]$这一段中，否则的话，$r \to x \to y \to u$，不经过$idom(u)$，违反了$idom(u)$的定义。综上两点，只能有$y = idom(u)$。所以任意$r$到$w$的路径都能找到$y$，也就是$idom(u)$，即，$idom(u)$支配$w$。

再证明$idom(u) = idom(w)$：因为树边上$a \to b$，则$a \to idom(b)$或者$idom(b) \to idom(a)$，根据$u \to w$，$idom(w) \to sdom(w) \to u$，所以只能有$idom(w) \le idom(u)$。因为$idom(u)$支配$w$，所以$idom(w) = idom(u)$。

4 时空复杂度证明

假设Flow Graph G的点数为n，边数为m。

Tarjan算法首先做了一次dfs，时间复杂度为$O(n)$。之后按顺序枚举了每一个点。$idom[x]$仅仅在$sdom[x]$处计算，总时间是点数$O(n)$，而$sdom[x]$的计算只枚举了所有终点为x的边，总时间是边数$O(m)$。每次计算均使用

了并查集操作，时间复杂度是$O(\alpha(n))$。综上，总的时间复杂度是$O((n+m)\alpha(n))$。

可以看到，Tarjan算法需要的$dfn[]$，$idom[]$，$sdom[]$，以及并查集都是$O(n)$的空间复杂度。存储图所用的边表是$O(m)$的空间，为了在$sdom[x]$处计算$idom(x)$，我们使用了$vector$数据结构，所有点都只会被$push_back$一次，计算完成后就会pop_back，所以空间复杂度也是$O(n)$。综上，总的空间复杂度是$O(n+m)$。

5 具体实现方式

大致分为以下几个步骤：

5.1 DFS

利用dfs构建搜索树T，计算每个节点的时间戳dfn，重标号。

5.2 计算半必经点

根据半必经点定理，按照时间戳dfn从大到小的顺序计算每个节点的半必经节点。

5.3 计算必经点

根据必经点定理，按照时间戳dfn从小到大的顺序，把半必经节点不等于必经节点的节点进行修正。

6 细节

对于$min(\{v|(v,w)\in E\ and\ v<w\}\bigcup\{sdom[u]|u>w,u$为$v$祖先且存在$(v,w)\in E\})$，其中的$min\{v|(v,w)\in E\ and\ v<w\}$可以直接计算，而$min\{sdom[u]|u>w,u$为$v$祖先且存在$(v,w)\in E\})$则使用并查集来维护。

每次将算好的点的$sdom$值与其在搜索树上的父亲合并，并维护他到并查集树根路径上$sdom$的最小值。因为在计算$sdom$时，是按照时间戳dfn从大到小的顺序，所以对于之后计算的$sdom[w]$，枚举所有终点为w的边，对于起点v，因为树上每一条从上到下的链（也就是从某个点到他的某个子孙

的链）上的dfn值是单调的，所以符合条件的u是从v的某个祖先开始到v的这一段，也就是并查集中点v到并查集树根的路径。于是利用并查集里记录的值，就得到了$min\{sdom[u]|u > w, u$为v祖先$\})$。

这样就计算出了$min\{sdom[u]|u > w, u$为v祖先且存在$(v, w) \in E\})$，与之前结果合并就得到了$sdom[w]$。