

第五章

R4. 比较和对照链路状态和距离矢量这两种路由选择算法。

假设一共有 n 个 路由器，

链路状态简称LS， 距离矢量简称DV

方面/算法	LS	DV
概述	以所有结点之间的连通性及所有链路的费用的完整信息为输入，计算本身可在某个场点进行或在多个场点重复进行。	没有结点拥有关于所有网络链路费用的完整信息，每个结点只和邻居交流，并维护到网络中所有其他结点的费用（距离）估计的向量，并向邻居提供这些向量。
报文复杂度	从某个点点亮全图，需要 $O(n^2)$ 报文，从每个点点亮全图，需要 $O(n^3)$ ；并且无论何时一条链路的费用发生变化，必须向所有结点发送新的链路费用。	只和邻居交流，复杂度低；并且仅当链路费用的改变引起该链路相邻结点之间的最路径发生变化时，才传播已改变费用的路径。
收敛速度	$O(n^2)$	收敛时间有变，总体比较慢 good news travels fast, bad news travels slow, 如果某条链路的开销突然有巨大增幅, 会造成路由选择环路, 被称为无穷计数
鲁棒性	强。路由器传播的是错误的链路开销；每个路由器只计算它自己的表。	弱。路由器传播的是错误的路径开销，产生黑洞效应；每个路由器的DV被其他人使用，导致错误通过网络传播

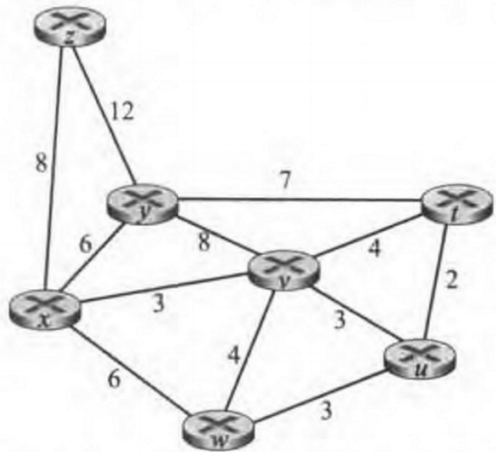
R6. 每个自治系统使用相同的 AS 内部路由选择算法是必要的吗？说明其原因。

不必要。

事实上，我们只需要保证相同AS中运行相同的域内选路算法协议，AS网关路由执行域间路由选路协议；

因为AS网关路由提供的不同AS间传输的选路算法，而不同的AS 中运行不同的内部路由选路算法不会影响到AS网关路由的选择。

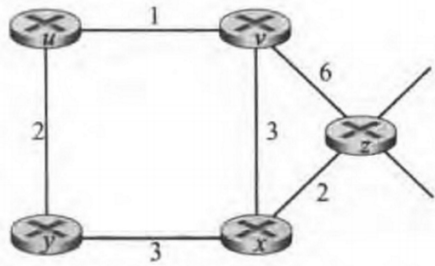
P3. 考虑下面的网络。对于标明的链路开销，用 Dijkstra 的最短路算法计算出从 x 到所有网络节点的最短路径。通过计算一个类似于表 5-1 的表，说明该算法是如何工作的。



步骤	N'	D(y), P(y)	D(z), P(z)	D(u),P(u)	D(v),P(v)	D(w),P(w)	D(t),P(t)
0	x	6,x	8,x	/	3,x	6,x	/
1	x,v	6,x	8,x	6,v		6,x	7,v
2	x,v,y		8,x	6,v		6,x	7,v
3	x,v,y,u		8,x			6,x	7,v
4	x,v,y,u,w		8,x				7,v
5	x,v,y,u,w,t		8,x				
6	x,v,y,u,w,t						

节点	最短路径	最短路径距离
y	x,y	6
z	x,z	8
v	x,v	3
u	x,v,u	6
w	x,w	6
t	x,v,t	7

P5. 考虑下图所示的网络，假设每个节点初始时知道到它的每个邻居的开销。考虑距离向量算法，并显示在节点 z 中的距离表表项。



黄色代表发生了变化，表格的左上角是表所属的节点名的大写。

time = 0

Z	x	y	z	u	v
x					
z	2		0		6

time = 1

Z	x	y	z	u	v
x	0	3	2		3
z	2	5	0	7	5

	x	y	z	u	v
x					
z	2		0		6
v					

x	0	3	2		3
z	2	5	0	7	5
v	3		6	1	0

V	x	y	z	u	v
x					
z					
u					
v	3		6	1	0

V	x	y	z	u	v
x	0	3	2		3
z	2		0		6
u		2		0	1
v	3	3	5	1	0

X	x	y	z	u	v
x	0	3	2		3
y					
z					
v					

X	x	y	z	u	v
x	0	3	2	4	3
y	3	0		2	
z	2		0		6
v	3		6	1	0

Y	x	y	z	u	v
x					
y	3	0		2	
u					

Y	x	y	z	u	v
x	0	3	2		3
y	3	0	5	2	3
u		2		0	1

U	x	y	z	u	v
y					
u		2		0	1
v					

U	x	y	z	u	v
y	3	0		2	
u	5	2	7	0	1
v	3		6	1	0

time = 2

Z	x	y	z	u	v
x	0	3	2	4	3
z	2	5	0	6	5
v	3	3	5	1	0

time = 3

Z	x	y	z	u	v
x	0	3	2	4	3
z	2	5	0	6	5
v	3	3	5	1	0

V	x	y	z	u	v
x	0	3	2	4	3
z	2	5	0	7	5
u	5	2	7	0	1
v	3	3	5	1	0

V	x	y	z	u	v
x	0	3	2	4	3
z	2	5	0	6	5
u	4	2	6	0	1
v	3	3	5	1	0

X	x	y	z	u	v
x	0	3	2	4	3
y	3	0	5	2	3
z	2	5	0	7	5
v	3	3	5	1	0

X	x	y	z	u	v
x	0	3	2	4	3
y	3	0	5	2	3
z	2	5	0	6	5
v	3	3	5	1	0

Y	x	y	z	u	v
x	0	3	2	4	3
y	3	0	5	2	3
u	5	2	7	0	1

Y	x	y	z	u	v
x	0	3	2	4	3
y	3	0	5	2	3
u	4	2	6	0	1

U	x	y	z	u	v
y	3	0	5	2	3
u	4	2	6	0	1
v	3	3	5	1	0

U	x	y	z	u	v
y	3	0	5	2	3
u	4	2	6	0	1
v	3	3	5	1	0

Over.

no changes over.

故而，各时间z的距离表表项为：

time=0

Z	x	y	z	u	v
x					
z	2		0		6
v					

time=2

Z	x	y	z	u	v
x	0	3	2	4	3
z	2	5	0	6	5
v	3	3	5	1	0

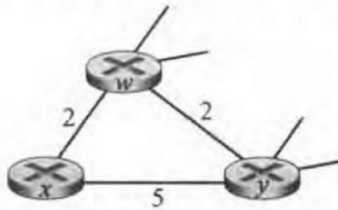
time=1

Z	x	y	z	u	v
x	0	3	2		3
z	2	5	0	7	5
v	3		6	1	0

time=3

Z	x	y	z	u	v
x	0	3	2	4	3
z	2	5	0	6	5
v	3	3	5	1	0

P7. 考虑下图所示的网络段。 x 只有两个相连邻居 w 与 y 。 w 有一条通向目的地 u （没有显示）的最低开销路径，其值为 5， y 有一条通向目的地 u 的最低开销路径，其值为 6。从 w 与 y 到 u （以及 w 与 y 之间）的完整路径未显示出来。网络中所有链路开销皆为正整数值。



a. 给出 x 对目的地 w 、 y 和 u 的距离向量。

节点	距离
w	2
y	5
u	7

b. 给出对于 $c(x, w)$ 或 $c(x, y)$ 的链路开销的变化，使得执行了距离向量算法后， x 将通知其邻居有一条通向 u 的新最低开销路径。

现在的路径为 $x-w-u$ ，想要 x 通知变化，则路径要有变化；

1. 如果减小 $c(x, y)$ ，由于有 $c > 0$ 的限制，去 $c(x, y)$ 最小为 1，最小费用的路径仍然是 $x-w-u = 7$ ，所以减小当

前链路中的费用，不会改变最小费用路径

2. 如果 $c(x, w)$ 增加到 7 时, $Dis(x - w - u) = 12 > Dis(x - y - u) = 11$, 因此 x 将通知其邻居有一条通向 u 的新最低费用路径为 $x - y - u$

- c. 给出对 $c(x, w)$ 或 $c(x, y)$ 的链路开销的变化, 使得执行了距离向量算法后, x 将不通知其邻居有一条通向 x 的新最低开销路径。

由b的分析我们可知, 想要不通知变化的情况:

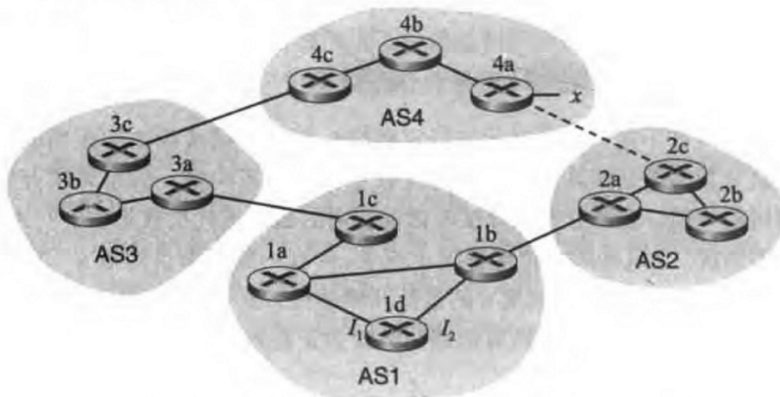
1. $c(x, w)$ 增加且 $c(x, w) < 7$,
2. $c(x, w)$ 减少
3. $c(x, y)$ 减少

P12. 描述在 BGP 中是如何检测路径中的环路的。

路由器通过BGP连接通告前缀时, 前缀包含"AS-PATH"属性, 这个属性中包含了已经通过的AS的列表, 而为了生成AS-PATH的值, 当一个前缀通过某个AS时, 该AS将其ASN加入AS-PATH, 如果一个路由器在AS-PATH列表中发现了自己的AS, 你们就检测出路径中有环路, 就拒绝该通告。

P14. 考虑下图所示的网络。假定 AS3 和 AS2 正在运行 OSPF 作为其 AS 内部路由选择协议。假定 AS1 和 AS4 正在运行 RIP 作为其 AS 内部路由选择协议。假定 AS 间路由选择协议使用的是 eBGP 和 iBGP。假定最初在 AS2 和 AS4 之间不存在物理链路。

- a. 路由器 3c 从哪个路由选择协议学习到了前缀 x ?
- b. 路由器 3a 从哪个路由选择协议学习到了前缀 x ?
- c. 路由器 1c 从哪个路由选择协议学习到了前缀 x ?
- d. 路由器 1d 从哪个路由选择协议学习到了前缀 x ?



- a. eBGP
- b. iBGP
- c. eBGP
- d. iBGP