

图应用

Kruskal算法：算法

11-F1

煮豆持作羹，漉菽以为汁
萁在釜下燃，豆在釜中泣
本是同根生，相煎何太急

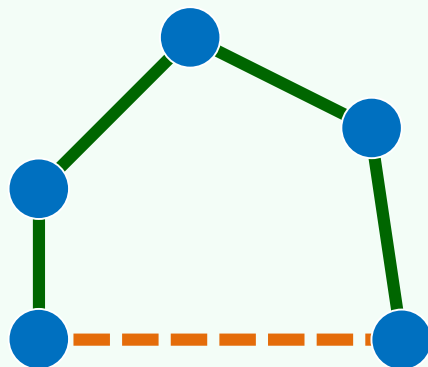
邓俊辉

deng@tsinghua.edu.cn

贪心策略

❖ 回顾Prim算法

- **最短边**，迟早会被采用
- **次短边**，亦是如此
- **再次短者**，则未必 //回路!

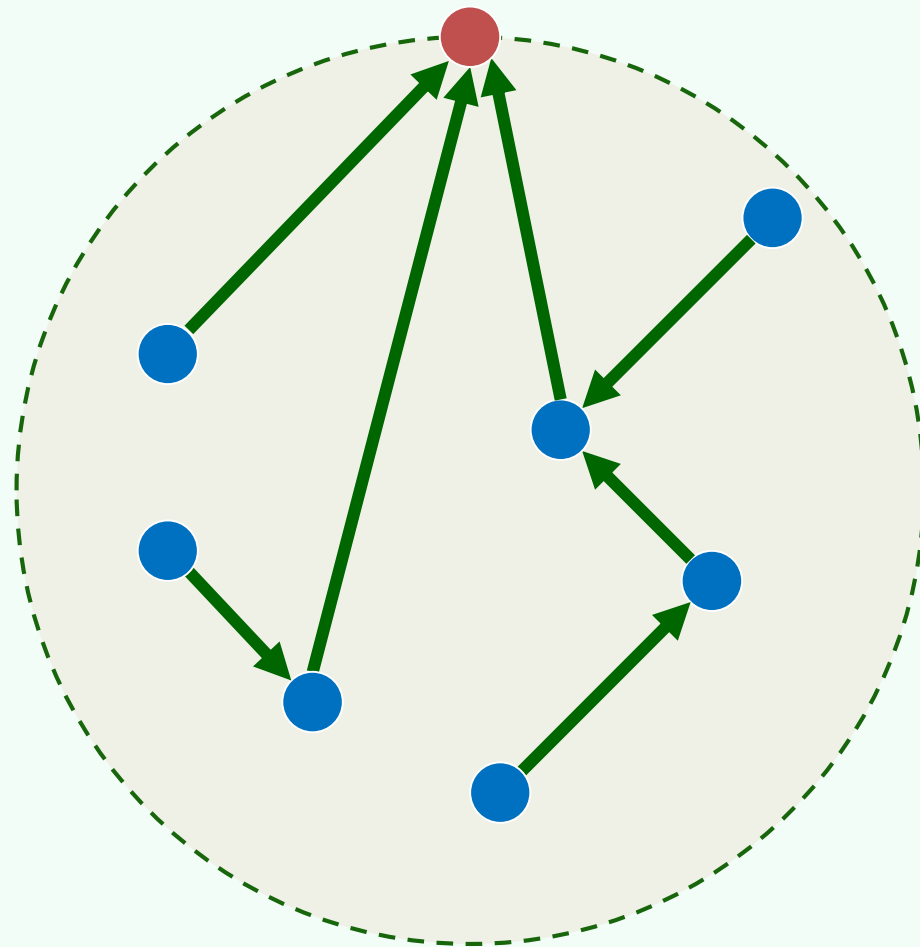


❖ Kruskal: 贪心原则

- 根据代价，从小到大依次尝试各边
- 只要“安全”，就加入该边

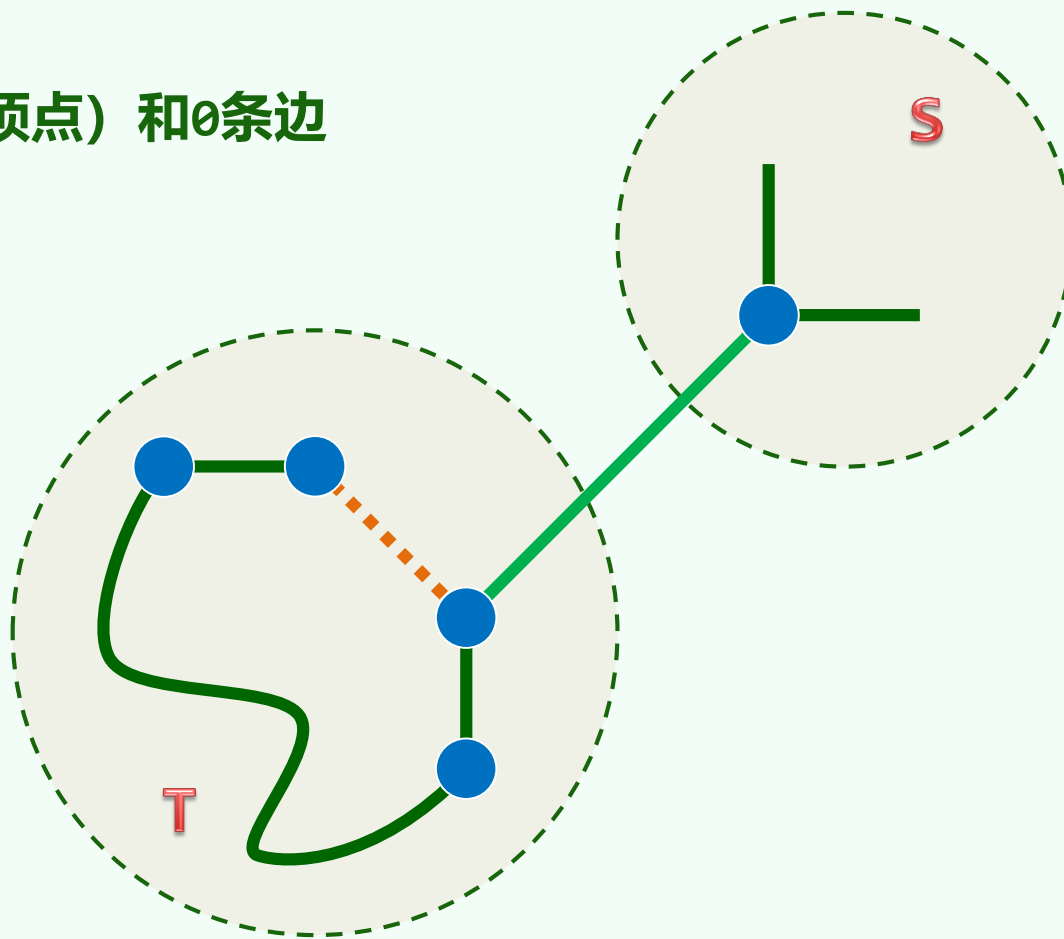
❖ 但是，每步局部最优 = 全局最优?

❖ 确实，Kruskal很幸运...



算法

- ❖ 维护 N 的一个森林: $F = (V; E') \subseteq N = (V; E)$
- ❖ 初始化:
 - $F = (V; \emptyset)$ 包含 n 棵树 (各含 1 个顶点) 和 0 条边
 - 将所有边按照代价排序
- ❖ 迭代, 直到 F 成为1棵树
 - 找到当前最廉价的边 e
 - 若 e 的顶点来自 F 中不同的树, 则
 - 令 $E' = E' \cup \{e\}$, 然后
 - 将 e 联接的2棵树合二为一
- ❖ 整个过程共迭代 $n-1$ 次, 选出 $n-1$ 条边



正确性

❖ 定理: Kruskal引入的每条边都属于**某棵**MST

❖ 设: 边 $e = (u, v)$ 的引入导致树 T 和 S 的合并

❖ 若: 将 $(T; V \setminus T)$ 视作原网络 N 的割

则: e 当属该割的一条**跨边**

❖ 在确定应引入 e 之前

- 该割的所有跨边都经Kruskal考察
- 且只可能因不短于 e 而被淘汰

❖ 故: e 当属该割的一条**极短跨边**

❖ 与Prim同理, 以上论述也不充分

为严格起见, 仍需归纳证明: Kruskal算法过程中不断生长的森林, 总是**某棵**MST的**子图**

