## Bài tập chương 4 - Problem 27

**P27. Consider the network shown in Problem P26. Using Dijkstra’s algorithm, and showing your work using a table similar to Table 4.3, do the following:**

**a.Compute the shortest path from t to all network nodes.**

**b.Compute the shortest path from u to all network nodes.**

**c.Compute the shortest path from v to all network nodes.**

**d.Compute the shortest path from w to all network nodes.**

**e.Compute the shortest path from y to all network nodes.**

**f. Compute the shortest path from z to all network nodes.**

P27: Hãy xem xét mạng được hiển thị trong P26. Sử dụng thuật toán Dijkstra và hiển thị công việc của bạn bằng bảng tương tự Bảng 4.3, làm theo các yêu cầu sau:

a. Tính đường đi ngắn nhất từ **t** đến tất cả các nút mạng.

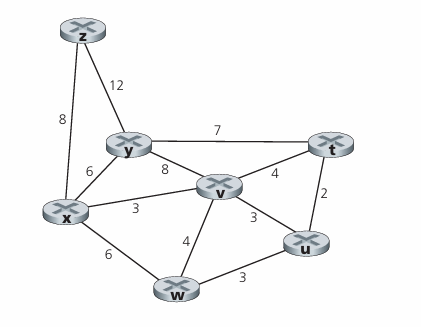
b. Tính đường đi ngắn nhất từ **u** đến tất cả các nút mạng.

c. Tính đường đi ngắn nhất từ **v** đến tất cả các nút mạng.

d. Tính đường đi ngắn nhất từ **w** đến tất cả các nút mạng.

e. Tính đường đi ngắn nhất từ **y** đến tất cả các nút mạng.

f. Tính đường đi ngắn nhất từ **z** đến tất cả các nút mạng.

* **Hình ảnh trong P26**

**Giải:**

1. Đường đi ngắn nhất từ **t** đến tất cả các nút mạng.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Step** | **N’** | **D(x),p(x)** | **D(u),p(u)** | **D(v),p(v)** | **D(w),p(w)** | **D(y),p(y)** | **D(z),p(z)** |
| **0** | t | ∞ | 2,t | 4,t | ∞ | 7,t | ∞ |
| **1** | tu | ∞ | **--** | 4,t | 5,u | 7,t | ∞ |
| **2** | tuv | 7,v | **--** | **--** | 5,u | 7,t | ∞ |
| **3** | tuvw | 7,v | **--** | **--** | **--** | 7,t | ∞ |
| **4** | tuvwx | **--** | **--** | **--** | **--** | 7,t | 15,x |
| **5** | tuvwxy | **--** | **--** | **--** | **--** | **--** | 15,x |
| **6** | tuvwxyz | **--** | **--** | **--** | **--** | **--** | -- |

1. Tính đường đi ngắn nhất từ **u** đến tất cả các nút mạng.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Step** | **N’** | **D(x),p(x)** | **D(t),p(t)** | **D(v),p(v)** | **D(w),p(w)** | **D(y),p(y)** | **D(z),p(z)** |
| **0** | u | ∞ | 2,u | 3,u | 3,u | ∞ | ∞ |
| **1** | ut | ∞ | **--** | 3,u | 3,u | 9,t | ∞ |
| **2** | utv | 6,v | **--** | **--** | 3,u | 9,t | ∞ |
| **3** | utvw | 6,v | **--** | **--** | **--** | 9,t | ∞ |
| **4** | utvwx | **--** | **--** | **--** | **--** | 9,t | 14,x |
| **5** | utvwxy | **--** | **--** | **--** | **--** | **--** | 14,x |
| **6** | utvwxyz | **--** | **--** | **--** | **--** | **--** | -- |

1. Tính đường đi ngắn nhất từ **v** đến tất cả các nút mạng.
2. Tính đường đi ngắn nhất từ **w** đến tất cả các nút mạng.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Step** | **N’** | **D(x),p(x)** | **D(u),p(u)** | **D(t),p(t)** | **D(w),p(w)** | **D(y),p(y)** | **D(z),p(z)** |
| **0** | v | 3,v | 3,v | 4,v | 4,v | 8,v | ∞ |
| **1** | vx | -- | 3,v | 4,v | 4,v | 8,v | 11,x |
| **2** | vxu | -- | **--** | 4,v | 4,v | 8,v | 11,x |
| **3** | vxut | -- | **--** | -- | 4,v | 8,v | 11,x |
| **4** | vxutw | **--** | **--** | **--** | 4,v | 8,v | 11,x |
| **5** | vxutwy | **--** | **--** | **--** | **--** | **--** | 11,x |
| **6** | vxutwyz | **--** | **--** | **--** | **--** | **--** | -- |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Step** | **N’** | **D(x),p(x)** | **D(u),p(u)** | **D(v),p(v)** | **D(t),p(t)** | **D(y),p(y)** | **D(z),p(z)** |
| **0** | w | 6,w | 3,w | 4,w | ∞ | ∞ | ∞ |
| **1** | wu | 6,w | **--** | 4,w | 5,u | ∞ | ∞ |
| **2** | wuv | 6,w | **--** | -- | 5,u | 12,v | ∞ |
| **3** | wuvt | 6,w | **--** | -- | -- | 12,v | ∞ |
| **4** | wuvtx | **--** | **--** | **--** | -- | 12,v | 14,x |
| **5** | wuvtxy | **--** | **--** | **--** | **--** | **--** | 14,x |
| **6** | wuvtxyz | **--** | **--** | **--** | **--** | **--** | -- |

1. Tính đường đi ngắn nhất từ **y** đến tất cả các nút mạng.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Step** | **N’** | **D(x),p(x)** | **D(u),p(u)** | **D(v),p(v)** | **D(w),p(w)** | **D(t),p(t)** | **D(z),p(z)** |
| **0** | y | 6,y | ∞ | 8,y | ∞ | 7,y | 12,y |
| **1** | yx | **--** | ∞ | 8,y | 12,x | 7,y | 12,y |
| **2** | yxt | **--** | 9,t | 8,y | 12,x | **--** | 12,y |
| **3** | yxtv | **--** | 9,t | -- | 12,x | **--** | 12,y |
| **4** | yxtvu | **--** | **--** | **--** | 12,x | **--** | 12,y |
| **5** | yxtvuw | **--** | **--** | **--** | **--** | **--** | 12,y |
| **6** | yxtvuwz | **--** | **--** | **--** | **--** | **--** | -- |

1. Tính đường đi ngắn nhất từ **z** đến tất cả các nút mạng.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Step** | **N’** | **D(x),p(x)** | **D(u),p(u)** | **D(v),p(v)** | **D(w),p(w)** | **D(y),p(y)** | **D(t),p(t)** |
| **0** | z | 8,z | ∞ | ∞ | ∞ | 12,z | ∞ |
| **1** | zx | **--** | ∞ | 11,x | 14,x | 12,z | ∞ |
| **2** | zxv | **--** | 14,v | -- | 14,x | 12,z | 15,v |
| **3** | zxvy | **--** | 14,v | -- | 14,x | **--** | 15,v |
| **4** | zxvyu | **--** | **--** | **--** | 14,x | **--** | 15,v |
| **5** | zxvyuw | **--** | **--** | **--** | **--** | **--** | 15,v |
| **6** | zxvyuwt | **--** | **--** | **--** | **--** | **--** | -- |

## Chương 5 - So sánh Aloha, Slotted Aloha, CSMA, CSMA/CD

* **Slotted ALOHA**

|  |  |
| --- | --- |
| **Giả thuyết** | **Hoạt động** |
| * Tất cả các frame có cùng kích thước * Thời gian được chia thành các slot có kích thước bằng nhau(thời gian để * truyền một frame) * Các node bắt đầu truyền chỉ ngay tại lúc bắt đầu slot * Các node được đồng bộ hóa * Nếu 2 hoặc nhiều node truyền trong slot, thì tất cả các node đều phát hiện   đụng độ | Khi node có được frame mới, nó sẽ truyền trong slot kế tiếp:   * Nếu không có đụng độ: node có thể gửi frame mới trong slot kế tiếp * Nếu có đụng độ: node truyền lại frame trong mỗi slot tiếp theo với xác   suất p cho đến khi thành công |

* **Mô hình giả định:**
* **C: collision slot**
* **E: empty slot**
* **A diagram of colorful squares

  Description automatically generated with medium confidenceS: successful**

|  |  |  |
| --- | --- | --- |
| **Ưu điểm** | **Nhược điểm** | **Hiệu suất** |
| * Node đơn kích hoạt có thể truyền liên tục với tốc độ tối đa của kênh * Phân cấp cao: chỉ có các slot trong các node cần được đồng bộ * Đơn giản | * Đụng độ, lãng phí slot * Các slot nhàn rỗi * Node phải phát hiện ra hiện tượng va chạm (nếu có) với thời giạn ngắn   hơn thời gian truyền   * Đồng bộ hóa | * Độ hiệu quả: % slot thành công so với tổng số slot, khi có nhiều node cùng   truyền một thời gian dài, mỗi node có một số lượng lớn các frame cần  truyền   * Hiệu suất cực đại: 1/e, xấp xỉ 37% |

* **Pure(unslotted) ALOHA**
* **Mô hình giả định**
* **A diagram of a diagram

  Description automatically generated with medium confidence**Frame được truyền tại thời điểm t0 va chạm với các frame được gửi trong khoảng thời gian [t0-1,t0+1]

|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| * Đơn giản hơn Slotted ALOHA * Không cần đồng bộ hóa * Truyền đi ngay lần đầu các frame đến | * Khả năng va chạm tang hơn so với Slotted ALOHA * Hiệu suất = 1/(2e) tức thấp hơn một nữa Slotted ALOHA |

* A diagram of a graph

  Description automatically generated**CSMA**

Lắng nghe trước khi truyền:

* Nếu kênh nhàn rỗi: truyền toàn bộ frame
* Nếu kênh truyền bận: trì hoãn truyền

Đụng độ có thể vẫn xảy ra: trễ lan truyền nghĩa là hai node không thể nghe thấy quá trình truyền lẫn nhau

Đụng độ: Toàn bộ thời gian truyền packet bị lãng phí

Lưu ý: Vai trò của khoảng cách và thời gian lan truyền có ảnh hưởng đến khả năng va chạm

* **A diagram of a collision

  Description automatically generatedCSMA/CD (collision detection)**

Đụng độ được phát hiện trong khoảng thời gian ngắn. Việc truyền đụng độ được bỏ qua, giảm lãng phí kênh truyền

Phát hiện đụng độ:

* Dễ dàng trong các mạng LAN hữu tuyến: Đo

cường độ tín hiệu, so sánh với các tín hiệu đã

được truyền và nhận

* Khó thực hiện trong mạng LAN vô tuyến: cường

độ tín hiệu được nhận bị áp đảo bởi cường độ

truyền cục bộ

Thuật toán Ethernet CSMA/CD

* NIC nhận datagram từ tầng network, tạo frame
* Nếu NIC cảm nhận được kênh rỗi, nó sẽ bắt đầu việc truyền frame. Nếu
* NIC cảm nhận kênh bận, đợi cho đến khi kênh rãnh, sau đó mới truyền
* Nếu NIC truyền toàn bộ frame mà không phát hiện việc truyền khác, NIC

được truyền toàn bộ frame đó!

* Nếu NIC phát hiện có sự truyền khác trong khi đang truyền, thì nó sẽ hủy

bỏ truyền và phát tín hiệu tắt nghẽn

* Sau khi hủy bỏ truyền, NIC thực hiện binary (exponential) backoff:
* Sau lần đụng độ thứ m, NIC chọn ngẫu nhiên số K trong khoảng {0,1,2, …, 2𝑚-1}. NIC sẽ đợi K·512 bit lần, sau đó trở lại bước 2- tức là NIC sẽ tiếp tục dò kênh
* Đụng độ nhiều thì sẽ có khoảng thời gian backoff dài hơn

Độ hiệu quả

* 𝑇𝑝𝑟𝑜𝑝 : độ trễ lan truyền lớn nhất giữa 2 node trong mạng LAN
* 𝑇𝑡𝑟𝑎𝑛𝑠: thời gian để truyền frame có kích thước lớn nhất
* Công thức

A black and white math equation

Description automatically generated

* Hiệu suất tiến tới 1
* Khi 𝑇𝑝𝑟𝑜𝑝 tiến tới 0
* Khi 𝑇𝑡𝑟𝑎𝑛𝑠 tiến tới vô cùng
* Hiệu suất tốt hơn ALOHA: đơn giản, chi phí thấp và điều khiển phân tán