

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

\_\_\_\_\_ \*



Báo cáo môn học:  
**Project 2**

---

## Tìm hiểu về mạng liên kết

---

*Sinh viên thực hiện:*  
Nguyễn Quang Quý

*Giáo viên hướng dẫn:*  
PGS.TS Nguyễn Khanh Văn

Hà Nội, Ngày 26 tháng 5 năm 2018

# Mục lục

1	Mở đầu . . . . .	1
2	Tổng quan về mạng liên kết . . . . .	1
2.1	Mạng liên kết là gì ? . . . . .	1
2.2	Mạng liên kết được dùng ở đâu ? . . . . .	2
2.3	Vai trò của mạng liên kết . . . . .	2
3	Hình trạng mạng (Topo) . . . . .	2
3.1	Direct Network . . . . .	3
3.2	Indirect Network . . . . .	5
4	Xây dựng topo cơ bản: Small World . . . . .	6
4.1	Mô hình . . . . .	6
4.2	Chương trình thực nghiệm . . . . .	8
5	Bài toán triển khai trung tâm dữ liệu . . . . .	13
5.1	Giới thiệu vấn đề . . . . .	13
5.2	Mô hình hóa bài toán . . . . .	14
5.3	Đề xuất hướng giải quyết . . . . .	16
6	Kết luận . . . . .	16

# 1 Mở đầu

Trong xã hội hiện đại, hệ thống kỹ thuật số đang trở nên phổ biến. Chúng ta dễ dàng nêu ra được các phát minh trong đó có sử dụng hệ thống số, đó là máy vi tính, điện thoại, các thiết bị di động, ...

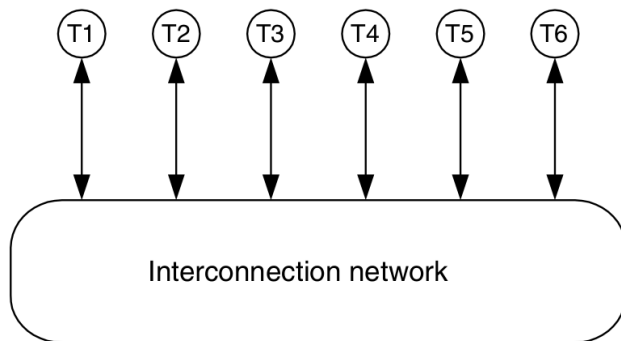
Một hệ thống số bao gồm ba khối cơ bản: logic, bộ nhớ và giao tiếp. Khối logic có nhiệm vụ chuyển đổi và kết hợp dữ liệu. Khối bộ nhớ lưu trữ dữ liệu để truy xuất sau. Và khối giao tiếp có nhiệm vụ chuyển dữ liệu từ vị trí địa lý này tới vị trí khác. Hiệu năng của hầu hết hệ thống số ngày nay bị giới hạn bởi khối giao tiếp của chúng. Cụ thể khối giao tiếp ở đây được thực hiện dưới mô hình mạng liên kết. Mạng liên kết có nhiệm vụ vận chuyển dữ liệu giữa các thành phần vật lý khác nhau của hệ thống số. Chính vì vậy để xây dựng được một hệ thống số hiệu quả, cần phải có sự hiểu biết về mạng liên kết. Trong tài liệu này, người viết xin phép được giới thiệu về các khái niệm chung trong mạng liên kết và việc triển khai mạng liên kết trong thực tế.

Em xin chân thành cảm ơn thầy và nhóm nghiên cứu đã hỗ trợ em hoàn thành tài liệu này.

## 2 Tổng quan về mạng liên kết

### 2.1 Mạng liên kết là gì ?

Mạng liên kết là một hệ thống vận chuyển dữ liệu có thể lập trình được giữa các thiết bị đầu cuối (terminal). Hình vẽ thể hiện một mạng liên kết, trong đó có 6 terminal, T1 tới T6, được kết nối với mạng. Khi terminal T3 muốn trao đổi dữ liệu với terminal T5, nó sẽ gửi một tin nhắn có chứa dữ liệu vào trong mạng và mạng sẽ làm nhiệm vụ vận chuyển tin nhắn tới T5. Mạng có thể lập trình được theo nghĩa nó có thể tạo kết nối với nhiều điểm khác nhau tại cùng một thời điểm. Mạng biểu diễn trong hình có thể chuyển tin nhắn từ T3 tới T5 trong một chu kỳ và sau đó sử dụng cùng tài nguyên đó để chuyển một tin nhắn mới từ T3 tới T1 trong chu kỳ tiếp theo. Mạng liên kết được coi là một hệ thống bởi nó bao gồm nhiều thành phần: buffer, channel, switch và control, các thành phần này làm việc cùng nhau để hoàn thành nhiệm vụ chính đó là vận chuyển dữ liệu.



Hình 1: Góc nhìn chức năng của một mạng liên kết. Terminals (được gán nhãn T1 tới T6) được kết nối với mạng sử dụng các kênh truyền.

## 2.2 Mạng liên kết được dùng ở đâu ?

Hầu hết các hệ thống số lớn (các hệ thống có nhiều thành phần cần trao đổi dữ liệu với nhau) đều có sử dụng mạng liên kết. Ứng dụng phổ biến của mạng liên kết đó là trong các hệ thống máy tính và giao tiếp giữa các switch. Trong hệ thống máy tính, chúng kết nối processor với bộ nhớ và thiết bị nhập xuất với bộ điều khiển nhập xuất. Chúng kết nối các cổng vào với các cổng ra trong giao tiếp với switch và với processor trong hệ thống điều khiển. Bất hệ thống nào xuất hiện sự trao đổi dữ liệu giữa hai hay nhiều thành phần của nó với nhau thì hệ thống có sử dụng tới mạng liên kết.

## 2.3 Vai trò của mạng liên kết

Hiện tại, mạng liên kết được ứng dụng rộng rãi trong các hệ thống máy tính và các hệ thống chuyển mạch thông tin liên lạc. Đặc biệt mạng liên kết được thiết kế sử dụng ở các mức độ khác nhau trong các hệ thống máy tính nhằm đáp ứng nhu cầu của các nhóm ứng dụng khác nhau như: tính toán hiệu năng cao, tính toán phân tán, ... Tùy thuộc vào số lượng các thiết bị được kết nối và khoảng cách giữa các thiết bị, mạng liên kết được chia ra làm bốn lĩnh vực ứng dụng chính:

- On-chip networks (OCNs) hay còn được nhắc tới với thuật ngữ network-on-chip (NoC): được sử dụng để kết nối bên trong các vi kiến trúc giữa các đơn vị chức năng, thanh ghi (register), bộ lưu trữ trung gian (caches), các bộ vi xử lý (processor) trong các module đa chip. OCNs hỗ trợ các kết nối giữa vài chục thiết bị đặt trong các vi mạch với khoảng cách tối đa khoảng vài centimet.
- System/storage area networks (SANs): Đây là mạng liên kết được sử dụng để kết nối các bộ vi xử lý liên kết (interprocessor) và các bộ nhớ (processor-memory) trong các hệ thống đa nhân và hệ thống đa máy tính (multicomputer). Ngoài ra loại mạng liên kết này cũng được sử dụng để kết nối các thành phần lưu trữ và thành phần xử lý vào ra trong môi trường gồm các máy chủ (server) và các trung tâm dữ liệu (data centers). Số lượng thiết bị được kết nối trong SANs có thể lên tới hàng nghìn thiết bị khác nhau phân bố với khoảng cách khoảng vài trăm met
- Local area networks (LANs): Đây là mạng liên kết được sử dụng để kết nối hệ thống máy tính cá nhân. Kết nối máy tính trong một cụm là một ví dụ điển hình. Ban đầu, các mạng LAN chỉ kết nối hàng trăm thiết bị, nhưng với cầu nối (bridges), mạng LAN có thể kết nối lên đến vài nghìn thiết bị. Khoảng cách kết nối tối đa bao phủ khu vực có đường kính vài kilomet đến vài chục kilomet.
- Wide area networks (WANs): WANs kết nối các hệ thống máy tính phân bố phân tán trên toàn thế giới. WANs cũng kết nối hàng triệu các máy tính với nhau trên khoảng cách lớn.

## 3 Hình trạng mạng (Topo)

Hình trạng mạng là mô hình thể hiện kết nối giữa các nút mạng với nhau thông qua các kênh truyền. Trong mỗi hệ thống cụ thể, việc lựa chọn một hình trạng mạng là yếu tố rất quan trọng do đây là yếu tố liên quan đến việc xây dựng hệ thống, lựa chọn các thiết bị chuyển tiếp trung gian có thông số phù hợp (số cổng kết nối, tốc độ xử lý và truyền tiếp gói tin ...). Không những

thể, hình trạng mạng còn ảnh hưởng đến việc lựa chọn/xây dựng giải thuật định tuyến. Mỗi mô hình mạng khác nhau thường yêu cầu các thuật toán định tuyến khác nhau để có thể sử dụng được những đặc điểm riêng biệt của một mô hình mạng. Do vậy, để đánh giá một mô hình mạng mới, người ta thường quan tâm đến chi phí khi xây dựng hệ thống trên thực tế và hiệu năng mà mô hình mạng đó đem lại.

Có hai loại hình trạng mạng: indirect network và direct network. Trong direct network, mỗi thiết bị được liên kết với một tập các thiết bị khác trong mạng. Trong trường hợp này, bất cứ sự trao đổi nào giữa hai thiết bị không phải là hàng xóm của nhau thì thông tin sẽ được chuyển qua các thiết bị trung gian trước khi đến được thiết bị đích. Thay vì kết nối trực tiếp giữa các thiết bị, trong mạng indirect network kết nối các thiết bị với nhau thông qua các switch. Nếu tồn tại nhiều switch, chúng sẽ được kết nối với nhau thông qua các liên kết điểm - điểm. Trong trường hợp này, thông tin trao đổi giữa các thiết bị đều phải thông qua một hoặc nhiều switch.

### 3.1 Direct Network

Direct network hay point-to-point network là một kiến trúc mạng phổ biến đáp ứng tốt nhu cầu làm việc với một lượng lớn vi xử lý. Một direct network bao gồm một tập các node, mỗi node được kết nối trực tiếp tới một tập các node khác trong mạng. Mỗi node là một máy tính sở hữu các tài nguyên riêng như: vi xử lý, bộ nhớ và các thiết bị hỗ trợ khác. Các node có thể có các chức năng khác nhau. Thành phần chung của các node đó là router, có nhiệm vụ xử lý thông điệp giao tiếp giữa các node. Vì lý do đó, direct network còn được gọi là router-based network. Mỗi router có nhiều kết nối trực tiếp tới các hàng xóm của nó. Thông thường, hai node là hàng xóm của nhau được kết nối bằng một cặp kênh vô hướng và có chiều ngược nhau. Kênh hai chiều cũng có thể được dùng để kết nối hai node hàng xóm với nhau.

Mỗi router hỗ trợ một số lượng các kênh vào ra. Internal channel hay port dùng để kết nối router với vi xử lý/ bộ nhớ của node. Thường sẽ chỉ có một cặp internal channel, tuy nhiên để tránh hiện tượng nghẽn giao tiếp giữa các vi xử lý/ bộ nhớ với router. External channel được sử dụng cho việc giao tiếp giữa các router. Thông thường mỗi node có một số lượng cố định các kênh vào ra và mỗi kênh vào sẽ có tương ứng kênh ra ghép với nó.

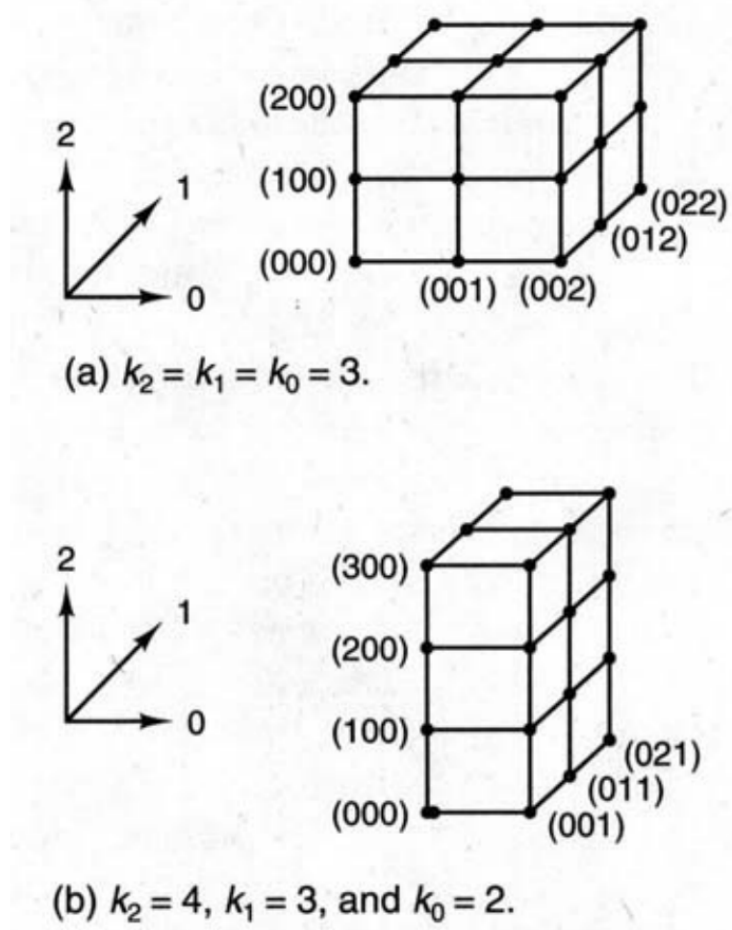
Direct network được mô hình hóa là một đồ thị  $G(N, C)$ ,  $N$  đại diện cho số lượng node và  $C$  là số lượng kênh. Những thuộc tính cần quan tâm trong direct network là:

- Bậc của node: số kênh mà node đang kết nối tới các hàng xóm của nó.
- Diameter: Khoảng cách lớn nhất trong những khoảng cách nhỏ nhất giữa các node.
- Regularity: Một mạng được xem là regular khi tất cả các node có cùng bậc.
- Symmetry: Một mạng là symmetric khi nó có dạng giống nhau khi quan sát từ mọi node.

### Các topo điển hình

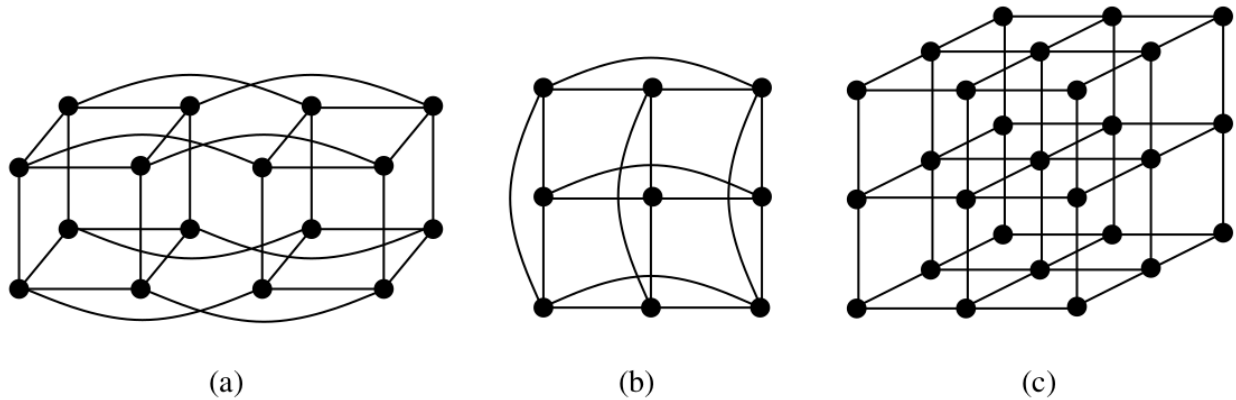
Các topo trong thực tế thường có cấu trúc liên kết trực giao. Mỗi node có ít nhất một liên kết với nó theo mỗi chiều.

**Lưới n chiều** Mạng gồm  $k_0 \times k_1 \times k_2 \times \dots \times k_{n-2} \times k_{n-1}$  node,  $k_i$  là số node theo dọc theo chiều  $i$ , với  $k_i \geq 2$  và  $0 \leq i \leq n-1$ . Mỗi node X được xác định bởi n tọa độ  $(x_{n-1}, x_{n-2}, \dots, x_1, x_0)$ , với  $0 \leq x_i \leq k_i - 1$  và  $0 \leq i \leq n-1$ . Hai node X và Y là hàng xóm khi và chỉ khi  $y_i = x_i$  với mọi  $0 \leq i \leq n-1$ , ngoại trừ một chiều  $j$  mà  $y_j = x_j \pm 1$ . Do đó, node sẽ có từ n tới  $2n$  hàng xóm, điều này phụ thuộc vào vị trí của nó trên lưới. Chính vì thế, topo này không có tính *regular*.



Hình 2: Lưới 3 chiều

**Mạng Torus (k-ary n-cube)** Trong cấu trúc mạng Torus, mọi node đều có cùng số lượng hàng xóm (bậc giống nhau). Topo này khác với lưới n chiều ở chỗ số lượng node ở mỗi chiều là bằng nhau và bằng  $k$ . Hai node X và Y là hàng xóm của nhau nếu và chỉ nếu  $y_i = x_i$  với mọi  $0 \leq i \leq n-1$ , ngoại trừ một chiều  $j$  mà  $y_j = x_j \pm 1 \bmod k$ . Sự khác biệt này giúp cho cấu trúc này có tính chất *regular* và *symmetry*. Từ đây, ta thấy node có  $n$  hàng xóm nếu  $k = 2$  và  $2n$  hàng xóm nếu  $k > 2$ . Khi  $n = 1$ , k-ary n-cube trở thành topo ring hai chiều với  $k$  node. Kênh truyền của mạng Torus có thể là một chiều hoặc hai chiều.

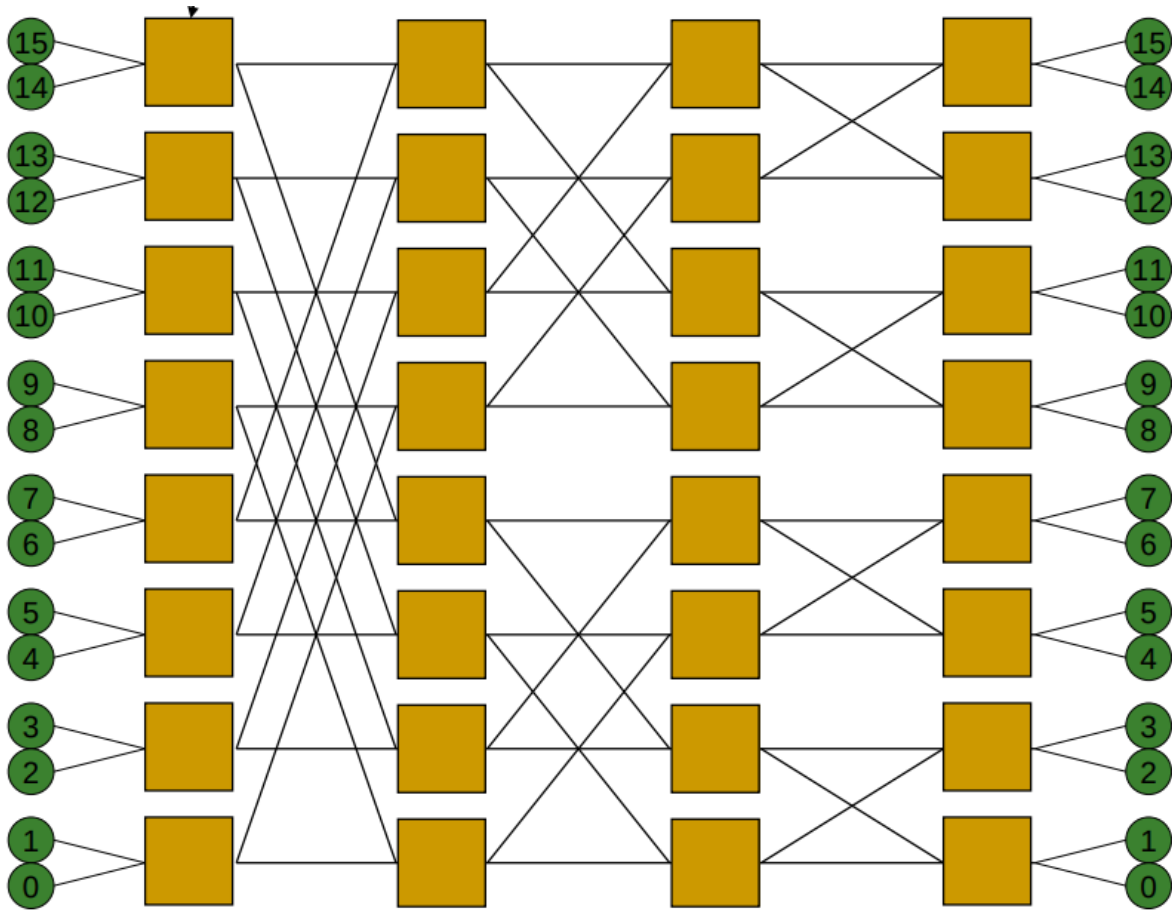


Hình 3: (a) 2-ary 4-cube (hypercube), (b) 3-ary 2-cube và (c) 3-ary 3-D mesh

### 3.2 Indirect Network

Indirect Network hay mạng dựa trên switch là một phân lớp khác của mạng liên kết. Thay vì cung cấp trực tiếp kết nối giữa các node với nhau, các node sẽ giao tiếp với nhau thông qua switch. Mỗi node sẽ có một adapter để kết nối với switch. Mỗi switch sẽ có một tập các port, mỗi port bao gồm một liên kết đi vào và một liên kết đi ra. Một số lượng port trong switch có thể được kết nối với các vi xử lý hoặc bỏ trống, số lượng port còn lại sẽ kết nối với các switch khác để tạo nên kết nối giữa các vi xử lý. Cấu trúc liên kết giữa các switch sẽ định nghĩa nhiều loại hình trạng mạng khác nhau.

Indirect Network cũng có thể được mô hình hóa dưới dạng một đồ thị  $G(N, C)$  -  $N$  là số lượng các switch,  $C$  là số lượng các liên kết một chiều hoặc hai chiều giữa các switch. Việc truyền thông điệp từ một node tới node khác yêu cầu phải đi các liên kết từ nguồn tới switch kết nối với nó, và liên kết giữa switch cuối cùng trên đường đi và node đích. Bởi vậy khoảng cách giữa hai node là khoảng cách giữa các switch kết nối trực tiếp đến những node đó cộng thêm hai đơn vị. Tương tự vậy, diameter sẽ là khoảng cách lớn nhất giữa hai switch kết nối với một số node cộng thêm hai đơn vị.



Hình 4: Indirect network với 2 đầu vào và 2 đầu ra

## 4 Xây dựng topo cơ bản: Small World

### 4.1 Mô hình

Từ thí nghiệm của Milgram, Kleinberg đã đề xuất một hướng tiếp cận mô hình Small-world có ứng dụng khá lớn vào các hệ thống Datacenter. Thuật toán dựa trên sự kết nối tự nhiên giữa các thực thể là con người với nhau trong một mạng xã hội.

**Đầu vào:** Với một topo theo dạng Grid hoặc Torus cho sẵn  $G(V, E)$  với  $V$  là tập đỉnh,  $E$  là tập cạnh. Cho trước tham số  $\alpha$  đóng vai trò là tham số cho xác suất ngẫu nhiên.

**Đầu ra:** Từ tham số ngẫu nhiên  $\alpha$ , tạo ra đồ thị mới được bổ sung thêm các kết nối dài giữa các điểm trong topo với nhau theo cặp.

**Phân tích:** Trong thuật toán ta xây dựng một tập các ứng cử viên candidates theo của mỗi node đang xét theo luật:

- Node đang xét không thuộc candidates.



- Hàng xóm (những node đã có liên kết với node đang xét) không thuộc candidates.
- Những node không còn khả năng nhận liên kết thì không thuộc candidates.

**Thuật toán** Với mỗi đỉnh nguồn  $u$  trong  $V$  ta xét:

- Với mỗi đỉnh  $v$  trong candidates, xác suất kết nối giữa  $u$  và  $v$  được định nghĩa như sau:

$$Pr[u \rightarrow v] \sim D^{-\alpha}(u, v) \quad (1)$$

- Từ một giá trị ngẫu nhiên  $\nabla \in (0, 1)$  ta sẽ chọn được đích  $v$  là node có nhãn lớn nhất thỏa mãn:

$$\sum Pr[u \rightarrow v] \leq \nabla \quad (2)$$

**Với**

- $D(u, v)$  là khoảng cách giữa 2 node  $u$  và  $v$  (tính theo bước lưới).
- Với mỗi node nguồn đang  $x$ , tập các đỉnh trong candidates là tập đầy nên:

$$1 = \sum Pr[u \rightarrow v] \sim \sum D^{-\alpha}(u, v) = \nabla \quad (3)$$

Từ (1) và (2) ta có:

$$Pr[u \rightarrow v] = \nabla^{-1} \cdot D^{-\alpha}(u, v) \quad (4)$$

## Với topo 2D-Grid

Bước lưới giữa hai node  $u(x_1; y_1)$  và  $v(x_2; y_2)$  là:  $D(u, v) = |x_1 - x_2| + |y_1 - y_2|$

## Với topo 2D-Torus

Với Torus, do có tính *regular* và *symmetric* nên ở đây  $\nabla$  là giống nhau đối với tất cả các node của đồ thị. Và khoảng cách giữa hai node  $u(x_1; y_1)$  và  $v(x_2; y_2)$  khi đó là  $D(u, v) = D_x + D_y$  với:

$$D_x = \begin{cases} |x_2 - x_1| & |x_2 - x_1| \leq \left\lceil \frac{M}{2} \right\rceil \\ M - |x_2 - x_1| & |x_2 - x_1| \geq \left\lceil \frac{M}{2} \right\rceil \end{cases}$$

$$D_y = \begin{cases} |y_2 - y_1| & |y_2 - y_1| \leq \left\lceil \frac{N}{2} \right\rceil \\ N - |y_2 - y_1| & |y_2 - y_1| \geq \left\lceil \frac{N}{2} \right\rceil \end{cases}$$

Ta có thể thấy ngay rằng, với tham số xác suất có giá trị càng lớn, sẽ kéo theo giá trị xác xuất kết nối giữa 2 node cũng trở lên càng nhỏ. Do đó, khả năng xuất hiện các kết nối dài cũng ngày càng thấp; thay vào đó, ta sẽ chỉ thấy các kết nối ngắn, và đồ thị sẽ thưa dần ra nếu ta tăng giá trị  $\alpha$  lên.

## 4.2 Chương trình thực nghiệm

**Xây dựng 2D-Grid** Topo Grid được xây dựng dựa trên nhận xét sau:

- Bốn node ở 4 góc có bậc là 2
- Các node khác 4 node trên mà thuộc hàng(cột) đầu hoặc hàng(cột) cuối cùng thì có bậc là 3
- Các node còn lại sẽ có bậc là 4

Thủ tục tạo 2D-Grid kích thước  $r$  hàng x  $c$  cột:

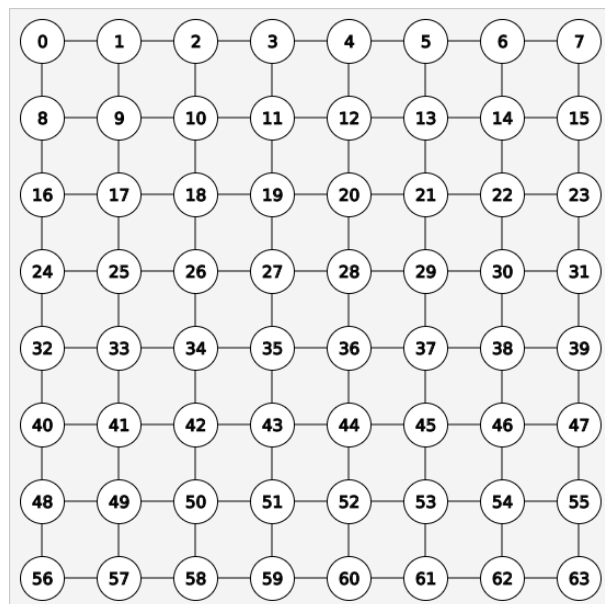
---

**Algorithm 1** Thuật toán xây dựng topo 2D-Grid kích thước  $r$  x  $c$

---

```
1: procedure INIT2DGRID( $r, c$ )
2:   for each node  $u$  in  $V$  do
3:     if  $u.y - 1 \geq 0$  then
4:       makeLink( $u$ , node( $u.y - 1, u.x$ ))
5:     end if
6:     if  $u.y + 1 < r$  then
7:       makeLink( $u$ , node( $u.y + 1, u.x$ ))
8:     end if
9:     if  $u.x + 1 < c$  then
10:      makeLink( $u$ , node( $u.y, u.x + 1$ ))
11:    end if
12:    if  $u.x - 1 \geq 0$  then
13:      makeLink( $u$ , node( $u.y, u.x - 1$ ))
14:    end if
15:  end for
16: end procedure
```

---



Hình 5: Grid 2D kích thước 8x8

**Xây dựng 2D-Torus** Với topo 2D-Torus thì do tính đối xứng, nên mỗi node trong mạng đều có bậc là 4. Do vậy thủ tục để tạo nên topo 2D-Torus ban đầu sẽ tạo nên các liên kết của topo Grid, sau đó sẽ tạo thêm các liên kết ở các node mà chưa đạt được bậc là 4 theo thủ tục sau:

---

**Algorithm 2** Thuật toán xây dựng topo 2D-Torus kích thước  $r \times c$

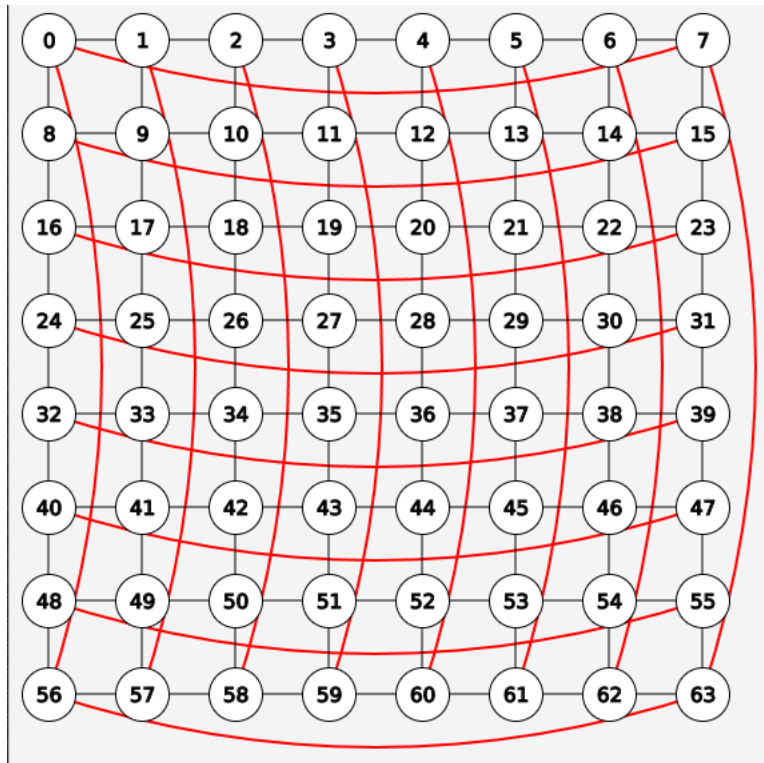
---

```

1: procedure PADDINGLINK( $r, c$ )
2:   for each node  $u$  in  $V$  do
3:     if  $u.y - 1 < 0$  then
4:       makeLink( $u$ , node( $c + u.y - 1$ ,  $u.x$ ))
5:     end if
6:     if  $u.y + 1 \geq r$  then
7:       makeLink( $u$ , node( $u.y + 1 - r$ ,  $u.x$ ))
8:     end if
9:     if  $u.x + 1 \geq c$  then
10:      makeLink( $u$ , node( $u.y$ ,  $u.x + 1 - c$ ))
11:    end if
12:    if  $u.x - 1 < 0$  then
13:      makeLink( $u$ , node( $u.y$ ,  $u.x - 1 + c$ ))
14:    end if
15:  end for
16: end procedure

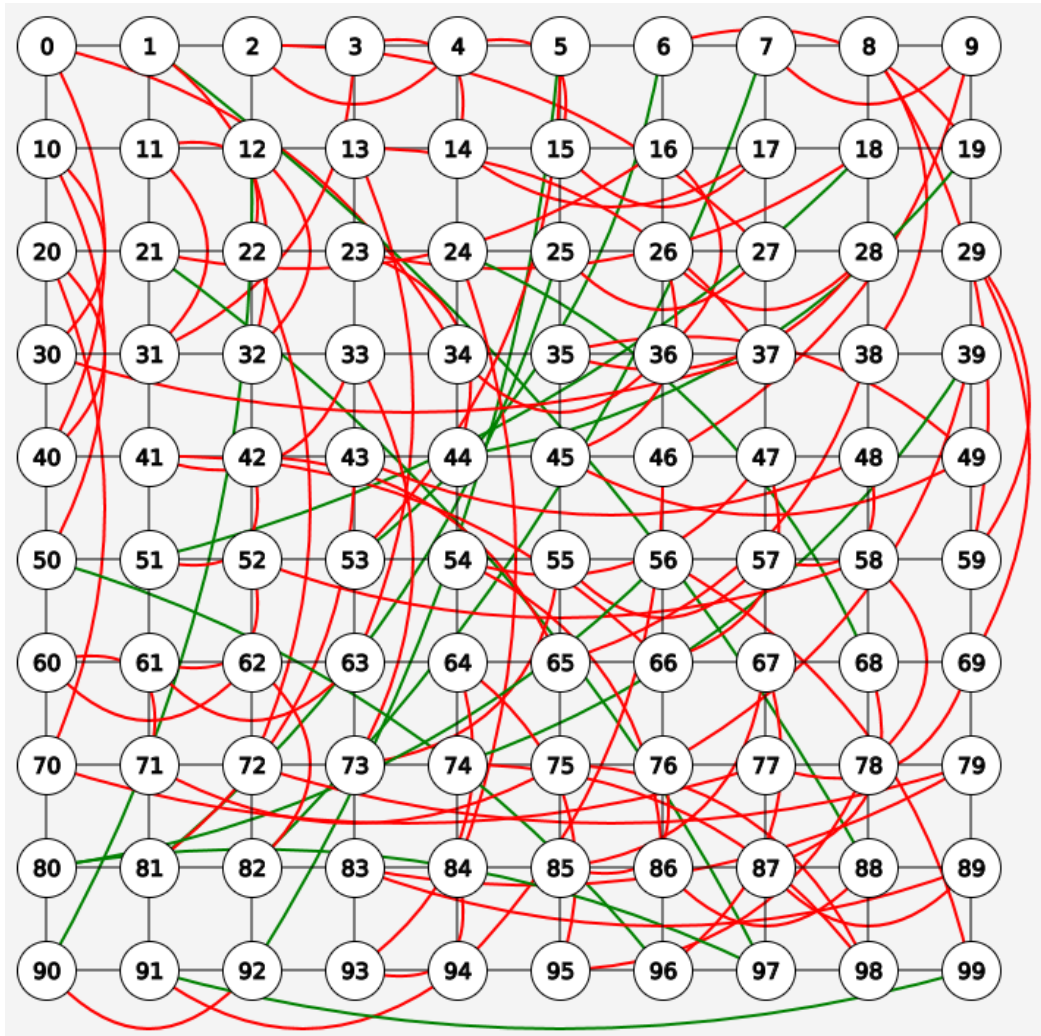
```

---

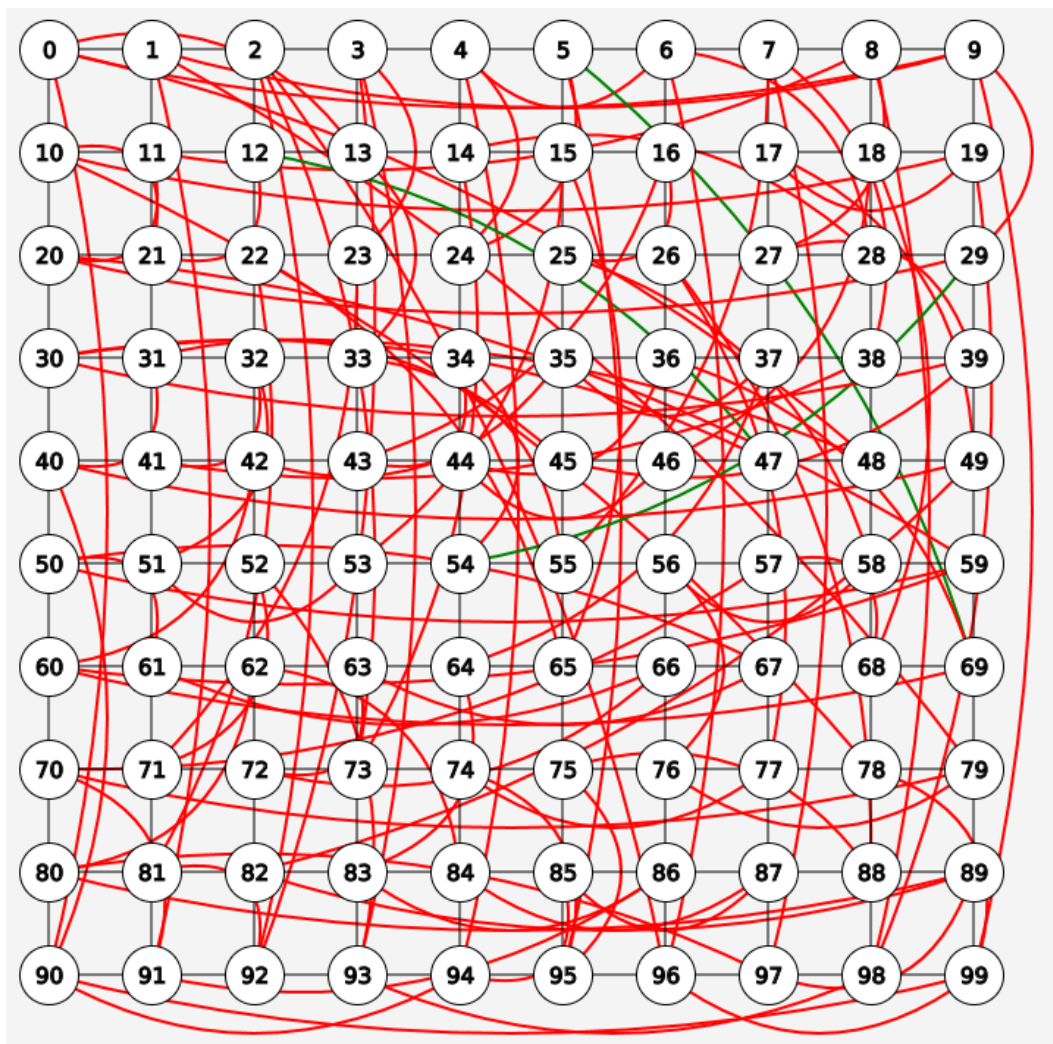


Hình 6: Torus 2D kích thước 8x8

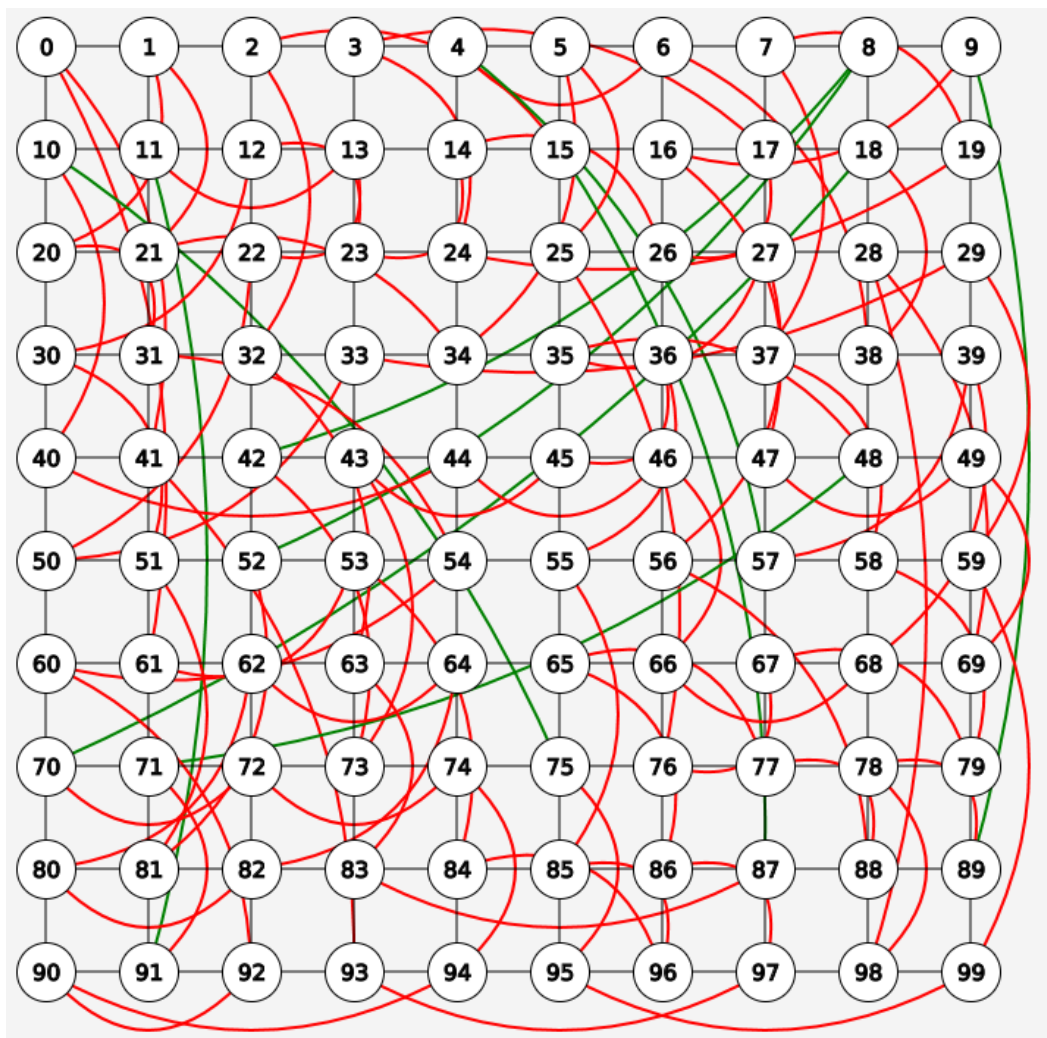
**Kết quả** Sau khi đã xây dựng được topo gốc chúng ta sẽ tiến hành cài đặt thuật toán trên mỗi topo. Một số kết quả thu được sau khi chạy thực nghiệm:



Hình 7: Small world 2D-Grid với  $\alpha = 1.6$

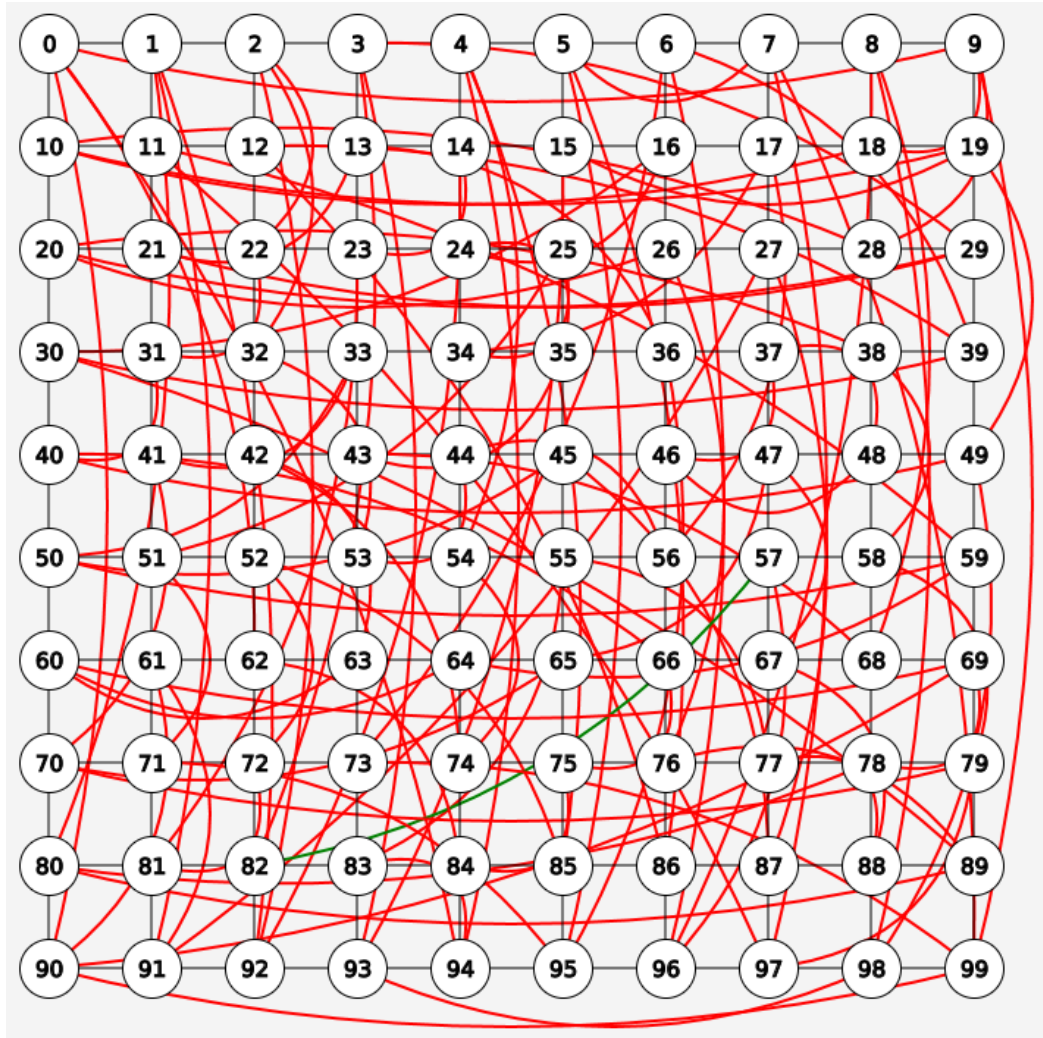


Hình 8: Small world 2D-Torus với  $\alpha = 1.6$



Hình 9: Small world 2D-Grid với  $\alpha = 2$





Hình 10: Small world 2D-Torus với  $\alpha = 2$

## 5 Bài toán triển khai trung tâm dữ liệu

### 5.1 Giới thiệu vấn đề

Một trung tâm dữ liệu được triển khai trong thực tế phải đạt được các tiêu chuẩn chung cơ bản đó là TIA-942A. Đây là tài liệu được coi là chuẩn chung cho mọi hệ thống trung tâm dữ liệu trên toàn thế giới. Trong việc triển khai trung tâm dữ liệu trong thực tế, bài toán mà chúng ta quan tâm đó là tối thiểu hóa chi phí thiết lập. Cụ thể ta có thể mô tả bài toán như sau: Xây dựng mạng liên kết với kích thước lớn nhất có thể với điều kiện vùng triển khai bị giới hạn trong một phạm vi diện tích biết trước và chi phí triển khai mạng liên kết không vượt quá một lượng cố định cho trước.

Trước khi bắt đầu mô hình hóa vấn đề, chúng ta cần tìm hiểu một số tiêu chuẩn TIA-942 cần quan tâm liên quan trực tiếp với vấn đề đã đề cập đề cập tới. Những tiêu chuẩn đó là:

- Chiều rộng của tủ Rack: 0,6m.

- Chiều dài của tủ Rack: 2,1m.
- Chiều cao của tủ Rack: 2,42m.
- Khoảng cách giữa hai tủ Rack theo chiều dài: 1,2m.
- Khoảng cách giữa hai tủ Rack theo chiều rộng: 1m.
- Dây nối giữa các switch/host giữa các tủ Rack được đi dây theo các góc vuông, tính theo khoảng cách Manhattan.
- Trong một tủ Rack sẽ chứa được tối đa 8 switch, mỗi switch có thể kết nối với tối đa 8 host.

## 5.2 Mô hình hóa bài toán

Với bài toán này ra đã biết trước được diện tích  $S_i$  ( $i = 1 \rightarrow N$ ) của mỗi sàn và chi phí tối đa được phép sử dụng. Các tủ Rack được phân bố vào các phòng theo chiến lược lấp đầy, nghĩa là khi một phòng đã chứa được tối đa số rack có thể thì ta sẽ cần phân bố số rack còn lại vào các sàn tiếp theo dựa trên cách thức đó. Sau khi đã phân bố rack, ta cần quan tâm tới cách thức đưa các node của topo vào các rack sao cho phù hợp. Dựa vào thực nghiệm của nhóm nghiên cứu đã tìm được hai giải pháp có thể mang lại sự tối ưu đó là đổ tràn và đổ đều:

- Đổ tràn là triển khai lần lượt các node mạng của topo vào tuần tự từng tủ rack theo lần lượt các phòng có diện tích  $S_i$
- Đổ đều là thực hiện chia đều số node mạng của topo vào các phòng có diện tích  $S_i$

Để kết nối các sàn của trung tâm dữ liệu với nhau, ta sử dụng Highway với các đặc tính sau:

- Giữa hai phòng bất kì luôn có một Highway để đảm bảo tính liên thông giữa chúng.
- Các node nằm ở hai phòng khác nhau mà có liên kết với nhau, thì dây cáp kết nối chúng nhất định phải đi qua Highway.
- Vị trí đặt Highway được cho là tối ưu đó là ở trung tâm của vùng phân bố Rack.

## Xác định chi phí

Ở đây ta xét các chi phí cơ bản để triển khai một hệ thống trung tâm dữ liệu, bao gồm: chi phí tủ Rack, chi phí switch và chi phí cáp nối. Trong đó chi phí cáp nối là yếu tố phức tạp cần khảo sát. Vì tổng chiều dài dây cáp có liên quan rất lớn đến chiến lược triển khai topo mạng vào diện tích của trung tâm dữ liệu, cũng như số lượng Highway, chiều dài và vị trí đặt các Highway.

## Khảo sát và đánh giá chi phí cable length

Như đã biết, trong một trung tâm dữ liệu sự kết nối giữa hai rack với nhau thực chất là sự kết nối giữa các switch/host trong hai rack với nhau. Do vậy ta cần tọa độ hóa được các switch này để có thể tính toán được total cable length.



Với một số lượng node  $m$  cho trước cùng với chiến lược ấn định là đổ tràn hoặc đổ đều và topo của mạng, ta có thể xác định được total cable length như sau:

- Tọa độ của một node được xác định theo 4 tham số đó là  $(x, y, z, f)$ :

- $x, y$  là tọa độ lưới của rack có chứa node trong sàn chứa rack đó.
- $z$  là chiều cao của node so với sàn.
- $f$  là chỉ số sàn chứa node này ( $f = 1 \rightarrow N$ )

Ở đây  $x, y, f$  là các số nguyên dương.

- Từ topo của mạng ta xác định được một ma trận 2 chiều với mục đích lưu trữ thông tin về các kết nối giữa các node của mạng:

- $w(i, j) = 1$ : node  $i$  và node  $j$  có liên kết trực tiếp với nhau.
- $w(i, j) = 0$ : node  $i$  và node  $j$  không có liên kết với nhau trực tiếp.

- Ta xây dựng thêm một ma trận 2 chiều  $d$  để lưu thông tin về khoảng cách giữa hai node  $i, j$  bất kì. Với mỗi cặp node  $i, j$  bất kì, sẽ có 3 trường hợp có thể xảy ra:

- $i, j$  nằm trên cùng một  $r$ :

$$d(i, j) = |z_i - z_j| \quad (5)$$

- $i, j$  nằm khác rack nhưng cùng một sàn ( $f_i = f_j$ ):

$$d(i, j) = |x_i - x_j| + |y_i - y_j| + z_i + z_j \quad (6)$$

- $i, j$  nằm khác rack và khác sàn:

$$\begin{aligned} d(i, j) = & |x_i - x_{Highway(f_i)}| + |y_i - y_{Highway(f_i)}| + z_i + \\ & |x_j - x_{Highway(f_j)}| + |y_j - y_{Highway(f_j)}| + z_j + L_{Highway} \end{aligned}$$

Với chiều dài Highway nối giữa sàn  $f_i$  và sàn  $f_j$  đó là  $L_{Highway}$  và tọa độ của đầu Highway ở sàn  $f_i$  là  $(x_{Highway(f_i)}, y_{Highway(f_i)}, 0, f_i)$ , sàn  $f_j$  là  $(x_{Highway(f_j)}, y_{Highway(f_j)}, 0, f_j)$

- Khi đó công thức tính cable length là:

$$CableLength = \sum_{i=1}^m \sum_{j=1}^m w(i, j) \cdot d(i, j) \quad (7)$$

Cuối cùng ta thu được bài toán như sau: Với vùng diện tích và chi phí tối đa để triển khai trung tâm dữ liệu là biết trước. Kích thước của mạng liên kết mà bên chủ đầu tư mong muốn là  $n_0$ . Đồng thời sử dụng chiến lược đổ tràn hoặc đổ đầy, hãy tìm:

- Kích thước mạng liên kết là lớn nhất có thể với các ràng buộc trên
- Topo được sử dụng khi đó.

### 5.3 Đề xuất hướng giải quyết

Giả sử sau khi trừ đi chi phí cho rack thì ta sẽ có chi phí tối đa dành cho node (switch/host) và cable length là  $C_0$ . Tập topo được sử dụng đã xác định trước, do đó ta cần tính toán xem với topo nào thì kết quả thu được là gần nhất với mong muốn đã đặt ra. Việc tính toán đó được thực hiện như sau:

- Với mỗi số lượng node  $n$ , ta xác định được cable length (CB) tương ứng dựa theo công thức (7)
- Sau đó so sánh  $cost = (n.cost_{node} + CB.cost_{link})$  với  $C_0$ :
  - Nếu  $cost > C_0$  thì ghi nhận số lượng node max đối với topo đang xét là  $n - 1$ .
  - Nếu  $cost < C_0$  thì tiếp tục tăng  $n$  cho tới khi đạt được  $cost > C_0$ .
  - Nếu  $cost = C_0$  thì số lượng node max sẽ là  $n$ .

Sau khi đã có được số lượng node max ứng với mỗi topo, ta sẽ chọn topo mà có số lượng node lớn nhất trong số đó. Và đó là kết quả mà ta cần tìm.

## 6 Kết luận

Tài liệu đã trình bày cơ bản về mạng liên kết và bài toán triển khai mạng trung tâm dữ liệu mà cụ thể là triển khai mạng liên kết. Về khía cạnh lý thuyết của mạng liên kết còn rất nhiều vấn đề hay mà người viết chưa đề cập tới do giới hạn về mặt thời gian. Ngoài các vấn đề liên quan tới việc triển khai cơ sở hạ tầng và lắp đặt, ở mạng liên kết còn nhiều vấn đề thú vị mà bạn đọc có thể quan tâm tới đó là vấn đề về độ trễ của mạng cũng như các thuật toán định tuyến. Bạn đọc có thể tham khảo thêm trong phần tài liệu tham khảo mà người viết cung cấp.

# Tài liệu tham khảo

- [1] William James Dally & Brian Towles. *Principles and Practices of Interconnection Networks*.
- [2] José Duato, Sudhakar Yalamanchili & Lionel M. Ni *Interconnection Networks An Engineering Approach*