

## Contents

<b>CHƯƠNG 2: Trao đổi thông tin trong hệ thống phân tán</b> .....	1
<b>Dạng 1: Giải thuật Schlosser</b> .....	1
<b>Dạng 2: Lan truyền ngẫu nhiên</b> .....	2
<b>CHƯƠNG 4: Đồng bộ và các giải thuật phân tán</b> .....	3
<b>Dạng 1: Giải thuật Cristian</b> .....	3
<b>Dạng 2: Giải thuật Berkeley</b> .....	4
<b>Dạng 3: Giải thuật trung bình (giải thuật đồng bộ)</b> .....	6
<b>Dạng 4: Giải thuật tham chiếu quảng bá (đồng bộ thời gian trong mạng không dây) (giao thức RBS)</b> .....	7
<b>Dạng 5: Đồng hồ Lamport</b> .....	8
<b>Dạng 6: Đồng hồ Vector</b> .....	10
<b>Dạng 7: Xác suất k tiến trình bị lỗi trong giải thuật không tập trung</b> .....	11
<b>Dạng 8: Bầu chọn trong môi trường không dây</b> .....	12
<b>CHƯƠNG 7: Phục hồi và tính chịu lỗi</b> .....	14
<b>Dạng 1: Bài toán đồng thuận Byzantine (Đồng thuận phân tán)</b> .....	14

1ms = 10.000 ticks ?

## CHƯƠNG 2: Trao đổi thông tin trong hệ thống phân tán

### Dạng 1: Giải thuật Schlosser

Số lượng chiều được xác định bằng cách lấy số bit tối thiểu có thể biểu diễn định danh của tất cả các thành viên.

Định danh mỗi thành viên được thể hiện bằng chuỗi bit với chiều dài bằng số lượng chiều của nhóm, **mỗi cạnh trong mạng phủ được gán nhãn với giá trị của nó thể hiện vị trí của bit đã thay đổi tính từ bên trái qua phải khi so sánh chuỗi bit định danh của hai thành viên.**

VD: Các nút mạng được thiết kế theo phương pháp Schlosser, định danh các nút thể hiện trên các hàng và cột tiêu đề. Hãy điền giá trị nhãn của các cạnh đến mỗi nút láng giềng của mình.

LG

Giá trị nút lớn nhất là 37 nên cần ít nhất 6bit nhị phân biểu diễn ( $2^6 - 1 = 63$ )

Đổi các số ra 6bit, ví dụ  $32(10) = 100000$ .

Giải thích hàng 1, xét từ trái qua phải thấy rằng:

33 (10) = 100001 thay đổi tại bit số 6 so với 32.

34 (10) = 100010 thay đổi tại bit số 5 so với 32

36 (10) = 100100 thay đổi tại bit số 4 so với 32 => Đây là các nút láng giềng của 32 với giá trị nhân tương ứng.

Nút	32 (100000)	33 (100001)	34 (100010)	35 (100011)	36 (100100)	37 (100101)	38 (100110)	39 (100111)
32		6	5		4			
33	6			5		4		
34	5	6					4	
35		5	6					4
36	4					6	5	
37		4			6			5
38			4		5			6
39				4		5	6	

## Dạng 2: Lan truyền ngẫu nhiên

Nhóm gồm N thành viên, một thành viên bất kỳ bắt đầu chuyển thông điệp cho các thành viên khác, gọi  $p_i$  là xác suất thành viên P chưa nhận được thông điệp sau vòng thứ i:

Với xác suất ở vòng 1 là:  $p_1 = 1 - \frac{1}{N-1}$

- Phương pháp đẩy:  $p_{i+1} = p_i \left(1 - \frac{1}{N-1}\right)^{N(1-p_i)} = p_i p_1^{N(1-p_i)}$
- Phương pháp kéo:  $p_{i+1} = p_i^2$
- Phương pháp đẩy và kéo:  $p_{i+1} = p_i^2 p_i \left(1 - \frac{1}{N-1}\right)^{N(1-p_i)}$

(Đẩy và kéo = Đẩy x kéo,  $p_1 = \left(1 - \frac{1}{N-1}\right)^2$ , nhưng từ  $p_2$  thì tính riêng đẩy và kéo xong nhân vào ?)

Chú ý khi làm bài tập:

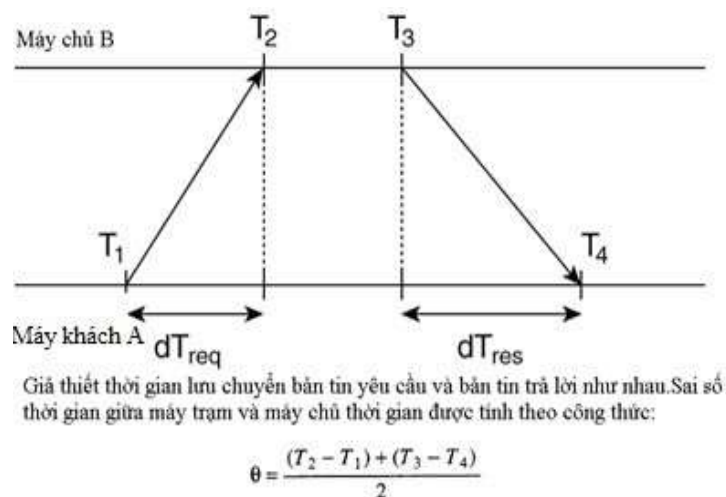
- Thường đề yêu cầu làm tròn 9 số thập phân, thêm 0 cho đủ.
- Giữ giá trị bước trước đó lại để tính xác suất tiếp, không tính theo cái đã làm tròn.

VD: 1 nhóm gồm N tiến trình được tổ chức truyền thông theo phương pháp lan truyền ngẫu nhiên. Một tiến trình trong nhóm khởi tạo lan truyền thông điệp cho tất cả tiến trình khác. Tổng số tiến trình N = 14107, kết quả làm tròn 9 số thập phân. Tính xác suất tiến trình P chưa nhận được thông điệp sau mỗi vòng lan truyền.

Vòng	Xác suất phương pháp đẩy	Xác suất phương pháp kéo	Xác suất phương pháp đẩy và kéo
1	0.999929108	0.999929108	0.999858221
2	0.999858216 (A)	0.999858221 (B)	0.999716458 (A * B)
3	0.999716448	0.999716463	0.999432991
4	0.999432986	0.999433006	....
5	0.998866394	0.998866339	....
6	0.997734594	0.997733953	....
7	0.995476639	0.995473041	....
8	0.990983432	0.990966574	....
9	0.982087382	0.982014751	....
10	0.964650409	0.964352972	....

## CHƯƠNG 4: Đồng bộ và các giải thuật phân tán

### Dạng 1: Giải thuật Cristian



Đồng bộ với các định nghĩa:

- Thời gian máy khách gửi yêu cầu:  $T_1$
- Thời gian máy chủ nhận yêu cầu:  $T_2$
- Thời gian máy chủ phản hồi:  $T_3$
- Thời gian máy khách nhận được kết quả:  $T_4$
- Độ lệch thời gian giữa máy khách và chủ (theo công thức trên ảnh)
- Thời gian bản tin di chuyển trên mạng:  $t = T_4 - T_3 + T_2 - T_1$
- Thời gian xử lý trên máy chủ:  $t = T_3 - T_2$

- Thời gian mới của máy khách sau đồng bộ:  $T = T_4 + \Theta$

VD: Máy khách đồng bộ thời gian vật lý với máy chủ NTP. Giả thiết thời gian tính toán sau khi máy khách nhận được phản hồi từ máy chủ bằng 0. Biết các tham số sau:

Thời gian máy khách gửi yêu cầu: 29/05/2024 12:39:21.116

Thời gian máy chủ gửi phản hồi: 29/05/2024 12:39:22.981

Thời gian máy chủ nhận yêu cầu: 29/05/2024 12:39:22.974

Thời gian máy khách nhận phản hồi: 29/05/2024 12:39:21.769

Hãy tính các tham số sau:

Tham số	Giá trị
Thời gian máy khách sau đồng bộ	
Độ lệch thời gian giữa chủ và khách (ms)	
Thời gian bản tin di chuyển trên mạng (ms)	
Thời gian xử lý trên máy chủ (ms)	

LG

Ngày, giờ và phút giống nhau, sau đồng bộ không làm thay đổi giá trị này nên chỉ quan tâm ms từ giây trở đi.

Từ đề:  $T_1 = 21116, T_2 = 22974, T_3 = 22981, T_4 = 21769$

Sai số thời gian giữa chủ và khách:  $\Theta = \frac{(T_2 - T_1) + (T_3 - T_4)}{2} = 1535$

Thời gian bản tin di chuyển trên mạng:  $t = T_4 - T_3 + T_2 - T_1 = 646$

Thời gian xử lý trên máy chủ:  $t = T_3 - T_2 = 7$

Thời gian máy khách sau đồng bộ:  $T = T_4 + \Theta = 23304 (23.304)$

Điền đáp án:

Tham số	Giá trị
Thời gian máy khách sau đồng bộ	29/05/2024 12:39:23.304
Độ lệch thời gian giữa chủ và khách (ms)	1535
Thời gian bản tin di chuyển trên mạng (ms)	646
Thời gian xử lý trên máy chủ (ms)	7

## Dạng 2: Giải thuật Berkeley

Nguyên lý:

- Máy chủ (thành viên điều phối) đặt thời gian trên máy của mình.
- Máy chủ gửi thời gian của mình cho các thành viên trong nhóm
- Từng thành viên trả về độ lệch thời gian giữa thành viên với máy chủ.

- Máy chủ tính trung bình của các giá trị trả về và coi đó là thời gian cần điều chỉnh.
- Máy chủ tính toán độ lệch thời gian cho mỗi thành viên và gửi yêu cầu để các thành viên cập nhật lại thời gian.

Đề cho số lượng máy khách (hoặc gộp cả máy chủ), thời gian máy chủ (điều phối) trước đồng bộ (P1) và thời gian máy khách (hoặc tiến trình) i (Pi) trước đồng bộ.

Thời gian sau đồng bộ bằng nhau trên tất cả các máy. Nên chuyển thời gian sang ms.

#### Các công thức:

- **Độ lệch thời gian mỗi i = thời gian Pi – P1**
- **Trung bình độ lệch = (tổng các độ lệch i) / (tổng số máy tính cả chủ)**
- **Độ lệch chung = trung bình độ lệch = giá trị điều chỉnh của máy chủ**
- **Giá trị điều chỉnh của i = trung bình độ lệch – độ lệch thời gian i**
- **Thời gian sau đồng bộ = thời gian Pi + giá trị điều chỉnh của i = thời gian P1**

VD: Có 5 máy khách với các tham số sau:

Thời gian máy chủ trước đồng bộ: 29/05/2024 12:26:52.986

Thời gian máy khách thứ 1 trước đồng bộ: 29/05/2024 12:28:00.600

Thời gian máy khách thứ 2 trước đồng bộ: 29/05/2024 12:27:39.828

Thời gian máy khách thứ 3 trước đồng bộ: 29/05/2024 12:27:54.000

Thời gian máy khách thứ 4 trước đồng bộ: 29/05/2024 12:26:22.098

Thời gian máy khách thứ 5 trước đồng bộ: 29/05/2024 12:26:46.068

Sử dụng giải thuật Berkeley để đồng bộ thời gian, điền đáp án vào bảng sau:

Tham số	Đáp án
Thời gian máy khách 1 cần điều chỉnh (ms)	
Thời gian máy khách 2 cần điều chỉnh (ms)	
Độ lệch chung (ms)	
Độ lệch thời gian của máy khách 1 với máy chủ (ms)	
Độ lệch thời gian của máy khách 2 với máy chủ (ms)	
Thời gian sau đồng bộ	

LG

Ngày và giờ giống nhau, sau đồng bộ theo trung bình ước chừng không làm thay đổi giá trị này nên không cần quan tâm.

Máy	Phút	Giây	Ms	Độ lệch	Điều chỉnh	Thời gian sau đồng bộ
Chủ	26	52.986	1612986	0	22944	

Khách 1	28	00.600	1680600	$1680600 - 1612986 =$ 67614	22944 – 67614 = -44670	1612986 + 22944 = 1635930 (27:15.930)
Khách 2	27	39.828	1659828	$1659828 - 1612986 =$ 46842	22944 – 46842 = -23898	
Khách 3	27	54.000	1674000	61014	-38070	
Khách 4	26	22.098	1582098	- 30888	53832	
Khách 5	26	46.068	1606068	- 6918	29862	
Trung bình độ lệch (Làm tròn hết)				$(0 + 67614 + 46842 +$ $... + -6918) / 6 = 22944$		

Điền đáp án:

Tham số	Đáp án
Thời gian máy khách 1 cần điều chỉnh (ms)	-44670
Thời gian máy khách 2 cần điều chỉnh (ms)	-23898
Độ lệch chung (ms)	22944
Độ lệch thời gian của máy khách 1 với máy chủ (ms)	67614
Độ lệch thời gian của máy khách 2 với máy chủ (ms)	46842
Thời gian sau đồng bộ	29/05/2024 12:27:15.930

### Dạng 3: Giải thuật trung bình (giải thuật đồng bộ)

Giải thuật trung bình giả thiết không tồn tại thành viên điều phối và cũng không đòi hỏi tất cả các thành viên đều phải tham gia, các thành viên trong hệ thống thống nhất chu kỳ đồng bộ thời gian, khoảng thời gian thứ  $i$  sẽ bắt đầu từ thời điểm  $T_0 + iR$  và chạy đến khi  $T_0 + (i+1)R$ , trong đó  $T_0$  là thời điểm xác định trước và  $R$  là khoảng thời gian của 1 chu kỳ đồng bộ.

**Ở mỗi chu kỳ, các thành viên quảng bá thời gian của mình, sau đó bắt đầu nhận bản tin quảng bá từ các thành viên khác trong khoảng thời gian  $S$ . Cách đơn giản nhất để tính thời gian mới là lấy trung bình của tất cả các giá trị nhận được, để tránh những giá trị gây nhiễu có thể loại bỏ  $M$  giá trị cao nhất và thấp nhất trước khi tính trung bình.**

VD: Tại 1 thời điểm các tiến trình đồng loạt quảng bá thời gian của mình cho các tiến trình khác trong nhóm, giả thuyết sau đó mỗi tiến trình đều nhận được các bản tin này. Thời gian đã quảng bá như sau:

Tiến trình 1: 22/04/2024 15:04:11.690

Tiến trình 2: 22/04/2024 14:55:08.000

Tiến trình 3: 22/04/2024 15:01:11.750

Tiến trình 4: 22/04/2024 14:58:07.940

Tiến trình 5: 22/04/2024 15:07:11.810

Tiến trình 6: 22/04/2024 15:01:07.880

Hãy tính thời gian cho mỗi tiến trình sau khi thực hiện giải thuật đồng bộ.

LG

Quan tâm giờ, phút và giây.

Tiến trình	Trước đồng bộ	Trước đồng bộ (ms)	Các bản tin được tính trung bình (loại chính nó và tmax, tmin)	Giá trị trung bình sau đồng bộ (ms)	Thời gian sau đồng bộ
1	15:04:11.690	54251690	3, 4, 6	$(54071750 + 53887940 + 54067880) / 3 = 54009190$	15:00:09.190
2	14:55:08.000	53708000	1, 3, 6	$(54251690 + 54071750 + 54067880) / 3 = 54130440$	15:02:10.440
3	15:01:11.750	54071750	1, 4, 6		
4	14:58:07.940	53887940	1, 3, 6		
5	15:07:11.810	54431810	3, 4, 6		
6	15:01:07.880	54067880	1, 3, 4		

Làm tương tự từ 3 -> 6, chú ý lúc điền phải thêm ngày nữa.

#### **Dạng 4: Giải thuật tham chiếu quảng bá (đồng bộ thời gian trong mạng không dây) (giao thức RBS)**

Một thành viên quảng bá thông điệp tham chiếu m, thông điệp này không chứa thời gian của nó mà chỉ là tín hiệu yêu cầu các thành viên thực hiện đồng bộ.

Độ lệch trung bình giữa 2 thành viên P, Q được tính theo công thức:

$$Offset[p, q] = \frac{\sum_{k=1}^M (T_{p,k} - T_{q,k})}{M}$$

**Với:**

- M là số lần gửi tín hiệu đồng bộ
- $T_{p,k}$  là thời gian nhận được bản tin tại nút P
- $T_{q,k}$  là thời gian nhận được bản tin tại nút Q

VD: Tiến trình P gửi tín hiệu đồng bộ thời gian vật lý đến tiến trình Q. Mỗi khi nhận được bản tin, tiến trình Q phản hồi thời gian nó nhận được bản tin, P cũng lưu lại thời gian nhận được bản tin từ Q.

Các mốc thời gian trao đổi bản tin như sau:

Q nhận từ P lúc 2024-05-29 12:48:09.780 và P nhận từ Q lúc 2024-05-29 12:48:10.230

Q nhận từ P lúc 2024-05-29 12:48:10.740 và P nhận từ Q lúc 2024-05-29 12:48:13.150

Q nhận từ P lúc 2024-05-29 12:48:13.510 và P nhận từ Q lúc 2024-05-29 12:48:14.830

Q nhận từ P lúc 2024-05-29 12:48:15.770 và P nhận từ Q lúc 2024-05-29 12:48:16.640

Q nhận từ P lúc 2024-05-29 12:48:18.180 và P nhận từ Q lúc 2024-05-29 12:48:20.970

Q nhận từ P lúc 2024-05-29 12:48:22.760 và P nhận từ Q lúc 2024-05-29 12:48:24.150

Q nhận từ P lúc 2024-05-29 12:48:24.960 và P nhận từ Q lúc 2024-05-29 12:48:27.780

Q nhận từ P lúc 2024-05-29 12:48:28.910 và P nhận từ Q lúc 2024-05-29 12:48:28.910

Q nhận từ P lúc 2024-05-29 12:48:30.290 và P nhận từ Q lúc 2024-05-29 12:48:31.590

Q nhận từ P lúc 2024-05-29 12:48:32.580 và P nhận từ Q lúc 2024-05-29 12:48:34.490

Tính độ lệch trung bình giữa P và Q.

LG

Chỉ quan tâm giây và ms vì ngày, giờ và phút không đổi.

P nhận từ Q ( $T_{p,k}$ )	Q nhận từ P ( $T_{q,k}$ )	Chênh lệch
10230	9780	450
13150	10740	2410
14830	13510	1320
16640	15770	870
20970	18180	2790
24150	22760	1390
27780	24960	2820
28910	28910	0
31590	30290	1300
34490	32580	1910
Trung bình độ lệch		$(450 + 2410 + 1320 + \dots + 1910) / 10 = 1526$

## Dạng 5: Đồng hồ Lamport

Cập nhật bộ đếm  $C_i$  cho tiến trình  $P_i$ :

- Trước khi thực hiện  $P_i$  gán  $C_i \leftarrow C_i + 1$
- Khi tiến trình  $P_i$  gửi thông điệp  $m$  cho tiến trình  $P_j$  nó đặt nhãn thời gian của thông điệp  $ts(m) = C_i$
- Khi nhận được thông điệp  $m$ , tiến trình  $P_j$  điều chỉnh bộ đếm của nó thành  $C_j \leftarrow \max\{C_j, ts(m)\}$  và chuyển lên tầng ứng dụng.



**Đề cho  $E(x, y)$ , tức sự kiện thứ  $y$  trên tiến trình  $x$ .**

VD: Thông điệp giữa các tiến trình:

$E(1,2)$  gửi và  $E(2,2)$  nhận;  $E(2,3)$  gửi và  $E(3,3)$  nhận;

$E(3,4)$  gửi và  $E(2,4)$  nhận;  $E(2,5)$  gửi và  $E(1,5)$  nhận.

Bộ đếm thời gian của mỗi tiến trình:

Tiến trình $x$	$E(x,1)$	$E(x,2)$	$E(x,3)$	$E(x,4)$	$E(x,5)$
P1	20	40	60	80	100
P2	15	30	45	60	75
P3	12	24	36	48	60

Sử dụng đồng bộ nhãn thời gian Lamport.

LG

Xét sự kiện  $E(1,2)$  gửi và  $E(2,2)$  nhận, vì bên nhận  $30 \leq 40 \Rightarrow$  thay giá trị nhãn thời gian thành  $40 + 1$  và tăng giá trị các nhãn đằng sau thêm  $41 - 30 = 11$ :

Tiến trình $x$	$E(x,1)$	$E(x,2)$	$E(x,3)$	$E(x,4)$	$E(x,5)$
P1	20	40	60	80	100
P2	15	41	56	71	86
P3	12	24	36	48	60

Xét sự kiện  $E(2,3)$  gửi và  $E(3,3)$  nhận, vì bên nhận  $36 \leq 56 \Rightarrow$  thay giá trị nhãn thời gian thành  $56 + 1$  và tăng giá trị các nhãn đằng sau thêm  $57 - 36 = 21$ :

Tiến trình $x$	$E(x,1)$	$E(x,2)$	$E(x,3)$	$E(x,4)$	$E(x,5)$
P1	20	40	60	80	100
P2	15	41	56	71	86
P3	12	24	57	69	81

Xét sự kiện  $E(3,4)$  gửi và  $E(2,4)$  nhận, vì bên nhận  $71 > 69$  nên giữ nguyên

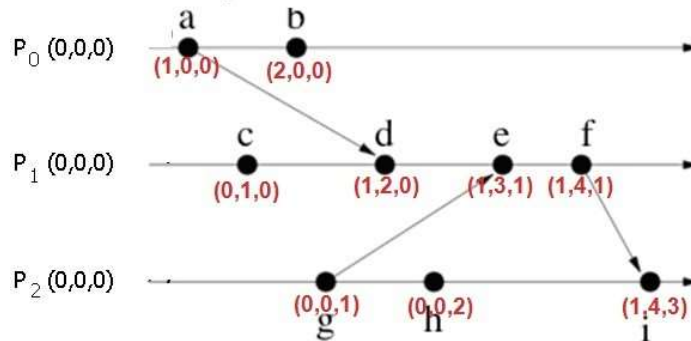
Xét sự kiện  $E(2,5)$  gửi và  $E(1,5)$  nhận, vì bên nhận  $100 > 86$  nên giữ nguyên.

Kết quả đồng bộ cuối:

Tiến trình $x$	$E(x,1)$	$E(x,2)$	$E(x,3)$	$E(x,4)$	$E(x,5)$
P1	20	40	60	80	100
P2	15	41	56	71	86
P3	12	24	57	69	81

## Dạng 6: Đồng hồ Vector

Nếu sử dụng chỉ mục cho mỗi tiến trình thì có thể biểu diễn lịch sử nhân quả dưới dạng vector, số lượng phần tử của vector bằng số lượng tiến trình và mỗi phần tử tương ứng với một tiến trình, giá trị ban đầu của các phần tử thường đặt bằng 0.



Mỗi tiến trình  $P_i$  duy trì một vector  $V_{Ci}$ , giá trị mỗi phần tử cho biết số lượng sự kiện đã xảy ra trên tất cả các tiến trình của hệ thống. Giá trị  $V_{Ci}[i]$  thể hiện số sự kiện đã xảy ra cho đến thời điểm hiện tại trên tiến trình  $P_i$ , nếu  $V_{Ci}[j] = k$  thì  $P_i$  hiểu rằng đã có  $k$  sự kiện xảy ra trên tiến trình  $P_j$ .

Thuật toán nhân thời gian Vector:

- Mỗi tiến trình  $P_i$  duy trì  $n$ -vector  $V_i$ , ban đầu tất cả phần tử bằng 0.
- Mục  $i$  trong  $V_i$  là số bước thực hiện trong tiến trình  $P_i$ .
- Tại mỗi bước, giá trị  $V_i[i]$  được tăng thêm:  $V_i[i] \leftarrow V_i[i] + 1$ , điều đó thể hiện đã có một sự kiện xảy ra.
- Khi tiến trình  $P_i$  gửi thông điệp  $m$  cho tiến trình  $P_j$ , nó đính kèm nhãn thời gian vector  $ts(m)=V_i$ .
- Tiến trình  $P_j$  nhận được thông điệp, giá trị mới của phần tử thứ  $k$  trong nhãn thời gian vector của tiến trình  $P_j$  sẽ là  $V_j[k] \leftarrow \max\{V_j[k], ts(m)[k]\}$ .

**Thường đề cho:** Nhãn thời gian của mỗi tiến trình được đặt theo cấu trúc  $(P1, ..., Pn)$  với  $n$  là số lượng tiến trình. Các sự kiện trên mỗi tiến trình thể hiện như sau:

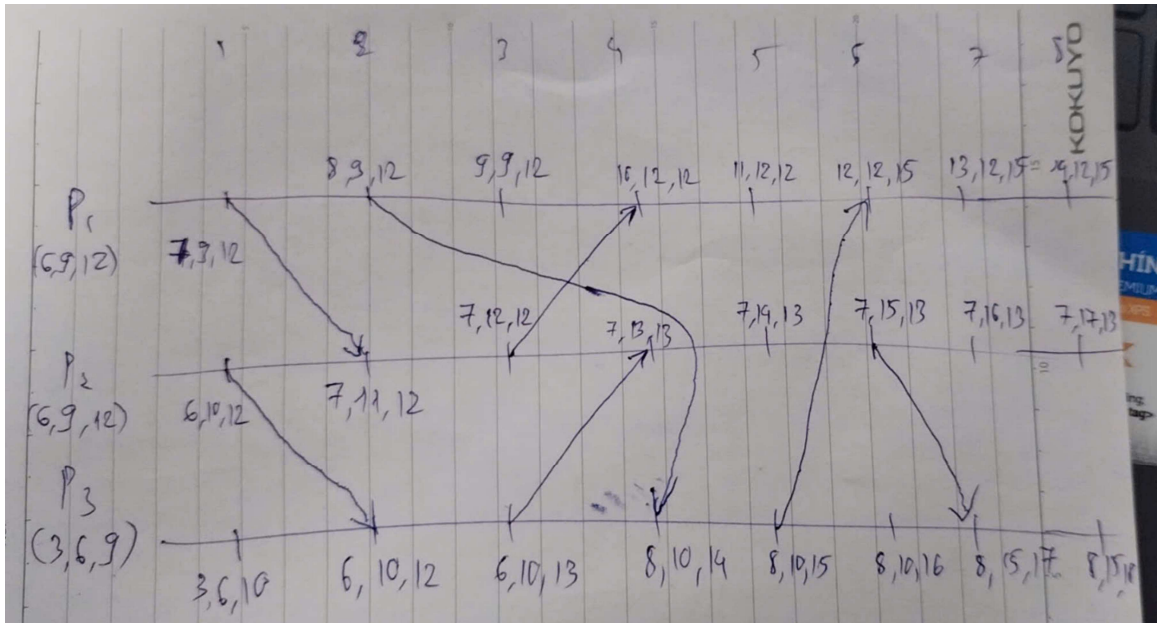
- $eX, Y$  thể hiện sự kiện.
- $X$  là số hiệu tiến trình.
- $Y$  là số thứ tự sự kiện trong tiến trình

VD:

- e2,1 gửi và e3,2 nhận
- e1,1 gửi và e2,2 nhận
- e3,3 gửi và e2,4 nhận
- e1,2 gửi và e3,4 nhận
- e2,3 gửi và e1,4 nhận
- e3,5 gửi và e1,6 nhận
- e2,6 gửi và e3,7 nhận

### Đáp án

TT	eX,0	eX,1	eX,2	eX,3	eX,4	eX,5	eX,6	eX,7	eX,8
P3	3,6,9	3,6,10	6,10,12	6,10,13	8,10,14	8,10,15	8,10,16	8,15,17	8,15,18
P2	6,9,12	6,10,12	7,11,12	7,12,12	7,13,13	7,14,13	7,15,13	7,16,13	7,17,13
P1	6,9,12	7,9,12	8,9,12	9,9,12	10,12,12	11,12,12	12,12,15	13,12,15	14,12,15



### Dạng 7: Xác suất k tiến trình bị lỗi trong giải thuật không tập trung

Dựa trên thuật toán bầu chọn:

- Mỗi tài nguyên sẽ được nhân bản N lần và mỗi bản sao sẽ có tiến trình điều phối riêng để kiểm soát truy nhập.
- Một tiến trình muốn truy nhập tài nguyên thì phải gửi yêu cầu đến N tiến trình điều phối đang hoạt động và sẽ được cấp quyền truy nhập  $\Leftrightarrow$  nhận được số phiếu đồng ý lớn hơn  $N/2$

Gọi p là xác suất bị lỗi của mỗi tiến trình điều phối, xác suất k trong m tiến trình điều phối bị lỗi sẽ là:

$$P(k) = C_m^k p^k (1 - p)^{m - k}$$

VD: Tổng số tiến trình là 15, xác suất mỗi tiến trình lỗi là 0.296560239. Tính xác suất 5 tiến trình bị lỗi (làm tròn 9 số thập phân)

$$P(5) = C_{15}^5 p^5 (1 - p)^{10} = 0.204356791$$

### Dạng 8: Bầu chọn trong môi trường không dây

Giải thuật tổ chức các tiến trình thành 1 cây logic, trong đó giữa 2 nút bất kỳ chỉ có 1 đường đi duy nhất. Giải thuật chia làm hai giai đoạn, thứ nhất là quảng bá thông điệp “Bầu chọn”, thứ hai là phản hồi thông điệp “Bầu chọn”.

Các bước:

- Khởi đầu bầu chọn:
  - Bất cứ tiến trình cũng có quyền khởi tạo bầu chọn, các tiến trình không cần biết định danh của các tiến trình khác trong hệ thống.
  - Nó gửi thông điệp "Bầu chọn" đến tất cả các tiến trình láng giềng (tức là có kết nối mạng trực tiếp).
- Nhận thông điệp bầu chọn:
  - Nếu đây là lần đầu nhận thông điệp này thì đánh dấu tiến trình gửi là cha và gửi tiếp “Bầu chọn” đến các láng giềng khác trừ cha và những tiến trình đã bị từ chối làm cha.
  - Nếu không phải lần đầu thì phản hồi đã nhận thông điệp, không nhận tiến trình gửi làm cha và vẫn gửi thông điệp cho các láng giềng khác.
- Phản hồi kết quả bầu chọn:
  - Nếu là tiến trình lá thì gửi định danh của mình cho tiến trình cha.
  - Nếu là tiến trình trung gian thì nó sẽ so sánh định danh của mình với định danh của các con để chọn tiến trình đại diện (có định danh cao nhất) cho nhánh của nó, sau đó gửi lên cha của nó.
- Tiến trình khởi sướng xác định được tiến trình nào được bầu chọn và quảng bá thông điệp kết quả bầu chọn cho tất cả các thành viên của hệ thống.

VD: Cho 20 tiến trình trong môi trường mạng không dây, mỗi tiến trình thể hiện bằng cặp tên và giá trị tham gia bầu chọn của tiến trình đó kèm theo mối quan hệ cha con với các tiến trình khác như sau:

tiến trình	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
nguồn gốc	13	19	16	22	25	30	26	31	36	41	35	40	47	52	44	49	57	61	53	55
cha	P1	P1	P2	P2	P2	P3	P3	P3	P4	P4	P4	P4	P5	P5	P5	P6	P6	P6	P6	P9

Nguồn từ tiến trình P1. Điền các thông điệp bầu chọn theo định dạng cặp tên tiến trình và giá trị bầu chọn phân cách bằng dấu phẩy (ví dụ P5,10)

LG

Dạng này có 2 cách điền đáp án, đối với cách 1 thì quy định hàng là tiến trình gửi, cột là tiến trình nhận, tùy đề.

Cách 1:

Tiến trình	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
P1																		P18,61		
P2	P18,61																			
P3	P20,55																			
P4		P13,47																		
P5		P14,52																		
P6		P18,61																		
P7			P7,26																	
P8			P8,31																	
P9			P20,55																	
P10				P10,41																
P11				P11,35																
P12				P12,40																
P13				P13,47																
P14					P14,52															
P15					P15,44															
P16					P16,49															
P17						P17,57														
P18						P18,61														
P19						P19,53														
P20									P20,55											

Cách 2:

Thông điệp	Tiến trình bầu chọn
Tiến trình được chọn	P18,61
P1	P18,61
P1	P20,55
P2	P13,47
P2	P14,52
P2	P18,61
P3	P7,26
P3	P8,31
P3	P20,55
P4	P10,41
P4	P11,35
P4	P12,40
P4	P13,47
P5	P14,52
P5	P15,44
P5	P16,49
P6	P17,57
P6	P18,61
P6	P19,53
P9	P20,55

## CHƯƠNG 7: Phục hồi và tính chịu lỗi

### Dạng 1: Bài toán đồng thuận Byzantine (Đồng thuận phân tán)

Giả sử hệ thống gồm N tiến trình, mỗi tiến trình  $i$  cung cấp giá trị  $V_i$  cho các tiến trình khác, mục tiêu cần đạt trong đồng thuận lỗi sẽ được thực hiện bằng cách cho phép mỗi tiến trình xây dựng vector  $V$  gồm N phần tử với  $V[i]=V_i$  nếu tiến trình  $i$  không lỗi và  $V[i]$  không xác định nếu tiến trình  $i$  bị lỗi. Quá trình tiến tới sự đồng thuận sẽ được thực hiện theo 4 bước sau:

1. Mỗi tiến trình  $i$  không lỗi gửi giá trị  $V_i$  cho các tiến trình khác sử dụng phương pháp truyền thông điểm-điểm tin cậy, tiến trình lỗi gửi giá trị bất kỳ.
2. Kết quả nhận được từ bước 1 sẽ tập hợp lại thành vector  $V$ .
3. Mỗi tiến trình gửi vector  $V$  của bước 2 cho các tiến trình khác.
4. Mỗi tiến trình kiểm tra phần tử thứ  $i$  trong các Vector nhận được ở bước 3, nếu kết quả kiểm tra chiếm đa số thì đặt giá trị vào Vector kết quả đồng thuận, nếu không thì đặt giá trị không xác định UNKNOWN.

VD: Điền giá trị các bước thực hiện giải thuật đồng thuận phân tán cho các tiến trình sau:

Tiến trình	1	2	3	4
Giá trị	180	281	382	483
Trạng thái	Không lỗi	Lỗi	Không lỗi	Không lỗi

Giả thiết tiến trình lỗi nhận được thông điệp từ các tiến trình không lỗi nhưng không gửi được thông điệp cho các tiến trình khác. Thứ tự giá trị được lưu trữ theo số thứ tự tiến trình tăng dần

LG

Các kết quả	Tiến trình 1	Tiến trình 2	Tiến trình 3	Tiến trình 4
Kết quả lần thứ nhất	180,Nil,382,483	180,281,382,483	180,Nil,382,483	180,Nil,382,483
Kết quả lần thứ hai 1	Nil,Nil,Nil,Nil	180,Nil,382,483	180,Nil,382,483	180,Nil,382,483
Kết quả lần thứ hai 2	180,Nil,382,483	180,Nil,382,483	Nil,Nil,Nil,Nil	Nil,Nil,Nil,Nil
Kết quả lần thứ hai 3	180,Nil,382,483	180,Nil,382,483	180,Nil,382,483	180,Nil,382,483
Kết quả cuối cùng	180,Unknown,382,483	180,Unknown,382,483	180,Unknown,382,483	180,Unknown,382,483

