

链路状态路由算法

VS.

路由震荡



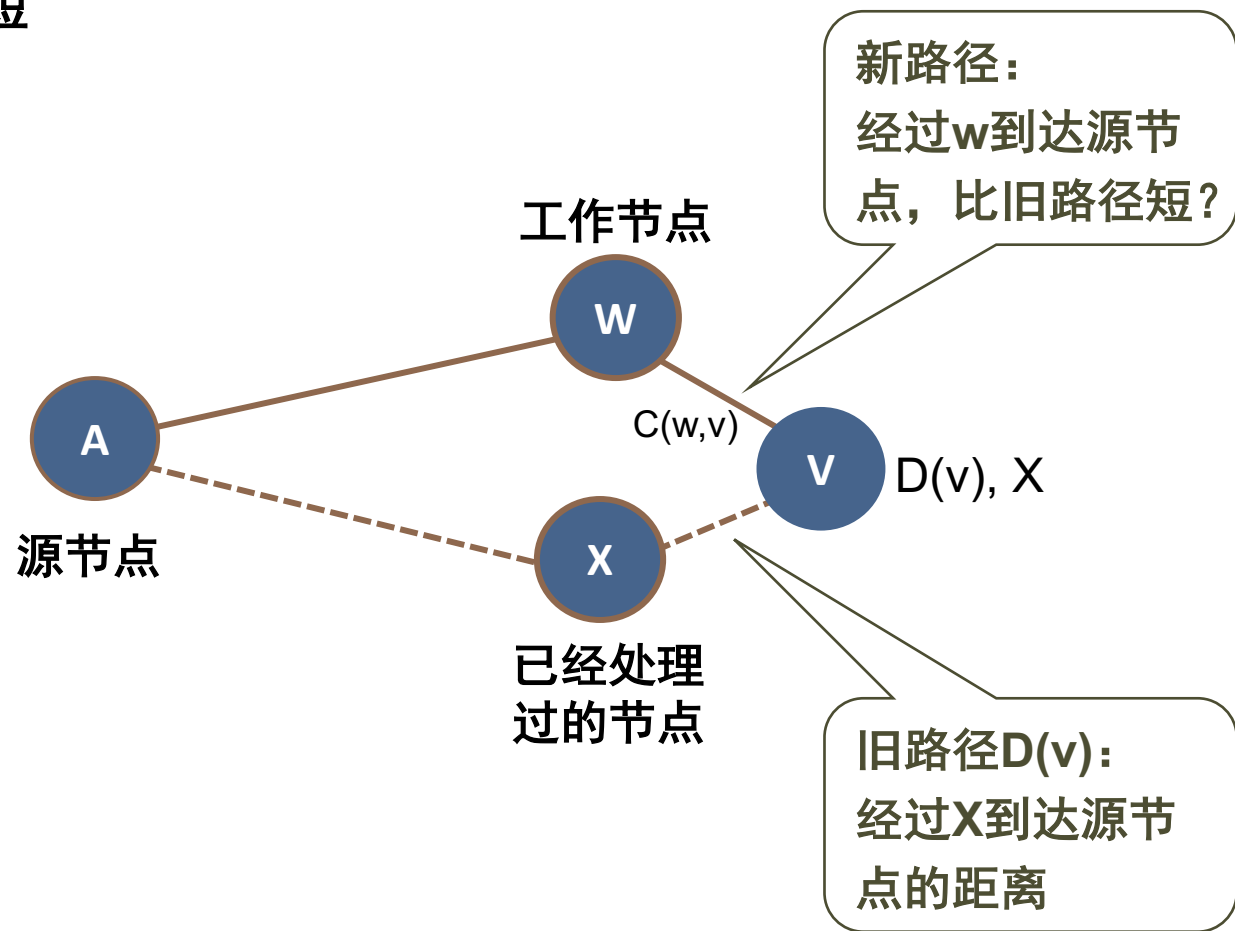
Dijkstra算法的迭代

在循环迭代中，算法试图找出所有节点到A的最短距离：

- 当前工作节点是w
- 检查所有w的邻居节点(例如v)，考察经过w到源节点的距离是否需要更新

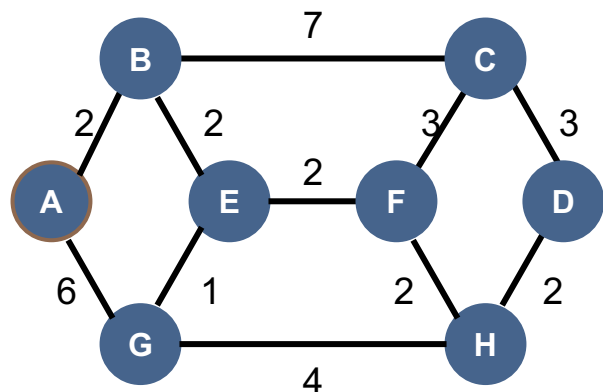
$$D(v) = \min\{ D(v), D(w) + C(w, v) \}$$

- $D(v)$ ：w的邻居通过其他节点到达源节点的距离
- $C(w, v)$ ：w和v的链路成本



通过迭代发现更好的路径

算法运行的基础：完整的网络拓扑信息

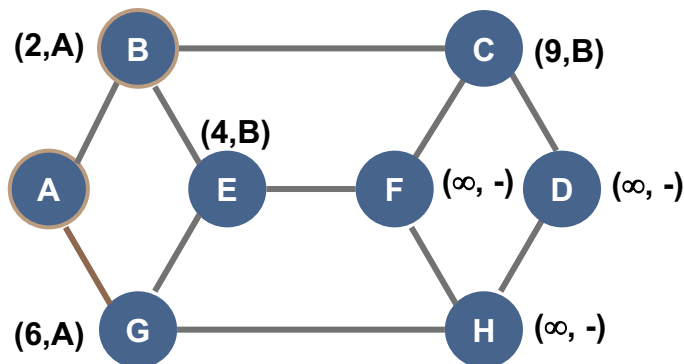


子网图

t1时刻:

- 节点G到源节点A的最短距离是6
- 从A出发到目标节点G的路径是：A-G

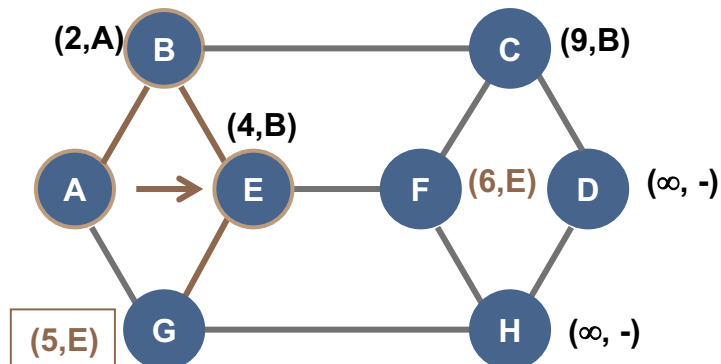
路径A-G，长度6



t2时刻:

- G通过邻居E到A的距离为E到A的距离加上G到E的链路成本，长度为5
- 更新从A出发到目标节点G的路径：A-B-E-G

新路径A-G，长度5



以链路时延作为链路成本

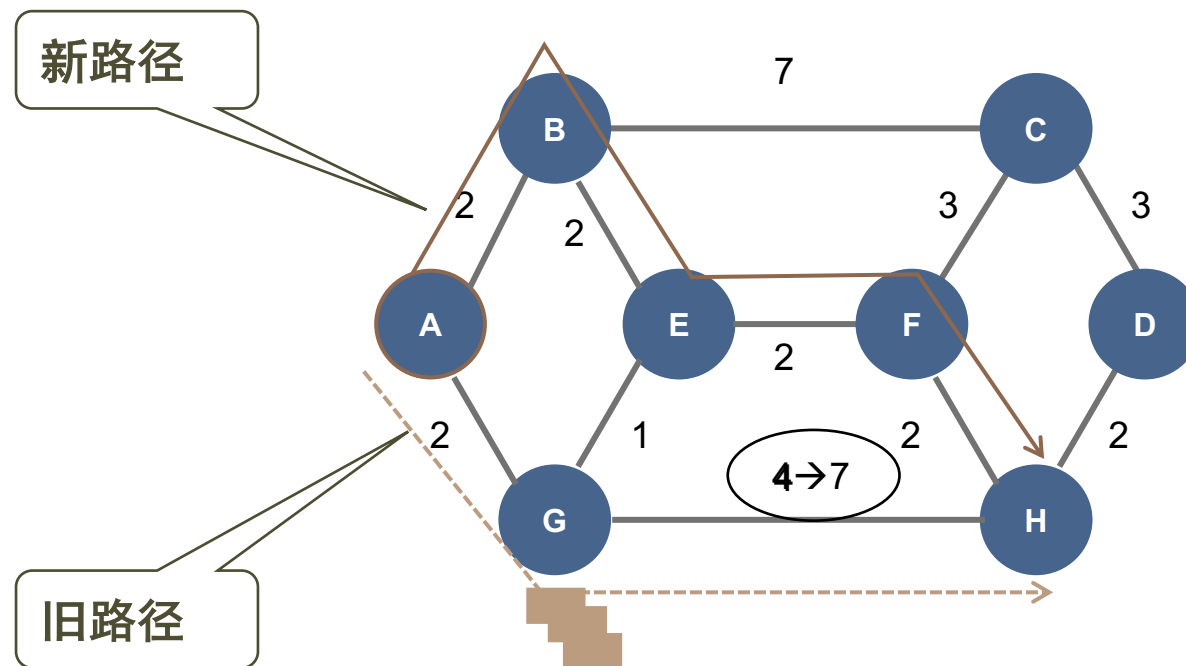
如果链路成本表示流量负载，那么通过ECHO消息测量链路成本时就考虑了网络流量负载这一因素。

优点

在相同链路条件下，选择轻负载的路径可减少排队时延

缺点

有可能造成路由来回震荡



- t0: A到H的最短路径选择了A-G-H
- t1: A到H的最短路径更新为A-B-E-F-H



基于链路状态的路由来回震荡

路由震荡：路由随着链路成本的变化来回切换。

- ① 初始时东西园区之间的包选择C-F（或E-I），所有包都通过C/F转发→C/F上的排队时延增加
- ② 路由更新时把路径切换至E-I（或C-F），所有的包都通过E/I转发→E/I上的排队时延增加
- ③
- ④ 两个校区之间的路由在路径1和路径2之间来回切换，带来了路由的不稳定

假设：某个大学由两个校区组成，如图建设了校园网。

- 校园网路由器采用了基于链路状态路由算法的路由协议
- 链路成本代表了链路时延

