

# Enrutamiento de vector de distancia

Cálculo de tabla de reenvío

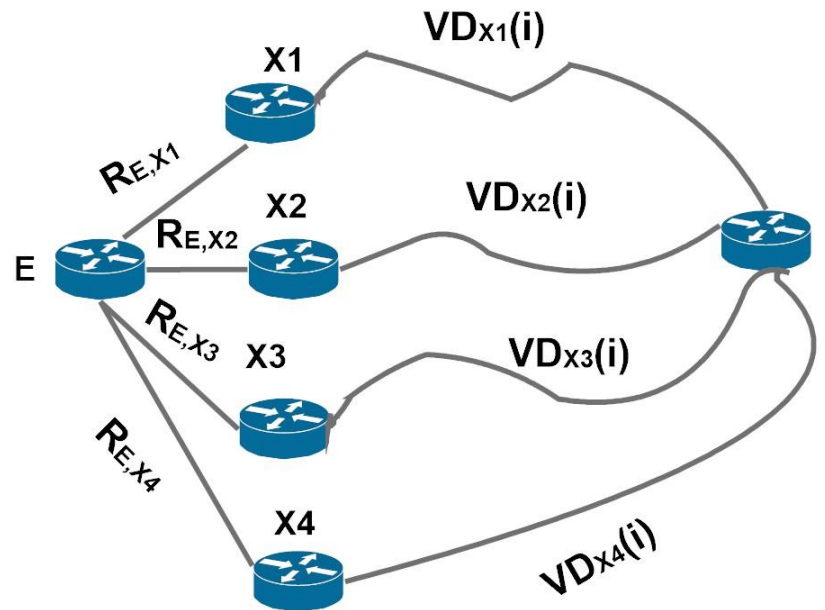
# Enrutamiento de Vector de Distancia

- Cada enrutador mantiene una **tabla de enrutamiento (o de reenvío)** indizada por cada enrutador en la subred.
  - ❑ Cada entrada comprende: la **línea preferida de salida hacia ese destino y una estimación del tiempo o distancia a ese destino.**
- A partir de su tabla de enrutamiento un enrutador  $E$  puede obtener **un vector de distancia** que contiene una lista de pares <destino, retardo estimado>
- El retardo de un enrutador a un vecino suyo, puede medirlo con
  - ❑ **paquetes de ECO** que el receptor simplemente marca con la hora y los regresa tan rápido como puede.

# Enrutamiento de Vector de Distancia

- Cada  $t$  mseg, cada enrutador envía a todos sus vecinos un vector de distancia y también recibe un vector de distancia de cada vecino.
- **Un poco de notación:**
  - ❑ El vector de distancia del enrutador  $X$  se denota con  $VD_X$ .
  - ❑  $VD_X$  es una función:  $VD_X(i)$  es la 'distancia estimada' para llegar al enrutador  $i$  desde  $X$ .
  - ❑ Si  $X$  vecino de  $E$ , el retardo de  $E$  a  $X$  se denota con  $R_{E,X}$ .
    - se usa paquete ECO para obtenerlo.
- Entonces la distancia estimada desde  $E$  enrutador a  $i$  a través de  $X$  es:  
$$R_{E,X} + VD_X(i).$$

# Enrutamiento de Vector de Distancia

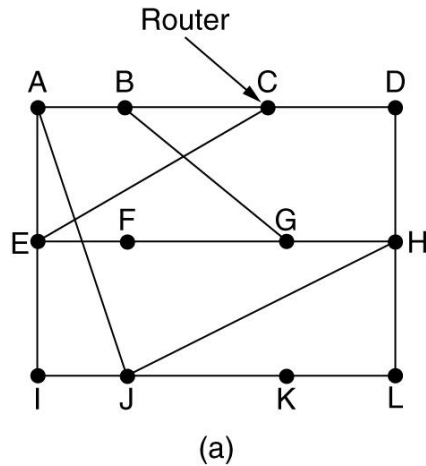


- Tengo estimación  $R_{E, X_n} + VD_{X_n}(i)$  de camino más corto de  $E$  a  $i$  que pasa por  $X_n$ , para todo  $n$  en  $\{1, \dots, 4\}$ .
- **¿cuál es la mejor de esas estimaciones?**
- Aquella que tiene el menor valor en  $\{R_{E, X_1} + VD_{X_1}(i), R_{E, X_2} + VD_{X_2}(i), R_{E, X_3} + VD_{X_3}(i), R_{E, X_4} + VD_{X_4}(i)\}$
- El vecino de  $E$  con la mejor de esas estimaciones conviene que sea la línea de salida a usar desde  $E$  para ir a  $i$ .

# Enrutamiento de Vector de Distancia

- El enrutador  $E$  estima la **distancia** desde  $E$  al enrutador de destino  $i$  de la siguiente manera:
  - $d(E, i) = \min\{R_{E,X} + VD_X(i) \mid X \text{ vecino de } E\}$
- El **mejor vecino** para ir de  $E$  a  $i$  se define como:
  - $MV(E, i) = \text{elegir } \{V : R_{E,V} + VD_V(i) = d(E, i)\}.$
  - *elegir* elige un elemento de un conjunto.
- ¿Cómo se actualiza **tabla de enrutamiento** de  $E$ ?
  - $E$  recibió de todo vecino  $X$  suyo:  $VD_X$  y  $R_{E,X}$
  - La tabla de enrutamiento de  $E$  en la fila del enrutador de destino  $i$  va a tener los valores:  $d(E, i)$  y  $MV(E, i)$ .
  - Observar que la vieja tabla de enrutamiento no se usa en este cálculo.

# Enrutamiento de Vector de Distancia



Se tienen vectores  $VD_A$ ,  $VD_I$ ,  $VD_H$  y  $VD_K$ . La última tabla es la de enrutamiento De J.

To	A	I	H	K	New estimated delay from J ↓ Line	
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K

JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6
---------------	----------------	----------------	---------------

Vectors received from J's four neighbors

New routing table for J	
8	A
20	A
28	I
20	H
17	I
30	I
18	H
12	H
10	I
0	–
6	K
15	K

(b)

(a) Una subred. (b) Input de A, I, H, K, y la nueva tabla de reenvío para J.

# Enrutamiento de Vector de Distancia

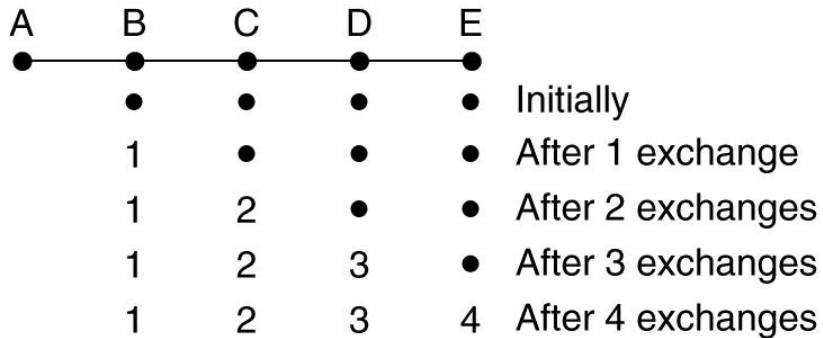
- **Evaluación del AEVD**
- Reacciona con rapidez a las buenas noticias, pero con lentitud ante las malas.

# Enrutamiento de Vector de Distancia

- Considere un enrutador cuya mejor ruta al destino  $X$  es larga. Si en el siguiente intercambio el vecino  $A$  informa repentinamente un retardo corto a  $X$ ,
  - ❑ el enrutador simplemente se conmuta a modo de usar la línea a  $A$  para enviar tráfico hasta  $X$ .
- Supongamos que la métrica de retardo es el número de saltos.
  - ❑ Las buenas noticias se difunden a razón de un salto por intercambio.
  - ❑ En una subred cuya ruta mayor tiene una longitud de  $N$  saltos, en un lapso de  $N$  intercambios todo el mundo sabrá sobre las líneas y enrutadores recientemente revividos.



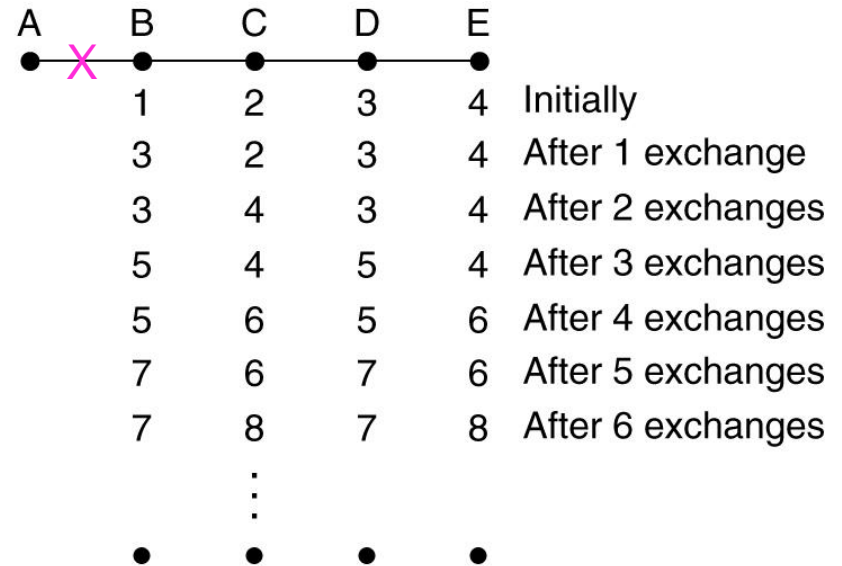
# Enrutamiento de Vector de Distancia



(a)

A volvió a la vida

Good news of a path  
to A spreads quickly



(b)

A se cayó

Bad news of no path to A  
is learned slowly

# Enrutamiento de Vector de Distancia

- La razón de porqué las malas noticias viajan con lentitud es: ningún enrutador jamás tiene un valor mayor en más de una unidad que el mínimo de todos sus vecinos.
  - Gradualmente todos los enrutadores elevan cuentas hacia el infinito, pero el número de intercambios requeridos depende del valor numérico usado para el **infinito**.
    - Si la métrica usada es el número de saltos, es prudente hacer que el infinito sea igual a la ruta más larga más 1.

# Enrutamiento de Vector de Distancia

- Si la métrica es el retardo de tiempo no hay un límite superior bien definido,
  - ❑ se necesita un valor alto para evitar que una ruta con un retardo grande sea tratada como si estuviera desactivada.
- Este es el **problema de la cuenta hasta el infinito**.
  - ❑ Se han hecho varios intentos para resolverlo, pero ninguno funciona bien en general.
  - ❑ La esencia del problema consiste en que cuando  $X$  indica  $VD_X(i)$  a  $E$ ,  $E$  *no tiene forma de saber* si el destino  $i$  está en alguna ruta en funcionamiento.