

[www.mientayvn.com](http://www.mientayvn.com)

Dịch tiếng anh chuyên ngành khoa học tự nhiên và kỹ thuật.

Dịch các bài giảng trong chương trình học liệu mở của học viện MIT, Yale.

Tìm và dịch tài liệu phục vụ cho sinh viên làm seminar, luận văn.

Tại sao mọi thứ đều miễn phí và chuyên nghiệp ???

Trao i tr c tuy n t i:

[www.mientayvn.com/chat\\_box\\_toan.html](http://www.mientayvn.com/chat_box_toan.html)

# GIỚI THIỆU LÝ THUYẾT MÔ PHÒNG VÀ MÔ HÌNH HÀNG CHỜ

## 1. Mục đích và các công cụ của mô phỏng

### 1.1. Khái niệm về mô phỏng ngẫu nhiên

Mô phỏng (*Simulation*) được ứng dụng rộng rãi trong kinh tế, kỹ thuật và nhiều lĩnh vực khác. Theo *Từ điển chính xác Oxford*, bản 1976, "mô phỏng có nghĩa là giả cách, ..., làm ra vẻ như, hành động như, bắt chước giống với, mang hình thức của, giả bộ như..., làm giả các điều kiện của tình huống nào đó thông qua một mô hình với mục đích huấn luyện hoặc tiện lợi".

Về mặt ý nghĩa kỹ thuật, mô phỏng (hay nói đúng hơn, phương pháp mô phỏng) hàm chứa việc áp dụng một mô hình nào đó để tạo ra kết quả, chứ không có nghĩa là thử nghiệm một hệ thống thực tế nào đó đang cần nghiên cứu hay khảo sát. Nếu mô hình có chứa các thành phần hay yếu tố ngẫu nhiên thì chúng ta có mô phỏng ngẫu nhiên.

Thuật ngữ "phương pháp Monte-Carlo" xuất hiện từ thế chiến thứ hai khi tiến hành các mô phỏng ngẫu nhiên trong quá trình phát kiến bom nguyên tử. Ngày nay, thuật ngữ này đôi khi cũng được dùng đồng nghĩa với thuật ngữ phương pháp mô phỏng ngẫu nhiên, như khi ta nói phương pháp Monte-Carlo tính tích phân chẳng hạn, tuy nhiên, nó không được sử dụng một cách rộng rãi.

Chúng ta xét mô phỏng trên hai quan điểm: nghệ thuật và kỹ thuật (với tư cách một công cụ), mà trong một số trường hợp rất khó phân định ranh giới rạch ròi. Trong chương này chúng ta nghiên cứu mô phỏng ngẫu nhiên về phương diện một số kỹ thuật, công cụ thường được sử dụng.

### 1.2. Các công cụ chủ yếu của mô phỏng

#### *Nguồn ngẫu nhiên (Source of randomness)*

Để áp dụng mô phỏng ngẫu nhiên trước hết cần phải có được một nguồn các *số ngẫu nhiên*. Các số ngẫu nhiên như vậy có thể được tạo ra bởi các hàm sinh số ngẫu nhiên.

Trong nhiều ngôn ngữ lập trình (như Visual C++ 6.0, hay Builder C++ 5.0,...), ta sẽ thấy có một cặp hàm dạng SRAND (*seed*) và RANDOM để phát sinh các số (được coi là) ngẫu nhiên. Hàm SRAND, có tham số là *seed* được gọi là *hạt mầm ngẫu nhiên*, đóng vai trò khởi tạo dãy số ngẫu nhiên. Còn hàm RANDOM là hàm sinh các số ngẫu nhiên sau khi có giá trị khởi tạo.

Thông thường, các nguồn này được coi như tồn tại một cách đương nhiên. Câu hỏi

đặt ra là chúng đã "đủ tốt" hay chưa? Trong giáo trình này chúng ta không đi sâu vào phân tích vấn đề trên. Một cách khái quát có thể nói rằng, các số được gọi là số ngẫu nhiên được tạo ra như vậy còn xa mới thực sự là ngẫu nhiên. Một cách chính xác hơn, chúng chỉ có thể gọi là các số *giả ngẫu nhiên* mà thôi. Chất lượng của nguồn ngẫu nhiên có thể ảnh hưởng rất lớn tới kết quả nghiên cứu khi sử dụng phương pháp mô phỏng ngẫu nhiên.

Xét về thực chất, các số giả ngẫu nhiên là các số có tính chất tất định (*deterministic*), nhưng chúng có tính chất giống với một dãy các giá trị thể hiện của các biến ngẫu nhiên độc lập, có phân phối đều. Ví dụ, xét dãy số: 13, 8, 1, 2, 11, 14, 7, 12, 13, 12, 17, 2, 11, 10, 3,... Dãy số này trông thì có vẻ ngẫu nhiên, nhưng thực chất là tuân theo một quy tắc (hãy phát hiện ra quy tắc này). Việc tìm kiếm các thuật giải (hay các quy tắc tất định) để phát sinh ra các số giả ngẫu nhiên đủ tốt là một lĩnh vực nghiên cứu chuyên sâu của Toán học và Tin học. Mặc dù trong thực tế, khi áp dụng mô phỏng ngẫu nhiên, người ta ít khi dùng các số ngẫu nhiên tuân theo luật phân phối xác suất đều  $U[0, 1)$  trên  $[0, 1)$ , nhưng nguồn số ngẫu nhiên loại này chính là cơ sở để mô phỏng các phân phối xác suất khác (xem mục 1.3).

### ***Mô hình ngẫu nhiên***

Hai lí do chính cho việc áp dụng mô phỏng ngẫu nhiên là:

- Tổng hợp dữ liệu theo sự phân loại nhất định.
- Đưa ra các dự báo.

Muốn áp dụng mô phỏng ngẫu nhiên cần phải có mô hình. Như vậy, mục đích của mô phỏng ngẫu nhiên cũng gần với mục đích của mô hình hoá (*modelling*). Có hai loại mô hình thường được áp dụng, đó là: mô hình cơ chế (*mechanistic model*) và mô hình tiện dụng (*convenient model*). Cả hai loại này đều có thể được sử dụng để trợ giúp các công việc nghiên cứu, khảo sát nhằm gia tăng sự nhận biết và tìm kiếm tri thức, dự báo và hỗ trợ việc ra quyết định.

Để ứng dụng một mô hình, ta có hai sự lựa chọn sau:

– Tiến hành các phân tích về mặt toán học để tìm hiểu hành vi của mô hình. Vấn đề này nhiều khi trở nên rất phức tạp với các hệ phi tuyến nhiều biến, do đó chúng ta cần đặt ra thêm các giả thiết. Tuy nhiên những giả thiết "chặt chẽ quá" của toán học đôi khi trở nên "đáng nghi ngờ" trong thực tế.

– Thí nghiệm với mô hình đang xem xét. Đối với các mô hình ngẫu nhiên các giá trị phản hồi (đầu ra) sẽ biến thiên, vì vậy chúng ta cần tạo ra hàng loạt các thể hiện (dữ liệu nhân tạo) với những bộ tham số khác nhau của mô hình.

Đôi khi cũng cần xem xét tới sự lựa chọn thứ ba, đó là tiếp cận lai (*hybrid approach*) của hai lựa chọn trên.

### **1.3. Mô phỏng một số phân phối xác suất**

#### ***Một số phân phối xác suất thường gặp***

Để áp dụng mô phỏng ngẫu nhiên cần biết một số kiến thức cơ bản mà chúng ta sẽ nhắc lại ngay sau đây. Biến ngẫu nhiên là một khái niệm quan trọng trong lý thuyết xác suất thống kê. Một cách giản lược, biến ngẫu nhiên (*random variable*), 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 chính: rời rạc và liên tục. Biến rời rạc có thể nhận các giá trị từ một tập hợp (có lực lượng) hữu hạn hoặc đếm được. Biến liên tục là một khái niệm toán học về loại biến ngẫu nhiên có thể nhận các giá trị dày sát nhau trên một hoặc một số khoảng / đoạn số thực nào đó (để trình bày vấn đề đơn giản, ở đây chúng ta chỉ nói tới biến ngẫu nhiên nhận các giá trị là số thực). Trong thực tế, không có một đại lượng ngẫu nhiên nào là liên tục theo nghĩa tuyệt đối, chẳng qua là chúng ta không nhận biết được (một cách cố ý hay không cố ý) khoảng cách giữa các giá trị rất sát nhau của nó mà thôi.

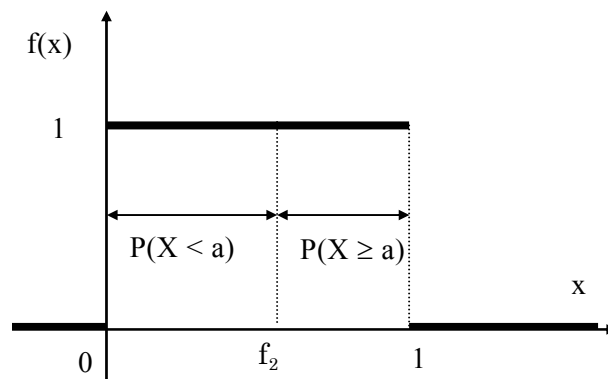
Phân phối xác suất của biến ngẫu nhiên rời rạc được minh họa qua ví dụ sau: Xét biến  $X$  có thể rơi vào một trong ba trạng thái được định lượng bởi các giá trị 6, 9, 12 với các xác suất tương ứng của các trạng thái là 0,3, 0,4 và 0,3. Chú ý rằng tổng các xác suất bằng 1 (100%) được phân phối vào các giá trị biến ngẫu nhiên  $X$  có thể lấy như trình bày trong bảng sau đây, được gọi là bảng phân phối xác suất.

Các giá trị của $X: x_i$	6	9	12
Xác suất tương ứng: $p_i$	0,3	0,4	0,3

(Chú ý:  $\sum p_i = 1$ )

Một số phân phối xác suất thường dùng của biến ngẫu nhiên liên tục và rời rạc được liệt kê dưới đây.

*Phân phối đều trong  $[0,1)$* :  $X$  nhận các giá trị thuộc nửa khoảng  $[0,1)$  với khả năng “như nhau”. Hàm mật độ xác suất  $f(x)$  của nó được biểu diễn trên hình III.1.



**Hình III.1. Đồ thị hàm mật độ phân phối đều**

*Phân phối Poát-xông*: Với một hệ thống hàng chờ một kênh (xem mục 3), số lượng  $X$  tín hiệu đến trong một khoảng thời gian là một biến ngẫu nhiên.

Giả sử số tín hiệu đến trung bình trong một khoảng thời gian đã biết được (kí hiệu  $\lambda$ ), thì với một số điều kiện nhất định có thể coi X tuân theo luật phân phối xác suất Poát-xông (*Poisson*) như sau:

Các giá trị của X: $x_i$	0   1   .....x..... $+\infty$
Xác suất $p_i$ tương ứng	$p(X = x) = \frac{\lambda^x e^{-\lambda}}{x!}$

$$\sum p_i = e^{-\lambda} \left[ \frac{\lambda^0}{0!} + \frac{\lambda^1}{1!} + \frac{\lambda^2}{2!} + \dots + \frac{\lambda^x}{x!} + \dots \right] = e^{-\lambda} \times e^{\lambda} = 1 \quad .$$

Chú ý rằng số đặc trưng cho giá trị trung bình của biến ngẫu nhiên X được gọi là kì vọng. Trong phân phối Poát-xông, kì vọng của X là  $\lambda$ . Số đặc trưng cho độ phân tán các giá trị của X xung quanh giá trị kì vọng của nó được gọi là độ lệch chuẩn  $\sigma$ . Với phân phối Poát-xông thì  $\sigma^2 = \lambda$ .

*Phân phối mũ:* Trên đây ta đã xét phân phối Poát-xông của số các tín hiệu đến trong một đơn vị thời gian. Một kiểu biến ngẫu nhiên thường xét là khoảng thời gian giữa hai tín hiệu liên tiếp sẽ tuân theo phân phối mũ. Đây là biến ngẫu nhiên liên tục chỉ nhận các giá trị không âm với hàm mật độ xác suất là  $f(\tau) = \lambda e^{-\lambda\tau}$ . Kí hiệu biến

ngẫu nhiên đang xét là  $\tau$  thì xác suất  $P(\tau \leq t) = \int_0^t \lambda e^{-\lambda\tau} d\tau$  có thể hiểu là xác suất cộng dồn cho tới t. Do đó hàm phân phối xác suất của  $\tau$  là:  $F(t) = \int_0^t f(\tau) d\tau = \int_0^t \lambda e^{-\lambda\tau} d\tau = -e^{-\lambda\tau} \Big|_0^t = 1 - e^{-\lambda t}$ .

*Phân phối chuẩn tắc N(0, 1):* Giả sử X là biến ngẫu nhiên có phân phối chuẩn tắc N(0,1). Lúc đó nó có kì vọng  $m = 0$  và độ lệch chuẩn  $\sigma = 1$ . Hàm phân phối xác suất của X có dạng:

$$F(x) = P(X \leq x) = \int_{-\infty}^x f(x) dx = \int_{-\infty}^x (1/\sqrt{2\pi}) \exp(-x^2 / 2) dx .$$

Cho X là biến ngẫu nhiên tuân theo luật phân phối chuẩn  $N(m, \sigma^2)$  có kì vọng m, độ lệch chuẩn  $\sigma$ . Lúc đó, thực hiện phép đổi biến  $Z = \frac{X - m}{\sigma}$  thì Z là một biến ngẫu nhiên tuân theo luật phân phối chuẩn tắc N(0,1).

**Mô phỏng các phân phối xác suất**

**Ví dụ 1:** Mô phỏng phân phối đều trên  $[0, 1)$

*Cách 1:* Dùng bảng số ngẫu nhiên (xem phụ lục 2A và 2B). Đây là các bảng số ghi lại các số (giá) ngẫu nhiên được phát sinh nhờ các hàm sinh số ngẫu nhiên trong máy tính. Chẳng hạn, sử dụng phụ lục 2B chúng ta nhận được một dãy số ngẫu nhiên: 0,10; 0,09; 0,73; 0,25 ...

*Cách 2:* Sử dụng các hàm sinh số ngẫu nhiên (*Random number generator*) đã được cài đặt trên máy tính.

Dù dùng bảng số ngẫu nhiên hay sử dụng các hàm sinh số ngẫu nhiên trong máy tính, ta cũng lấy ra hoặc tính được liên tiếp các số ngẫu nhiên  $x_i$  trong  $[0, 1)$  với  $i = 1, 2, \dots, n$ . Tần số các giá trị này rơi vào  $k$  khoảng nhỏ với độ dài bằng nhau  $1/k$  được chia ra từ  $[0, 1)$  là gần như nhau ( $\approx n/k$ ). Với  $n$  lớn thì các tần số đó càng sát gần  $n/k$ . Vì vậy ta coi các giá trị phát sinh được là các thể hiện của biến ngẫu nhiên  $X$  tuân theo phân phối đều trên  $[0, 1)$ .

Trong trường hợp cần mô phỏng biến  $Y$  phân phối đều trên  $[a, b)$ , ta chỉ việc tính  $y_i = a + (b - a)x_i$ . Chú ý rằng để phát sinh các số ngẫu nhiên nhận giá trị nguyên  $0, 1, 2, \dots, N$ , chỉ cần áp dụng công thức  $y_i = [(N + 1)x_i]$ , trong đó vế phải là phần nguyên của  $(N + 1)x_i$ . Một số bảng số ngẫu nhiên nguyên hay hàm sinh số ngẫu nhiên nguyên cài đặt sẵn trong các hệ máy tính cũng giúp giải quyết vấn đề này.

**Ví dụ 2:** Mô phỏng phân phối rời rạc với luật phân phối xác suất sau

Các giá trị của $X: x_i$	6	9	12
Xác suất $p_i$	0,3	0,4	0,3

Muốn mô phỏng phân phối trên, trước hết cần tạo ra một dãy các chữ số ngẫu nhiên bằng cách tra bảng số ngẫu nhiên hay dùng hàm sinh số ngẫu nhiên đã được cài đặt trong máy tính. Chẳng hạn ta có thể chọn dãy sau 1009732533 7652013586 3467354876 ... lấy từ hàng đầu bảng số ngẫu nhiên trong phụ lục 2B. Ta quy định nếu các chữ số 0, 1, 2 xuất hiện thì coi  $X = 6$ , nếu 3, 4, 5, 6 xuất hiện thì coi  $X = 9$ , còn nếu có 7, 8, 9 xuất hiện thì coi  $X = 12$ . Lúc đó ứng với 10 chữ số đầu tiên của dãy trên  $a_1 a_2 \dots a_{10} = 1009732533$  ta có bảng sau đây cho biết các giá trị của  $X$  có thể lấy:

$a_i$	1	0	0	9	7	3	2	5	3	3
Các giá trị của $X: x_i$	6	6	6	12	12	9	6	9	9	9

Như vậy, đã có 10 giá trị (thể hiện) của  $X$  được tạo ra. Tương tự, có thể tạo ra các thể hiện khác của  $X$ . Do tần suất (hay xác suất thực nghiệm) của mỗi chữ số ngẫu nhiên từ 0 tới 9 trong bảng số ngẫu nhiên là khoảng 10% nên tần suất (xác suất thực nghiệm)  $X$  nhận giá trị 6, 9 và 12 theo thứ tự là 30%, 40% và 30%. Do đó có thể coi  $P(X = 6) = 30\%$ ,  $P(X = 9) = 40\%$ ,  $P(X = 12) = 30\%$ .

Vậy muốn mô phỏng phân phối của  $X$  phải phát sinh ra một loạt các giá trị (các thể hiện)  $x_i$  của biến ngẫu nhiên  $X$  tuân theo quy luật phân phối đã cho.

**Ví dụ 3:** Mô phỏng phân phối mũ.

Giả sử biến ngẫu nhiên  $\tau$  tuân theo phân phối mũ với hàm phân phối xác suất là  $F(t) = P(\tau \leq t) = 1 - e^{-\lambda t}$ . Đây chính là xác suất để  $\tau$  nhận giá trị không lớn hơn một số  $t$  cho trước;  $\lambda$  là tham số đã cho của phân phối mũ.

Nếu  $r$  là biến ngẫu nhiên có phân phối đều trên  $[0, 1)$  thì  $P(r \geq e^{-\lambda t}) = 1 - e^{-\lambda t} = P(\tau \leq t)$  (xem hình III.1). Do đó,  $P(\ln r \geq -\lambda t) = P(-\frac{1}{\lambda} \ln r \leq t) = P(\tau \leq t)$ . Vậy để phát sinh ra các giá trị ngẫu nhiên (các thể hiện) của  $\tau$  thì trước hết cần phát sinh ra các giá trị ngẫu nhiên  $r$  và tính  $\tau = -\frac{1}{\lambda} \ln r$ . Chẳng hạn, từ bảng số ngẫu nhiên (phụ lục 2B), nếu lấy

$r = 0,10$  và  $\lambda = 5$  thì  $\tau = -0,2 \times \ln r = -0,2 \times \ln 0,1 = 0,46$ . Tiếp theo, nếu lấy  $r = 0,09$  thì  $\tau = -0,2 \times \ln 0,09 = 0,482$ . Cứ như vậy ta thu được một dãy các thể hiện của  $\tau$ .

## 2. Áp dụng mô phỏng ngẫu nhiên

### 2.1. Vai trò của phương pháp mô phỏng

Nhiều bài toán thực tế chứa các yếu tố ngẫu nhiên, bất ổn định không giải được bằng các phương pháp giải tích. Nếu chúng ta áp dụng các phương pháp giải tích, thì trong nhiều trường hợp buộc phải công nhận những giả thiết chặt chẽ không được thoả mãn trên thực tế, và do đó lời giải tìm được cũng ít có giá trị thực tiễn. Phương pháp mô phỏng được dùng rộng rãi để giải các bài toán loại đó, nhất là những bài toán liên quan đến hệ thống lớn, bất ổn định, hàm chứa nhiều yếu tố ngẫu nhiên.

Chúng ta cần áp dụng phương pháp mô phỏng trong các tình huống sau đây:

- Khi không tìm được mô hình giải tích nào thích hợp.
  - Các hoạt động của hệ thống thường bị ngắt quãng, đứt đoạn không theo quy luật nào cả.
  - Mô phỏng là phương pháp duy nhất cho chi phí tiết kiệm và tốn ít thời gian.
- Tuy nhiên phương pháp mô phỏng có một số điểm hạn chế sau:
- Không đưa ra được lời giải chính xác.
  - Khó xác định được sai số.
  - Mô phỏng chỉ sử dụng khi môi trường có tính bất ổn định.
  - Mô phỏng chỉ tạo ra các phương án đánh giá chứ không đưa ra được kỹ thuật tìm lời giải tối ưu.
  - Mô phỏng đôi khi rất đắt tiền.

## 2.2. Các bước cần tiến hành khi áp dụng mô phỏng

- Xác định vấn đề hay hệ thống cần mô phỏng.
- Xác định mô hình mô phỏng.
- Đo và thu thập số liệu cần thiết cho mô hình.
- Chạy mô phỏng.
- Phân tích kết quả mô phỏng, nếu cần thì phải sửa lại phương án đã được đánh giá qua chạy mô phỏng.
- Chạy mô phỏng để kiểm chứng phương án mới.
- Kiểm tra tính đúng đắn của mọi kết luận về hệ thống thực tế được rút ra sau khi chạy mô phỏng.

Trên đây là các bước cần làm khi áp dụng mô phỏng ngẫu nhiên để tìm ra các phương án hợp lý cho các bài toán thực tế. Ngoài ra, mô phỏng còn được áp dụng để giải quyết nhiều vấn đề khác.

## 2.3. Một số ví dụ về áp dụng phương pháp mô phỏng

**Ví dụ 1:** Cần lựa chọn một trong hai chiến lược để phát triển sản phẩm, với các số liệu thu thập được cho trong ba bảng III.1, III.2 và III.3.

**Bảng III.1. Xác suất thời gian phát triển sản phẩm**

Thời gian phát triển sản phẩm	Xác suất	
	Chiến lược I	Chiến lược II
6	0,2	0,4
9	0,3	0,4
12	0,5	0,2

**Bảng III.2. Chi phí lợi nhuận**

Chi phí/giá bán	Chiến lược I	Chiến lược II
Chi phí cố định	600.000	1.500.000
Chi phí biến thiên/đơn vị	7,5	6,75
Giá bán/đơn vị sản phẩm	10	10



**Bảng III.3. Doanh số phụ thuộc thời gian phát triển sản phẩm**

Doanh số	Xác suất		
	6 tháng	9 tháng	12 tháng
1.000.000	0,2	0,4	0,5
1.500.000	0,8	0,6	0,5

Vấn đề đặt ra là áp dụng phương pháp mô phỏng để tính lợi nhuận trung bình của từng chiến lược, sau đó kiểm tra kết quả (so sánh với kết quả lí thuyết).

Như vậy có năm phân phối xác suất cần mô phỏng ứng với năm biến ngẫu nhiên:  $X_1$  – thời gian phát triển sản phẩm (theo chiến lược) I,  $X_2$  – thời gian phát triển sản phẩm II,  $X_3$  – doanh số cho thời gian 6 tháng,  $X_4$  – doanh số cho thời gian 9 tháng và  $X_5$  – doanh số cho thời gian 12 tháng. Trong ví dụ này, để trình bày đơn giản về vấn đề mô phỏng các phân phối xác suất của các biến trên, ta dùng mười số ngẫu nhiên, mỗi số gồm mười chữ số ngẫu nhiên rút ra từ bảng số ngẫu nhiên – phụ lục 2A (vì vậy các chữ số 0, 1, 2,..., 9 mỗi số chiếm khoảng 10%).

$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$	$a_9$	$a_{10}$
1	5	8	1	9	2	2	3	9	6
2	0	6	8	5	7	7	9	8	4
8	2	6	2	1	3	0	8	9	2
8	3	7	4	8	3	6	0	4	9
4	6	3	7	5	6	7	4	8	8
0	9	2	8	1	0	5	5	8	2
7	2	9	5	0	8	8	5	7	9
9	5	8	6	1	1	1	6	3	2
7	0	5	5	5	0	8	7	6	7
6	4	7	2	3	8	2	9	3	4

Ta quy định  $a_1$  ứng với  $X_1$ ,  $a_2$  ứng với  $X_2$ ,  $a_6$  ứng với  $X_3$ ,  $a_8$  ứng với  $X_4$  và  $a_{10}$  ứng với  $X_5$ . Ngoài ra cũng quy định:

$$a_1 = \begin{cases} 0,1 & \text{thì } X_1 = 6 \text{ tháng (thời gian phát triển sản phẩm I)} \\ 2,3,4 & \text{thì } X_1 = 9 \text{ tháng} \\ 5,6,7,8,9 & \text{thì } X_1 = 12 \text{ tháng} \end{cases}$$

$$a_2 = \begin{cases} 0,1,2,3 & \text{thì } X_2 = 6 \text{ tháng (thời gian phát triển sản phẩm II)} \\ 4,5,6,7 & \text{thì } X_2 = 9 \text{ tháng} \\ 8,9 & \text{thì } X_2 = 12 \text{ tháng} \end{cases}$$

$$a_6 = \begin{cases} 0,1 & \text{thì } X_3 = 10^6 \text{ (doanh số 6 tháng phát triển sản phẩm)} \\ 2,3,\dots, 9 & \text{thì } X_3 = 1,5 \cdot 10^6 \end{cases}$$

$$a_8 = \begin{cases} 0,1,2,3 & \text{thì } X_4 = 10^6 \text{ (doanh số 9 tháng phát triển sản phẩm)} \\ 4,5,\dots, 9 & \text{thì } X_4 = 1,5 \cdot 10^6 \end{cases}$$

$$a_{10} = \begin{cases} 0,1,2,3,4 & \text{thì } X_5 = 10^6 \text{ (doanh số 12 tháng phát triển sản phẩm)} \\ 5,6,\dots, 9 & \text{thì } X_5 = 1,5 \cdot 10^6 \end{cases}$$

Cần nhắc lại một số công thức trong lĩnh vực quản trị kinh doanh như sau:

$$+ \text{ Lợi nhuận} = (\text{Doanh số} - \text{Điểm hoà vốn}) \times (\text{Lợi nhuận} / \text{đơn vị sản phẩm})$$

$$+ \text{ Điểm hoà vốn} = (\text{Chi phí cố định}) / (\text{Lợi nhuận} / \text{đơn vị sản phẩm})$$

$$+ \text{ Lợi nhuận} / \text{đơn vị sản phẩm} = (\text{Giá bán/đơn vị sản phẩm}) - (\text{chi phí} / \text{đơn vị sản phẩm})$$

Các tính toán mô phỏng được tổng hợp trong bảng III.4.

**Bảng III.4. Kết quả tính toán mô phỏng**

Số ngẫu nhiên										Thời gian		Doanh số		Lợi nhuận	
a <sub>1</sub>	a <sub>2</sub>				a <sub>6</sub>		a <sub>8</sub>		a <sub>10</sub>	I	II	I	II	I	II
1	5	8	1	9	2	2	3	9	6	6	9	1,5.10 <sup>6</sup>	10 <sup>6</sup>	3,15.10 <sup>6</sup>	1,75.10 <sup>6</sup>
2	0	6	8	5	7	7	9	8	4	9	6	1,5.10 <sup>6</sup>	1,5.10 <sup>6</sup>	3,15.10 <sup>6</sup>	3,38.10 <sup>6</sup>
8	2	6	2	1	3	0	8	9	2	12	6	10 <sup>6</sup>	1,5.10 <sup>6</sup>	1,9.10 <sup>6</sup>	3,38.10 <sup>6</sup>
8	3	7	4	8	3	6	0	4	9	12	6	1,5.10 <sup>6</sup>	1,5.10 <sup>6</sup>	3,15.10 <sup>6</sup>	3,38.10 <sup>6</sup>
4	6	3	7	5	6	7	4	8	8	9	9	1,5.10 <sup>6</sup>	1,5.10 <sup>6</sup>	3,15.10 <sup>6</sup>	3,38.10 <sup>6</sup>
0	9	2	8	1	0	5	5	8	2	6	12	10 <sup>6</sup>	10 <sup>6</sup>	1,9.10 <sup>6</sup>	1,75.10 <sup>6</sup>
7	2	9	5	0	8	8	5	7	9	12	6	1,5.10 <sup>6</sup>	1,5.10 <sup>6</sup>	3,15.10 <sup>6</sup>	3,38.10 <sup>6</sup>
9	5	8	6	1	1	1	6	3	2	12	9	10 <sup>6</sup>	1,5.10 <sup>6</sup>	1,9.10 <sup>6</sup>	3,38.10 <sup>6</sup>
7	0	5	5	5	0	8	7	6	7	12	6	1,5.10 <sup>6</sup>	10 <sup>6</sup>	3,15.10 <sup>6</sup>	1,75.10 <sup>6</sup>
6	4	7	2	3	8	2	9	3	4	12	9	10 <sup>6</sup>	1,5.10 <sup>6</sup>	1,9.10 <sup>6</sup>	3,38.10 <sup>6</sup>

$$\text{Điểm hoà vốn của chiến lược I} = \frac{600.000}{10 - 7,5} = 240.000$$

$$\text{Điểm hoà vốn của chiến lược II} = \frac{1.500.000}{10 - 6,75} = 461538$$

**Bảng III.5. So sánh lợi nhuận giữa chiến lược I và II**

Tổng lợi nhuận	Chiến lược I	Chiến lược II
		26,5 × 10 <sup>6</sup>
Lợi nhuận trung bình (Σ lợi nhuận/ 10)	2,65 × 10 <sup>6</sup>	2,891 × 10 <sup>6</sup>

Cần chú ý rằng trong bảng III.5 là kết quả tính toán khi chạy mô phỏng 10 lượt ứng với 10 số đã chọn ra. Nếu ta lấy càng nhiều số ngẫu nhiên thì độ chính xác đạt được càng cao. Vì vậy, nếu việc tính toán trên đây được lập trình và chạy trên máy tính với hàng trăm, hàng ngàn lượt thì độ chính xác sẽ rất cao.

Qua các phân tích trên ta thấy

*Để tiến hành mô phỏng cần phải có:*

– Cơ sở dữ liệu (*DataBase*)

– Cơ sở tri thức (*KnowledgeBase*)

Kiểm tra kết quả mô phỏng trên bằng cách so sánh với kết quả lí thuyết được thực hiện như sau:

Doanh số chiến lược I =  $0,2 \times (0,2 \times 10^6 + 0,8 \times 1,5 \times 10^6) + 0,3 \times (0,4 \times 10^6 + 0,6 \times 1,5 \times 10^6) + 0,5 \times (0,5 \times 10^6 + 0,5 \times 1,5 \times 10^6) = 1,295 \times 10^6$ . Lợi nhuận trung bình chiến lược I =  $(1,295 - 0,24) \times 2,5 \times 10^6 = 2,637 \times 10^6$ . Kết quả tính toán mô phỏng là  $2,65 \times 10^6$  rất sát với kết quả này.

Tương tự ta tính được doanh số và lợi nhuận trung bình cho chiến lược II ( $2,84 \times 10^6$ ), và rút ra được kết luận về độ chính xác của tính toán mô phỏng.

**Ví dụ 2:** Tìm xác suất p để bao lồi của 4 điểm lấy bất kì trong vòng tròn đơn vị là một hình tam giác (*bài toán Sylvester*).

Có lẽ cách đơn giản nhất để giải bài toán này là áp dụng mô phỏng ngẫu nhiên theo các bước sau đây:

i) Gán cho biến đếm *Counter* giá trị ban đầu bằng 0.

ii) Tiến hành một đợt gieo ngẫu nhiên tám số thực  $0 \leq r_i \leq 1$  và  $0 \leq \varphi_i \leq 2\pi$  (để gieo  $\varphi_i$  ta lấy số ngẫu nhiên thuộc  $[0, 1)$  gieo được nhân thêm với  $2\pi$ ),  $i = 1, 2, 3, 4$ . Đặt  $x_i = r_i \sin \varphi_i$ ,  $y_i = r_i \cos \varphi_i$ , ta có 4 điểm nằm trong hình tròn đơn vị. Đặt  $A = (x_1, y_1)$ ,  $B = (x_2, y_2)$ ,  $C = (x_3, y_3)$ ,  $D = (x_4, y_4)$ .

iii) Ta tính diện tích 4 tam giác ABC, ABD, ACD và BCD. Nếu ta có diện tích của một tam giác bằng tổng diện tích ba tam giác còn lại thì ta được bao lồi của bốn điểm A, B, C và D là một tam giác. Ta tăng giá trị của biến đếm *Counter* lên thêm 1, nếu trái lại biến đếm giữ nguyên giá trị cũ và quay về bước ii).

Quá trình cứ thế tiếp diễn cho tới khi số đợt gieo đạt tới một giá trị khá lớn được chọn từ trước (chẳng hạn 10000 đợt hoặc 20000 đợt, hoặc 100000 đợt). Mỗi lần biến đếm *Counter* sẽ có giá trị kết thúc khác nhau. Lấy tỉ số của số đó và số đợt, ta có tần suất xuất hiện của sự kiện "*bao lồi của 4 điểm là tam giác*". Số tần suất này theo luật số lớn là giá trị gần đúng của xác suất cần tính.

Theo các tài liệu chuyên khảo, lời giải đúng của bài toán là:  $p = 35/(12\pi^2) \approx 0,29552$ . Rõ ràng, trong trường hợp này, ta nên áp dụng mô phỏng ngẫu nhiên để tính ra tần suất (việc dễ thực hiện), thay thế cho việc tính xác suất theo lí thuyết (việc khó thực hiện).

Sau đây là văn bản chương trình máy tính với ngôn ngữ lập trình C giải bài toán Sylvester.

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <conio.h>
```

```
#define PI 3.14159265358979
```

```

const double esp =4.5e-12;
struct diem {double x,y;};

/* Tao bo so ngau nhien bang cach tron – shuffling */
int rd(){return rand()%10;}
double radm()
{return rand()%100+0.1*rd()+0.001*rd()+0.0001*rd();}

/* Chuong trinh chinh */
void main()
{
clrscr(); long int count = 0, reps; diem d[4]; double r, goc;
printf("\n Provide number of repetitions:");
scanf("%ld",&reps); printf("\n reps= %ld",reps); srand(19587);

/* Gieo ngau nhien 4 diem va tinh dien tich bon tam giac */
for(long int i=0;i<reps;i++)
{
for (int j=0;j<4;j++)
{
r=radm(); goc=radm()/100;
d[j].x=r*cos(goc); d[j].y=r*sin(goc);
}
double d12=sqrt(pow(d[0].x-d[1].x,2)+pow(d[0].y-d[1].y,2));
double d13=sqrt(pow(d[0].x-d[2].x,2)+pow(d[0].y-d[2].y,2));
double d14=sqrt(pow(d[0].x-d[3].x,2)+pow(d[0].y-d[3].y,2));
double d23=sqrt(pow(d[1].x-d[2].x,2)+pow(d[1].y-d[2].y,2));
double d24=sqrt(pow(d[1].x-d[3].x,2)+pow(d[1].y-d[3].y,2));
double d34=sqrt(pow(d[2].x-d[3].x,2)+pow(d[2].y-d[3].y,2));
double p123=(d12+d23+d13)/2;
double p124=(d12+d24+d14)/2;
double p134=(d13+d34+d14)/2;
double p234=(d23+d24+d34)/2;
double s123=p123*(p123-d12)*(p123-d13)*(p123-d23);
double s124=p124*(p124-d12)*(p124-d14)*(p124-d24);
double s134=p134*(p134-d13)*(p134-d14)*(p134-d34);
double s234=p234*(p234-d23)*(p234-d24)*(p234-d34);

```

```

/* Cac truong hop bao loi cua 4 diem la tam giac */
if(s123>0&& s123>0&& s134>0&& s234>0)
{
s123=sqrt(s123);s124=sqrt(s124);
s134=sqrt(s134);s234=sqrt(s234);
if(fabs(s123-(s124+s134+s234))<esp)
count++;
else if(fabs(s124-(s123+s134+s234))<esp)
count++;
else if(fabs(s134-(s123+s124+s234))<esp)
count++;
else if(fabs(s234-(s123+s124+s134))<esp)
count++;
}
else count++;
}
printf("\n Number of repetitions = %ld",reps);
printf("\n Number of successes = %ld",count);
printf("\n Probability to compute= %0.9f", count*1.0/reps);
getch();
}

```

Các kết quả chạy chương trình trong bốn lần như sau:

esp = 4.5e-12	esp = 4.5e-12	esp = 4.5e-12	esp = 4.5e-12
số lần lặp 10000	số lần lặp 20000	số lần lặp 100000	số lần lặp 200000
số lần thành công 3050	số lần thành công 5941	số lần thành công 29594	số lần thành công 58993
xác suất: 0.30500	xác suất: 0.29705	xác suất: 0.29594	xác suất: 0.294965

**Ví dụ 3:** Giải bài toán tối ưu toàn cục sau (xem lại mục 4.2 chương I về bài toán tối ưu toàn cục phi tuyến).

$$f(x) = 4x_1^2 - 2,1x_1^4 + x_1^6 / 3 + x_1x_2 - 4x_2^2 + 4x_2^4 \rightarrow \text{Min}$$

với điều kiện ràng buộc (miền ràng buộc D):

$$\begin{cases} -2,5 \leq x_1 \leq 2,5 \\ -1,5 \leq x_2 \leq 1,5 \end{cases}$$

Bài toán trên đây có 4 cực tiểu địa phương và 2 cực tiểu toàn cục, có tên gọi là

“Bài toán lưng lạc đà”.

Ta tìm phương án tối ưu toàn cục bằng phương pháp mô phỏng tôi SA (*Simulated Annealing*). Phương pháp (SA) mô phỏng quá trình một vật thể rắn sau khi bị nung nóng ở nhiệt độ rất cao được để nguội từ từ về một nhiệt độ rất thấp. Lúc đó hàm năng lượng của vật thể sẽ đạt mức thấp nhất. Thuật giải SA áp dụng mô phỏng ngẫu nhiên (bằng lí thuyết xích Markov, như sẽ trình bày trong chương IV, có thể chứng minh được quá trình SA sẽ hội tụ theo xác suất về lời giải tối ưu toàn cục) như sau:

#### *Bước khởi tạo*

Ta xuất phát từ một phương án  $X$  bất kì ban đầu thoả điều kiện ràng buộc. Lấy nhiệt độ  $T = T_{\text{ban đầu}}$  khá cao ( $T_{\text{ban đầu}} = 10000$ , chẳng hạn).

#### *Các bước lặp*

Tại mỗi mức nhiệt độ  $T$  thực hiện các bước sau:

i) Chọn  $X' \in D$  và thuộc một lân cận đủ nhỏ của  $X$ .

ii) Xét  $\Delta f = f(X') - f(X)$ . Nếu  $\Delta f < 0$  thì đặt  $X := X'$ . Nếu trái lại khi  $\Delta f > 0$  thì chấp nhận  $X := X'$  với xác suất  $p = \exp(-\Delta f / (K_b \times T))$ , trong đó  $K_b$  là hằng số Boltzmann ( $K_b = 1,38 \cdot 10^{23}$ ),  $T$  là nhiệt độ hiện thời trong quá trình nguội.

Quy trình i) và ii) lặp lại một số lần  $L$  đủ lớn (chẳng hạn  $L = 200, 300, \dots$ ).

Sau đó tính mức nhiệt độ mới theo công thức  $T := \alpha T$  ( $\alpha \approx 1$ , chẳng hạn như  $\alpha = 0,95$  hay  $0,99 \dots$ ). Thuật toán dừng khi  $T \leq T_{\text{cuối}}$  ( $T_{\text{cuối}}$  là giá trị đã chọn trước  $\approx 0$ ).

Sau đây là văn bản chương trình *annealing.cpp*:

```
/* Su dung ky thuat annealing – mo phong toi
giai bai toan toi uu toan cuc co rang buoc */
```

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <math.h>
```

```
/* Tinh gia tri ham so can cuc tieu hoa */
float f(float x,float y)
{
float fg = 4*pow(x,2)-2.1*pow(x,4)+pow(x,6)/3;
fg = fg +x*y -4*pow(y,2)+4*pow(y,4);
return fg;
}
```

```

/* Kiem tra cac dieu kien rang buoc */
int constraint(float x,float y)
{
    float fg;
    fg = x + 2.5;
    if (fg<0) goto ZERO;
    fg = 2.5 - x;
    if (fg<0) goto ZERO;
    fg = y + 1.5;
    if (fg<0) goto ZERO;
    fg = 1.5 - y;
    if (fg<0) goto ZERO;
    return 1;
    ZERO: return 0;
}

/* Thu tuc tim diem thay the mo phong qua trinh annealing */
void sa(float x0,float y0,int k,float T,float Tlast,float alfa,float delta)
{
    float x1,y1,deltaf,u,p,ux,uy;
    int l=1; srand(27556);
    printf("\n starting value of the function=%f",f(x0,y0)); getch();
    printf("\n obtained at x=%f,y=%f",x0,y0);getch();
    do
    {
        do
        {
            ux=float(random(10000))/10000-0.5;
            x1=x0-delta*ux;
            uy=float(random(10000))/10000-0.5;
            y1=y0-delta*uy;
            if(constraint(x1,y1)==1)
            {
                deltaf=f(x1,y1)-f(x0,y0);
                if(deltaf<=0) {x0=x1;y0=y1;}
            }
            else
            {
                u=float(random(10000))/10000;

```



```

        p=exp(-deltaf/T);
        if(u<=p) {x0=x1;y0=y1;}
    }
}
l=l+1;
}while(l<=k);
T=T*alfa;
}while(T>Tlast);
printf("\n Optimal value fMin=%f",f(x0,y0)); getch();
printf("\n obtained at x=%f,y=%f",x0,y0);getch();
}
/* Chuong trinh chinh tim diem toi uu */
void main()
{
    clrscr();
    float x0,y0,T,Tlast,k,alfa,delta;
    printf("\n Within the range -2.5 to 2.5 provide value x0=");
    scanf("%f",&x0);
    printf("\n Within the range -1.5 to 1.5 provide value y0=");
    scanf("%f",&y0);
    printf("\n Specify number of iterations at each T level=");
    scanf("%d",&k);
    printf("\n Specify a very high value for Tstart=");
    scanf("%f",&T);
    printf("\n Specify a very low value for Tlast=");
    scanf("%f",&Tlast);
    printf("\n Specify reduction coefficient alfa=");
    scanf("%f",&alfa);
    printf("\n Specify neighbourhood radius delta=");
    scanf("%f",&delta);
    sa(x0,y0,k,T,Tlast,alfa,delta);
    getch();
}

```

Kết quả chạy chương trình máy tính với thuật giải SA là:

Hạt mầm (Seed)	Phương án ban đầu	Phương án tối ưu	Giá trị $f_{Min}$
27556	(0, 0)	(-0.0898613, 0.7124848)	-1.0316283
19587	(0.1, 0.1)	(0.0898837, -0.7125957)	-1.0316284

(với  $\alpha = 0.997$ ,  $\delta = 0.01$ ,  $T_{\text{ban đầu}} = 10000$ ,  $L = 500$ ,  $T_{\text{cuối}} = 0.0001$ ).

### 3. Một số vấn đề về mô hình hàng chờ

#### 3.1. Một số yếu tố cơ bản của hệ thống hàng chờ

Như đã biết, trong nhiều hoạt động sản xuất kinh doanh cũng như trong đời sống chúng ta áp dụng các hệ dịch vụ đám đông hay hệ phục vụ công cộng. Chúng có tên gọi chung là hệ thống hàng chờ (*Waiting Line System*). Chẳng hạn các xí nghiệp sửa chữa máy móc, các cửa hàng, các bến xe, bến cảng, trạm tổng đài, các hệ thống điện tử viễn thông, dịch vụ Internet,... là các ví dụ về hệ thống hàng chờ.

#### *Mô hình hàng chờ*

Trong các hệ thống hàng chờ thường xuyên diễn ra hai quá trình: quá trình nảy 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. 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 (*Waiting Line Model*) 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.

#### *Các phương pháp giải bài toán 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 những bài toán 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.

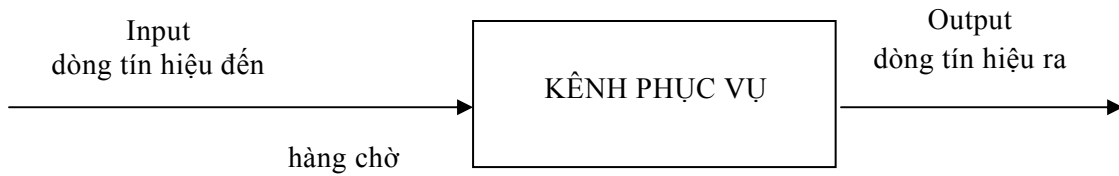
*Bước 3:* Chạy mô phỏng kiểm chứng (*test simulation*) mô hình và so sánh kết quả kiểm chứng với các kết quả đã biết được trong thực tế. Phân tích kết quả chạy mô phỏng kiểm chứng, nếu cần thì phải sửa lại phương án đã được đánh giá qua chạy mô phỏng.

*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 Lí thuyết hàng chờ (*Waiting Line Theory*) còn gọi là Lí thuyết hệ phục vụ công cộng hay Lí 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, kĩ thuật, ... đã được giải quyết thành công nhờ áp dụng phương pháp mô phỏng mô hình hàng chờ.

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

Hệ thống hàng chờ tổng quát được minh hoạ như trên hình III.2.



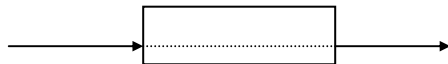
**Hình III.2. Hệ thống hàng chờ**

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

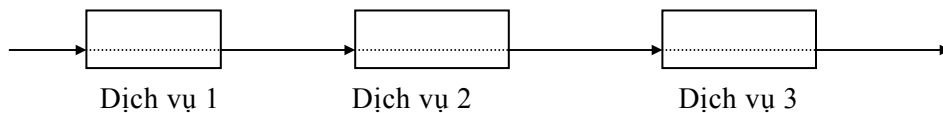
**a. Bố trí vật lí của hệ thống**

Hệ thống hàng chờ có một số dạng bố trí vật lí (*physical layout*) như minh hoạ trên hình III.3.

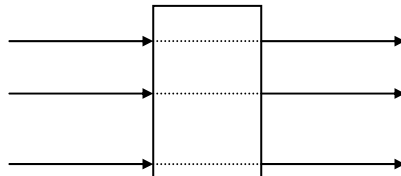
*Single Channel – Single Server* (Một kênh phục vụ, một loại dịch vụ)



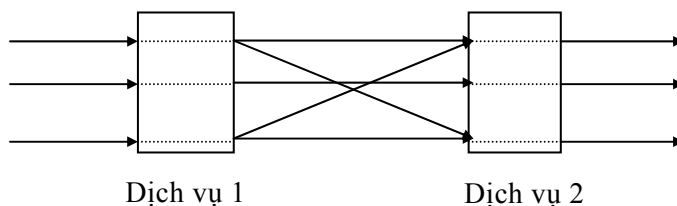
*Single Channel – Multi Server* (Một kênh phục vụ, nhiều loại dịch vụ)



*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 III.3. Các dạng hệ thống hàng chờ**

Trên hình III.3, các kênh phục vụ được hiểu là những thiết bị kĩ thuật hoặc con người hoặc những tổ hợp các thiết bị kĩ thuật và con người được tổ chức quản lí 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.

### ***b. 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 bố 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,...

### ***c. 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 trước (1 phút, 3 phút, 5 phút, 30 phút,...) tuân theo luật phân phối Poát-xông  $\mathcal{P}(\lambda)$ . Ở đây, tham số  $\lambda$  đặc trưng cho số tín hiệu đến (trung bình) trong khoảng thời gian trên. Ví dụ, số khách vào siêu thị (trung bình) là 100 người trong 1 giờ. Có nghĩa là, số khách vào siêu thị là biến ngẫu nhiên  $X$  có phân phối Poát-xông với  $\lambda = 100$ . Hoặc, với số cuộc gọi (trung bình) đến tổng đài trong vòng 1 phút là 3 (tín hiệu) thì có  $X \sim \mathcal{P}(3)$ .

Một cách chính xác hơn, trong những trường hợp trên, ta có dòng tín hiệu đến là dòng Poát-xông dừng (còn gọi là *dòng tới gián*) với các tính chất trên sau:

– *Tính không hậu quả*: Một dòng tín hiệu có tính không hậu 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  $\tau$ .

### 3.2. Các chỉ số cần khảo sát

Đối với một hệ thống hàng chờ, cần tìm cách để đánh giá được các chỉ số sau:

– A (*Arrival rate*): cường độ dòng tín hiệu đến hay số tín hiệu đến trung bình trong một khoảng thời gian. Ví dụ: A = 6 (6 khách hàng đến trong 2 tiếng); A = 20 (20 cú điện thoại đến tổng đài trong 1 phút).

– S (*Service rate*): cường độ phục vụ hay số tín hiệu trung bình được phục vụ trên một đơn vị thời gian. Ví dụ: S = 7 (hệ thống có thể phục vụ 7 khách trong 1 giờ); S = 25 (tổng đài phục vụ được 25 cú điện thoại trong 2 phút).

– L<sub>q</sub> (*Number in queue* hay *Length of queue*): số tín hiệu trung bình trong hàng chờ.

– L<sub>s</sub> (*Number in system* hay *Length of system*): số tín hiệu trung bình trong toàn hệ thống (như vậy L<sub>s</sub> ≥ L<sub>q</sub>).

– W<sub>q</sub> (*Waiting time in queue*): thời gian chờ trung bình trong hàng chờ của một tín hiệu.

– W<sub>s</sub> (*Waiting time in system*): thời gian chờ trung bình trong hệ thống của một tín hiệu.

– P<sub>w</sub> (*Probability the system is busy*): xác suất hệ thống bận (đang hoạt động) hay còn gọi là hệ số (chỉ số) sử dụng của toàn hệ thống (*Utilization factor*).

### 3.3. Tính toán các chỉ số

Với mục đích tìm hiểu bước đầu, sau đây chúng ta chỉ xét các hệ thống hàng chờ với một loại dịch vụ. Bằng phương pháp giải tích (xem mục 3.1), có thể tìm được công thức tính toán các chỉ số với điều kiện: các giả thiết của mô hình được thỏa mãn.

**Mô hình một kênh phục vụ thỏa mãn: số tín hiệu đến có phân phối Poát-xông, thời gian phục vụ có phân phối mũ**

Các công thức (I) sau đây đã được chứng minh (bằng phương pháp giải tích):

$$L_q = \frac{A^2}{S(S-A)}; L_s = \frac{A}{S-A}; L_q = \frac{A}{S} \cdot L_s;$$

$$W_q = \frac{A}{S(S-A)}; W_s = \frac{1}{S-A}; P_w = \frac{A}{S}.$$

Chẳng hạn với A = 3, S = 4 thì:

$$L_q = \frac{9}{4(4-3)} = 2,25; L_s = \frac{3}{1} = 3;$$

$$W_q = \frac{3}{4} = 0,75; W_s = 1; P_w = \frac{3}{4} = 0,75.$$

**Mô hình một kênh phục vụ thỏa mãn: số tín hiệu đến là phân phối Poát-xông,**

**thời gian phục vụ có phân phối bất kì.**

Các công thức (II) sau đây đã được chứng minh:

$$Lq = \frac{A^2\sigma^2 + (A/S)^2}{2(1-A/S)}; \quad Ls = Lq + \frac{A}{S};$$
$$Wq = \frac{Lq}{A}; \quad Ws = Wq + \frac{1}{S}; \quad Pw = \frac{A}{S}.$$

Trong đó  $\sigma$  độ lệch chuẩn thời gian phục vụ một tín hiệu. Chú ý rằng, nếu thời gian phục vụ tuân theo phân phối mũ thì  $\sigma = \frac{1}{S}$  và cũng là thời gian trung bình phục vụ một tín hiệu. Có thể nhắc lại rằng:

$$m = \int_0^{\infty} tf(t)dt = \frac{1}{S} \quad \text{và} \quad \sigma = \sqrt{\int_0^{\infty} (t-m)^2 f(t)dt} = \frac{1}{S}.$$

Lúc này các công thức (II) trở về (I).

**Mô hình một kênh phục vụ thỏa mãn: số tín hiệu đến có phân phối Poát-xông, thời gian phục vụ có phân phối mũ, hàng chờ có giới hạn số tín hiệu tối đa M**

Các công thức (III) sau đây đã được chứng minh:

$$P_0 = \frac{1-A/S}{1-(A/S)^{M+1}}; \quad Pw = 1 - P_0;$$

với  $P_0$  là xác suất không có tín hiệu nào trong hệ thống (hệ số không sử dụng);

$P_M = (A/S)^M P_0$  là tỉ lệ % tín hiệu không được phục vụ do hệ thống “đầy”;

$$Ls = \frac{Pw - M(A/S)P_M}{1-(A/S)}; \quad Lq = Ls - \frac{A(1-P_M)}{S};$$

$$Ws = \frac{Ls}{A(1-P_M)}; \quad Wq = Ws - \frac{1}{S}.$$

Chú ý rằng nếu  $M = +\infty$  thì (III) trở về (I).

**Mô hình nhiều kênh phục vụ thỏa mãn: tín hiệu đến có phân phối Poát-xông, thời gian phục vụ là phân phối mũ.**

$P_0$  – xác suất tất cả các kênh phục vụ đều không có tín hiệu, tìm được bằng cách tra phụ lục 3 dựa trên tỉ số  $A/kS$  ( $k$  số kênh phục vụ) hoặc tính trực tiếp từ công thức sau:

$$P_0 = \frac{1}{\sum_{n=0}^{k-1} \frac{1}{n!} \left(\frac{A}{S}\right)^n + \frac{1}{k!} \left(\frac{A}{S}\right)^k \frac{kA}{kA-S}} \text{ với } kS > A;$$

$$P_w = \frac{1}{k!} \left(\frac{A}{S}\right)^k \frac{kS}{kS-A} P_0;$$

$$L_s = \frac{AS(A/S)^k}{(k-1)!(kS-A)^2} P_0 + \frac{A}{S}; \quad L_q = L_s - \frac{A}{S}; \quad W_s = \frac{L_s}{A}; \quad W_q = \frac{L_q}{A}.$$

### **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 thiết “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 không phải kiểu Poát-xông, cường độ phục vụ phụ thuộc vào số tín hiệu trong hàng chờ ... và việc giải quyết những 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 lí, 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, 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 các 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 lí thuyết thích hợp.

### **3.4. Áp dụng mô phỏng cho một số hệ thống hàng chờ**

**Ví dụ 1:** Bài toán hệ dịch vụ hàng chờ ba kênh với dòng tới gián có từ chối.

Cho biết: dòng tín hiệu đến là dòng Poát-xông dừng (còn gọi là *dòng tới gián*). Giãn cách thời gian giữa thời điểm đến của hai nhu cầu (tín hiệu) liên tiếp có phân phối mũ với tham số  $\mu = 5$ , tức là có hàm mật độ  $f(t) = 5e^{-5t}$ . Nếu tín hiệu xuất hiện mà có ít nhất một trong ba kênh không bận (kênh số 1 hoặc kênh số 2 hoặc kênh số 3 không bận) thì tín hiệu được phục vụ tại kênh không bận với số thứ tự nhỏ nhất; nếu trái lại (khi cả ba kênh đều bận) thì tín hiệu bị từ chối. Biết thời gian phục vụ mỗi nhu cầu là 0,5 phút, hãy xác định kì vọng toán số nhu cầu được phục vụ trong khoảng thời gian 4 phút.



Như vậy, cần áp dụng mô hình hàng chờ *MultiChannel – SingleServer System* (Hệ thống nhiều kênh phục vụ – một loại dịch vụ) theo quy tắc *First in first out* (FIFO: Tín hiệu đến trước được phục vụ xong trước). Thời gian giữa hai tín hiệu liên tiếp có phân phối mũ với hàm mật độ xác suất  $f(t) = 5e^{-5t}$ . Trong bài toán này (nhằm đơn giản các bước tính toán) thời gian phục vụ mỗi tín hiệu được coi là không đổi và bằng 0,5 phút.

Chúng ta sẽ áp dụng mô phỏng để xác định số nhu cầu trung bình cần được phục vụ trong khoảng thời gian 4 phút như trình bày sau đây.

Kí hiệu  $T_i$  là thời điểm đến của tín hiệu thứ  $i$ ,  $T_{ki}$  là thời điểm kết thúc dịch vụ của tín hiệu thứ  $i$  (nếu có), tại kênh thứ  $k$  ( $k = 1, 2, 3$ ). Thời điểm đến của nhu cầu tiếp theo là  $T_i = T_{i-1} + \tau_i$  với  $\tau$  tuân theo luật phân phối mũ có hàm mật độ  $f(t) = 5e^{-5t}$  và hàm phân phối là  $F(t) = 1 - e^{-5t} = P(\tau \leq t)$ .

Lúc đó  $T_1 = 0$ ,  $T_{11} = T_1 + 0,5$ . Kết quả này cho biết thời điểm đến của tín hiệu thứ nhất là  $T_1 = 0$  và được kênh 1 phục vụ. Kết thúc phục vụ tín hiệu 1 là thời điểm  $T_{11} = T_1 + 0,5 = 0,5$ . Máy đếm ghi nhận 1 đơn vị là số tín hiệu đã được phục vụ.

Để tìm  $T_2$  theo công thức  $T_2 = T_1 + \tau_2$ , ta phát sinh  $\tau_2$  theo cách đã biết ở mục 3.1.3: Trước hết, phát sinh số ngẫu nhiên  $r_2$  có 2 chữ số sau dấu phẩy  $0 \leq r_i \leq 1$  (theo bảng số ngẫu nhiên – phụ lục 2B) ta có  $r_2 = 0,10$ . Sau đó tính  $\tau_2 = -\frac{1}{5} \ln r_2$  và  $T_2 = T_1$

$-\frac{1}{5} \ln r_2 = 0 - 0,2 \ln 0,1 = 0,46$ . Vậy tín hiệu tiếp theo phải vào kênh 2 vì kênh 1 còn đang bận. Máy đếm ghi thêm 1 đơn vị thời điểm kết thúc phục vụ tín hiệu 2 là  $T_{22} = T_2 + 0,5 = 0,46 + 0,5 = 0,96$ .

Tiếp tục phát sinh  $r_3 = 0,09$ , ta có  $\tau_3 = -0,2 \ln 0,09 = 0,482$ . Do đó thời điểm đến của tín hiệu 3 là  $T_3 = T_2 + \tau_3 = 0,46 + 0,482 = 0,942$ . Lúc này kênh 1 đã được giải phóng do đã phục vụ xong tín hiệu 1, nên tín hiệu 3 được tiếp nhận vào kênh 1. Tại thời điểm kết thúc phục vụ tín hiệu 3 là  $T_{13} = T_3 + 0,5 = 0,942 + 0,5 = 1,442$  máy đếm lại ghi tiếp 1 đơn vị. Thực hiện tính toán tương tự, kết quả tổng hợp được ghi trong bảng III.6.

**Bảng III.6. Tính toán mô phỏng tìm số nhu cầu được phục vụ**

Thứ tự tín hiệu	Số ngẫu nhiên $r_i$	$-\ln r_i$	$\tau_i =$ $-1/5 \ln r_i$	Thời điểm đến $T_i$	Thời điểm $T_{ki}$ kết thúc phục vụ tại kênh k			Đếm số tín hiệu	
					1	2	3	nhận	bỏ
1				0	0,5			1	
2	0,10	2,30	0,46	0,46		0,96		1	
3	0,09	2,44	0,482	0,942	1,442			1	
4	0,73	0,32	0,064	1,006		1,506		1	
5	0,25	1,39	0,278	1,284			1,784	1	
6	0,33	1,11	0,222	1,506	2,006			1	
7	0,76	0,27	0,054	1,560		2,060		1	
8	0,52	0,65	0,13	1,690					1
9	0,01	4,6	0,92	2,61	3,11			1	
10	0,35	1,05	0,21	2,82		3,32		1	
11	0,86	0,15	0,03	2,85			3,35	1	
12	0,34	1,08	0,216	3,066					1
13	0,67	0,40	0,08	3,146	3,646			1	
14	0,35	1,05	0,21	3,356		3,856		1	
15	0,48	0,73	0,146	3,502			4,022	1	
16	0,76	0,27	0,054	3,556					1
17	0,80	0,22	0,044	3,600					1
18	0,95	0,05	0,01	3,61					1
19	0,9	0,10	0,02	3,63					1
20	0,91	0,09	0,018	3,648	4,148			1	
21	0,17	1,77	0,354	4,002				1	
								14	6

Phân tích kết quả tính toán ta thấy trong 20 nhu cầu đến thì chỉ có 14 nhu cầu được phục vụ. Tính toán tương tự 6 lần nữa ta có kết quả:

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$
14	15	14	12	13	15

Vậy số nhu cầu trung bình được hệ phục vụ trong vòng 4 phút vào khoảng

$$\bar{x} = (14 + 15 + 14 + 12 + 13 + 15)/6 = 13,83.$$

***Giải bài toán dịch vụ ba kênh có từ chối trên máy tính***

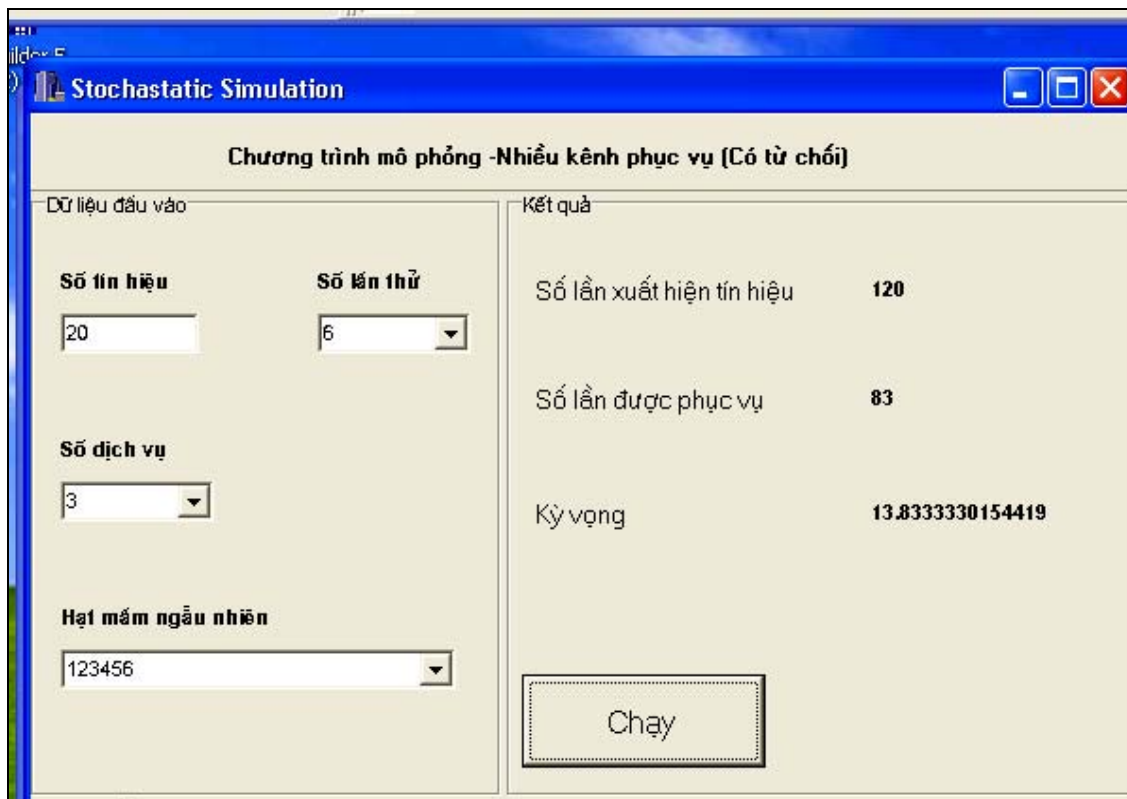
Một phần mềm máy tính với tên gọi MOPHONG1 phiên bản 1.0 đã được thiết kế dựa trên ngôn ngữ Builder C++ 5.0 để giải bài toán một dịch vụ với nhiều kênh phục vụ

có từ chối như đã trình bày trong ví dụ 1 với các điều kiện sau:

- Số kênh phục vụ  $n$  có thể lấy giá trị từ 2 tới 10.
- Dòng tín hiệu đến là dòng Poát-xông, thời gian dẫn cách giữa hai tín hiệu liên tiếp tuân theo phân phối mũ với hàm mật độ xác suất là  $f(t) = 5e^{-5t}$ .
- Thời gian trung bình phục vụ (xử lí) một tín hiệu là 0,5 phút.
- Tính số tín hiệu được phục vụ trong số  $m$  tín hiệu đến ( $m$  có thể lấy giá trị từ 10 tới 100).
- Thực hiện  $k$  lần mô phỏng ( $k$  lần thử,  $k$  có thể lấy giá trị từ 4 tới 10).

Mục đích của việc xây dựng phần mềm này nhằm vào: phục vụ dạy và học môn Mô phỏng ngẫu nhiên, cũng như tiếp tục nâng cấp phiên bản 1.0 để mô phỏng được hệ nhiều kênh phục vụ – nhiều loại dịch vụ có từ chối trong trường hợp tổng quát khi dòng tín hiệu đến và thời gian phục vụ tín hiệu có phân phối bất kì.

Để chạy phần mềm MOPHONG1 chúng ta cần cài đặt Builder C++ 5.0 vào máy tính. Sau khi kích chuột vào biểu tượng của phần mềm, chọn *File > Open Project > Look in > Mophong1 (Thư mục lưu trữ phần mềm) > Mp1.bpr*. Sau đó chọn *Run* trên thanh công cụ để chạy phần mềm (xem hình III.4).



**Hình III.4. Chạy phần mềm MOPHONG 1**

Chúng ta nhập số kênh phục vụ  $n = 3$  vào ô số (kênh) dịch vụ, số tín hiệu phát

sinh  $m = 20$  vào ô số tín hiệu, số lần mô phỏng  $k = 6$  vào ô số lần thử. Ngoài ra ta phải chọn hạt mầm là một số nguyên (đủ lớn) nhằm khởi tạo hàm sinh số ngẫu nhiên có phân phối đều trong  $[0, 1)$ , chẳng hạn số 123456 để ghi vào ô hạt mầm ngẫu nhiên, nhằm từ đó mô phỏng dòng Poát-xông các tín hiệu đến. Sau đó chúng ta kích chuột vào nút *Chạy* để chạy chương trình.

Kết quả ta thấy tổng cộng số tín hiệu đến là 120, trong đó có 83 tín hiệu được phục vụ (do đó có 37 tín hiệu bị từ chối). Vậy trung bình trong 20 tín hiệu đến có 13,833 tín hiệu được phục vụ. Kết quả này so với kết quả tính toán trên giấy cho ví dụ trên là khá sát nhau.

*Chú ý:* Việc tính toán dựa vào bảng số ngẫu nhiên cho trong phụ lục hoặc dựa vào hàm sinh số ngẫu nhiên trong máy tính không cho kết quả hoàn toàn giống nhau trong các lần chạy mô phỏng khác nhau. Điều này xảy ra là vì các bộ số ngẫu nhiên tạo được không giống nhau. Với các hạt mầm khác nhau thì hàm sinh số ngẫu nhiên cũng cho các bộ số ngẫu nhiên khác nhau và do đó các kết quả cuối cùng cũng không trùng nhau. Muốn kết quả mô phỏng ổn định hơn cần chọn số tín hiệu đến  $m$  đủ lớn.

Ngoài ra, chúng ta có thể chọn số kênh phục vụ tùy ý, chẳng hạn  $n = 5$ . Kết quả chạy phần mềm với hạt mầm 123456 cho biết trong số 20 tín hiệu đến trung bình có 18,167 tín hiệu được phục vụ.

**Ví dụ 2:** Một công ti điều hành một kho nguyên liệu để cấp phát cho các đốc công của 10 phân xưởng. Hiện tại, hai nhân viên phục vụ đang được công ti giao cho nhiệm vụ cấp phát nguyên liệu. Bộ phận quản lí công ti muốn cân nhắc liệu có nên thêm một nhân viên phục vụ nữa hay không.

Rõ ràng rằng số đốc công (tín hiệu cần phục vụ) là hữu hạn, phân phối của số tín hiệu đến trong một đơn vị thời gian cũng không theo kiểu Poát-xông và thời gian phục vụ tín hiệu cũng không tuân theo luật phân phối mũ. Do đó, không thể tìm ra được lời giải giải tích thông qua một mô hình nhiều kênh phục vụ với các giả thiết như vậy. Phương pháp “duy nhất” là tìm cách áp dụng mô phỏng.

#### *Số liệu thu thập được*

– Quan sát trong vòng một tháng vào các ngày làm việc, mỗi ngày một giờ vào các thời điểm ngẫu nhiên, các số liệu về thời gian phục vụ một tín hiệu và tần suất tương ứng đã được thu thập (bảng III.7).

– Tính thời gian phục vụ trung bình cho một tín hiệu (tính kì vọng):

$$8 \times 0,1 + 9 \times 0,2 + 10 \times 0,3 + 11 \times 0,4 = 10 \text{ (phút).}$$

– Ngoài ra cũng đã khảo sát được: giãn cách thời gian trung bình giữa hai tín hiệu liên tiếp là 5 phút và số lượng tín hiệu trung bình đến trong một khoảng 5 phút là một tín hiệu.

**Bảng III.7. Thời gian phục vụ một tín hiệu và tần suất**

Thời gian phục vụ một tín hiệu (phút)	Số lượng	Tần suất/ Xác suất thực nghiệm
8	15	0,1
9	30	0,2
10	45	0,3
11	60	0,4
	$\Sigma = 150$	$\Sigma = 1$

Cần chú ý rằng, dòng tín hiệu đến chưa chắc tuân theo phân phối Poát-xông và thời gian phục vụ một tín hiệu không nhất thiết tuân theo phân phối mũ. Do đó, không áp dụng được công thức của mục 3.3 mà phải dùng mô phỏng để giải quyết vấn đề: cần bố trí bao nhiêu kênh phục vụ (nhân viên phục vụ) trong kho cấp phát nguyên liệu là hợp lý nhất? Như vậy mô phỏng có khả năng xử lý các tình huống, sự kiện như trong thực tế xảy ra chứ không bắt chúng tuân theo các phân phối xác suất nhất định hay theo các hành vi gò ép.

*Mô phỏng hệ thống hàng chờ*

– Mô phỏng tín hiệu đến: trung bình 5 phút có 1 tín hiệu đến. Chúng ta dùng 24 số ngẫu nhiên sau lấy ra từ bảng số ngẫu nhiên (phụ lục 2A), mỗi số gồm 10 chữ số để mô phỏng 24 khoảng 5 phút (như vậy tổng cộng là 120 phút, mỗi số dùng để mô phỏng một khoảng 5 phút từ 9h – 11h): 1581922396, 2068577984, 8262130892, 8374856049, 4637567488, 0928105582, 7295088579, 9586111652, 7055508767, 6472382984, 4112077556, 3440672486,..., 5973470495. Nếu chữ số 7 xuất hiện trong số 10 chữ số đã chọn, ta coi như 1 tín hiệu đến trong khoảng thời gian tương ứng (vì trung bình trong 5 phút có 1 tín hiệu đến cũng giống như trung bình trong một số có 10 chữ số có một chữ số 7). Chẳng hạn trong khoảng 5 phút đầu không có tín hiệu nào, khoảng 5 phút thứ hai có 2 tín hiệu đến.

Ta thấy số tín hiệu đến chỉ có thể là 0, 1, 2, 3, 4 tín hiệu. Thời điểm đến của tín hiệu trong mỗi khoảng 5 phút được quy định như sau tùy theo số tín hiệu đến trong khoảng đó (bảng III.8). Chẳng hạn, nếu có hai tín hiệu đến, thì thời điểm đến là vào đầu phút thứ nhất và đầu phút thứ ba.

**Bảng III.8. Quy định thời điểm đến của tín hiệu**

Phút	1	2	3	4	5
Nếu 1 tín hiệu đến	*				
Nếu 2 tín hiệu	*		*		
Nếu 3 tín hiệu	*		*		*
Nếu 4 tín hiệu		*	*	*	*

– Mô phỏng thời gian phục vụ X một tín hiệu: Ta mô phỏng phân phối xác suất ở bảng III.7 theo cách đã biết. Trước hết lấy ba số ngẫu nhiên có 10 chữ số: 9846413446, 8306646692, 0661684251 (hàng 4 từ dưới lên, phụ lục 2A). X = 8 phút nếu xuất hiện chữ số 0; X = 9 phút nếu xuất hiện chữ số 1, 2; X = 10 phút nếu xuất hiện chữ số 3, 4, 5; X = 11 phút nếu xuất hiện chữ số 6, 7, 8 hoặc 9. Bảng III.9 tổng hợp kết quả mô phỏng số tín hiệu đến và thời gian phục vụ các tín hiệu.

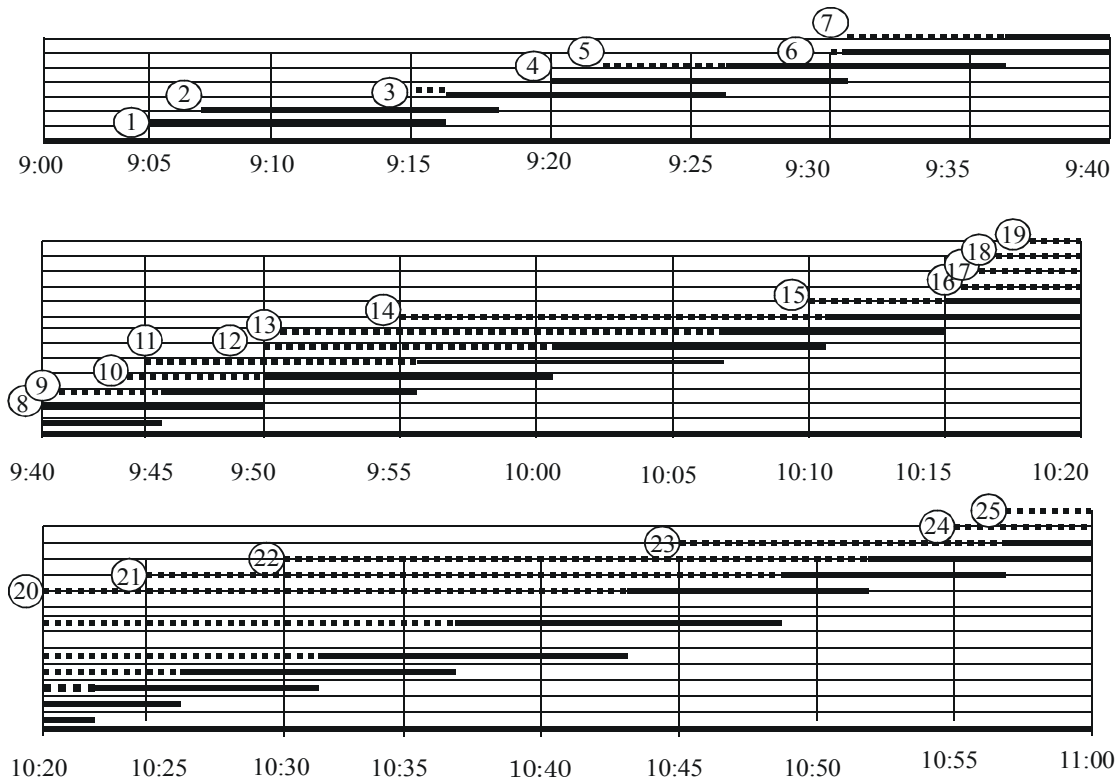
**Bảng III. 9. Kết quả mô phỏng số tín hiệu đến và thời gian phục vụ tín hiệu**

Chu kì	Số ngẫu nhiên 10 chữ số										Số tín hiệu đến	Thời gian phục vụ
1	1	5	8	1	9	2	2	3	9	6	0	
2	2	0	6	8	5	7	7	9	8	4	2	11, 11
3	8	2	6	2	1	3	0	8	9	2	0	
4	8	3	7	4	8	5	6	0	4	9	1	10
5	4	6	3	7	5	6	7	4	8	8	2	11, 10
6	0	9	2	8	1	0	5	5	8	2	0	
7	7	2	9	5	0	8	8	5	7	9	2	9, 10
8	9	5	8	6	1	1	1	6	5	2	0	
9	7	0	5	5	5	0	8	7	6	7	3	10, 10, 11
10	6	4	7	2	3	8	2	9	3	4	1	11
11	4	1	1	2	0	7	7	5	5	6	2	10, 8
12	3	4	4	0	6	7	2	4	8	6	1	11
13	1	8	8	2	4	1	2	9	6	3	0	
14	0	6	8	4	0	1	2	0	0	6	0	
15	0	9	3	3	1	4	7	9	1	4	1	11
16	7	4	5	7	4	7	7	4	6	8	4	10, 11, 11, 11
17	5	4	3	5	8	8	0	7	8	8	1	9
18	9	6	7	0	8	5	2	9	1	3	1	8
19	1	2	9	1	2	6	5	7	3	0	1	11
20	4	8	9	0	0	3	1	3	0	5	0	
21	0	0	9	9	5	2	0	8	5	8	0	
22	3	0	9	0	9	0	8	8	7	2	1	11
23	2	0	3	9	5	9	3	1	8	1	0	
24	5	9	7	3	4	7	0	4	9	5	2	9,11

Để tiến hành minh họa quá trình tính toán mô phỏng cho số tín hiệu đến trong từng khoảng thời gian 5 phút và thời gian phục vụ mỗi tín hiệu, chúng ta quy ước các kí hiệu sau:

○ Tín hiệu đến      ..... Chờ      ————— Được phục vụ

Hình III.5 tổng hợp kết quả tính toán mô phỏng cho hệ thống chờ hai kênh phục vụ – một dịch vụ.



**Hình III.5. Tổng hợp kết quả mô phỏng hệ thống chờ**

Theo hình III.5 có thể thấy, trong thời gian 120 phút có 25 tín hiệu đến (25 số 7 xuất hiện ở các chu kì 2, 4, 5, 7, 9, 10, 11, 12, 15, 16, 17, 18, 19, 22, 24). Tổng thời gian đợi 213 phút, vì vậy thời gian đợi trung bình =  $\frac{213}{25} = 8,52$  phút.

Chúng ta phân tích ý nghĩa kinh tế của tính toán mô phỏng như sau: Mô hình hàng chờ trên có hai kênh phục vụ (hai nhân viên phục vụ). Giả sử rằng lương 7\$/1 giờ/1 nhân viên phục vụ, lương của mỗi đốc công là 12\$/1 giờ/1 đốc công. Theo dữ kiện của bài toán, thời gian trung bình giữa hai lần tín hiệu đến liên tiếp là 5 phút. Như vậy, trong trong 8 giờ có 96 lần đi, và phải đợi 8,52 phút mỗi lần. Do đó, tổng hao hụt / mỗi ngày làm việc do thời gian đốc công lãng phí do phải đợi là:  $(8,52 \times 12\$ \times 96) / 60 = 163,56\$$ . Ngoài ra, tổng chi lương / ngày cho hai nhân viên phục vụ là:  $7\$ \times 8 \times 2$

người = 112\$. Từ đó, tổng chi phí / ngày cho hệ thống hàng chờ trên là \$ 275,56.

Tương tự, nếu bố trí ba kênh phục vụ (ba nhân viên phục vụ) thì có thể tính được thời gian trung bình chờ đợi là 1,88 phút và tổng chi phí cho hệ thống ba kênh là 204\$. Nếu dùng bốn kênh phục vụ thì thời gian chờ đợi trung bình là 0 phút, tổng chi phí là 224\$. Vậy hệ thống hàng chờ trên nên dùng ba kênh phục vụ là tốt nhất. Nói cách khác, ban điều hành công ti nên điều thêm một nhân viên phục vụ tới làm việc tại kho cấp phát nguyên liệu nhằm giảm bớt chi phí “cơ hội”.

#### *So sánh với lời giải lí thuyết*

So sánh lời giải dùng mô phỏng với lời giải dựa trên lí thuyết mô hình hàng chờ (lúc đó cần giả sử rằng: số tín hiệu đến tuân theo luật Poát-xông, thời gian phục vụ một tín hiệu tuân theo phân phối mũ).

Khi tra phụ lục 3 tìm  $P_0$  dựa vào tỉ số  $A/(kS)$  với  $k = 3$  là số kênh phục vụ, ta được

$$P_0 = 0,11435 \quad (A = 12, S = 6; A/(kS) = 0,66);$$

$$L_s = \frac{AS(A/S)^k}{(k-1)!(kS-A)^2} P_0 + \frac{A}{S} = 2,915; \quad L_q = L_s - \frac{A}{S} = 0,915;$$

$$W_q = \frac{L_q}{A} = 0,076 \text{ (giờ)} = 4,5 \text{ phút.}$$

Trong khi đó theo kết quả mô phỏng thì thời gian chờ trung bình là 1,88 phút. Tuy nhiên, nếu chạy mô phỏng cho thời gian dài hơn (10 tới 15 giờ) thông qua chương trình trên máy tính thì có kết quả thời gian chờ trung bình là 2,63 phút tương đối ổn định.

Từ các phân tích trên có thể thấy, trong các tình huống chúng ta không xử lí được các điều kiện toán học phức tạp ẩn chứa trong bài toán hàng chờ, lựa chọn duy nhất là áp dụng mô phỏng để giải bài toán.

#### ***Phần mềm mô phỏng hệ thống hàng chờ***

Phần mềm máy tính với tên gọi MOPHONG2 phiên bản 1.0 đã được thiết kế dựa trên ngôn ngữ Builder C++ 5.0 để mô phỏng hệ thống hàng chờ như đã trình bày trong ví dụ 2 với các điều kiện sau:

– Số kênh phục vụ  $n$  có thể lấy giá trị từ 2 tới 10.

– Dòng tín hiệu đến theo quy luật: cứ trung bình 5 phút có một tín hiệu đến, nhưng chưa biết đây có phải là phân phối Poát-xông hay không. Ngoài ra, thời điểm đến của các tín hiệu trong vòng 5 phút cũng chưa biết rõ, nên chúng được mô phỏng căn cứ bảng III.8.



– Thời gian phục vụ (xử lý) một tín hiệu tuân theo phân phối rời rạc đã biết:

Thời gian xử lý một tín hiệu (phút)	8	9	10	11
Xác suất	0,1	0,2	0,3	0,4

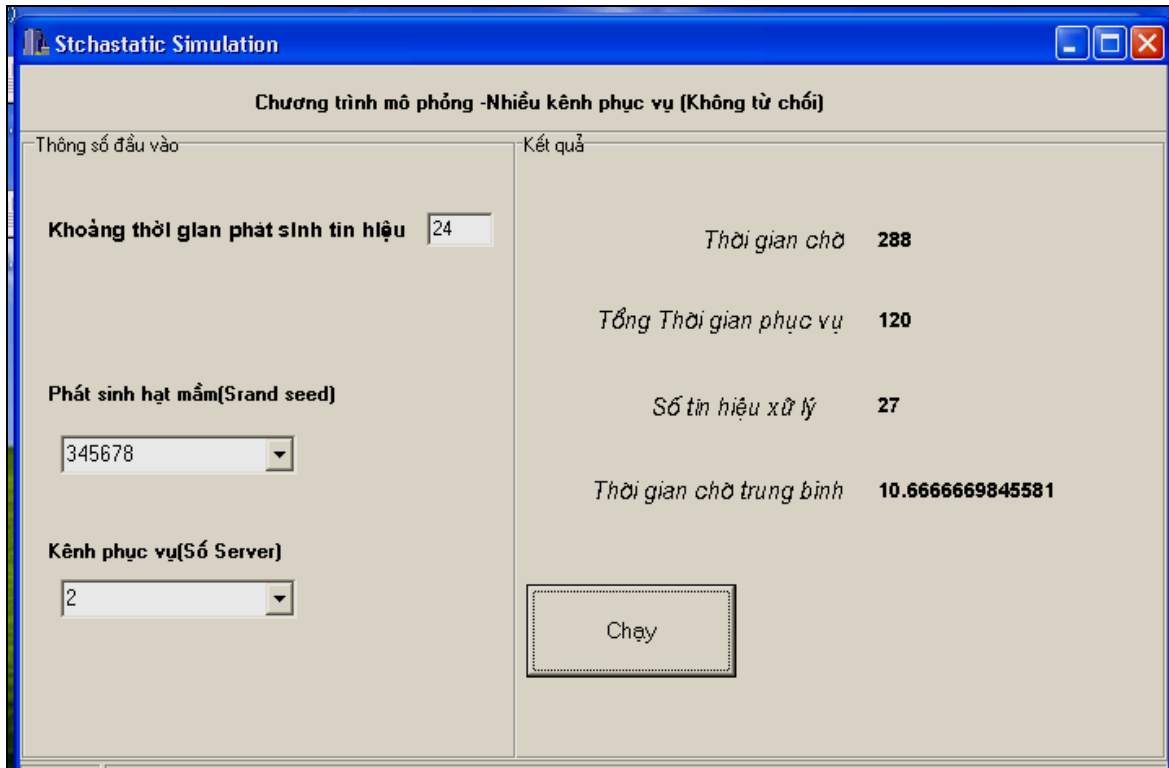
– Khoảng thời gian mô phỏng có thể chọn  $k$  chu kì, mỗi chu kì 5 phút ( $k$  có thể chọn giá trị 12, 24, 32, 48, ...)

– Cần tính thời gian chờ trung bình (phút) cho mỗi tín hiệu đi đến hệ thống dịch vụ trên đây.

Mục đích của việc xây dựng phần mềm này là phục vụ dạy và học môn Mô phỏng ngẫu nhiên, cũng như tiếp tục nâng cấp phiên bản 1.0 để mô phỏng được hệ nhiều kênh phục vụ – nhiều loại dịch vụ không có từ chối trong trường hợp tổng quát khi dòng tín hiệu đến và thời gian phục vụ tín hiệu có phân phối bất kì.

Để chạy phần mềm MOPHONG2, cũng tương tự như chạy phần mềm MOPHONG1 chúng ta cần cài đặt Builder C++ 5.0 vào máy tính. Sau khi kích chuột vào biểu tượng của phần mềm, chọn *File > Open Project > Look in > Mophong2 (Thư mục lưu trữ phần mềm) > Mp2.bpr*. Sau đó chọn *Run* trên thanh công cụ để chạy phần mềm (xem hình III.6).

Chúng ta nhập số kênh phục vụ  $n = 3$  vào ô số (kênh) dịch vụ, khoảng thời gian mô phỏng với  $m = 24$  chu kì 5 phút vào ô khoảng thời gian phát sinh tín hiệu. Ngoài ra, ta phải chọn hạt mầm là một số nguyên (đủ lớn) nhằm khởi tạo hàm sinh số ngẫu nhiên có phân phối đều trong  $[0, 1)$ , chẳng hạn số 345678 để ghi vào ô hạt mầm ngẫu nhiên. Sau đó chúng ta kích chuột vào nút *Chạy* để chạy chương trình.



**Hình III.6. Chạy phần mềm MOPHONG2**

Kết quả ta thấy tổng cộng có 27 tín hiệu đi tới hệ thống dịch vụ, tổng thời gian chờ là 288 phút, nên trung bình một tín hiệu phải chờ là 10,667 phút. Kết quả này so với kết quả tính toán trên giấy sử dụng bảng số ngẫu nhiên (phụ lục 2A) là khá sát nhau.

Cần nhắc lại rằng việc tính toán dựa vào bảng số ngẫu nhiên cho trong phụ lục 2A và dựa vào hàm sinh số ngẫu nhiên trong máy tính không cho kết quả hoàn toàn giống nhau do các bộ số ngẫu nhiên tạo được không giống nhau. Với các hạt mầm khác nhau thì hàm sinh số ngẫu nhiên cũng cho các bộ số ngẫu nhiên khác nhau và do đó các kết quả cuối cùng cũng không trùng nhau. Chẳng hạn, nếu thay hạt mầm 345678 (cho kết quả thời gian chờ trung bình 10,667 phút) bằng hạt mầm 456789 thì thời gian chờ trung bình sẽ chỉ là 4,158 phút. Nếu lấy trung bình của hai kết quả này thì được con số sát hơn với kết quả tính toán đã biết dựa trên phụ lục 2A.

Từ các phân tích trên ta thấy, muốn kết quả mô phỏng ổn định hơn cần chọn số chu kỳ mô phỏng  $m$  đủ lớn và chạy cho nhiều hạt mầm khác nhau để tính giá trị trung bình của kết quả các lần chạy đó. Vấn đề này thực chất là vấn đề *bổ trí thí nghiệm mô phỏng* trên máy tính, cũng là một hướng trong nghiên cứu về mô phỏng ngẫu nhiên hiện nay đang được tiến hành trong nhiều lĩnh vực.