

# Lời nói đầu

Trong thực tế, nhiều khách hàng phải xếp thành hàng để đợi mua vé, các cuộc gọi điện thoại phải đợi để được liên lạc tại các tổng đài điện thoại và các tác vụ có thể phải đợi để nhận được điều khiển của CPU trong các máy tính. Trong một mạng máy tính, nhiều người cùng chia sẻ tài nguyên nhưng chỉ có thể sử dụng tài nguyên cho mỗi công việc tại mỗi thời điểm, vì thế các công việc khác phải xếp xếp hàng. Các gói tin có thể phải đợi trong các bộ đệm tại một nút mạng nào đó trên mạng máy tính. Tất cả các ví dụ trên đã và đang được nghiên cứu nhờ sử dụng một lý thuyết toán học của các xếp hàng có tên là lý thuyết xếp hàng (lý thuyết hàng chờ). Lý thuyết xếp hàng giúp ta xác định được các số đo hiệu năng của các hàng xếp.

Lý thuyết xếp hàng có nguồn gốc từ đầu thế kỷ 20 với các nghiên cứu khởi đầu của nhà toán học người Đan Mạch A. K. Erlang trên các mạng điện thoại và sự sáng tạo của các mô hình Markov do nhà toán học người Nga A. A. Markov. Ngày nay lý thuyết xếp hàng được sử dụng rộng rãi cho các ứng dụng khác nhau và nó vẫn đang được nghiên cứu và phát triển.

Lý thuyết này có thể áp dụng vào giải quyết các bài toán kinh tế, những bài toán phụ thuộc vào các yếu tố ngẫu nhiên. Lý thuyết này cũng phù hợp với các kiến thức cơ sở của môn học "Các mô hình ngẫu nhiên và ứng dụng" do cô giáo TS Nguyễn Thị Ngọc Anh trực tiếp giảng dạy. Vì vậy,

---

em đã chọn đề tài "Áp dụng lý thuyết xếp hàng vào giải quyết bài toán Bán vé tàu" để làm đề tài nghiên cứu cho bản thân. Nội dung chính của tiểu luận gồm:

*Chương 1: Giới thiệu lý thuyết hàng chờ*

*Chương 2: Giải quyết bài toán bán vé tàu*

*Chương 3: Chương trình*

Do khả năng có hạn nên tiểu luận không tránh khỏi những thiếu sót và hạn chế. Rất mong nhận được sự góp ý và chỉ bảo của Cô giáo để nhóm có thể hoàn chỉnh hơn nữa những hiểu biết của mình.

# Mục lục

## Lời nói đầu

<b>1</b>	<b>Kiến thức cơ sở</b>	<b>1</b>
1.1	Biến ngẫu nhiên . . . . .	1
1.2	Phân phối Poisson . . . . .	1
1.2.1	Định nghĩa . . . . .	1
1.2.2	Giá trị đặc trưng của phân phối Poisson . . . . .	2
1.3	Phân phối mũ . . . . .	2
1.3.1	Định nghĩa . . . . .	2
1.3.2	Hàm phân phối xác suất . . . . .	2
1.3.3	Các giá trị đặc trưng . . . . .	3
<b>2</b>	<b>Lý thuyết xếp hàng</b>	<b>4</b>
2.1	Giới thiệu chung . . . . .	4
2.2	Định nghĩa . . . . .	4
2.3	Các phương pháp giải mô hình hàng chờ . . . . .	7
2.4	Các yếu tố cơ bản của hệ thống hàng chờ . . . . .	8
2.4.1	Bố trí vật lý của hệ thống . . . . .	9
2.4.2	Nguyên tắc phục vụ . . . . .	10
2.4.3	Các phân phối xác suất của các dòng tín hiệu, dòng phục vụ . . . . .	11
2.5	Một số đặc trưng của hệ thống xếp hàng . . . . .	12
2.6	Một số điểm hạn chế của các mô hình hàng chờ . . . . .	13
<b>3</b>	<b>Áp dụng giải quyết bài toán bán vé tàu</b>	<b>15</b>
3.1	Phát biểu bài toán . . . . .	15
3.2	Hệ thống xếp hàng M/M/C/K . . . . .	15
3.3	Giải quyết bài toán . . . . .	16
3.3.1	Các đại lượng của mô hình M/M/1/K . . . . .	17

3.3.2	Các đại lượng của mô hình M/M/C/K . . . . .	18
<b>4</b>	<b>Chương trình</b>	<b>20</b>
4.1	Một số mã nguồn chính . . . . .	20
4.1.1	Khởi tạo . . . . .	20
4.1.2	Thiết lập ma trận sinh G . . . . .	21
4.1.3	Tìm phân phối dừng $\pi_i$ . . . . .	23
4.1.4	Tính số khách trung bình toàn bộ hệ thống . . . . .	24
4.1.5	Lợi nhuận . . . . .	24
4.2	Chạy chương trình . . . . .	25
	<b>Kết luận</b>	<b>26</b>
	<b>Tài liệu tham khảo</b>	<b>27</b>

# Chương 1

## Kiến thức cơ sở

### 1.1 Biến ngẫu nhiên

Biến ngẫu nhiên là một khái niệm quan trọng trong xác suất thống kê. Một cách giản lược, biến ngẫu nhiên còn gọi là đại lượng ngẫu nhiên, được hiểu là biến nhận giá trị tùy thuộc vào kết quả của phép thử (phép đo, quan sát, thí nghiệm) mà không thể đoán trước được.

Biến ngẫu nhiên chia làm hai loại: rời rạc và liên tục. Biến rời rạc là biến ngẫu nhiên nhận các giá trị từ một tập hợp hữu hạn hoặc đếm được. Biến liên tục là biến ngẫu nhiên nhận giá trị trên một tập con liên tục của tập số thực.

### 1.2 Phân phối Poisson

#### 1.2.1 Định nghĩa

**Định nghĩa 1.1.** Một biến ngẫu nhiên  $X$  được gọi là có phân bố Poisson với tham số  $\lambda > 0$ , ký hiệu:  $X \sim \rho(X)$ , nếu  $X$  có hàm trọng số xác suất:

$$P_X(x) = P(X = x) = \frac{\lambda^x}{x!} e^{-\lambda}, \quad x = 0, 1, \dots$$

### 1.2.2 Giá trị đặc trưng của phân phối Poisson

- Kỳ vọng:

$$\begin{aligned} E(X) &= \sum_{x=0}^{+\infty} x \cdot P_X(x) = \sum_{x=0}^{+\infty} x \cdot \frac{\lambda^x}{x!} e^{-\lambda} \\ &= \sum_{x=0}^{+\infty} \frac{\lambda^x}{(x-1)!} e^{-\lambda} = \lambda \cdot e^{-\lambda} \sum_{x=0}^{+\infty} \frac{\lambda^{x-1}}{(x-1)!} \end{aligned}$$

Đặt  $n = x - 1$ , ta có:

$$E(X) = \lambda \cdot e^{-\lambda} \sum_{n=0}^{+\infty} \frac{\lambda^n}{n!} = \lambda \cdot e^{-\lambda} \cdot e^{\lambda} = \lambda$$

- Phương sai:

$$V(X) = E(X^2) - (E(X))^2 = E(X^2) - \lambda^2$$

$$E(X^2) = \sum_{x=0}^{+\infty} x^2 \cdot \frac{\lambda^x}{x!} e^{-\lambda} = \lambda^2 + \lambda$$

$$V(X) = \lambda^2 + \lambda - \lambda^2 = \lambda$$

## 1.3 Phân phối mũ

### 1.3.1 Định nghĩa

**Định nghĩa 1.2.** Biến ngẫu nhiên  $X$  được gọi là có phân phối mũ với tham số  $\mu > 0$ , ký hiệu:  $X \sim \text{Exp}(\mu)$  và hàm mật độ có dạng:

$$f_X(x) = \begin{cases} \mu \cdot e^{-\mu x} & , \quad x \geq 0 \\ 0 & , \quad x < 0 \end{cases}$$

### 1.3.2 Hàm phân phối xác suất

Hàm phân phối xác suất được định nghĩa bởi hàm sau:

$$F_X(x) = P(X < x) = \int_{-\infty}^{+\infty} f_X(t) dt$$

- $x < 0 : F_X(x) = \int_{-\infty}^{+\infty} 0dt = 0$

- $x \geq 0 : F_X(x) = \int_{-\infty}^0 0dt + \int_0^x \mu \cdot e^{-\mu t} dt = 0 - e^{-\mu t} \Big|_0^x = -e^{-\mu x} + 1$

Vậy  $F_X(x) = \begin{cases} 1 - e^{-\mu x} & , \ x \geq 0 \\ 0 & , \ x < 0 \end{cases}$

### 1.3.3 Các giá trị đặc trưng

- Kỳ vọng:

$$E(X) = \int_{-\infty}^{+\infty} x \cdot f_X(x) dx = \int_{-\infty}^0 x \cdot 0 dx + \int_0^{+\infty} x \cdot e^{-\mu x} dx = \frac{1}{\mu}$$

- Phương sai:

$$V(X) = E(X^2) - (E(X))^2 = \int_0^{+\infty} x^2 \cdot e^{-\mu x} dx - \frac{1}{\mu^2} = \frac{2}{\mu^2} - \frac{1}{\mu^2} = \frac{1}{\mu^2}$$

## Chương 2

# Lý thuyết xếp hàng

### 2.1 Giới thiệu chung

Lý thuyết xếp hàng là một trong các công cụ toán học mạnh mẽ cho việc phân tích, ước lượng trong các hệ thống xếp hàng. Lý thuyết xếp hàng thông thường được áp dụng cho các hệ thống lý tưởng để đưa ra các kết quả gần đúng cho một mô hình thực tế. Tính chất chung của các giải pháp ứng dụng lý thuyết xếp hàng là làm rõ lưu lượng dòng vào, để cung cấp dự báo những danh giới lớn hơn trên những kết quả nghiên cứu thu được. Chúng rất hữu ích cho việc xác định tính đúng đắn của các phương pháp.

### 2.2 Định nghĩa

Lý thuyết xếp hàng là một phần của lý thuyết xác suất thống kê, nó được định nghĩa là một bộ các quy tắc và luật đề cập đến việc tắc nghẽn và phương pháp giải quyết tắc nghẽn như: dự đoán độ trễ, trễ nhỏ nhất, độ dài hàng chờ, số kênh phục vụ cần thiết,... Lý thuyết xếp hàng có rất nhiều ứng dụng từ việc nghiên cứu xe cộ lưu thông trên đường phố, phục vụ khách hàng tại các siêu thị, nhà hàng...

Một quá trình xếp hàng là:



- Dòng khách hàng tới
- Khả năng phục vụ của server
- Nếu khách hàng tới chưa được phục vụ thì sẽ xếp hàng chờ
- Hệ thống xếp hàng gồm khách hàng trong xếp hàng và khách hàng đang được phục vụ.

Chuẩn ký hiệu cho hệ thống xếp hàng được sắp xếp như sau  
 Dòng vào / dòng phục vụ / số lượng server / số lượng khách hàng lớn nhất trong hệ thống / quy tắc xếp hàng.

- Dòng vào: lượng khách tới hệ thống,
- Dòng phục vụ: lượng khách được phục vụ xong ra khỏi server,
- Số server: số kênh phục vụ,
- Lượng khách lớn nhất: tổng lượng khách đang phục vụ và trong hàng chờ,
- Quy tắc xếp hàng: Một số nguyên tắc phục vụ thường được áp dụng trong các hệ thống xếp hàng là FIFO (First in first out), LIFO (Last in first out), FCFS (First come first serve), có ưu tiên, không ưu tiên, Random Order...

Trong các hệ thống hàng chờ thường xuyên diễn ra hai quá trình: quá trình này sinh các yêu cầu (một yêu cầu còn được coi là một tín hiệu cần được phục vụ) và quá trình phục vụ các yêu cầu ấy. Song trong quá trình phục vụ của các hệ thống, do nhiều nguyên nhân khác nhau, thường xảy ra các tình trạng sau: Trong nhiều trường hợp, quá trình phục vụ không đáp ứng các yêu cầu và do đó dẫn đến kết quả là nhiều yêu cầu phải chờ để được phục vụ. Ngược lại, trong một số tình huống khác, khả năng phục vụ của hệ thống vượt quá số yêu cầu cần được phục vụ, với kết quả là hệ thống không sử dụng hết phương tiện phục vụ.

Người quản trị hệ thống phải xác định cho được những chi phí vô ích. Những chi phí vô ích này tạo thành tính không hiệu quả của hệ thống. Có

hai dạng chi phí vô ích:

- Chi phí của khách hàng phải chờ trong hệ thống trước khi được phục vụ. Chi phí này có thể hiểu được một cách tương đương là trong cùng một khoảng thời gian quản lý  $T$ , nếu khách hàng chờ lâu thì lượng khách hàng chờ tới trong khoảng thời gian  $T$  giảm.
- Chi phí cho các trạm phục vụ khách hàng nhưng lại không có khách hàng. Như vậy trong khoảng thời gian quản lý  $T$ , tỷ lệ thời gian phục vụ khách hàng tạo thành hiệu suất  $U$  của một trạm phục vụ. Hiệu suất càng gần 1 thì chi phí vô ích càng nhỏ và ngược lại, nếu hiệu suất gần bằng 0 thì chi phí vô ích càng lớn.

Đây là hai loại chi phí ngược nhau: nếu giảm chi phí vô ích từ phía khách hàng tức là giảm thời gian chờ của khách hàng thì phải tăng số trạm phục vụ; và như vậy làm tăng chi phí vô ích từ phía phục vụ. Ngược lại nếu muốn giảm chi phí vô ích từ phía phục vụ thì phải giảm số trạm phục vụ nhưng điều này lại làm tăng thời gian chờ của khách hàng. Rõ ràng muốn tăng tính hiệu quả hoạt động của hệ thống thì cần phải cân đối tổng thể từ hai loại chi phí này.

Vì vậy bài toán đặt ra là:

- Phân tích bản chất của quá trình diễn ra trong các hệ thống hàng chờ và thiết lập các mối liên hệ về lượng giữa các đặc trưng của các quá trình ấy. Điều đó có nghĩa là cần thiết lập hay lựa chọn một mô hình hàng chờ phản ánh được bản chất của hệ thống.
- Trên cơ sở các mối liên hệ đã được xây dựng và các số liệu thu được từ hệ thống, cần tính toán, phân tích và đưa ra các quyết định nhằm tìm ra các giá trị thích hợp cho các tham số điều khiển / thiết kế của hệ thống để thiết kế hay điều khiển các hoạt động của hệ thống hoạt động một cách có hiệu quả hơn.

## 2.3 Các phương pháp giải mô hình hàng chờ

Để tìm lời giải cho một mô hình hàng chờ người ta thường sử dụng hai phương pháp: phương pháp giải tích và phương pháp mô phỏng trên máy tính.

Phương pháp giải tích để giải mô hình hàng chờ gồm các bước sau:

- Bước 1: Phân tích hệ thống, chủ yếu là phân tích bản chất của dòng yêu cầu / tín hiệu đến và các trạng thái của hệ thống.
- Bước 2: Thiết lập hệ phương trình trạng thái cho các xác suất trạng thái (xác suất để hệ thống ở một trạng thái nào đó tại thời điểm  $t$ ).
- Bước 3: Giải hệ phương trình để tìm các xác suất trạng thái. Từ đó thiết lập các mối quan hệ giữa các chỉ tiêu cần phân tích.
- Bước 4: Tính toán, phân tích các chỉ tiêu, trên cơ sở đó đưa ra các nhận xét và các quyết định.

Phương pháp giải tích thường sử dụng các giả thiết rất chặt chẽ của Toán học về các đặc trưng của hệ thống, vì vậy nó có một số hạn chế nhất định khi giải các bài toán thực tế.

Trong khi đó, phương pháp mô phỏng / mô phỏng ngẫu nhiên để giải mô hình hàng chờ được áp dụng cho các bài toán dịch vụ đám đông không giải được bằng công cụ giải tích, nhất là liên quan đến hệ thống lớn, bất ổn định, hàm chứa nhiều yếu tố ngẫu nhiên, không tuân theo các giả thiết quá chặt chẽ của Toán học. Trong nhiều trường hợp phương pháp mô phỏng cho ta tiết kiệm được thời gian và chi phí nghiên cứu. Tuy phương pháp mô phỏng chỉ tạo ra các phương án đủ tốt để đánh giá hoạt động của hệ thống chứ không đưa ra được kĩ thuật tìm lời giải tốt nhất, nó tỏ ra rất thành công khi giải quyết nhiều bài toán hàng chờ nảy sinh từ thực tiễn.

Các bước cần tiến hành khi áp dụng phương pháp mô phỏng bao gồm:

- Bước 1: Xác định bài toán hay hệ thống hàng chờ cần mô phỏng và mô hình mô phỏng.
- Bước 2: Đo và thu thập số liệu cần thiết để khảo sát thống kê các số đặc trưng / các yếu tố cơ bản của mô hình.
- Bước 3: Chạy mô phỏng kiểm chứng (test simulation) mô hình và so sánh kết quả kiểm chứng với các kết quả đã biết được trong thực tế. Phân tích kết quả chạy mô phỏng kiểm chứng, nếu cần thì phải sửa lại phương án đã được đánh giá qua chạy mô phỏng.
- Bước 4: Chạy mô phỏng để kiểm chứng phương án cuối cùng và kiểm tra tính đúng đắn của mọi kết luận về hệ thống thực tế được rút ra sau khi chạy mô phỏng. Triển khai hoạt động của hệ thống hàng chờ dựa trên phương án tìm được.

Từ những phân tích trên đây có thể thấy Lý thuyết hàng chờ (Waiting Line Theory) còn gọi là Lý thuyết hệ phục vụ công cộng hay Lý thuyết hệ dịch vụ đám đông là lĩnh vực rất quan trọng của Toán ứng dụng / Vận trù học. Nhiều bài toán thực tế trong các lĩnh vực hệ thống dịch vụ, kỹ thuật, ... đã được giải quyết thành công nhờ áp dụng phương pháp mô phỏng mô hình hàng chờ.

## 2.4 Các yếu tố cơ bản của hệ thống hàng chờ

Hệ thống hàng chờ tổng quát được minh họa như hình sau.



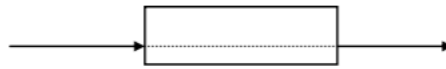
Hình 2.1: Hệ thống hàng chờ

Các yếu tố cơ bản của hệ thống hàng chờ bao gồm

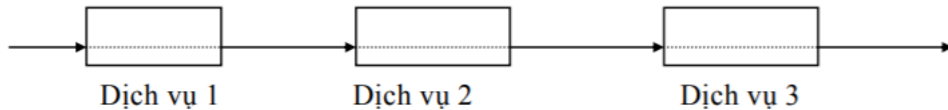
#### **2.4.1    Bố trí vật lý của hệ thống**

Hệ thống hàng chờ có một số dạng bố trí vật lí (phisical layout) như minh họa trên hình 2.1.

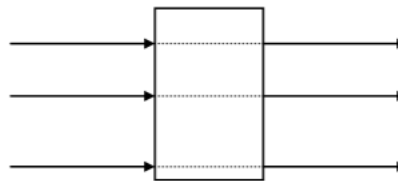
*Single Channel – Single Server* (Một kênh phục vụ, một loại dịch vụ)



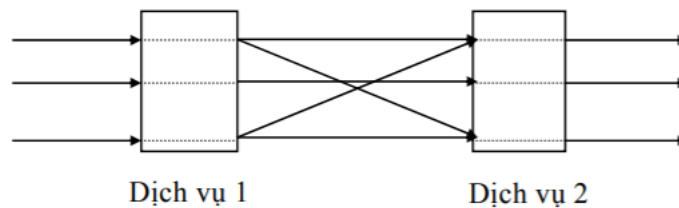
*Single Channel – Multi Server* (Một kênh phục vụ, nhiều loại dịch vụ)



*Multi Channel – Single Server* (Nhiều kênh phục vụ, một loại dịch vụ)



*Multi Channel – Multi Server* (Nhiều kênh phục vụ, nhiều loại dịch vụ)



Hình 2.2: Các dạng hệ thống hàng chờ

Trên hình 1.2, các kênh phục vụ được hiểu là những thiết bị kỹ thuật hoặc con người hoặc những tổ hợp các thiết bị kỹ thuật và con người được tổ chức quản lý một cách thích hợp nhằm phục vụ các yêu cầu / các tín hiệu đến hệ thống. Chẳng hạn, ở các trạm điện thoại tự động, kênh phục vụ là các đường dây liên lạc cùng các thiết bị kỹ thuật khác phục vụ cho việc đàm thoại.

### 2.4.2 Nguyên tắc phục vụ

Nguyên tắc phục vụ (hay nội quy) của hệ thống là cách thức nhận các yêu cầu vào các kênh phục vụ. Nguyên tắc phục vụ cho biết trường hợp nào thì yêu cầu được nhận vào phục vụ và cách thức phân bổ các yêu cầu vào các kênh như thế nào. Đồng thời nguyên tắc phục vụ cũng cho biết

trong trường hợp nào yêu cầu bị từ chối hoặc phải chờ và giới hạn của thời gian chờ.

Một số nguyên tắc phục vụ thường được áp dụng trong các hệ thống hàng chờ là

- FIFO (First in first out),
- LIFO (Last in first out),
- FCFS (First come first serve),
- Có ưu tiên,
- Không ưu tiên,...

### 2.4.3 Các phân phối xác suất của các dòng tín hiệu, dòng phục vụ

Số tín hiệu đến trong một khoảng thời gian cũng như thời gian phục vụ từng tín hiệu nói chung là những biến ngẫu nhiên, và do đó, chúng tuân theo các quy luật phân phối xác suất. Các quy luật phân phối xác suất này được thiết lập căn cứ các số liệu thực nghiệm thu thập từ các quan sát, thí nghiệm, hay từ cơ sở dữ liệu sẵn có.

Đối với dòng tín hiệu đầu vào, thông thường chúng ta giả sử rằng số tín hiệu đến trong vòng một khoảng thời gian nào đó được ấn định được trước (1 phút, 3 phút, 5 phút, 30 phút,...) tuân theo luật phân phối Poisson  $P(\lambda)$ . Ở đây, tham số  $\lambda$  đặc trưng cho tín hiệu đến (trung bình) là 100 người trong 1 giờ. Có nghĩa là, số khách vào siêu thị là biến ngẫu nhiên  $X$  có phân phối Poisson với  $\lambda = 100$ . Hoặc, với số cuộc gọi (trung bình) đến tổng đài trong vòng 1 phút là 3 (tín hiệu) thì có  $X \sim P(3)$ .

Một cách chính xác hơn, trong những trường hợp trên, ta có dòng tín hiệu đến là dòng Poisson dừng (còn gọi là dòng tối giản) với các tính chất trên sau:

- *Tính không hiệu quả*: Một dòng tín hiệu có tính không hiệu quả nếu xác suất xuất hiện một số tín hiệu nào đó trong một khoảng thời gian nhất định không phụ thuộc vào việc đã có bao nhiêu tín hiệu đã xuất hiện và xuất hiện như thế nào trước khoảng thời gian đó.
- *Tính đơn nhất*: Dòng tín hiệu có tính đơn nhất nếu xét trong khoảng thời gian khá bé thì sự kiện "có nhiều hơn một tín hiệu xuất hiện" hầu như không xảy ra, Về mặt thời gian ta có thể xem dòng tín hiệu có tính đơn nhất nếu thời điểm xuất hiện các tín hiệu không trùng nhau.
- *Tính dừng*: Dòng tín hiệu có tính dừng nếu xác suất xuất hiện một số tín hiệu nào đó trong khoảng thời gian  $\tau$  chỉ phụ thuộc vào độ dài của  $\tau$  chứ không phụ thuộc vào điểm khởi đầu của  $\tau$ .

## 2.5 Một số đặc trưng của hệ thống xếp hàng

Bất kỳ một hệ thống xếp hàng nào cũng được mô tả bởi đặc trưng sau: Quá trình đến: Nếu các khách hàng đến xếp hàng vào các thời điểm  $t_1, t_2, \dots, t_j$  thì các biến số ngẫu nhiên  $P_j = t_j - t_{j-1}$  được gọi là các thời điểm giữa các lần đến. Các thời điểm này thường được giả thuyết là các biến số ngẫu nhiên độc lập và được phân bố đồng nhất IID (Independent and Identically Distributed). Các quá trình đến thông dụng nhất là:



- $M$  = quá trình mũ (là quá trình Markov hay quá trình không có bộ nhớ).
- $Er(Erlangr)$  = là quá trình Erlang bậc  $r$ : trung tâm dịch vụ ở đây được biểu diễn bởi một dãy có  $r$  giai đoạn trễ, mỗi giai đoạn có cùng thời gian dịch vụ trung bình và có phân phối mũ. Không có các xếp hàng tại bất cứ một giai đoạn phục vụ nào vì yêu cầu tiếp theo sẽ không được phục vụ nếu như yêu cầu trước đó chưa hoàn thành dịch vụ ở tất cả  $r$  giai đoạn.
- $Hr$  = quá trình siêu số mũ bậc  $r$ .
- $D$  = quá trình tất định.
- $G$  = quá trình chung.

## 2.6 Một số điểm hạn chế của các mô hình hàng chờ

Các mô hình hàng chờ giới thiệu ở trên là những mô hình tiện lợi nhất được áp dụng khá rộng rãi. Tuy nhiên, do các mô hình này công nhận các giả thuyết "quá chặt chẽ" ít xảy ra trên thực tế, nên các chuyên gia trong lĩnh vực Toán ứng dụng/Vận trù học/Khoa học quản lý cũng đã đề xuất xem xét nhiều mô hình khác. Đó là các mô hình với các giả thiết như: số tín hiệu cần phục vụ là hữu hạn, dòng tín hiệu đến là Poisson, cường độ phục vụ phụ thuộc vào số tín hiệu trong hàng chờ ... và việc giải quyết mô hình như vậy cần tới sự trợ giúp của phương pháp mô phỏng ngẫu nhiên.

Ngay cả khi các giả thiết khá chặt chẽ của bốn mô hình đã nêu trong mục này (cũng như một số mô hình tương tự khác) là hợp lý, thì việc các mô hình hàng chờ đưa ra các lời giải với trạng thái vững (*steady state solutions*) cũng ít có ý nghĩa thực tế. Trong nhiều ứng dụng thực tiễn. Trong nhiều ứng dụng thực tiễn, các hệ thống hàng chờ không bao giờ đạt tới các trạng thái vững. Chẳng hạn, trong một hệ thống hàng chờ, cường độ tín hiệu đến trung bình thay đổi nhiều lần trong ngày không cho phép hệ thống đạt được trạng thái vững.

Do đó, để giải quyết nhiều bài toán hàng chờ trong lĩnh vực dịch vụ

đám đông và các lĩnh vực khác, cần áp dụng phương pháp mô phỏng để tìm ra lời giải có tính thực tiễn cho các mô hình hàng chờ khi hệ thống không thể đạt tới trạng thái vững hoặc khi không có các mô hình lý thuyết thích hợp.

## Chương 3

# Áp dụng giải quyết bài toán bán vé tàu

### 3.1 Phát biểu bài toán

Giả sử dòng khách hàng đến mua vé tại một ga tàu với  $M$  quầy phục vụ là dòng Poisson với tham số  $\lambda$  là số khách hàng /1 phút, ví dụ  $\lambda = 6$  (có nghĩa là khách hàng đến phòng bán vé với các thời điểm tuân theo luật phân phối mũ tham số  $\lambda = 6$ ). Ngoài ra, còn biết nguyên tắc phục vụ là FCFS (First come first serve) và thời gian phục vụ tại mỗi quầy có luật phân phối mũ với kì vọng  $t$  (phút). Suy ra,  $\mu = \frac{1}{t}$ .

### 3.2 Hệ thống xếp hàng M/M/C/K

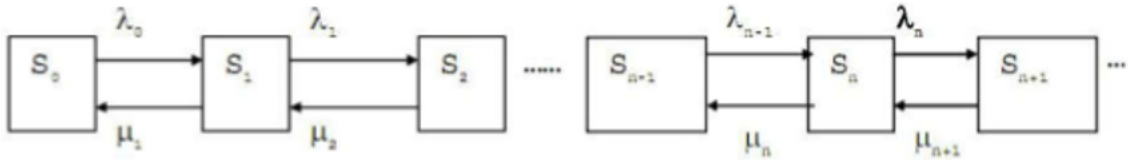
- Có tiến trình đến là một tiến trình phân phối Poisson.
- Hệ thống phục vụ có thời gian dịch vụ là một biến ngẫu nhiên phân phối mũ.
- $C$  là số trạm phục vụ (server) khách hàng.
- $K$  là số lượng khách có thể chứa tối đa trong hệ thống .

Hệ thống này được xây dựng bằng cách cho tốc độ đến hoặc tốc độ dịch vụ phụ thuộc vào số các khách hàng trong hệ thống. Ta có sơ đồ biểu diễn hệ thống xếp hàng M/M/C/K như sau:



Hình 3.1: Sơ đồ biểu diễn hệ thống xếp hàng M/M/C/K.

Với mô hình của hệ thống trên ta có lược đồ chuyển trạng thái của mô hình này như sau:



Hình 2.2: Sơ đồ chuyển trạng thái trong quá trình sinh-tử.

Hình 3.2: Sơ đồ chuyển trạng thái trong quá trình sinh-tử.

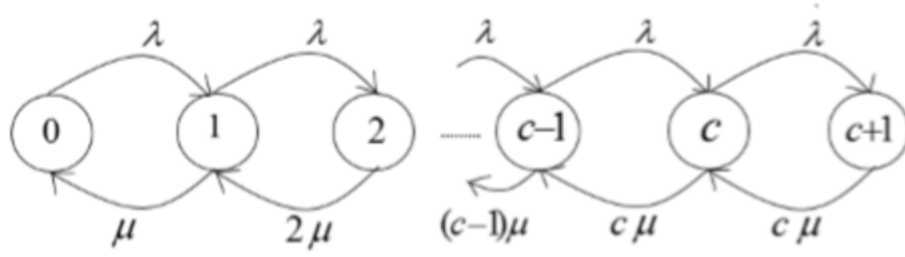
### 3.3 Giải quyết bài toán

Bài toán có  $k + 1$  trạng thái

$$\lambda_i = \lambda, \forall i = 1, 2, \dots, n$$

$$\mu_n = \begin{cases} n\mu, & n = 1, 2, \dots, c - 1 \\ c\mu, & n = c, c + 1, \dots \end{cases}$$

Mô hình hệ thống xếp hàng M/M/C/K. begincenter



Hình 3.3: Mô hình hệ thống xếp hàng M/M/C/K.

Ta thiết lập được ma trận sinh sau:

$$G = \begin{bmatrix} -\lambda_0 & \lambda_0 & & & & \\ \mu_1 & -(\mu_1 + \lambda_1) & \lambda_1 & & & \\ & \mu_2 & -(\mu_2 + \lambda_2) & \lambda_2 & & \\ & & \ddots & \ddots & \ddots & \\ & & & & \lambda_n & \\ & & & & \mu_n & -\mu_n \end{bmatrix}$$

Phân phối dừng của lượng người trong hệ thống bán vé là vectơ  $\pi$  là nghiệm của hệ

$$\begin{cases} \pi G = 0 \\ \sum_{i=0}^{k+1} \pi_i = 1 \end{cases}$$

Từ đó ta suy ra được phân phối dừng của hệ thống

### 3.3.1 Các đại lượng của mô hình M/M/1/K

1. Số lượng khách hàng trung bình trong hệ thống  $L$ :

$$L = \begin{cases} \rho \frac{1+k \cdot \rho^{k+1} - (k+1)\rho^k}{(1-\rho)(1-\rho^{k+1})} & , \quad \rho \neq 1 \\ \frac{k}{2} & , \quad \rho = 1 \end{cases}$$

2. Số lượng khách hàng trong xếp hàng  $L_q$ :

$$L_q = \begin{cases} L - \frac{\rho(1-\rho^k)}{1-\rho^{k+1}} & , \quad \rho \neq 1 \\ \frac{k(k-1)}{2(k+1)} & , \quad \rho = 1 \end{cases}$$

3. Thời gian trung bình của một khách hàng phải mất trong hệ thống  $W_s$ :

$$W_s = \frac{L}{\lambda(1 - \rho)}$$

4. Thời gian trung bình của một khách hàng phải mất để xếp hàng  $W_q$ :

$$W_q = W_s - \frac{1}{\mu}$$

5. Tỷ lệ hoạt động có ích của hệ thống  $\rho$ :  $\rho = \frac{\text{dòng vào}}{\text{dòng ra}}$

$$\rho = \frac{\lambda}{\mu}$$

6. Tỷ lệ thời gian nhàn rỗi của hệ thống  $p_0$ :

$$p_0 = \begin{cases} \frac{1-\rho}{1-\rho^{k+1}} & , \quad \rho \neq 1 \\ \frac{1}{k+1} & , \quad \rho = 1 \end{cases}$$

7. Xác suất khi hệ thống phục vụ hết khách hàng  $p_n$ :

$$p_n = p_0 \cdot \rho^n$$

### 3.3.2 Các đại lượng của mô hình M/M/C/K

1. Số lượng khách hàng trung bình trong hệ thống  $L$ :

$$L = L_q + c - p_0 \sum_{n=0}^{c-1} \frac{(c-n)(\rho c)^n}{n!}$$

2. Số lượng khách hàng trung bình trong xếp hàng  $L_q$ :

$$L_q = \frac{p_0 r^c \rho}{c!(1-\rho)^2} [1 - \rho^{K-c+1} - (1-\rho)(K-c+1)\rho^{K-c}],$$

trong đó  $\rho = \frac{\lambda}{c\mu} \neq 1$ .

3. Thời gian trung bình của một khách hàng phải mất trong hệ thống  $W_s$ :

$$W_s = \frac{L}{\lambda(1 - p_k)}$$

4. Thời gian trung bình của một khách hàng phải mất để xếp hàng  $W_q$ :

$$W_q = \frac{L_q}{\lambda(1 - p_k)}$$

5. Cường độ dòng đến (dòng vào)  $\lambda$ :

$$\lambda_n = \begin{cases} \lambda & , \quad 0 \leq n < k \\ 0 & , \quad n \geq k \end{cases}$$

6. Cường độ dòng phục vụ (dòng ra)  $\mu$ :

$$\mu_n = \begin{cases} n\mu & , \quad 0 \leq n < c \\ c\mu & , \quad c \leq n \leq k \end{cases}$$

7. Tỷ lệ thời gian nhàn rỗi của hệ thống  $p_0$ :

$$p_0 = \begin{cases} \left[ \sum_{n=0}^{c-1} \left( \frac{1}{n!} r^n \right) + \left( \frac{r^c}{c!} \right) \left( \frac{1 - \rho^{K-c+1}}{1 - \rho} \right) \right]^{-1} & , \quad \rho \neq 1 \\ \left[ \sum_{n=0}^{c-1} \left( \frac{1}{n!} r^n \right) + \left( \frac{r^c}{c!} \right) (K - c + 1) \right]^{-1} & , \quad \rho = 1 \end{cases}$$

8. Xác suất khi hệ thống phục vụ hết khách hàng  $p_n$ :

$$p_n = \begin{cases} \frac{1}{n!} \left( \frac{\lambda}{\mu} \right)^n p_0 & , \quad 0 \leq n < c \\ \frac{1}{c^{k-c} c!} \left( \frac{\lambda}{\mu} \right)^n p_0 & , \quad c \leq n \leq k \end{cases}$$

## Chương 4

### Chương trình

Xây dựng chương trình trên nền tảng C.NET giúp người dùng chỉ cần nhập những thông số input cơ bản của hệ thống sẽ đưa ra kết quả là Phân phối dừng và lợi nhuận thu được từ đó có thể đánh giá, cân nhắc đưa ra lựa chọn thay đổi hệ thống sao cho hợp lý.

#### 4.1 Một số mã nguồn chính

##### 4.1.1 Khởi tạo

- mohinh.a - Dòng vào a (giá trị  $\lambda$ )
- mohinh.b - Dòng ra b (giá trị  $\mu$ )
- mohinh.c - Số kênh c (giá trị  $c$ )
- mohinh.d - Số khách tối đa d (giá trị  $k$ )
- ln - Lợi nhuận
- L - Số khách trung bình toàn bộ hệ thống



```
private void button1_Click(object sender, EventArgs e)
{
    listBox1.Items.Clear();
    if(textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "" || textBox4.Text == "")
    {
        MessageBox.Show("Yêu cầu nhập đầy đủ dữ liệu");
        return;
    }
    mohinh.a = Convert.ToInt16(textBox1.Text);
    mohinh.b = Convert.ToInt16(textBox2.Text);
    mohinh.c = Convert.ToInt16(textBox3.Text);
    mohinh.d = Convert.ToInt16(textBox4.Text);
    string[] matran = new string[mohinh.d + 1];
    matran = mohinh.matran();
    listBox1.Items.Add("Ma trận sinh G");
    for (int i = 0; i < matran.Length; i++)
    {
        listBox1.Items.Add(matran[i]);
    }
    listBox1.Items.Add(""); listBox1.Items.Add("");
    float[] pio = mohinh.buildPi();
    float L = 0;

    for(int i=0; i< pio.Length; i++)
    {
        L += i * pio[i];
        listBox1.Items.Add("pi("+i+") = "+pio[i]);
    }
    float ln = 0;
    float r = (float)mohinh.a / (float)mohinh.b;
    float Lq = (L - r);
    float Wq = Lq / mohinh.a;
    float W = Wq + 1 / mohinh.b;
    listBox1.Items.Add("");
    if (textBox5.Text != "") {
        ln = L * 24 * (Convert.ToInt16(textBox5.Text));
        listBox1.Items.Add("Lợi nhuận = " + ln);
    }
    listBox1.Items.Add("Số khách trung bình toàn bộ hệ thống là " + L);
    //listBox1.Items.Add("Số khách trung bình trong hàng chờ là " + Lq.ToString());
    //listBox1.Items.Add("Thời gian chờ trung bình trong hàng chờ " + Wq.ToString());
    //listBox1.Items.Add("Thời gian chờ trung bình trong toàn bộ hệ thống " + W.ToString());
}
}
```

### 4.1.2 Thiết lập ma trận sinh G

Bài toán có  $k + 1$  trạng thái

$$\lambda_i = \lambda, \forall i = 1, 2, \dots, n$$

$$\mu_n = \begin{cases} n\mu, & n = 1, 2, \dots, c - 1 \\ c\mu, & n = c, c + 1, \dots \end{cases}$$

Ta thiết lập được ma trận sinh sau:

$$G = \begin{bmatrix} -\lambda_0 & \lambda_0 & & & & \\ \mu_1 & -(\mu_1 + \lambda_1) & \lambda_1 & & & \\ & \mu_2 & -(\mu_2 + \lambda_2) & \lambda_2 & & \\ & & \ddots & \ddots & \ddots & \\ & & & & \lambda_n & \\ & & & & \mu_n & -\mu_n \end{bmatrix}$$

```
class MHNN
{
    public int a;
    public int b;
    public int c;
    public int d;
    public string[] matran()
    {
        string[] g = new string[this.d + 1];
        for(int i = 0; i <= this.d; i++)
        {
            string hang = "";
            int mul = 0;
            if(i < this.c) { mul = i * this.b; } else { mul = this.c * this.b; }
            for(int j = 0; j <= this.d; j++)
            {
                if (i == 0)
                {
                    if(j == i) { hang += "-" + this.a + " "; }
                    else if(j == i + 1) { hang += this.a + " "; }
                    else { hang += "0 "; }
                }
                else if(i == this.d)
                {
                    if (j == i - 1) { hang += mul + " "; }
                    else if (j == i) { hang += "-" + mul + " "; }
                    else { hang += "0 "; }
                }
                else
                {
                    if(j == i - 1) { hang += mul + " "; }
                    else if (j == i) { hang += "-" + (int) (this.a + mul) + " "; }
                    else if (j == i + 1) { hang += this.a + " "; }
                    else { hang += "0 "; }
                }
            }
            g[i] = hang;
        }
        return g;
    }
}
```

### 4.1.3 Tìm phân phối dừng $\pi_i$

Phân phối dừng là nghiệm của hệ:

$$\begin{cases} \pi.G & = 0 \\ \pi_0\lambda_0 & = \pi_1\mu_1 \\ \pi_1\lambda_1 & = \pi_2\mu_2 \\ \dots & \\ \pi_{k-1}\lambda_{k-1} & = \pi_k\mu_k \end{cases}$$

trong đó  $\lambda_i = \lambda, \forall i$ . Ta tính các  $\pi_i, \forall i > 0$  theo  $\pi_0$ .

Tiếp đến ta thay  $\pi_i$  theo  $\pi_0$  vào biểu thức điều kiện:  $\sum_{i=0}^k \pi_i = 1$

Ta thu được  $\pi_0 = \frac{1}{1+tong}$  với giá trị:

$$tong = \frac{\lambda}{\mu_1} + \frac{\lambda.\lambda}{\mu_1.\mu_2} + \dots + \frac{\lambda^k}{\mu_1.\mu_2...\mu_k}$$

Cuối cùng ta tính các đại lượng bởi công thức:

$$\pi_n = \frac{\lambda}{\mu_n} \pi_{n-1}, \quad \text{với } n = 1, 2, \dots, k$$

$$\implies \pi_1 = \frac{\lambda}{\mu_1} \pi_0; \quad \pi_2 = \frac{\lambda}{\mu_2} \pi_1; \quad \dots; \quad \pi_k = \frac{\lambda}{\mu_k} \pi_{k-1}$$

```
public float[] buildPi()
{
    float[] pii = new float[this.d + 1];
    float[] mul = new float[this.d + 2];
    float tong = 0;
    //Tinh mang MUL
    for (int i = 1; i <= this.d + 1; i++)
    {
        if (i < this.c) { mul[i] = i * this.b; } else { mul[i] = this.c * this.b; }
    }
    // Tính pi[0]
    for (int i = 1; i <= this.d; i++)
    {
        string str = "";
        float phanso = 1;
        for (int j = 0; j < i; j++)
        {
            phanso *= this.a / mul[j+1];
        }
        tong += phanso;
    }
    pii[0] = 1 / (1 + tong);
    // Xác định các pi còn lại
    for (int i = 1; i <= this.d; i++)
    {
        pii[i] = (this.a / mul[i]) * pii[i - 1];
    }
    return pii;
}
```

#### 4.1.4 Tính số khách trung bình toàn bộ hệ thống

Công thức áp dụng:

$$L = \sum_{i=0}^k i \cdot \pi_i$$

```
for(int i=0; i< pio.Length; i++)
{
    L += i * pio[i];
    listBox1.Items.Add("pi("+i+" ) = "+pio[i]);
}
```

#### 4.1.5 Lợi nhuận

Công thức áp dụng: Tính lợi nhuận sau 24 giờ khi có lợi nhuận đầu vào

$$\text{Lợi nhuận} = \text{Lợi nhuận}_{input} * 24 * L$$

```
if (textBox5.Text != "") {
    ln = L * 24 * (Convert.ToInt16(textBox5.Text));
    listBox1.Items.Add("Lợi nhuận = " + ln);
}
```

## 4.2 Chạy chương trình



Mô hình ngẫu nhiên

# HỆ THỐNG XẾP HÀNG

Dòng vào  (đv/giờ)

Dòng ra  (đv/giờ)

Số kênh

K

Lợi nhuận  (/đv)

Process

Kết Quả

Hình 4.1: Giao diện khi chưa input



Mô hình ngẫu nhiên

# HỆ THỐNG XẾP HÀNG

Dòng vào  (đv/giờ)

Dòng ra  (đv/giờ)

Số kênh

K

Lợi nhuận  (/đv)

Process

Kết Quả

Ma trận sinh G

-6	6	0	0
9	-15	6	0
0	18	-24	6
0	0	18	-18

$\pi(0) = 0.509434$   
 $\pi(1) = 0.3396226$   
 $\pi(2) = 0.1132075$   
 $\pi(3) = 0.03773585$

Lợi nhuận = 81509.45  
Số khách trung bình toàn bộ hệ thống là 0.6792454

Hình 4.2: Giao diện khi đã input

Mã nguồn đầy đủ:

<https://github.com/NguyenThienDong/MHNN>

# Kết luận

Việc xây dựng mô hình mô phỏng trong thực tế là rất khó khăn và phức tạp. Lý thuyết xếp hàng có nhiều ứng dụng quan trọng trong việc phân tích và giải quyết một số bài toán thực tế. Trong tiểu luận này nhóm em đã cố gắng áp dụng quá trình sinh-tử của lý thuyết hàng chờ để giải quyết một vấn đề nhỏ trong bài toán bán vé tàu. Tuy nhiên, vì thời gian có hạn, sự am hiểu về môn học còn hạn chế nên nội dung còn nhiều thiếu sót. Nguyên vọng tiếp theo của bản thân là cố gắng tìm hiểu sâu hơn về Lý thuyết xếp hàng cùng với các ứng dụng thực tế và đa dạng hơn. Một lần nữa chúng em xin chân thành cảm ơn cô giáo TS. Nguyễn Thị Ngọc Anh đã giúp chúng em trong quá trình tiếp cận môn học và thực hiện tiểu luận này.

# Tài liệu tham khảo

- [1] Ngô Hoàng Long, *Mô hình ngẫu nhiên và ứng dụng.*, , 2018.
- [2] Trần Lộc Hùng, *Bài giảng Mô phỏng ngẫu nhiên (Cao học CNTT)*, Huế, 2007.
- [3] Trần Lộc Hùng, *Bài giảng Mô phỏng ngẫu nhiên (Cao học CNTT)*, Huế, 2007.
- [4] Đặng Hùng Thắng, *Quá trình ngẫu nhiên và tính toán ngẫu nhiên*, Nhà xuất bản Đại học quốc gia Hà Nội, 2007.