# 作业

09019204 曹邹颖

1. 比较和对照集中式和分布式路由选择算法的性质，并给出一个路由选择协议的例子, 该路由选择协议分别采用分布式方法和集中式方法。

答：集中式路由选择算法是用完整的、全局性的网络知识计算出从源到目的地之间的最低开销路径，以所有节点之间的连通性与所有链路的开销为算法的输入，例OSPF协议。

分布式路由选择算法以迭代、分布式的方式计算出最低开销路径，每个节点仅有与其直接相连链路的开销知识即可开始工作，然后通过迭代计算过程以及与相邻节点的信息交换，一个节点逐渐计算出到达某目的节点或一组目的节点的最低开销路径，例距离向量（DV）算法。

标答：

集中式方法需要用完整的、全局性的网络信息来计算从源到目的之间的最短路径，也就是需要知道网络中每条链路的费用。在LS算法中，每个节点通过广播与所有其他节点交换信息，但它仅告诉它们与它直接相连链路的费用。Dijkstra 算法计算从源节点到网络中所有其他节点的最低费用路径。Dijkstra 算法是迭代算法，经算法的第 k 次迭代后，可知道到 k 个目的节点的最低费用。常见的LS协议是OSPF协议。

分布式的路由算法通过迭代计算并与相邻节点交换信息，逐渐计算出到达某目的节点或一组目的节点的最短路径。在DV算法中，每个节点仅与它的直接相连邻居交换信息，但它为它的邻居提供了从自己到网络中（它所知道的）所有其他节点的最低费用估计。常见的协议是RIP协议。

1. 为什么在因特网中用到了不同的AS间与AS内部协议? 分别列举出AS间的路由协议和AS内的路由协议。

答：因为在AS之间选择路由，策略问题起主导作用，所用路由的质量（如性能）通常 是次要关心的问题，而在AS内部，一切都是在相同的管理控制名义之下进行的，策略 问题起着微不足道的作用，反而更多地关注路由的性能。同时，可扩展性在AS间路由 选择中也是一个关键问题，但一个AS内，若变得太大可以将其分为两个AS，并在两个 新的AS之间执行AS间路由选择，所以AS内可扩展性不是关注的焦点。综上，AS内与 AS间路由选择目标之间有众多差别，为此要采用不同的AS间与AS内部协议。

AS间的路由协议：OSPF

AS内的路由协议：BGP

标答：由于路由器数量太多了，各个单位又希望自己管理自己的设备，因特网使用了层次路由策略，也就是将这些路由器划分成自治系统AS(Autonomous System)来进行管理。类似于人类社会的国家，一个国家一般使用相同的法律。在AS之间，策略问题起主导作用。一个给定的AS产生的流量不能穿过另一个特定的AS；一个特定AS或者要控制它承载的其他AS之间穿越的流量。而在AS内部，有着统一的管理，策略问题在AS内部选择路由中起着微不足道的作用。

AS内部路由选择协议：RIP，开放最短路优先（OSPF）协议

AS间路由选择协议：边界网关协议（BGP），BGP包括iBGP和eBGP，iBGP用于在域内传递广域路由信息，而eBGP用于计算域间路由。

1. R8. 是否判断题: 当一台OSPF路由器发送它的链路状态信息时, 它仅向那些直接相邻的节点发送. 解释理由

R8. True or false: When an OSPF route sends its link state information, it is sent only to those

nodes directly attached neighbors. Explain.

答：错误，因为不仅仅是向那些直接相邻的节点发送，而是所有其他节点都接收到它的 链路状态信息。OSPF是一种链路状态协议, 使用洪泛链路状态信息和Dijkstra最低开销 路径算法，通过让每个节点向AS内所有其他节点广播链路状态分组，来完成已知所有 链路开销与网络拓扑，从而用作链路状态路由选择算法的输入。

标答：错误。OSPF是一种链路状态协议, 使用洪泛链路状态信息和Dijkstra算法，注意Dijkstra是传输链路距离给邻居的。运行OSPF时, 路由器向自治系统内所有其他路由器广播路由选择信息, 而不仅仅是向相邻路由器广播.

1. 在发送主机执行Traceroute程序, 收到哪两种类型的报文? Traceroute的过程是怎样的？

答：ICMP“TTL过期”告警报文（类型11编号0），ICMP“目的端口不可达”报文（类 型3编号3），Traceroute的过程：源主机向目的主机发送一系列IP数据报，每个数据 报里都携带了一个具有不可达UDP端口号的UDP报文段，第一个数据报的TTL为１， 第二个数据报的TTL为２，第三个数据报的TTL为３，依次类推。当第ｎ个数据报到达 第ｎ台路由器时，第ｎ台路由观察到该数据报正好TTL过期，从而丢弃该数据报并发送 一个“TTL过期”告警报文（类型11编号0）给源主机，里面包含了路由器名字和它的 IP地址。直到数据报之一最终到达目的主机，返回一个“目的端口不可达”报文（类型 3编号3），此时源主机不再发送另外的探测分组。

标答：ICMP警告报文和端口不可达的ICMP报文。

Traceroute过程：为了判断源和目的地之间所有路由器的名字和地址，源主机中的Traceroute向目的地主机发送一系列普通的IP数据报。这些数据报每个携带了一个具有不可达UDP端口号的UDP报文段。第一个数据报TTL为1，第二个的TTL为2，以此类推。该源主机为每个数据报启动定时器。当第n个数据报到达第n台路由器时，第n台路由器观察到这个数据报的TTL正好过期。根据IP协议规则，路由器丢弃该数据报并发送一个ICMP警告报文给源主机。该警告报文包含了路由器的名字和它的IP地址。当该ICMP报文返回源主机时，源主机从定时器得到往返时延，从ICMP报文中得到第n台路由器的名字与IP地址。

由于TTL逐次加1，因此有数据报最终到达目的主机。因为该数据报包含了一个具有不可达端口号的UDP报文段，该目的主机将向源发送一个端口不可达的ICMP报文。当源主机收到该特别的ICMP报文时，知道它不需要再发送另外的探测分组。

1. 执行ping命令时，调用了什么协议？ping的功能是什么？

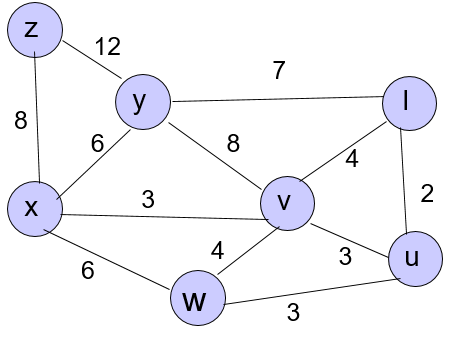
答：调用了ICMP协议，ping的功能是查看一个主机是否可达。

标答：ping使用了ICMP协议，ping是专门用来测试网络点对点传输速度及点对点传输是否能够互连互通。即：你的电脑发送四个数据包，发送到对方IP，对方再返回告诉你它收到你的数据，这个数据一来一回的时长，就是连接的延时，单位一般是毫秒。

1. 考虑下面的网络。使用Dijkstra算法和一个类似于5-1的表来说明你做的工作：

A 计算出从x到所有网络节点的最短路径。

B 计算出从u到所有网络节点的最短路径。



答：

1. 从x开始

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | z | y | w | v | u | I |
| 1 | 8 | 6 | 6 | 3 | ∞ | ∞ |
| 2 | 8 | 6 | 6 |  | 6 | 7 |
| 3 | 8 |  | 6 |  | 6 | 7 |
| 4 | 8 |  |  |  | 6 | 7 |
| 5 | 8 |  |  |  |  | 7 |
| 6 | 8 |  |  |  |  |  |

从x到v的最短路径：3；

从x到y的最短路径：6；

从x到w的最短路径：6；

从x到u的最短路径：6；

从x到I的最短路径：7；

从x到z的最短路径：8；

1. 从u开始

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | I | v | w | x | y | z |
| 1 | 2 | 3 | 3 | ∞ | ∞ | ∞ |
| 2 |  | 3 | 3 | ∞ | 9 | ∞ |
| 3 |  |  | 3 | 6 | 9 | ∞ |
| 4 |  |  |  | 6 | 9 | ∞ |
| 5 |  |  |  |  | 9 | 14 |
| 6 |  |  |  |  |  | 14 |

从u到I的最短路径：2；

从u到v的最短路径：3；

从u到w的最短路径：3；

从u到x的最短路径：6；

从u到y的最短路径：9；

从u到z的最短路径：14；

答：

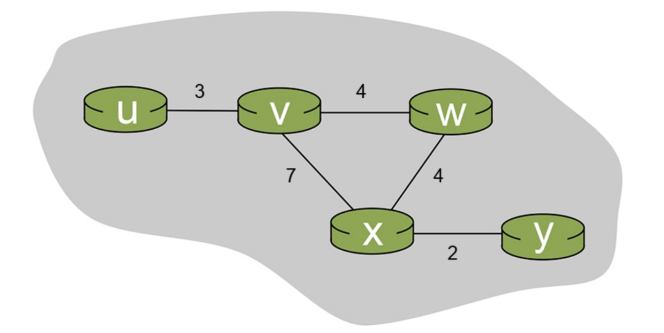
A:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 步骤 | N’ | D(v),p(v) | D(w),p(w) | D(l),p(l) | D(u),p(u) | D(y),p(y) | D(z),p(z) |
| 0 | x | 3,x | 6,x | ∞ | ∞ | 6,x | 8,x |
| 1 | xv |  | 6,x | 7,v | 6,v | 6,x | 8,x |
| 2 | xvw |  |  | 7,v | 6,v | 6,x | 8,x |
| 3 | xvwu |  |  | 7,v |  | 6,x | 8,x |
| 4 | xvwuy |  |  | 7,v |  |  | 8,x |
| 5 | xvwuyl |  |  |  |  |  | 8,x |
| 6 | xvwuylz |  |  |  |  |  |  |

B:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 步骤 | N’ | D(v),p(v) | D(w),p(w) | D(l),p(l) | D(x),p(x) | D(y),p(y) | D(z),p(z) |
| 0 | u | 3,u | 3,u | 2,u | ∞ | ∞ | ∞ |
| 1 | ul | 3,u | 3,u |  | ∞ | 9,l | ∞ |
| 2 | ulv |  | 3,u |  | 6,v | 9,l | ∞ |
| 3 | ulvw |  |  |  | 6,v | 9,l | ∞ |
| 4 | ulvwx |  |  |  |  | 9,l | 14,x |
| 5 | ulvwxy |  |  |  |  |  | 14,x |
| 6 | ulvwxyz |  |  |  |  |  |  |

1. 考虑下图所示的网络，假设每个节点初始时知道它的每个邻居的开销。使用距离向量算法，并显示每次迭代后节点u的距离表表项。



答：算法的运行以**同步**的方式显现出来，其中所有节点**同时**从其邻居接收报文，计算 其新距离向量，如果距离向量发生了变化则通知其邻居。

每次迭代后节点u的距离表表项：

到...开销

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | u | v | w | x | y |
| u | 0 | 3 | ∞ | ∞ | ∞ |
| v | ∞ | ∞ | ∞ | ∞ | ∞ |

从

到...开销

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | u | v | w | x | y |
| u | 0 | 3 | 7 | 10 | ∞ |
| v | 3 | 0 | 4 | 7 | ∞ |

从

到...开销

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | u | v | w | x | y |
| u | 0 | 3 | 7 | 10 | 12 |
| v | 3 | 0 | 4 | 7 | 9 |

从