TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

50 NĂM XÂY DỰNG VÀ PHÁT TRIỂN

1956 - 2006

KỶ NIỆM 45 NĂM THÀNH LẬP BỘ MÔN TỰ ĐỘNG HOÁ XNCN

GS. TS. NGUYỄN CÔNG HIỀN
TS. NGUYỄN PHẠM THỤC ANH

MÔ HÌNH HOÁ HỆ THỐNG VÀ MÔ PHỎNG

NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
50 NĂM XÂY DỰNG VÀ PHÁT TRIỂN
KỶ NIỆM 45 NĂM THÀNH LẬP BỘ MÔN TỰ ĐỘNG HOÁ XNCN
1956 - 2006

GS. TS. NGUYỄN CÔNG HIỀN
TS. NGUYỄN PHẠM THỤC ANH

MÔ HÌNH HOÁ HỆ THỐNG VÀ MÔ PHỎNG

NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT
HÀ NỘI

LỜI NÓI ĐẦU

Mô hình hoá và mô phỏng là một phương pháp nghiên cứu khoa học được ứng dụng rất rộng rãi: từ nghiên cứu, thiết kế, chế tạo đến vận hành các hệ thống. Ngày nay nhờ sự trợ giúp của máy tính có tốc độ tính toán cao và bộ nhớ lớn mà phương pháp mô hình hoá được phát triển mạnh mẽ, đưa lại hiệu quả to lớn trong nghiên cứu khoa học và thực tiễn sản xuất. Mô hình hoá và mô phỏng được ứng dụng không những vào lĩnh vực khoa học công nghệ mà còn ứng dụng có hiệu quả vào nhiều lĩnh vực khác như quân sự, kinh tế và xã hội v.v... Ngày nay có nhiều công trình nghiên cứu về những vấn đề cơ bản của mô hình hoá và mô phỏng [1, 2, 3, 6,7, 12] cũng như ứng dụng kỹ thuật mô phỏng vào các lĩnh vực khác nhau [8, 9, 10, 20, 26].

Mô hình hoá và mô phỏng là một công cụ mạnh của cán bộ nghiên cứu, cán bộ kỹ thuật để giải các bài toán kỹ sư, quy hoạch, tối ưu hoá v.v... Phương pháp mô hình hoá và mô phỏng được dùng phổ biến trong các trường đại học, các viện nghiên cứu cũng như ở các cơ sở sản xuất và đã đưa lại hiệu quả to lớn.

Cuốn sách này trình bày những kiến thức cơ bản về mô hình hoá và mô phỏng cũng như ứng dụng của nó trong các lĩnh vực khác nhau.

Cuốn sách được dành cho các kỹ sư, cán bộ kỹ thuật trong lĩnh vực khoa học công nghệ đồng thời cũng có thể làm tài liệu tham khảo cho sinh viên của các trường đại học và cao đẳng.

Phân công viết cuốn sách này như sau:

Nguyễn Công Hiền viết từ Chương 1 đến Chương 10

Nguyễn Phạm Thục Anh viết Chương 11.

Mô hình hoá và mô phỏng là một lĩnh vực rộng lớn và đang phát triển. Do khuôn khổ cuốn sách có hạn, nên tác giả chỉ lựa chọn một số vấn đề cơ bản để trình bày trong cuốn sách này. Hy vọng sẽ cung cấp cho các độc giả những cơ sở ban đầu để nghiên cứu và phát triển kỹ thuật mô hình hoá và mô phỏng.

Các tác giả tỏ lòng biết ơn các ý kiến góp ý và nhận xét quý báu của các bạn đọc. Các ý kiến xin gửi về: Nhà xuất bản Khoa học và Kỹ thuật, 70 Trần Hưng Đạo, Hà Nội.

Các tác giả

GS. TS. Nguyễn Công Hiền

TS. Nguyễn Phạm Thục Anh

Chương 1
VAI TRÒ CỦA MÔ HÌNH HOÁ HỆ THỐNG

1.1. Khái niệm chung

Ngày nay khó có thể tìm thấy lĩnh vực hoạt động nào của con người mà không sử dụng phương pháp mô hình hoá ở những mức độ khác nhau. Điều này đặc biệt quan trọng đối với lĩnh vực điều khiển các hệ thống kỹ thuật và xã hội, bởi vì điều khiển chính là quá trình thu nhận thông tin từ hệ thống, nhận dạng hệ thống theo một mô hình nào đó và đưa ra quyết định thích hợp để điều khiển hệ thống. Quá trình này được tiếp diễn liên tục nhằm đưa hệ thống vận động theo một mục tiêu định trước [1, 2, 3, 4, 5, 12].

Quá trình phát triển khoa học kỹ thuật đi theo các bước cơ bản sau đây: quan sát - thu thập dữ liệu - nghiên cứu lý thuyết - thực nghiệm - tổ chức sản xuất. Mô hình hoá là một phương pháp khoa học trợ giúp cho các bước nói trên.

Phương pháp mô hình hoá và mô phỏng được phát triển từ Đại chiến thế giới thứ hai vào những năm 40 của thế kỷ 20. Lúc đó người ta ứng dụng phương pháp mô phỏng để nghiên cứu phản ứng hạt nhân nhằm chế tạo bom nguyên tử [3].

Mô hình hoá và mô phỏng là một trong những kỹ thuật được dùng rộng rãi nhất trong khoa học kỹ thuật cũng như khoa học quản lý. Theo điều tra khảo sát tại Trường Case Wesstern Reserve University (Mỹ, năm 1978) [1], cho thấy đối với bậc học Master thì "mô phỏng" là môn học đứng thứ 5 trong 15 môn học hàng đầu (đứng sau môn phương pháp thống kê, khoa học dự báo, phân tích hệ thống và hệ thống thông tin), còn đối với bậc học Tiến sĩ thì mô phỏng là môn học đứng thứ 2 cùng với môn "lập trình tuyến tính" (chỉ sau môn phương pháp thống kê).

Theo số liệu đã dẫn ở [1] thì khảo sát ở 137 hãng sản xuất lớn cho thấy trong 40 kỹ thuật mà hãng sử dụng thì "kỹ thuật mô phỏng" đứng thứ 2 (84% hãng đã sử dụng) chỉ sau " phân tích thống kê " (93% hãng đã sử dụng).

Những số liệu nêu trên là vào những năm 70 của thế kỷ 20, ngày nay kỹ

thuật máy tính, công nghệ thông tin đã có bước tiến nhảy vọt nên kỹ thuật mô phỏng được phát triển lên một mức cao hơn, phong phú hơn.

Nhờ có máy tính điện tử mà phương pháp mô hình hoá và mô phỏng phát triển nhanh chóng và được ứng dụng vào nhiều lĩnh vực khoa học kỹ thuật cũng như khoa học xã hội khác nhau. Nhờ có phương pháp mô hình hoá và mô phỏng người ta có thể phân tích, nghiên cứu các hệ thống phức tạp, xác định các đặc tính, hành vi hoạt động của các hệ thống. Các kết quả mô phỏng được dùng để thiết kế, chế tạo cũng như xác định chế độ vận hành của hệ thống. Nhờ có phương pháp mô hình hoá và mô phỏng mà người ta có thể đưa ra nhiều "kịch bản" khác nhau để từ đó lựa chọn phương án ttối ưu.

Đối với các hệ thống phức tạp, phi tuyến, ngẫu nhiên, các tham số biến đổi theo thời gian, phương pháp giải tích truyền thống không thể cho ta lời giải chính xác được. Lúc này phương pháp mô hình hoá và mô phỏng phát huy thế mạnh của mình và trong nhiều trường hợp nó là giải pháp duy nhất để nghiên cứu các hệ thống phức tạp nói trên.

Mục 1.2 trình bày một số định nghĩa cơ bản dùng trong mô hình hoá. Mục 1.3 nói về hệ thống và mô hình hệ thống. Cuối cùng mục 1.4 bàn về triển vọng phát triển của phương pháp mô hình hoá và mô phỏng.

1.2. Một số định nghĩa cơ bản

Trước khi đi vào nghiên cứu chi tiết, chúng ta hãy thống nhất một số định nghĩa cơ bản dùng trong cuốn sách này:

• **Đối tượng** (*Object*) là tất cả những sự vật, sự kiện mà hoạt động của con người có liên quan tới và cần nghiên cứu nó [1, 2,3].

• **Hệ thống** *(System)* Thuật ngữ *hệ thống* được dùng với nghĩa rất rộng tuỳ thuộc vào mục đích nghiên cứu, vì vậy khó có thể đưa ra được một định nghĩa đầy đủ. Sau đây chúng ta bắt đầu với một định nghĩa tương đối đơn giản, về sau có thể mở rộng tuỳ thuộc và mục đích nghiên cứu và hệ thống cụ thể. *Hệ thống* là tập hợp các đối tượng, sự kiện (con người, máy móc) mà giữa chúng có mối quan hệ nhất định [1, 2, 3].

• **Trạng thái của hệ thống** *(State of System)* là tập hợp các biến số, tham số dùng để mô tả hệ thống tại một thời điểm và điều kiện nhất định.

• **Mô hình** (*Model*) là một sơ đồ phản ánh đối tượng hoặc hệ thống. Con

người dùng sơ đồ đó để nghiên cứu, thực nghiệm nhằm tìm ra các quy luật hoạt động của đối tượng hoặc hệ thống. Hay nói một cách khác *mô hình* là đối tượng thay thế của đối tượng gốc (đối tượng thực tế) dùng để nghiên cứu về đối tượng gốc [1, 2, 3].

• **Mô hình hoá** *(Modeling)* là thay thế đối tượng gốc bằng một mô hình để nhằm thu nhận các thông tin về đối tượng bằng cách tiến hành các thực nghiệm, tính toán trên mô hình. Lý thuyết xây dựng mô hình và nghiên cứu mô hình để hiểu biết về đối tượng gốc gọi là lý thuyết mô hình hoá [1, 2, 3, 4, 5].

Mô hình hoá là một phương pháp khoa học để nghiên cứu đối tượng. Nếu như các quá trình xảy ra trong mô hình đồng nhất – theo các chỉ tiêu định trước – với các quá trình xảy ra trong đối tượng gốc thì người ta nói rằng mô hình đồng nhất với đối tượng. Lúc này người ta có thể tiến hành các thực nghiệm trên mô hình để thu nhận các thông tin về đối tượng [1, 2, 3].

• **Mô phỏng** *(Simulation, Imitation)* là phương pháp mô hình hoá dựa trên việc xây dựng mô hình số (*numerical model*) và dùng phương pháp số (*numerical methods*) để tìm các lời giải. Chính vì vậy, máy tính số là công cụ duy nhất và hữu hiệu để thực hiện việc mô phỏng hệ thống [1, 2, 3, 12].

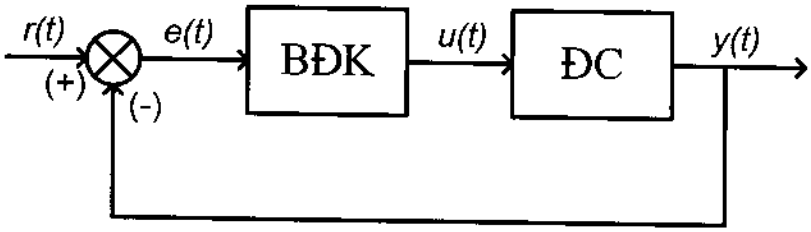
Lý thuyết cũng như thực nghiệm đã chứng minh rằng chúng ta chỉ có thể xây dựng được các mô hình gần đúng với đối tượng mà thôi, vì trong quá trình mô hình hoá bao giờ cũng phải chấp nhận một số giả thiết nhằm giảm bớt độ phức tạp của mô hình, để mô hình có thể ứng dụng thuận tiện trong thực tế. Mặc dầu vậy mô hình hoá luôn luôn là một phương pháp hữu hiệu để con người nghiên cứu đối tượng, nhận biết các quá trình, các quy luật tự nhiên. Đặc biệt ngày nay nhờ có sự trợ giúp đắc lực của kỹ thuật máy tính, kỹ thuật tin học, người ta đã phát triển các phương pháp mô hình hoá cho phép xây dựng các mô hình ngày càng gần với đối tượng nghiên cứu, đồng thời việc thu nhận lựa chọn xử lý các thông tin về mô hình rất thuận tiện, nhanh chóng và chính xác.

Chính vì vậy, mô hình hoá là một phương pháp nghiên cứu khoa học mà tất cả các kỹ sư đều phải nghiên cứu và ứng dụng vào thực tiễn hoạt động của mình.

1.3. Hệ thống và mô hình hệ thống

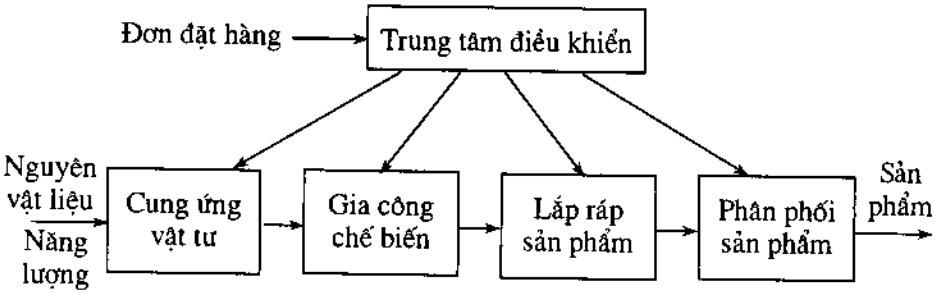
Đầu tiên chúng ta xem xét một số ví dụ về các hệ thống tương đối đơn giản. Hình 1.1. trình bày hệ thống tự động điều khiển tốc độ động cơ. Tín hiệu vào

của hệ thống *r(t)* là tốc độ đặt mong muốn, tín hiệu ra của hệ thống *y (t)* là tốc độ thực tế của động cơ. Sai lệch tốc độ *e(t) = y (t) − r(t)* được đưa vào bộ điều khiển *BĐK* để tạo ra tín hiệu điều khiển *u(t)* tác động vào động cơ *ĐC* nhằm duy trì tốc độ động cơ ở mức mong muốn.



Hình 1.2. Hệ thống điều chỉnh tự động tốc độ động cơ.

Hình 1. 2 trình bày hệ thống điều khiển quá trình sản xuất. Hệ thống sản xuất gồm nhiều hệ con chức năng như: cung cấp vật tư, năng lượng; gia công, chế biến; lắp ráp, hoàn thiện sản phẩm; phân phối, tiêu thụ. Điều khiển quá trình sản xuất là trung tâm điều khiển. Đầu vào của hệ thống là đơn đặt hàng của khách hàng, đầu ra của hệ thống là sản phẩm cuối cùng.



Hình 1.2. Hệ thống điều khiển quá trình sản xuất.

Từ hình 1.1 và hình 1.2 chúng ta thấy rằng trong hệ thống có nhiều phần tử thường được gọi là các *thực thể (Entity)*, mỗi một thực thể có các *thuộc tính (Attribute)* của nó. Một quá trình gây ra thay đổi trong hệ thống được gọi là một *hoạt động (Activity)*. Một tác động làm thay đổi trạng thái của hệ thống được gọi là một *sự kiện (Event)*. Tập hợp các biến phản ánh trạng thái của hệ thống tại một thời điểm được gọi là *biến trạng thái (State variable)*.

Tuỳ theo mục đích nghiên cứu mà hệ thống được mô tả với mức độ chi tiết khác nhau. Một số hệ thống với các đặc tính cơ bản được trình bày trong bảng 1.1.

Bảng 1.1

Hệ thống	*Thực thể*	*Thuộc tính*	*Hoạt động*	*Sự kiện*	*Biến trạng thái*
Giao thông	Xe buýt	Tốc độ Khoảng cách	Lái xe	Xe đến bến	Số khách hàng chờ ở bến
Ngân hàng	Khách hàng	Kiểm tra tài khoản	Rút gửi tiền	Số khách hàng	Số nhân viên phục vụ
Thông tin liên lạc	Thông tin	Thời lượng liên lạc	Truyền tin	Thông tin truyền đến	Số người đợi liên lạc
Siêu thị	Khách hàng	Danh mục mua sắm	Tính trả tiền	Khách hàng đến siêu thị	Số khách hàng rời siêu thị
Trạm lắp ráp sản phẩm	Sản phẩm	Kích thước, trọng lượng	Lắp ráp sản phẩm	Sản phẩm hoàn thiện	Số sản phẩm hoàn thiện

Có hai con đường để nghiên cứu hệ thống: nghiên cứu trên hệ thực và nghiên cứu trên mô hình thay thế của nó. Rõ ràng rằng nghiên cứu trên hệ thực cho ta kết quả trung thực và khách quan. Tuy nhiên trong nhiều trường hợp tiến hành nghiên cứu trên hệ thực gặp nhiều khó khăn như sẽ được trình bày dưới đây, do đó phương pháp tốt nhất và thuận tiện nhất là nghiên cứu trên mô hình của nó. Chính vì vậy phương pháp mô hình hoá rất được chú ý nghiên cứu và phát triển và phương pháp này đóng vai trò quan trọng trong sự nghiệp phát triển khoa học và kỹ thuật.

1.4. Triển vọng phát triển của phương pháp mô hình hoá hệ thống

Trước đây phương pháp giải tích được dùng để mô hình hoá hệ thống. Tuy máy tính đã giúp cho việc tính toán được thuận lợi như tăng khối lượng tính toán, giảm thời gian tính v.v... nhưng bản thân phương pháp giải tích gặp nhiều khó khăn khi mô tả hệ thống như thường phải chấp nhận nhiều giả thiết để đơn giản hoá mô hình, do đó các kết quả nghiên cứu có độ chính xác không cao.

Ngày nay bên cạnh phương pháp giải tích nói trên, phương pháp mô phỏng (*Imitation, Simulation*) được phát triển mạnh mẽ và ứng dụng rất rộng rãi. Các mô hình được xây dựng dựa trên phương pháp mô phỏng được gọi là *mô hình*

mô phỏng (hay còn gọi là mô hình số). Phương pháp mô phỏng cho phép đưa vào mô hình nhiều yếu tố gần sát với thực tế. Đồng thời mô hình được giải trên các máy tính có tốc độ tính nhanh, dung lượng lớn, do đó các kết quả thu được có độ chính xác cao. Vì vậy phương pháp mô phỏng đã tạo điều kiện để giải các bài toán phức tạp như bài toán mô hình hoá các hệ thống lớn, hệ thống ngẫu nhiên, phi tuyến có các thông số biến thiên theo thời gian [1, 2, 3, 4,5, 12].

Phương pháp mô phỏng đặc biệt phát huy hiệu quả khi cần mô hình hoá các hệ thống lớn mà đặc điểm cơ bản của nó là có cấu trúc phân cấp, cấu trúc hệ con, giữa các hệ con và trung tâm điều khiển có sự trao đổi thông tin với nhau. Phương pháp mô phỏng cũng tỏ ra hữu hiệu khi mô hình hoá các hệ thống có các yếu tố ngẫu nhiên, có thông tin không đầy đủ, các thông tin sẽ được bổ sung trong quá trình mô phỏng, trong quá trình trao đổi thông tin giữa người điều khiển với đối tượng.

Phương pháp mô phỏng được ứng dụng để mô hình hoá trong nhiều lĩnh vực khác nhau như: kỹ thuật, kinh tế, xã hội, sinh học, đặc biệt là các hệ thống lớn, phức tạp, có nhiều yếu tố ngẫu nhiên tác động.

Tóm lại, mô hình hoá là một phương pháp nghiên cứu khoa học đang phát triển và rất có triển vọng.

Ở giai đoạn thiết kế hệ thống, mô hình hoá giúp người thiết kế lựa chọn cấu trúc, các thông số của hệ thống để tổng hợp hệ thống. Ở giai đoạn chế tạo, mô hình hoá giúp cho việc lựa chọn vật liệu và công nghệ chế tạo. Ở giai đoạn vận hành hệ thống, mô hình hoá giúp cho người điều khiển giải các bài toán điều khiển tối ưu, dự đoán các trạng thái của hệ thống. Đặc biệt trong trường hợp kết hợp hệ chuyên gia (*Expert System*) với mô hình hoá người ta có thể giải được nhiều bài toán điều khiển, tiết kiệm được thời gian cũng như chi phí về vật chất và tài chính.

Phương pháp mô hình hoá thường được dùng trong các trường hợp sau đây:

1. Khi nghiên cứu trên hệ thống thực gặp khó khăn do nhiều nguyên nhân gây ra như :

- Giá thành nghiên cứu trên hệ thống thực quá đắt

Ví dụ: Nghiên cứu kết cấu tối ưu, độ bền, khả năng chống dao động của ôtô, tàu thuỷ, máy bay người ta phải tác động các lực đủ lớn đến mức phá huỷ

các đối tượng nghiên cứu nói trên để từ đó đánh giá các chỉ tiêu kỹ thuật đã đề ra. Như vậy giá thành nghiên cứu sẽ rất đắt. Bằng cách mô hình hoá trên máy tính chúng ta có thể thu được nhiều phương án khác nhau để từ đó lựa chọn phương án ttối ưu của các thiết bị nói trên.

• Nghiên cứu trên hệ thống thực đòi hỏi thời gian quá dài

Ví dụ: Nghiên cứu đánh giá độ tin cậy, đánh giá tuổi thọ trung bình của hệ thống kỹ thuật (thông thường tuổi thọ trung bình của hệ thống kỹ thuật khoảng 30 ~ 40 năm) hoặc nghiên cứu quá trình phát triển dân số trong quãng thời gian 20 - 50 năm v.v... Nếu chờ đợi quãng thời gian dài như vậy mới có kết quả nghiên cứu thì không còn tính thời sự nữa. Bằng cách mô phỏng hệ thống và cho hệ thống "vận hành" tương đương với quãng thời gian nghiên cứu. Bằng cách thay đổi các dữ liệu ban đầu và điều kiện vận hành người ta có thể đưa ra các "kịch bản" khác nhau để đánh giá, so sánh và lựa chọn các chỉ tiêu kinh tế kỹ thuật của hệ thống.

• Nghiên cứu trên hệ thống thực ảnh hưởng đến sản xuất hoặc gây nguy hiểm cho người và thiết bị.

Ví dụ: Nghiên cứu quá trình cháy trong lò hơi của nhà máy nhiệt điện, trong lò luyện clanhke của nhà máy ximăng v.v... người ta phải thay đổi chế độ cấp nhiên liệu (than, dầu), tăng giảm lượng gió cấp, thay đổi áp suất trong lò v.v... Việc làm các thí nghiệm như vậy sẽ cản trở sản xuất bình thường, trong nhiều trường hợp có thể xảy ra cháy, nổ gây nguy hiểm cho người và thiết bị. Bằng cách mô phỏng hệ thống, người ta có thể cho hệ thống "vận hành" với các bộ thông số, các chế độ vận hành khác nhau để tìm ra lời giải tối ưu. Trong trường hợp này mô hình cũng được dùng để đào tạo và huấn luyện các cán bộ kỹ thuật trước khi họ tham gia vào vận hành hệ thống thực.

• Trong một số trường hợp không cho phép làm thực nghiệm trên hệ thống thực

Ví dụ: Nghiên cứu các hệ thống làm việc ở những nơi độc hại, nguy hiểm, dưới hầm sâu, đáy biển hoặc nghiên cứu trên cơ thể con người v.v... Trong những trường hợp này dùng phương pháp mô phỏng là giải pháp duy nhất để nghiên cứu hệ thống.

2. *Phương pháp mô hình hoá cho phép đánh giá độ nhạy của hệ thống* khi thay đổi tham số hoặc cấu trúc của hệ thống cũng như đánh giá phản ứng của hệ

thống khi thay đổi tín hiệu điều khiển. Những số liệu này dùng để thiết kế hệ thống hoặc lựa chọn thông số tối ưu để vận hành hệ thống.

Ví dụ: Cần nghiên cứu độ nhạy của biến động giá thành sản phẩm khi giá điện năng tăng lên.

3. Phương pháp mô hình hoá cho phép nghiên cứu hệ thống ngay cả khi chưa có hệ thống thực

Trong trường hợp chưa có hệ thống thực thì nghiên cứu trên mô hình là biện pháp duy nhất để đánh giá các chỉ tiêu kỹ thuật của hệ thống, lựa chọn cấu trúc và các thông số tối ưu của hệ thống v.v...

Ví dụ: Trước khi xây dựng nhà máy thuỷ điện lớn người ta phải dùng phương pháp mô hình hoá để nghiên cứu, lựa chọn kết cấu và thông số kỹ thuật đập chính của nhà máy.

Ngày nay nhờ có những tiến bộ vượt bậc của kỹ thuật máy tính và công nghệ thông tin mà phương pháp mô hình hoá và mô phỏng phát triển lên một mức cao và đi theo một số hướng sau đây:

+ Mô phỏng các hệ thống lớn phức tạp, đặc biệt là các hệ phi tuyến, ngẫu nhiên. Ứng dụng kỹ thuật đồ hoạ 3 chiều, kỹ thuật tạo hình ảnh động (hoạt hình) để xây dựng những chương trình mô phỏng sinh động, trực quan rất thuận tiện cho việc nghiên cứu và hiển thị các kết quả mô phỏng.

+ Mô phỏng các hệ thống sản xuất như quy hoạch nguồn lực sản xuất, lập kế hoạch sản xuất, quản lý kho v.v...

+ Mô phỏng các hệ thống dịch vụ như trạm sửa chữa ôtô, phòng khám bệnh, nhà hàng, siêu thị v.v...

+ Mô phỏng các hệ thống trò chơi (*game*), lĩnh vực này được ứng dụng trong giải trí, quân sự, kinh doanh v.v...

+ Mô phỏng các hệ thống đào tạo như phòng thí nghiệm ảo, lớp học điện tử (*E-Learning*), phòng đào tạo lái xe ôtô, máy bay, tàu thuỷ, huấn luyện vận hành các hệ thống kỹ thuật phức tạp v.v...

Nhìn chung kỹ thuật mô hình hoá và mô phỏng ngày càng phát triển không những trong lĩnh vực khoa học kỹ thuật mà còn được ứng dụng thành công tronh nhiều lĩnh vực khác như kinh tế, xã hội, quân sự, y tế, giáo dục, kinh doanh, giải trí v.v...

Bài tập

1.1. Hãy xác định các thành phần của hệ thống là thực thể, thuộc tính, hoạt động, sự kiện và biến trạng thái của cảng biển sau đây: tàu đến cảng sẽ cập bến nếu như còn chỗ trống, ngược lại sẽ phải sắp hàng chờ đến lượt. Tàu được các cần cẩu bốc dỡ hàng. Khi hàng bốc xong tàu rời bến ngay

1.2. Cũng như câu 1.1 nhưng hệ thống là quán càphê, trạm rửa xe.

1.3. Hãy lấy ví dụ chứng minh những khó khăn gặp phải khi nghiên cứu trên hệ thực và những ưu điểm khi chuyển sang nghiên cứu bằng phương pháp mô phỏng.

Chương 2
KHÁI NIỆM CƠ BẢN VỀ MÔ HÌNH HOÁ HỆ THỐNG

2.1. Khái niệm chung

Ngày nay phương pháp *tiếp cận hệ thống* được dùng để phân tích và tổng hợp các hệ thống lớn. Khác với phương pháp truyền thống trước đây đi từ phần tử đến hệ thống, phương pháp tiếp cận hệ thống đi từ phân tích chung toàn hệ thống đến cấu tạo từng phân tử, đi từ xác định mục tiêu toàn hệ thống đến chức năng nhiệm vụ từng phần tử cụ thể, xác định mối tương quan giữa các phần tử trong hệ thống, giữa hệ thống đang xét với các hệ thống khác và với môi trường xung quanh. Người ta định nghĩa *hệ thống S* là tập hợp các phần tử có quan hệ với nhau, đó chính là đối tượng cần nghiên cứu. Môi trường xung quanh *E* là tập hợp các thực thể ngoài hệ thống, có tác động qua lại với hệ thống. Tuỳ thuộc vào mục đích nghiên cứu mà người ta xác định hệ thống *S* và môi trường *E* tương ứng [1, 12].

Khi tiến hành mô hình hoá điều quan trọng là xác định mục tiêu mô hình hoá, trên cơ sở đó xác định hệ thống *S*, môi trường *E* và mô hình *M*. Bước tiếp theo là xác định cấu trúc của hệ thống tức là tập các phần tử và quan hệ giữa chúng trong hệ thống.

Cấu trúc của hệ thống có thể được xem xét trên hai phương diện: từ phía ngoài và từ phía trong. Từ phía ngoài tức là xem xét các phần tử cấu tạo thành hệ thống và mối quan hệ giữa chúng, hay nói cách khác đó là *phương pháp tiếp cận cấu trúc*. Từ phía trong, tức là phân tích đặc tính chức năng của các phần tử cho phép hệ thống đạt tới mục tiêu đã định, hay nói một cách khác đó là *phương pháp tiếp cận chức năng*.

Khi xem xét sự vận động của hệ thống theo thời gian *S(t)*, có nghĩa là hệ thống chuyển từ trạng thái này sang trạng thái khác trong không gian trạng thái *Z*, người ta quan tâm đến chức năng hoạt động của hệ thống. Để đánh giá chức năng của hệ thống người ta phải xác định các chỉ tiêu đánh giá, hoặc là tập các chỉ tiêu riêng, hoặc là chỉ tiêu tổng hợp cho toàn hệ thống.

Tiếp cận hệ thống cho phép chúng ta xây dựng mô hình hệ thống lớn có tính đến nhiều yếu tố tác động trong nội bộ hệ thống S cũng như giữa hệ thống với môi trường *E*.

Người ta có thể chia quá trình mô hình hoá ra làm hai giai đoạn: giai đoạn thiết kế tổng thể (*Macro Design*) và giai đoạn thiết kế cụ thể *(Micro Design)*. Trong giai đoạn thiết kế tổng thể, trên cơ sở các dữ liệu của hệ thống thực *S* và của môi trường *E* người ta xây dựng mô hình hệ thống và mô hình môi trường thoả mãn các chỉ tiêu đánh giá định trước. Còn trong giai đoạn thiết kế cụ thể, trên cơ sở mô hình đã được lựa chọn người ta xác định các điều kiện ràng buộc, xây dựng các chương trình mô phỏng trên máy tính và thực hiện việc mô phỏng để tìm các đặc tính kinh tế kỹ thuật của hệ thống thực.

Mục 2.2 nói về đặc điểm của mô hình hoá hệ thống. Mục 2.3 trình bày cách phân loại hệ thống. Cuối cùng mục 2.4 trình bày một số nguyên tắc khi xây dựng mô hình.

2.2. Đặc điểm của mô hình hoá hệ thống

Cùng với sự phát triển của các phương pháp lý thuyết, các phương pháp thực nghiệm để nghiên cứu, phân tích và tổng hợp hệ thống ngày càng được hoàn thiện. Đối với một hệ thống thực có hai phương pháp cơ bản để nghiên cứu thực nghiệm, nghiên cứu trên hệ thực và nghiên cứu trên mô hình của nó. Nghiên cứu thực nghiệm trên hệ thực cho ta các số liệu khách quan, trung thực. Ở đây phải giải quyết vấn đề lấy mẫu thống kê, ước lượng tham số, phân tích và xử lý dữ liệu v.v... Tuy nhiên việc nghiên cứu thực nghiệm trên hệ thực trong nhiều trường hợp gặp khó khăn như đã được trình bày ở mục 1.4, trong trường hợp này nghiên cứu thực nghiệm trên các mô hình là phương pháp có nhiều triển vọng [1, 2, 12].

Nhìn chung các đối tượng thực có cấu trúc phức tạp và thuộc loại hệ thống lớn vì vậy mô hình của chúng cũng được liệt vào các hệ thống lớn và có những đặc điểm cơ bản sau đây [12]:

a) Tính mục tiêu: Tuỳ theo yêu cầu nghiên cứu có thể có mô hình chỉ có một mục tiêu là để nghiên cứu một nhiệm vụ cụ thể nào đó, hoặc mô hình đa mục tiêu nhằm khảo sát một số chức năng, đặc tính của đối tượng thực tế.

b) Độ phức tạp: Độ phức tạp thể hiện ở cấu trúc phân cấp của mô hình, các mối

quan hệ qua lại giữa các hệ con với nhau và giữa hệ thống S với môi trường *E*.

c) Hành vi của mô hình: Hành vi của mô hình là con đường để mô hình đạt được mục tiêu đề ra. Tuỳ thuộc có yếu tố ngẫu nhiên tác động vào hệ thống hay không mà ta có mô hình tiền định hoặc ngẫu nhiên. Theo hành vi của hệ thống có thể phân ra mô hình liên tục hoặc gián đoạn. Nghiên cứu hành vi của mô hình người ta biết được xu hướng vận động của đối tượng thực.

d) Tính thích nghi: Tính thích nghi là đặc tính của hệ thống có tổ chức cấp cao, hệ thống có thể thích nghi với sự thay đổi của các tác động vào hệ thống. Tính thích nghi của mô hình thể hiện ở khả năng mô hình phản ánh được các tác động của môi trường tới hệ thống và khả năng giữ ổn định của mô hình khi các tác động đó thay đổi.

e) Tính điều khiển được: Ngày nay nhiều phương pháp tự động hoá đã được ứng dụng trong mô hình hoá hệ thống. Sử dụng các biện pháp lập trình người ta có thể điều khiển mô hình theo mục tiêu định trước, thực hiện khả năng đối thoại giữa người với mô hình để thu nhận thông tin và ra quyết định điều khiển.

g) Khả năng phát triển của mô hình: Khi tiến hành mô hình hoá hệ thống bao giờ cũng xuất hiện bài toán nghiên cứu sự phát triển của hệ thống trong tương lai. Vì vậy mô hình phải có khả năng mở rộng, thu nạp thêm các hệ con, thay đổi cấu trúc để phù hợp với sự phát triển của hệ thống thực.

h) Độ tin cậy - Độ chính xác: Mô hình hoá là thay thế đối tượng thực bằng mô hình của nó để thuận tiện trong việc nghiên cứu. Vì vậy mô hình phải phản ánh trung thực các hiện tượng xảy ra trong đối tượng. Các kết quả thực nghiệm trên mô hình phải có độ chính xác, độ tin cậy thoả mãn yêu cầu đề ra.

Cần phải nhấn mạnh rằng kết quả mô hình hoá phụ thuộc rất nhiều vào khả năng và kinh nghiệm của người nghiên cứu. Một mặt, người nghiên cứu phải am hiểu đối tượng, nắm vững các hiện tượng, các quy luật xảy ra trong hệ thống thực. Mặt khác, người nghiên cứu phải biết lựa chọn phương pháp mô hình hoá thích hợp với từng đối tượng cụ thể, đồng thời phải có khả năng thực hiện mô hình trên máy tính - tức khả năng lập trình để giải các bài toán về mô hình hoá.

2.3. Phân loại mô hình hệ thống

Có thể căn cứ vào nhiều dấu hiệu khác nhau để phân loại mô hình. Có thể liệt kê từng cặp mô hình như sau:

Mô hình tiền định - Mô hình ngẫu nhiên

Mô hình tĩnh - Mô hình động

Mô hình tuyến tính - Mô hình phi tuyến

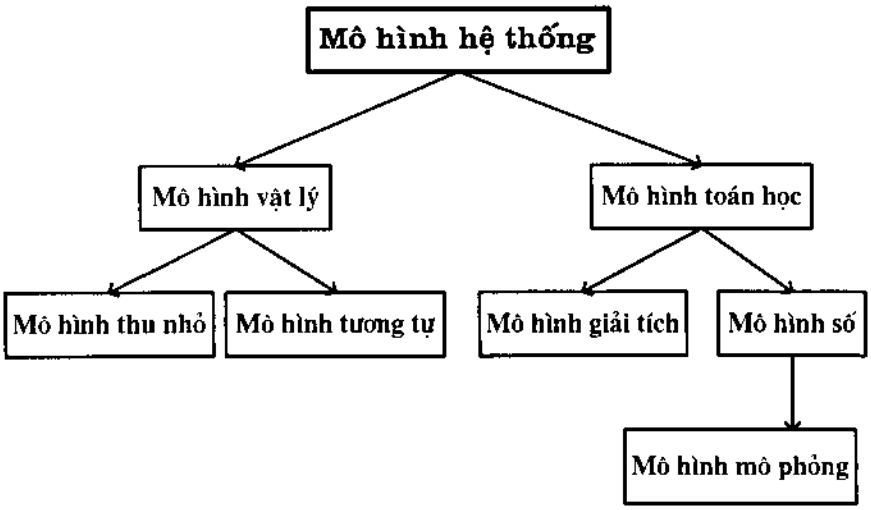
Mô hình liên tục - Mô hình gián đoạn

Mô hình vật lý - Mô hình toán học

Mô hình giải tích - Mô hình mô phỏng

Có thể liệt kê thêm các cặp mô hình như trên nữa, tuy nhiên sau đây chúng ta đưa ra một cách phân loại mô hình như trình bày trên hình 2-1.

Mô hình được chia làm hai nhóm chính: mô hình vật lý và mô hình toán học hay còn gọi là mô hình trừu tượng. Từ hai nhóm chính đó lại có thể chia ra thành các loại mô hình cụ thể hơn.



Hình 2.1. Sơ đồ phân loại mô hình.

Mô hình vật lý là mô hình được cấu tạo bởi các phần tử vật lý. Các thuộc tính của đối tượng được phản ánh bằng các định luật vật lý xẩy ra trong mô hình. Nhóm mô hình vật lý được chia thành mô hình thu nhỏ và mô hình tương tự.

Mô hình vật lý thu nhỏ có cấu tạo giống như đối tượng thực nhưng có kích thước nhỏ hơn cho phù hợp điều kiện của phòng thí nghiệm. Ví dụ người ta chế tạo lò hơi của nhà máy nhiệt điện có kích thước nhỏ đặt trong phòng thí nghiệm để nghiên cứu quá trình cháy trong lò hơi, hoặc xây dựng mô hình đập thuỷ điện

có kích thước nhỏ trong phòng thí nghiệm để nghiên cứu các chế độ thủy văn của đập thủy điện. Ưu điểm của loại mô hình này là các quá trình vật lý xẩy ra trong mô hình giống trong đối tượng thực, có thể đo lường quan sát các đại lượng vật lý một cách trực quan với độ chính xác cao. Nhược điểm của của loại mô hình vật lý thu nhỏ là giá thành đắt, vì vậy chỉ được dùng khi thật cần thiết.

Mô hình vật lý tương tự được cấu tạo bằng các phần tử vật lý không giống với đối tượng thực nhưng các quá trình xẩy ra trong mô hình tương đương với quá trình xảy ra trong đối tượng thực. Ví dụ có thể nghiên cứu quá trình dao động điều hoà của con lắc đơn bằng mô hình tương tự là mạch dao động *R-L-C* vì dòng điện dao động điều hoà trong mạch *R-L-C* hoàn toàn tượng tự quá trình dao động điều hoà của con lắc đơn, hoặc người ta có thể nghiên cứu đường dây tải điện (có thông số phân bố rãi) bằng mô hình tương tự là mạch bốn cực *R-L-C* (có thông số tập trung). Ưu điểm của loại mô hình này là giá thành rẻ, cho phép chúng ta nghiên cứu một số đặc tính chủ yếu của đối tượng thực.

Mô hình toán học thuộc loại mô hình trừu tượng. Các thuộc tính của đối tượng được phản ánh bằng các biểu thức, phương trình toán học. Mô hình toán học được chia thành mô hình giải tích và mô hình số. Mô hình giải tích được xây dựng bởi các biểu thức giải tích. Ưu điểm của loại mô hình này là cho kết quả rõ ràng, tổng quát. Nhược điểm của mô hình giải tích là thường phải chấp nhận một số giả thiết đơn giản hoá để có thể biểu diễn đối tượng thực bằng các biểu thức giải tích, vì vậy loại mô hình này chủ yếu được dùng cho các hệ tiền định và tuyến tính.

Mô hình số được xây dựng theo phương pháp số tức là bằng các chương trình chạy trên máy tính số. Ngày nay nhờ sự phát triển của kỹ thuật máy tính và kỹ thuật tin học, người ta đã xây dựng các mô hình số có thể mô phỏng được quá trình hoạt động của đối tượng thực. Những mô hình loại này được gọi là mô hình *mô phỏng (Simulation)*. Ưu điểm của loại mô hình mô phỏng là có thể mô tả các yếu tố ngẫu nhiên và tính phi tuyến của đối tượng thực do đó mô hình càng gần với đối tượng thực. Ngày này mô hình mô phỏng được ứng dụng rất rộng rãi.

Mô hình phải đạt được hai tính chất cơ bản sau đây:

• *Tính đồng nhất:* Mô hình phải đồng nhất với đối tượng thực mà nó phản ánh theo những tiêu chuẩn định trước.

• *Tính thực dụng:* Có khả năng sử dụng mô hình để nghiên cứu đối tượng.

Rõ ràng rằng, để tăng tính đồng nhất trong mô hình phải đưa vào nhiều yếu tố phản ánh đầy đủ các mặt của đối tượng. Nhưng như vậy nhiều khi mô hình trở nên quá phức tạp và cồng kềnh đến nỗi không thể dùng để tính toán được, có nghĩa là mất tính chất thực dụng của nó. Nếu quá chú trọng tính thực dụng, xây dựng mô hình đơn giản thì sai lệch giữa mô hình và đối tượng thực sẽ lớn, điều đó dẫn đến kết quả nghiên cứu không chính xác. Vì vậy tuỳ thuộc mục đích nghiên cứu mà người ta lựa chọn tính đồng nhất và tính thực dụng của mô hình một cách thích hợp.

2.4. Một số nguyên tắc khi xây dựng mô hình

Việc xây dựng mô hình toán học phụ thuộc vào đặc điểm của hệ thống thực, vì vậy khó có thể đưa ra những nguyên tắc chặt chẽ mà chỉ có thể đưa ra những nguyên tắc có tính định hướng cho việc xây dựng mô hình. Sau đây là một số nguyên tắc chính

a) Nguyên tắc xây dựng sơ đồ khối

Nhìn chung hệ thống thực là một hệ thống lớn phức tạp, vì vậy người ta tìm cách phân chúng ra thành nhiều hệ con, mỗi hệ con đảm nhiệm một số chức năng của hệ lớn. Như vậy mỗi hệ con được biểu diễn bằng một khối, tín hiệu ra của khối trước chính là tín hiệu vào của khối sau (xem hình 2 ở phần Mở đầu)

b) Nguyên tắc thích hợp

Tùy theo mục đích nghiên cứu mà người ta lựa chọn một cách thích hợp giữa tính đồng nhất và tính thực dụng của mô hình. Có thể bỏ bớt một số chi tiết không quan trọng để mô hình bớt phức tạp và việc giải các bài toán trên mô hình dễ dàng hơn.

c) Nguyên tắc về độ chính xác

Yêu cầu về độ chính xác phụ thuộc vào mục đích nghiên cứu. Ở giai đoạn thiết kế tổng thể độ chính xác không đòi hỏi cao, nhưng khi nghiên cứu thiết kế những bộ phận cụ thể thì độ chính xác của mô hình phải đạt được yêu cầu cần thiết.

d) Nguyên tắc tổ hợp

Tuỳ theo mục đích nghiên cứu mà người ta có thể phân chia hoặc tổ hợp các bộ phận của mô hình lại với nhau. Ví dụ khi mô hình hoá một phân xưởng

để nghiên cứu quá trình sản xuất sản phẩm thì ta coi các máy móc là thực thể của nó. Nhưng khi nghiên cứu quá trình điều khiển nhà máy thì ta tổ hợp phân xưởng như là một thực thể của toàn nhà máy.

Bài tập

2.1. Hãy nêu các khó khăn gặp phải khi tiến hành nghiên cứu trên các hệ thống thực sau đây: nghiên cứu quá trình lão hoá của vật liệu, nghiên cứu quá trình phát triển dân số của một quốc gia, nghiên cứu quá trình cháy trong lò hơi nhà máy nhiệt điện, nghiên cứu quá trình biến dạng của cột điện cao thế.

2.2. Hãy lấy ví dụ về mô hình vật lý thu nhỏ và mô hình vật lý tương tự.

2.3. Cho các hệ thống sau đây:

a) Siêu thị

b) Đường dây tải điện cao áp

c) Trạm lắp ráp thiết bị điện tử

d) Mô hình dòng sông

e) Cửa hàng ăn

Nếu muốn mô hình hoá các hệ thống trên thì nên dùng loại mô hình nào, mô hình giải tích hay mô phỏng?

Chương 3
PHƯƠNG PHÁP MÔ PHỎNG

3.1. Khái niệm chung về mô phỏng

Khi có một mô hình toán học của hệ thống thực người ta có thể tìm các thông tin về hệ thống bằng nhiều cách. Trong trường hợp mô hình tương đối đơn giản người ta có thể dùng phương pháp giải tích, ngược lại người ta phải dùng phương pháp số. Phương pháp giải tích cho ta lời giải tổng quát còn phương pháp số cho ta lời giải của từng bước tính với những điều kiện xác định, muốn lời giải đạt độ chính xác cao số bước tính phải được tăng lên đủ lớn. Đối với các hệ thống lớn, có cấu trúc phức tạp, có quan hệ tác động qua lại giữa các hệ con với trung tâm điều khiển, giữa hệ thống với môi trường bên ngoài, có các yếu tố ngẫu nhiên tác động v.v. thì phương pháp giải tích tỏ ra bất lực. Trong trường hợp này người ta phải dùng phương pháp mô phỏng *(Imitation, Simulation)*. Như vậy phương pháp mô phỏng đòi hỏi khối lượng tính toán rất lớn, điều này chỉ có thể giải quyết được khi ứng dụng các máy tính có tốc độ nhanh. Nhờ có sự phát triển của máy tính mà phương pháp mô phỏng ngày càng được hoàn thiện.

Mục 3.2 trình bày về bản chất của phương pháp mô phỏng. Mục 3.3 nói về các bước mô phỏng. Mục 3.4 trình bày về mô hình nguyên lý. Mục 3.5 nói về ưu nhược điểm của phương pháp mô phỏng và so sánh với phương pháp giải tích. Mục 3.6 trình bày về các ngôn ngữ và thiết bị mô phỏng. Cuối cùng mục 3.7 là các phương pháp mô phỏng và phạm vi ứng dụng.

3.2. Bản chất của phương pháp mô phỏng

Phương pháp mô phỏng có thể được định nghĩa như sau [3]:

"Mô phỏng là quá trình xây dựng mô hình toán học của hệ thống thực và sau đó tiến hành tính toán thực nghiệm trên mô hình để mô tả, giải thích và dự đoán hành vi của hệ thống thực "

Theo định nghĩa này có ba điểm cơ bản mà mô phỏng phải đạt được. Thứ

nhất là phải có mô hình toán học tốt tức mô hình có tính đồng nhất cao với hệ thực đồng thời mô hình được mô tả rõ ràng thuận tiện cho người sử dụng. Thứ hai là phải có khả năng làm thực nghiệm trên mô hình tức là có khả năng thực hiện các chương trình tính trên máy tính để xác định các thông tin về hệ thực. Cuối cùng là khả năng dự đoán hành vi của hệ thực tức có thể mô tả sự phát triển của hệ thực theo thời gian.

Phương pháp mô phỏng được đề xuất vào những năm 80 của thế kỷ 20, từ đó đến nay phương pháp mô phỏng đã được nghiên cứu, hoàn thiện và ứng dụng thành công vào nhiều lĩnh vực khác nhau như kỹ thuật, kinh tế, xã hội.v.v.

Bản chất của phương pháp mô phỏng là xây dựng một mô hình số (*Model Numerically)*, tức mô hình được thể hiện bằng các chương trình máy tính, sau đó tiến hành các "*thực nghiệm*" trên mô hình để tìm ra các đặc tính của hệ thống được mô phỏng. Số lần "thực nghiệm" về lý thuyết được tăng lên vô cùng lớn.

Quá trình mô hình hoá được tiến hành như sau. Gọi hệ thống được mô phỏng là *S*. Bước thứ nhất người ta mô hình hoá hệ thống *S* với các mối quan hệ nội tại của nó. Để thuận tiện trong việc mô hình hoá, người ta thường chia hệ *S* thành nhiều hệ con theo các tiêu chí nào đó $S = s_1, s_2, \ldots s_i, \ldots, s_n$. Tiếp đến người ta mô tả toán học các hệ con cùng các quan hệ giữa chúng. Thông thường giữa các hệ con có mối quan hệ trao đổi năng lượng và trao đổi thông tin. Bước thứ hai người ta mô hình hoá môi trường xung quanh *E*, nơi hệ thống *S* làm việc, với các mối quan hệ tác động qua lại giữa *S* và *E*. Khi đã có mô hình của *S* và *E*, người ta tiến hành các "*thực nghiệm*" trên mô hình, tức là cho *S* và *E* làm việc ở một điều kiện xác định nào đó. Kết quả người ta thu được một bộ thông số của hệ thống, hay thường gọi là xác định được một *điểm làm việc* của hệ thống. Các "thực nghiệm" đó được lặp lại nhiều lần và kết quả mô phỏng được đánh giá theo xác suất thống kê. Kết quả mô phỏng càng chính xác nếu số lần " thực nghiệm", còn gọi là " bước mô phỏng " càng lớn. Về lý thuyết " bước mô phỏng " phải tiến đến vô cùng lớn, nhưng trong thực tế số " bước mô phỏng " là hữu hạn nhưng phải đủ lớn và phụ thuộc vào yêu cầu của độ chính xác. Các bước nghiên cứu mô phỏng được trình bày ở mục 3.3 phản ánh bản chất của phương pháp mô phỏng.

Sau đây là một số lĩnh vực mà phương pháp mô phỏng được ứng dụng và phát huy ưu điểm của mình:

• Thiết kế và phân tích hệ thống điều khiển sản xuất, lập kế hoạch sản xuất;

• Đánh giá phần cứng, phần mềm của hệ thống máy tính;

• Quản lý và xác định chính sách dự trữ mua sắm vật tư của hệ thống kho vật tư, nguyên liệu;

• Đánh giá và phân tích hệ thống phòng thủ quân sự, xác định chiến lược phòng thủ tấn công;

• Thiết kế và phân tích hệ thống thông tin liên lạc, đánh giá khả năng làm việc của mạng thông tin;

• Thiết kế và phân tích các hệ thống giao thông như đường sắt, đường bộ, hàng không, cảng biển;

• Đánh giá phân tích thiết kế các cơ sở dịch vụ như bệnh viện, bưu điện, nhà hàng, siêu thị;

• Phân tích hệ thống tài chính, kinh tế.

Phương pháp mô phỏng được ứng dụng vào các giai đoạn khác nhau của việc nghiên cứu thiết kế và vận hành các hệ thống:

a) Phương pháp mô phỏng được ứng dụng vào giai đoạn nghiên cứu hệ thống trước khi tiến hành thiết kế nhằm xác định độ nhạy của hệ thống đối với sự thay đổi cấu trúc và tham số của hệ thống.

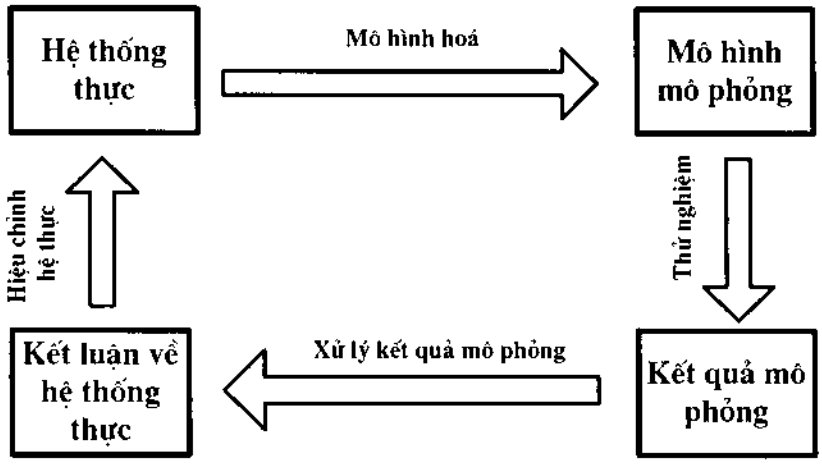
b) Phương pháp mô phỏng được ứng dụng vào giai đoạn thiết kế hệ thống để phân tích và tổng hợp các phương án hệ thống, chọn cấu trúc hệ thống thoả mãn các chỉ tiêu cho trước.

c) Phương pháp mô phỏng được ứng dụng vào giai đoạn vận hành hệ thống để đánh giá khả năng hoạt động, giải bài toán vận hành tối ưu, chẩn đoán các trạng thái đặc biệt của hệ thống.

Hình 3.1 trình bày quá trình nghiên cứu bằng phương pháp mô phỏng và quan hệ giữa hệ thống thực với các kết quả mô phỏng.

Từ hình 3.1 thấy rằng để nghiên cứu hệ thực chúng ta phải tiến hành mô hình hoá tức xây dựng mô hình mô phỏng. Khi có mô hình mô phỏng sẽ tiến hành làm các thực nghiệm để thu được kết quả mô phỏng. Thông thường kết quả mô phỏng có tính trừu tượng của toán học nên phải thông qua xử lý kết quả mô phỏng chúng ta mới thu được các thông tin, kết luận về hệ thực. Sau đó dùng các thông tin và kết luận trên để hiệu chỉnh hệ thực theo mục đích nghiên cứu

đề ra ban đầu. Quá trình nghiên cứu mô phỏng sẽ được trình bày cụ thể trong mục tiếp theo dưới đây.



Hình 3.1. Quá trình nghiên cứu bằng phương pháp mô phỏng.

3.3. Các bước nghiên cứu mô phỏng

Khi tiến hành nghiên cứu mô phỏng thông thường phải thực hiện qua 10 bước như được trình bày trên hình 3.2

Bước 1: Xây dựng mục tiêu mô phỏng và kế hoạch nghiên cứu

Điều quan trọng trước tiên là phải xác định rõ mục tiêu nghiên cứu mô phỏng. Mục tiêu đó được thể hiện bằng các chỉ tiêu đánh giá, bằng hệ thống câu hỏi cần được trả lời.

Bước 2: Thu thập dữ liệu và xác định mô hình nguyên lý

Tuỳ theo mục tiêu mô phỏng mà người ta thu thập các thông tin, các dữ liệu tương ứng của hệ thống *S* và môi trường *E*. Trên cơ sở đó xây dựng mô hình nguyên lý M_{nl} Mô hình nguyên lý là mô hình toán học phản ánh bản chất của hệ thống *S*. Các mô hình nguyên lý thường dùng được trình bày trong mục 3.4 dưới đây.

Bước 3: Hợp thức mô hình nguyên lý M_{nl}

Hợp thức mô hình nguyên lý là kiểm tra tính đúng đắn hợp lý của mô hình. Mô hình nguyên lý phải phản ánh đúng bản chất của hệ thống *S* và môi trường *E*

nhưng đồng thời phải tiện dụng không quá phức tạp cồng kềnh. Nếu mô hình nguyên lý M_{nl} không đạt yêu cầu phải thu thập thêm thông tin và dữ liệu để tiến hành xây dựng lại mô hình. Phần hợp thức mô hình nguyên lý sẽ được trình bày trong Chương 7.

Bước 4: Xây dựng mô hình mô phỏng M_M trên máy tính

Mô hình mô phỏng M_{mp} là những chương trình chạy trên máy tính, còn được gọi là *mô hình số* hay *mô hình mô phỏng*. Các chương trình này được viết bằng các ngôn ngữ thông dụng như FORTRAN, PASCAL, C^{++} hoặc các ngôn ngữ chuyên dụng để mô phỏng như GPSS, SIMSCRIPT, $SIMPLE^{++}$.v.v.

Bước 5: Chạy thử

Sau khi cài đặt chương trình, người ta tiến hành chạy thử xem mô hình mô phỏng có phản ánh đúng các đặc tính của hệ thống *S* và môi trường *E* hay không. Ở giai đoạn này cũng tiến hành sửa chữa các lỗi về lập trình.

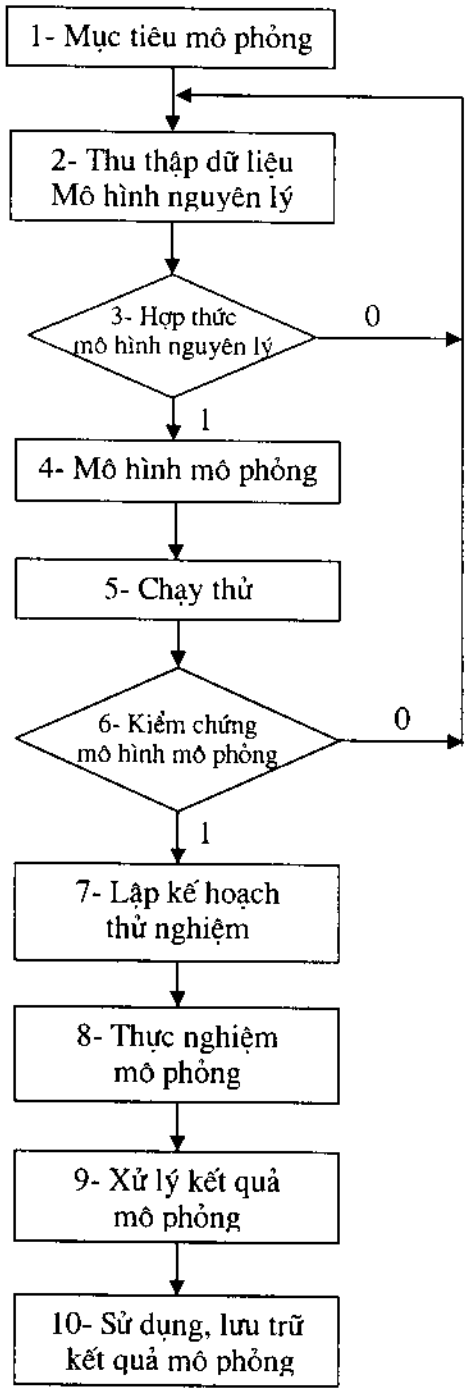
Bước 6: Kiểm chứng mô hình mô phỏng

Sau khi chạy thử người ta có thể kiểm chứng và đánh giá mô hình mô phỏng có đạt yêu cầu hay không, nếu không phải quay lại từ bước 2.

Kiểm chứng và hợp thức hoá mô hình là hai thủ tục quan trọng để xác nhận mô hình chúng ta xây dựng nên có thể dùng được hay không. *Kiểm chứng* là kiểm tra xem lập trình có đúng không, chương trình tính có thể chạy được không, dữ liệu vào ra có thuận lợi và chính xác hay không. *Hợp thức hoá mô hình* là đánh giá xem mô hình có phản ánh bản chất của hệ thực hay không, kết quả mô phỏng có đáp ứng được yêu cầu nghiên cứu hay không. *Kiểm chứng và hợp thức hoá mô hình* sẽ được trình bày trong Chương 7.

Bước 7: Lập kế hoạch thực nghiệm mô phỏng

Ở bước này người ta phải xác định một số điều kiện cho mô phỏng. Đầu tiên là xác định điều kiện đầu, điều kiện cuối hay còn gọi là chiều dài mô phỏng (*run*). Tiếp đến xác định số lần thử nghiệm hay còn gọi là số lần chạy mô phỏng độc lập (*replications*). Để cho các dữ liệu mô phỏng hoàn toàn độc lập với nhau, mỗi lần chạy mô phỏng người ta dùng một hạt giống ngẫu nhiên (*random seed*) khác nhau. Cuối cùng xác định thời gian mô phỏng của từng bộ phận hoặc toàn bộ mô hình. Căn cứ vào kết quả mô phỏng (ở bước 9) mà người ta hiệu chỉnh kế hoạch thực nghiệm để đạt được kết quả với độ chính xác theo yêu cầu. Các vấn đề về thực nghiệm mô phỏng sẽ được trình bày trong Chương 9.



Hình 3.2. Các bước nghiên cứu mô phỏng.

Bước 8: Thực nghiệm mô phỏng

Cho chương trình chạy thực nghiệm theo kế hoạch đã được lập ở bước 7. Đây là bước thực hiện việc mô phỏng. Các kết quả lấy ra từ bước này chính là dữ liệu đầu ra của mô phỏng.

Bước 9: Xử lý kết quả mô phỏng

Thực nghiệm mô phỏng thường cho nhiều dữ liệu có tính thống kê xác xuất. Vì vậy để có được kết quả cuối cùng với độ chính xác theo yêu cầu phải dùng phương pháp xác suất thống kê để xử lý các dữ liệu đầu ra. Các kết quả này phải được biểu diễn dưới dạng tường minh thuận lợi cho việc lưu trữ và sử dụng. Bước xử lý kết quả đóng vai trò quan trọng trong quá trình mô phỏng, vấn đề này sẽ được trình bày trong Chương 8.

Bước 10: Sử dụng và lưu trữ kết quả

Sử dụng kết quả mô phỏng vào mục đích đã định và lưu giữ dưới dạng các tài liệu để có thể sử dụng nhiều lần.

3.4. Mô hình nguyên lý

Trong các bước nghiên cứu mô phỏng ở mục 3.3 có đề cập đến mô hình nguyên lý và mô hình mô phỏng. Sau đây trình bày chi tiết hơn về mô hình nguyên lý.

Mô hình nguyên lý (hay còn gọi là mô hình khái niệm - Conceptual Model) là mô hình toán học phản ánh bản chất của hệ thực. Để xây dựng mô hình nguyên lý, người thực hiện mô hình hoá cần phải hiểu biết sâu sắc về hệ thực và có trình độ toán học cao để có thể dùng ngôn ngữ toán học mô tả hệ thực. Các kỹ sư công nghệ thường khó đạt được cả hai yêu cầu trên, nên trong thực tế người ta chọn những mô hình toán học (đã được các nhà toán học nghiên cứu và phát triển) phù hợp với hệ thực để làm mô hình nguyên lý. Sau đây liệt kê một số mô hình toán học thường được dùng để xây dựng mô hình nguyên lý:

- Mô hình bài toán tối ưu hoá.
- Mô hình bài toán quy hoạch tuyến tính
- Mô hình bài toán vận tải
- Mô hình trò chơi
- Mô hình sơ đồ mạng lưới (PERT)
- Mô hình độ tin cậy

- Mô hình quản lý dự trữ

- Mô hình hàng đợi (phục vụ đám đông) v.v...

Trong thực tế có nhiều loại mô hình toán học được ứng dụng vào các lĩnh vực cụ thể khác nhau. Sau đây chỉ trình bày tóm tắt một số mô hình nói trên. Chủ yếu là trình bày phương pháp luận và các ứng dụng của nó để bạn đọc có định hướng trong việc lựa chọn mô hình nguyên lý. Bạn đọc có thể tìm thấy các mô hình toán học được trình bày chi tiết trong các tài liệu tham khảo như [22, 23, 24], riêng mô hình hàng đợi sẽ được trình bày kỹ trong Chương 10 của cuốn sách này.

a). Mô hình bài toán tối ưu hoá [22, 23].

Bài toán tối ưu hoá chiếm một vị trí quan trọng trong việc giải bài toán điều khiển. Mục đích của tối ưu hoá là nhằm chọn phương án điều khiển thoả mãn một *tiêu chuẩn* nào đó. Việc lựa chọn tiêu chuẩn phụ thuộc vào mục tiêu chung được đặt ra đối với vấn đề điều khiển. Ví dụ, mục tiêu là sản lượng nhiều nhất, lợi nhuận cao nhất, giá thành sản phẩm thấp nhất v.v..Trong thực tế, khi giải bài toán điều khiển thường xuất hiện nhiều tiêu chuẩn, nhiều khi các tiêu chuẩn đó lại mâu thuẫn nhau. Ví dụ, khi tăng sản lượng có thể làm giảm chất lượng hoặc khi đạt được thời gian sản xuất ngắn thì chi phí sản xuất lại tăng lên.

Để giải bài toán tối ưu hoá thông thường có ba cách sau:

- Cách thứ nhất là chọn một tiêu chuẩn để tối ưu hoá, còn các tiêu chuẩn khác thì đặt ở một ngưỡng nào đó. Ví dụ có n tiêu chuẩn $A1, A2,... An$, đầu tiên ta tiến hành tối ưu hoá theo tiêu chuẩn A1 và cho trị số ngưỡng đối với các tiêu chuẩn khác, sau đó tối ưu hoá đối với $A2, A3$ v.v...

- Cách thứ hai là xây dựng một tiêu chuẩn hỗn hợp là một hàm đối với các tiêu chuẩn ban đầu, trong thực tế thường là một hàm tuyến tính

$$f(A1, A2, A3,..., An) = \alpha_1 A1 + \alpha_2 A2 + \alpha_3 A3 + ... + \alpha_n An.$$

Các hệ số trọng lượng $\alpha_1, \alpha_2,..., \alpha_n$ (chúng có thể là dương hay âm) được chọn tùy theo quan niệm của người đặt bài toán về tầm quan trọng của các tiêu chuẩn. Theo cách này người ta phải giải nhiều bài toán tối ưu hoá ứng với cách chọn các hệ số trọng lượng khác nhau.

- Cách thứ ba là biến đổi quy mô bài toán. Khi xét bài toán ở quy mô nhỏ có thể tính đến nhiều tiêu chuẩn cụ thể, nhưng khi xét ở quy mô lớn hơn người ta chỉ chọn một số tiêu chuẩn chung nhất để cho bài toán trở nên đơn giản hơn.

Giải bài toán tối ưu hoá tức là giải bài toán tìm cực trị của một hàm $f(x_1, x_2,..., x_n)$ nào đó trong miền S xác định của các tham số $x_1, x_2,..., x_n$. Để tìm cực trị ta phải tìm các điểm mà ở đó đạo hàm riêng triệt tiêu

$$\frac{\partial f}{\partial x_i} = 0, (i = 1, 2,..., n)$$

Điều kiện đạo hàm riêng triệt tiêu là điều kiện cần nhưng chưa đủ của điểm cực trị vì có thể chỉ là *điểm cực trị cục bộ*. Cần phải kiểm tra để tìm điểm lớn nhất hoặc nhỏ nhất đó là *điểm cực trị tuyệt đối*. Có nhiều phương pháp giải bài toán tìm cực trị, phổ biến nhất là phương pháp thừa số Lagrăng và phương pháp gradien.

Trong nhiều trường hợp điểm cực trị đạt được ngay trên biên giới của miền S, lúc này ta có bài toán cực trị có điều kiện: tìm điểm cực trị của hàm f với điều kiện thoả mãn hệ phương trình giới hạn sau

$$g_1(x_1, x_2,..., x_n) = 0,..., g_m(x_1, x_2,..., x_n) = 0.$$

Các bước giải bài toán như sau: thay hàm f bằng hàm $F = f + \lambda_1 g_1 + \lambda_2 g_2 + ... + \lambda_m g_m$, *trong đó* $\lambda_1, \lambda_2,..., \lambda_m$ là các hệ số Lagrăng tạm thời chưa biết. Tiếp đến giải hệ phương trình

$$\frac{\partial F}{\partial x_i} = 0, \ (i = 1, 2,..., n),$$

cùng với hệ phương trình giới hạn để tìm các nghiệm x và λ, sau đó tính giá trị của hàm f tại tất cả các điểm $x = (x_1, x_2,..., x_n)$ để tìm cực trị. Phương pháp này đòi hỏi phải giải nhiều hệ phương trình khá phức tạp.

Trong thực tế phổ biến nhất là dùng phương pháp gradien. Thủ tục tìm cực trị như sau: đầu tiên chọn một điểm A_0 bất kỳ trong miền S, sau đó tính gradien tại điểm A_0

$$\overline{g} = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \cdots, \frac{\partial f}{\partial x_n} \right).$$

Tiếp đó thực hiện một bước di chuyển bằng vectơ $\overline{c}_0 \overline{g}_0$ từ điểm A_0 đến điểm A_1, trong đó hằng số $c_0 > 0$ nếu muốn di chuyển đến điểm cực đại và $c_0 < 0$ nếu muốn di chuyển đến điểm cực tiểu. Tại A_1 lặp lại thủ tục trên cho đến khi tìm được điểm cực trị. Phương pháp gradien chỉ cho phép tìm được điểm cực trị cục bộ, sau đó phải tiến hành so sánh để tìm điểm cực trị tuyệt đối.

b). Mô hình bài toán quy hoạch tuyến tính [22, 24]

Trong thực tế thường gặp các trường hợp riêng của bài toán tối ưu hoá khi cần tìm cực trị của bài toán với hàm mục tiêu f và phương trình giới hạn của miền S là g đều là những hàm tuyến tính. Trong trường hợp này bài toán có tên là *bài toán quy hoạch tuyến tính.*

Ví dụ bài toán được đặt ra là tìm kế hoạch x_i để sản xuất sản phẩm thứ i sao cho đạt được lợi nhuận tối đa với điều kiện là tổng số các sản phẩm của từng loại không ít hơn một giá trị cho trước.

c). Mô hình bài toán vận tải

Mô hình bài toán vận tải là trường hợp riêng của bài toán quy hoạch tuyến tính. Quan trọng nhất là bài toán vận tải đảm bảo *tiêu chuẩn có giá thành nhỏ nhất.* Bài toán đặt ra như sau: có m địa điểm gửi hàng đi, trong đó tương ứng có $a_1, a_2,..., a_m$ đơn vị hàng hoá và n địa điểm nhận hàng có nhu cầu tương ứng là $b_1, b_2,..., b_n$ đơn vị hàng hoá. Ngoài ra còn biết giá thành c_{ij} của việc chuyên chở một đơn vị hàng từ mỗi địa điểm gửi hàng đến mỗi địa điểm nhận hàng. Yêu cầu xác định số lượng các đơn vị hàng hoá x_{ij} cần phải chuyên chở từ điểm gửi hàng thứ i đến điểm nhận hàng thứ j ($i = 1, 2,...m; j = 1, 2,...n$) sao cho tổng giá thành chuyên chở $f = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$ là cực tiểu và nhu cầu của tất cả các điểm nhận hàng đều được thoả mãn. Thông thường giả thiết rằng tổng số lượng hàng hoá có trong các địa điểm gửi hàng bằng tổng số nhu cầu của các điểm nhận hàng. Để giải bài toán, người ta xây dựng ma trận các giá thành chuyên chở $C = \|c_{ij}\|$, trong đó một phần tử c_{ij} của ma trận này gọi là được chọn nếu ta xác định được một kế hoạch chuyên chở x_{ij} từ điểm gửi hàng thứ i đến điểm nhận hàng thứ j. Ma trận kế hoạch $X = \|x_{ij}\|$ chứa các phương án chuyên chở $x_{ij} > 0$. Mục tiêu giải bài toán là tìm ma trận X sao cho hàm mục tiêu f đạt cực tiểu.

d). Mô hình trò chơi [22, 23]

Trong thực tế chúng ta thường phải lựa chọn các quyết định trong điều kiện thiếu thông tin hoặc trong tình thế những người tham gia lựa chọn quyết định có quyền lợi mâu thuẫn nhau. Ví dụ như giải bài toán giữa đầu tư và lợi nhuận, chọn phương án bố trí lực lượng phòng thủ v.v.. *Mô hình trò chơi* là công cụ hữu hiệu để giải các bài toán nêu trên trong những *tình thế xung khắc* và các bên tham gia theo đuổi những mục đích đối lập.

Kết cục của trò chơi được đánh giá định lượng, ví dụ được là +1, thua là -1 và hoà là 0. Trò chơi có thể theo cặp hay theo nhóm (nhiều bên tham gia), những người tham gia trò chơi được gọi là *đấu thủ*. Được nghiên cứu đầy đủ nhất là *trò chơi theo cặp có tổng số bằng không*, tức bên này được bao nhiêu thì bên kia thua bấy nhiêu. Trò chơi được phát triển do thực hiện liên tiếp các *nước đi* nào đó. *Nước đi cá nhân* là nước đi do đấu thủ phân tích tình thế mà đưa ra. *Nước đi ngẫu nhiên* là nước đi phụ thuộc vào yếu tố ngẫu nhiên nào đó. *Chiến lược* của đấu thủ là tập hợp các quy tắc dùng để phân tích tình thế và chọn các nước đi.

Xét trò chơi theo cặp, nếu biết cặp chiến lược (*A*, *B*) (của mình và của đối phương) thì hoàn toàn xác định được kết cục của trò chơi tức là xác định được phần được của bên này và phần thua của bên kia. Trò chơi được gọi là hữu hạn nếu mỗi đấu thủ chỉ có một số hữu hạn chiến lược. Kết quả của trò chơi theo cặp có tổng số bằng không có thể đưa vào một ma trận mà các dòng và các cột tương ứng với các chiến lược khác nhau, còn các phần tử của ma trận là phần được của một đấu thủ (tức phần thua của bên kia). Ma trận này được gọi là *ma trận trả tiền* hay là *ma trận trò chơi*. Xét trò chơi *m x n* với ma trận sau đây:

	B_1 B_1.. B_1
A_1	α_{11} α_{12}... α_{1n}
A_1	α_{21} α_{22}... α_{2n}
...	.....
A_1	α_{m1} α_{m2}... α_{mn}

Nếu đấu thủ thứ nhất áp dụng chiến lược A_i thì đấu thủ thứ hai sẽ cố gắng chọn một chiến lược tương ứng sao cho phần được của đấu thủ thứ nhất là cực tiểu.

Trị số cực tiểu này bằng $\min_j a_{ij}$, chúng ta ký hiệu nó là α_i. Theo quan điểm của đấu thủ thứ nhất (khi đối phương có bất kỳ đáp ứng nào) thì nên cố gắng tìm được chiến lược sao cho α_i có giá trị cực đại. Giá trị cực đại này được gọi là *giá trị thấp nhất của trò chơi* và chúng ta ký hiệu là α. Vì giá trị α được tính theo công thức sau:

$$\alpha = \max_i \min_j \alpha_{ij}$$

nên nó được gọi là giá trị *maximin*, và chiến lược tương ứng với nó là *chiến lược maximin*. Nếu đấu thủ thứ nhất duy trì chiến lược này khi đối phương áp dụng bất kỳ chiến lược nào thì phần được của đấu thủ thứ nhất được đảm bảo không nhỏ hơn α (tuỳ theo dấu của α, đây có thể là phần thua và trong trường hợp này là phần thua nhỏ nhất). Chiến lược maximin có thể hiểu là chiến lược đảm bảo lợi nhuận tối đa.

Tương tự như vậy có thể xác định phần thua cực tiểu (trong thực tế có thể là phần được) của đấu thủ thứ hai:

$$\beta = \min_j \max_i \alpha_{ij}$$

Trị số β được gọi là *giá trị cao của trò chơi* hay giá trị *minimax*. Tương ứng với nó là *chiến lược minimax* của đấu thủ thứ hai. Người ta chứng minh được rằng đối với trò chơi theo cặp có tổng số bằng không luôn tồn tại một cặp *chiến lược tối ưu*. Khi áp dụng chiến lược tối ưu, phần được (có thể là âm) của đấu thủ thứ nhất được gọi là *giá trị của trò chơi* và được ký hiệu là γ. Giá trị của trò chơi nằm giữa giá trị thấp và giá trị cao của trò chơi $\alpha \le \gamma \le \beta$. Các chiến lược được áp dụng hỗn hợp để có chiến lược tối ưu được gọi là *chiến lược hữu ích*. Giải một trò chơi có nghĩa là tìm ra cặp chiến lược tối ưu và giá trị trò chơi của nó. Mô phỏng mô hình trò chơi sẽ cho phép chúng ta tìm được chiến lược tối ưu trong sản xuất, kinh doanh hoặc nghiên cứu khoa học công nghệ.

e). Mô hình sơ đồ mạng lưới (PERT) [22, 23, 24]

Mô hình sơ đồ mạng lưới (PERT – Program Evaluation Review Technics) là một công cụ toán học dùng để biểu diễn, nghiên cứu và điều khiển các tổ hợp phức tạp của những công việc có liên quan tương hỗ với nhau. Một trong những mục tiêu chủ yếu của việc điều khiển và quản lý hệ thống là trong những điều kiện ràng buộc nhất định phải hoàn thành những khối lượng công việc cho trước với khoảng thời gian ngắn nhất. Trong các lĩnh vực kinh tế, công nghệ, thiết kế, nghiên cứu khoa học v.v. người ta thường phải lập kế hoạch thực hiện dự án từ khi xây dựng mục tiêu, thực hiện các giai đoạn trung gian cho đến khi kết thúc dự án. Bài toán quy hoạch và tối ưu hoá thời hạn thực hiện một dự án có thể được giải bằng phương pháp sơ đồ mạng lưới.

Sơ đồ mạng lưới thực chất là một *graph có hướng* gồm có một *tập đỉnh (nút)*

$X = x_1, x_2 ... x_n$ và một *tập cạnh có hướng (cung)* $A = a_1, a_2 ... a_m$. *Đỉnh* tương

ứng với các *sự kiện* và *cạnh* tương ứng với các *công việc*. *Sự kiện* là kết quả thực hiện một số công việc. Sự kiện chỉ xẩy ra khi tất cả các công việc liên quan đến sự kiện đó được hoàn thành. *Công việc* ở đây được hiểu theo nghĩa rộng: có thể là một nhiệm vụ cụ thể nào đó, đòi hỏi chi phí thực sự về thời gian và nguồn lực vật chất; cũng có thể không phải là nhiệm vụ cụ thể mà chỉ là mối liên hệ logic giữa các sự kiện, nó chỉ rõ khả năng bắt đầu một công việc nào đó chỉ có thể thực hiện được sau khi một số công việc khác đã được hoàn thành, trong trường hợp này người ta gọi là *công việc biểu kiến* và được biểu diễn bằng cạnh nét đứt trên sơ đồ mạng lưới; đó cũng có thể là thời gian chờ đợi do quy trình công nghệ yêu cầu nhưng không đòi hỏi tiêu phí nguồn lực vật chất.

Một đặc trưng quan trọng của sơ đồ mạng lưới là cho phép ta xác định *đường găng*. Trong sơ đồ mạng lưới, xét quá trình bắt đầu từ sự kiện *i* và kết thúc ở sự kiện *j*, thời hạn sớm nhất để có thể bắt đầu sự kiện *j* được gọi là *thời gian găng*. Đường nối các sự kiện tương ứng với thời gian găng được gọi là *đường găng*, trên sơ đồ mạng lưới đường găng được biểu diễn bằng mũi tên kép. Các sự kiện trên đường găng phải được hoàn thành đúng thời hạn đã định để đảm bảo dự án được thực hiện đúng hạn. Đường găng chỉ cho ta thấy cần phải tập trung nỗ lực thực hiện những công việc nào để hoàn thành dự án đúng tiến độ.

Khi phân tích sơ đồ mạng lưới theo chỉ tiêu thời gian ta thấy rằng muốn giảm thời gian hoàn thành dự án thì phải rút ngắn thời gian thực hiện các công việc trên đường găng. Những công việc khác không găng cho phép có thời gian dự trữ trong một giới hạn nào đó mà không ảnh hưởng tới thời hạn hoàn thành dự án.

Đối với một dự án, rõ ràng tồn tại các sơ đồ mạng lưới có mức chi phí khác nhau. Bài toán tối ưu hoá rút ngắn đường găng nhằm đạt được một trong hai mục đích sau đây: đảm bảo hoàn thành dự án không quá một thời hạn cho phép hoặc đảm bảo hoàn thành dự án với thời hạn tối ưu trên cơ sở một chi phí quy dẫn nào đó của dự án. Người ta thực hiện nhiều biện pháp để rút ngắn đường găng như: phân bổ lại lực lượng thi công và nguồn lực vật chất cho hợp lý, ưu tiên cho các công việc trên đường găng, ứng dụng các công nghệ tiền tiến v.v..

d). Mô hình độ tin cậy [22, 23]

Độ tin cậy là một đặc trưng quan trọng của hệ thống kỹ thuật. Vì vậy người ta thường dùng phương pháp mô hình hoá để nghiên cứu các biện pháp đảm bảo và nâng cao độ tin cậy của hệ thống kỹ thuật.

Độ tin cậy của hệ thống kỹ thuật là khả năng của hệ thống đảm bảo các chỉ tiêu kinh tế kỹ thuật trong điều kiện làm việc cho trước và thời gian vận hành cho trước. *Hỏng hóc* là tình trạng hệ thống không còn làm việc tin cậy được nữa, như vậy hỏng hóc đối lập với độ tin cậy.

Các đặc trưng cơ bản của độ tin cậy là:

- Xác suất làm việc tin cậy $P(t)$ hay thường được gọi là độ tin cậy. Xác suất làm việc tin cậy được tính như sau:

$$P(t) = P(T>t),$$

trong đó: t - thời gian xác định độ tin cậy của hệ thống;

T - thời gian làm việc của hệ thống kể từ lúc bắt đầu làm việc đến thời điểm xẩy ra hỏng hóc đầu tiên.

- Cường độ hỏng hóc $\lambda(t)$ là số lần xẩy ra hỏng hóc trên một đơn vị thời gian. Thông thường người ta xét hệ thống đang làm việc ở giai đoạn ổn định, lúc này có thể coi cường độ hỏng hóc là hằng số $\lambda = const$.

- Thời gian làm việc tin cậy T_{tb}, là thời gian trung bình của hệ thống làm việc tin cậy cho đến khi xẩy ra hỏng hóc đầu tiên.

Khi $\lambda = const$, ta có các quan hệ sau:

$$P(t) = e^{-\lambda t}\ ;\ T_{tb} = \frac{1}{\lambda}$$

Các hệ thống kỹ thuật được chia ra hệ thống có phục hồi và hệ thống không phục hồi.

Hệ thống có phục hồi là hệ thống khi xẩy ra hỏng hóc sẽ được sửa chữa, phục hồi lại chức năng ban đầu. Việc sửa chữa phục hồi có thể được thực hiện nhiều lần. Đặc trưng quan trọng của hệ thống có phục hồi là *hệ số sẵn sàng*.

Hệ thống không phục hồi là hệ thống chỉ làm việc từ khi bắt đầu cho đến khi xẩy ra hỏng hóc.

Dự phòng là biện pháp quan trọng để nâng cao độ tin cậy của hệ thống kỹ thuật. Có hai phương pháp dự phòng là *dự phòng nóng* và *dự phòng nguội*. Người ta dùng phương pháp mô hình hoá để đánh giá độ tin cậy của hệ thống kỹ thuật, tìm các biện pháp nâng cao độ tin cậy, xác định sơ đồ đấu dây của hệ thống cũng như các biện pháp dự phòng trong hệ thống.

e). Mô hình quản lý dự trữ [22, 24]

Chính sách quản lý dự trữ nguyên liệu, nhiên liệu, vật tư kỹ thuật, phụ tùng thay thế có ảnh hưởng rất lớn đến hiệu quả của các hệ thống sản xuất.

Người sản xuất mong muốn có dự trữ lớn để dễ dàng trong điều hành sản xuất. Ngược lại, người quản lý tài chính và kinh doanh lại mong muốn có dự trữ thấp để giảm giá thành, giảm chi phí bảo quản sản phẩm và tăng vốn lưu động. *Mô hình quản lý dự trữ* sẽ giúp chúng ta thoả mãn các yêu cầu trái ngược nêu trên và đưa ra chính sách dự trữ tối ưu. Chính sách quản lý dự trữ phải trả lời hai câu hỏi cơ bản sau đây để đảm bảo sản xuất liên tục và sử dụng đồng vốn tối ưu:

- Khi nào phải đặt hàng ?
- Lượng đặt hàng bao nhiêu ?

Có hai hệ thống dự trữ với chu kỳ đặt hàng và lượng đặt hàng khác nhau.

Hệ thống thứ nhất là hệ thống có số lượng đặt hàng cố định và chu kỳ đặt hàng thay đổi. Trong hệ thống này người ta sẽ đặt hàng khi lượng dự trữ giảm xuống đến mức báo động hay còn gọi là *điểm đặt hàng* còn lượng đặt hàng là một lượng cố định nào đó. Điểm đặt hàng thay đổi tuỳ theo mức tiêu thụ dự trữ.

Hệ thống thứ hai là hệ thống có chu kỳ đặt hàng cố định và số lượng đặt hàng thay đổi. Trong hệ thống này, cứ sau một chu kỳ cố định, hàng tháng hàng quý chẳng hạn, người ta đặt một lượng hàng sao cho lượng dự trữ đạt được mức cố định nào đó được gọi là *mức tái tạo dự trữ*. Như vậy lượng đặt hàng thay đổi tuỳ theo mức tiêu thụ dự trữ.

Người ta dùng phương pháp mô hình hoá để mô phỏng quá trình sản xuất, xác định mức tiêu thụ dự trữ và điểm đặt hàng cũng như số lượng đặt hàng tối ưu.

3.5. Ưu nhược điểm của phương pháp mô phỏng và so sánh với phương pháp giải tích

Như đã trình bày ở trên, phương pháp mô phỏng ngày càng được ứng dụng rộng rãi để nghiên cứu, phân tích và tổng hợp các hệ phức tạp.

Phương pháp mô phỏng có các ưu điểm sau đây:

• Có khả năng nghiên cứu các hệ thống phức tạp, có các yếu tố ngẫu nhiên, phi tuyến, đối với những hệ thống này phương pháp giải tích thường không có hiệu lực. Ưu điểm nổi bật của phương pháp mô phỏng là tính linh hoạt, người ta có thể đưa vào chương trình mô phỏng nhiều đặc tính của hệ thống mà phương pháp giải tích không thực hiện được.

• Có thể đánh giá các đặc tính của hệ thống làm việc trong các điều kiện dự kiến trước, hoặc ngay cả khi hệ thống còn đang trong giai đoạn thiết kế khảo sát, hệ thống chưa tồn tại.

• Có thể so sánh, đánh giá các phương án khác nhau của hệ thống.

• Có thể nghiên cứu các giải pháp điều khiển hệ thống.

• Có thể nghiên cứu trong một khoảng thời gian ngắn đối với hệ thống có thời gian hoạt động dài như hệ thống kinh tế, hệ thống xã hội.

Các nhược điểm của phương pháp mô phỏng là:

• Phương pháp mô phỏng đòi hỏi công cụ mô phỏng đắt tiền như máy tính, phần mềm chuyên dụng.

• Phương pháp mô phỏng thường sản sinh ra khối lượng lớn các dữ liệu có tính thống kê xác suất, do đó đòi hỏi phải có những chuyên gia thành thạo về phân tích dữ liệu để xử lý kết quả mô phỏng.

Khi quyết định dùng phương pháp mô phỏng để nghiên cứu hệ thống phải phân tích kỹ ưu nhược điểm và điều kiện cần thiết để thực hiện phương pháp này, đồng thời so sánh với phương pháp giải tích nếu có thể được.

Khi cho một mô hình toán học, có thể dùng phương pháp giải tích hoặc mô phỏng để thu được lời giải (thông tin) về mô hình. Tuy nhiên mỗi một phương pháp có ưu nhược điểm riêng của mình.

Hình 3.3 trình bày các điểm khác biệt cơ bản giữa mô hình giải tích và mô hình mô phỏng.

• Phương pháp giải tích cho một lời giải tổng quát và tường minh với giả thiết là các thông số của mô hình không thay đổi trong suốt quá trình được khảo sát.

• Phương pháp mô phỏng (tức phương pháp số), chỉ cho lời giải của từng bước tính, mỗi bước ứng với một điều kiện nhất định của mô hình, muốn có kết quả chính xác phải tăng số bước tính lên đủ lớn (theo lý thuyết là vô cùng lớn) và lời giải nhận được cũng chỉ ở dạng các "đánh giá" theo xác suất.

• Đối với phương pháp giải tích thường phải chấp nhận nhiều giả thiết để có thể biểu diễn các hệ thống và sự kiện bằng các biểu thức giải tích như bỏ qua hiện tượng phi tuyến, bão hoà v.v... Trong lúc đó phương pháp mô phỏng cho phép tính đến nhiều yếu tố như phi tuyến, ngẫu nhiên do đó mô tả hệ thống gần với hệ thực hơn.

Hệ thống thực

↓

Mô hình

Mô hình giải tích	*Mô hình mô phỏng*
• Lời giải tổng quát và tường minh. • Mô hình giải được khi phải chấp nhận các giả thiết để đơn giản hoá mô hình. • Lời giải "chính xác" chỉ khi được biểu diễn bằng các công thức có thể tính được. • Phạm vi ứng dụng: hệ tuyến tính, tiền định.	• Lời giải số từng bước là các "đánh giá". • Mô hình giải được không cần nhiều các giả thiết để đơn giản hoá mô hình. • Lời giải mang tính "đánh giá" sẽ cho đánh giá chính xác nếu bước tính tăng lên đủ lớn. Phạm vi ứng dụng: hệ tuyến tính, phi tuyến, ngẫu nhiên.

Hình 3.3. So sánh giữa mô hình giải tích và mô hình mô phỏng.

Cần phải nhấn mạnh rằng nếu trong mô hình có các yếu tố ngẫu nhiên thì phương pháp giải tích không thể giải được. Trong trường hợp này phương pháp mô phỏng là giải pháp duy nhất để nghiên cứu loại mô hình ngẫu nhiên.

3.6. Các ngôn ngữ và thiết bị mô phỏng

Khi tiến hành mô phỏng chúng ta phải xây dựng mô hình mô phỏng M_{mp} trên máy tính. Mô hình M_{mp} là một tập hợp các chương trình chạy trên máy tính được gọi là phần mềm mô phỏng, những chương trình này thường được viết bằng ngôn ngữ cấp cao thông dụng như PASCAL, C^{++}, VISUAL BASIC...

Tuy nhiên đối với các hệ thống phức tạp viết các chương trình mô phỏng như vậy gặp rất nhiều khó khăn và mất nhiều thời gian.

Trong thực tế người ta đã phát triển nhiều phần mềm mô phỏng chuyên dụng được gọi là ngôn ngữ mô phỏng *(Simulation Language)* và thiết bị mô

phỏng *(Simulator)*.

Ngôn ngữ mô phỏng bao gồm nhiều khối chuẩn, người sử dụng chỉ cần nạp các thông số cần thiết, nối các khối theo một logic định trước, cho mô hình chạy trong thời gian mô phỏng và nhận được các kết quả dưới dạng bảng số hoặc đồ thị.

Sử dụng các ngôn ngữ mô phỏng có rất nhiều ưu điểm như:

- Thời gian xây dựng mô hình ngắn.
- Dễ dàng thay đổi cấu trúc và thông số của mô hình.
- Dễ gỡ rối, sửa chữa sai sót.
- Các kết quả được xử lý tốt, thuận tiện cho việc sử dụng.

Sau đây sẽ điểm qua các ngôn ngữ mô phỏng chính hiện được sử dụng nhiều.

a) GPSS - General Purpose Simulation System

Ngôn ngữ GPSS do IBM sản xuất năm 1972. Sau đó được cải tiến nhiều lần, GPSS / H năm 1977, GPSS / PC năm 1984, GPSS / PC có thể chạy trên máy tính PC. GPSS có trên 60 khối chuẩn. Đây là ngôn ngữ hướng quá trình (*Process oriented language)*, có các khối để biểu diễn quá trình, các hình ảnh mô phỏng chuyển động theo quá trình mô phỏng (*Concurrent graphics animation*) rất thuận tiện cho việc theo dõi quá trình mô phỏng.

b) SIMSCRIPT

SIMSCRIPT được sản xuất năm 1962 sau đó được cải tiến nhiều lần với nhiều phiên bản (Version) khác nhau như SIMSCRIPT 1.5, SIMSCRIPT II.5. Đây là ngôn ngữ hướng quá trình và sự kiện (*Process and event -oriented)*.

c) SIMPLE ++ - Simulation Production Logistics Engineering Design.

SIMPLE ++ là ngôn ngữ hướng hướng đối tượng, hiện nay ngôn ngữ này được dùng rất phổ biến vì có những đặc điểm sau:

- Cấu trúc hướng đối tượng (*Object -Oriented*)
- Hình ảnh mô phỏng chuyển động *(Animation*).
- Kết quả được biểu diễn bằng bảng số, đồ thị dễ dàng so sánh.
- Dễ dàng mô phỏng các hệ thống kỹ thuật và thương mại phức tạp.
- Có thể nối với các phần mềm chuyên dụng khác như MRP (*Manufacturing Resource Planning)*.

- Người sử dụng có thể định nghĩa các đối tượng mới và dễ dàng lập trình mô phỏng.

Ngoài ra còn nhiều ngôn ngữ mô phỏng khác như SIGMA, SLAM (*Simulation Language for Alternative Modelling*), MODSIM, AUTOMOD v.v...

Thiết bị mô phỏng (Simulator) là một phần mềm chuyên dụng mô phỏng một hệ thống cụ thể. Thiết bị mô phỏng có rất ít hoặc không đòi hỏi phải lập trình như ngôn ngữ mô phỏng ở trên. Thuộc loại này có thiết bị mô phỏng dùng để huấn luyện lái máy bay, tàu thuỷ, ôtô v.v... Ngày nay những nhà máy lớn như nhà máy điện, ximăng, lọc dầu v.v... thường đặt thiết bị mô phỏng để huấn luyện cho người vận hành và giải bài toán tìm chế độ vận hành tối ưu. Những thiết bị mô phỏng loại này thường có giá thành tương đối đắt, phạm vi ứng dụng hạn chế vì chỉ dùng để mô phỏng một hệ thống cụ thể nhưng đưa lại hiệu quả to lớn trong huấn luyện cũng như vận hành hệ thống nên được dùng ở những nơi quan trọng. Một số loại thiết bị mô phỏng thường dùng hiện nay là SIMFACTORY, NETWORK v.v...

3.7. Các phương pháp mô phỏng và phạm vi ứng dụng

Tuỳ theo trạng thái của hệ thống thay đổi liên tục hay gián đoạn theo thời gian mà người ta phân biệt thành hệ thống liên tục hay gián đoạn. Đứng về mặt mô hình mà xét, người ta có thể chọn một trong hai mô hình liên tục hay gián đoạn để mô hình hệ thống. Vì vậy không nhất thiết phải có sự tương đương giữa loại hệ thống và loại mô hình. Việc phân biệt mô hình liên tục hay gián đoạn trở nên quan trọng khi tiến hành mô phỏng, đặc biệt là khi lập trình trên máy tính để thực hiện việc mô phỏng bởi vì kỹ thuật tính dùng cho các loại mô hình sẽ rất khác nhau. Chính vì vậy có hai phương pháp mô phỏng chủ yếu là *phương pháp mô phỏng liên tục và mô phỏng gián đoạn.*

Phương pháp mô phỏng liên tục (Continuous Simulation) thường được dùng cho hệ liên tục mà mô hình của nó là mô hình giải tích thường được biểu diễn bằng các hệ phương trình vi phân. Nếu phương trình vi phân tương đối đơn giản, nó có thể được giải bằng phương pháp giải tích và cho lời giải tổng quát là một hàm của giá trị của biến trạng thái tại thời điểm $t = 0$. Có nhiều trường hợp phương pháp giải tích không giải được. Trong trường hợp này, người ta phải dùng phương pháp số như phương pháp tích phân Runge-Kutta để giải phương trình vi phân và cho lời giải đặc biệt của biến trạng thái tại thời điểm $t = 0$.

Phương pháp mô phỏng liên tục sẽ được trình bày trong Chương 4 tiếp sau đây.

Phương pháp mô phỏng gián đoạn hay còn có tên là *Phương pháp mô phỏng các sự kiện gián đoạn (Discrete-Event Simulation)* thường được dùng cho hệ gián đoạn. Trong những hệ này sự kiện xẩy ra tại các thời điểm gián đoạn và làm thay đổi trạng thái của hệ thống. Phương pháp mô phỏng gián đoạn là nội dung chính của cuốn sách này, nó sẽ được trình bày kỹ trong các chương tiếp theo.

Ngoài hai phương pháp mô phỏng chính kể trên trong các tài liệu [1,2,3] còn giới thiệu các phương pháp mô phỏng khác như:

- Phương pháp mô phỏng hỗn hợp liên tục-gián đoạn (Combined Discrete - Continuous Simulation)

Có một số hệ thống không hoàn toàn gián đoạn cũng không hoàn toàn liên tục, đó là các hệ thống mà trong đó các trạng thái có thể thay đổi một cách liên tục hoặc gián đoạn. Ví dụ lò nung phôi thép, trong hệ thống này nhiệt độ lò nung thay đổi một cách liên tục nhưng số phôi thép đưa vào hoặc lấy ra khỏi lò nung thay đổi thay đổi một cách gián đoạn. Để mô phỏng hệ thống này ta phải dùng phương pháp mô phỏng hỗn hợp.

- Phương pháp mô phỏng Monte - Carlo (Monte - Carlo Simulation)

Phương pháp mô phỏng Monte - Carlo dựa trên số ngẫu nhiên $U(0,1)$ để giải các bài toán xác suất thống kê. Ví dụ dùng phương pháp này để xác định giá trị tới hạn trong phương pháp thử Kolmogorov - Smirnov để đánh giá dữ liệu đầu vào (xem mục 6.4).

Các phương pháp mô phỏng này được coi như là những trường hợp riêng của hai phương pháp mô phỏng chính nêu trên.

Bài tập

3.1. Hãy kể các lĩnh vực có thể dùng phương pháp mô phỏng để nghiên cứu và phân tích ưu nhược điểm khi dùng phương pháp này.

3.2. Hãy phân tích ưu nhược điểm và phạm vi ứng dụng của phương pháp giải tích và phương pháp mô phỏng.

3.3. Hãy kể ra một vài ngôn ngữ mô phỏng, phân tích ưu nhược điểm và phạm vi ứng dụng của chúng.

Chương 4
MÔ PHỎNG HỆ THỐNG LIÊN TỤC

4.1. Khái niệm chung về mô hình hệ thống liên tục

Hệ thống liên tục là hệ thống mà trong đó các trạng thái và thuộc tính của hệ thống thay đổi một cách liên tục. Mô hình toán học của hệ thống liên tục thường là hệ phương trình vi phân. Trường hợp đơn giản nhất đó là hệ phương trình vi phân tuyến tính với hệ số hằng và được giải một cách dễ dàng bằng phương pháp giải tích. Tuy nhiên, khi mô hình có phần tử phi tuyến như bão hoà, trễ, vùng chết v.v... thì phương pháp giải tích khó hoặc không thể giải được. Trong trường hợp này hợp lý nhất là dùng phương pháp mô phỏng để giải bài toán. Người ta có thể dùng máy tính tương tự [2, 10] hoặc máy tính số [1, 13] để mô phỏng hệ thống liên tục. Mục 4.2 và 4.3 trình bày cách dùng máy tính tương tự và số để mô phỏng hệ thống liên tục. Mục 4.4 nói về biến đổi Z, cơ sở để xây dựng hàm truyền số. Mục 4.5, 4.6 và 4.7 trình bày về hàm truyền số của hệ gián đoạn và liên tục cũng như trình tự tìm hàm truyền số. Mục 4.8 trình bày về cách chọn bước cắt mẫu, một thông số quan trọng trong phương pháp dùng máy tính để mô phỏng hệ liên tục. Mục 4.9 và 4.10 trình bày về toán tử tích phân số và cách dùng toán tử này để tìm phương trình sai phân của hệ điều khiển tự động. Cuối cùng mục 4.11 là ví dụ minh hoạ.

4.2. Dùng máy tính tương tự để mô phỏng hệ thống liên tục

Máy tính tương tự đã có quá trình phát triển lâu dài và đã góp phần giải các bài toán của hệ thống liên tục tuyến tính cũng như phi tuyến. Máy tính tương tự được dùng rộng rãi nhất là máy tính tương tự điện tử mà phần tử cơ bản của nó là các khuếch đại thuật toán. Điện áp của máy tính biểu thị biến số của mô hình toán học. Khuếch đại thuật toán có thể làm thành các bộ cộng, bộ tích phân và bộ đảo dấu điện áp do đó nó có thể giải các phương trình vi phân dùng để mô hình hoá hệ thống liên tục. Máy tính tương tự bị hạn chế là độ chính xác không cao do nhiều nguyên nhân: do độ chính xác của phép đo điện áp, do hiện tượng trôi điểm zero của khuếch đại thuật toán v.v... Nói chung độ chính xác của máy tính tương tự không vượt quá 0,1%. Một hạn chế quan trọng khác của máy tính

tương tự là đối với từng hệ thống cụ thể phải lắp mạch máy tính tương ứng, nếu hệ thống phức tạp thì thời gian lắp ráp và hiệu chỉnh máy tính sẽ đáng kể, hơn nữa máy tính không có khả năng phát triển mềm dẻo khi muốn thay đổi cấu trúc của hệ thống. Từ khi có máy tính số, máy tính tương tự ít được sử dụng vào mô phỏng. Tuy nhiên máy tính tương tự còn được sử dụng trong một số trường hợp như làm thiết bị mô phỏng của hệ thống sản xuất hoá chất, sinh học hoặc dùng trong mô phỏng hỗn hợp.

Xét một hệ thống liên tục được mô hình hoá bằng phương trình vi phân tuyến tính sau:

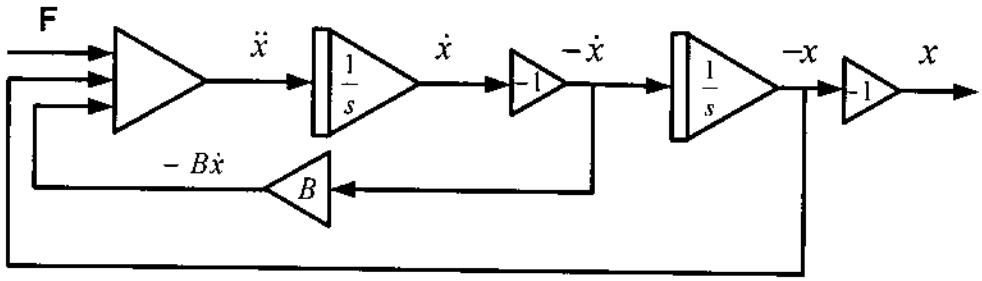
$$\frac{d^2x}{dt^2} + B\frac{dx}{dt} + x = F(t) \qquad (4-1)$$

Giả thiết rằng các điều kiện đầu bằng không và các hệ số trong phương trình vi phân đều là hằng số.

Với $B > 1$, có thể viết (4 - 1) thành phương trình sau:

$$\ddot{x} = -B\dot{x} - x + F(t) \qquad (4-2)$$

Dựa vào phương trình (4 - 2) ta có thể xây dựng được sơ đồ khối của máy tính tương tự như hình 4.1 để giải phương trình trên



Hình 4.1. Sơ đồ khối của máy tính tương tự để giải phương trình vi phân (4 - 2).

Để nhận được đáp ứng của hệ thống người ta phải đặt tín hiệu vào $F(t)$ vào bộ cộng. Qua hai khối tích phân ta nhận được tín hiệu ra x, tức lời giải của phương trình. Hiệu chỉnh hệ số B để được đặc tính ra mong muốn.

4.3. Dùng máy tính số để mô phỏng hệ thống liên tục

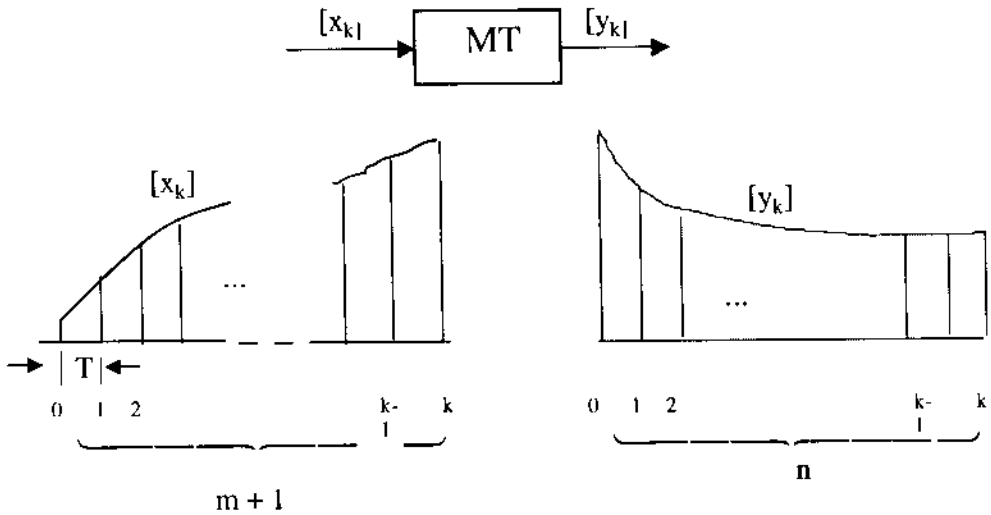
Từ khi máy tính số ra đời cho đến nay đã hơn nửa thế kỷ, máy tính số đã

phát triển rất nhanh và được ứng dụng vào hầu hết các lĩnh vực hoạt động của con người. Do ngày nay chủ yếu là dùng máy tính số nên từ đây về sau *máy tính số* được gọi tắt là *máy tính - MT (Computer)*. Trong lĩnh vực mô hình hoá, MT là công cụ chủ yếu để thực hiện việc mô phỏng hệ thống. Sau đây chúng ta sẽ nghiên cứu *MT* làm việc như thế nào trong việc mô hình hoá hệ thống.

4.3.1. Phương trình máy tính

Dùng *MT* để mô hình hoá hệ thống có nghĩa là đưa vào *MT* các dữ liệu ban đầu, *MT* xử lý các dữ liệu đó theo chức năng hoạt động của hệ thống *S*, đầu ra của *MT* cho ta các trạng thái của hệ thống *S* theo thời gian.

Tín hiệu vào $[X_K]$ và tín hiệu ra $[Y_K]$ của *MT* đều là những tín hiệu số (gián đoạn). Sau đây chúng ta xét quan hệ giữa chúng



Hình 4.2. Quan hệ giữa tín hiệu vào ra của máy tính.

Bước gián đoạn hoá *T* (bước cắt mẫu) là nhịp làm việc của *MT*.

Dãy tín hiệu vào $[x_k] = x(0), x(T), x(2T)... x(kT)$.

Dãy tín hiệu ra $[y_K] = y(0), y(T), y(2T)... y(kT)$.

Khi khảo sát ta chấp nhận giả thiết là thời gian tính của *MT* không đáng kể nên có thể bỏ qua, có nghĩa là dãy tín hiệu *ra* $[y_k]$ hoàn toàn đồng bộ với dãy tín hiệu vào $[x_k]$.

Tín hiệu ra ở thời điểm *k* tức *y(kT)* phụ thuộc vào giá trị của *n* tín hiệu ra và *m+1* tín hiệu vào xảy ra trước đó. Các giá trị của *m* tín hiệu vào và *n* tín hiệu ra được lưu trữ trong bộ nhớ của máy tính.

Như vậy quan hệ giữa tín hiệu ra và tín hiệu vào của *MT* được viết như sau [13]:

$$y(kT) = \sum_{i=o}^{m} b_{m-i} x(kT - iT) - \sum_{j=1}^{n} a_{n-j} y(kT - jT); \qquad (4 - 3)$$

trong đó: a_n, b_m - các hệ số;

$i = 0, m; j = 0, n;$ với $m \leq n$.

Phương trình (4 - 3) được gọi là *phương trình máy tính*, biểu thị quan hệ tuyến tính giữa tín hiệu ra với tín hiệu vào của MT.

Chú ý rằng trong phương trình (4 - 3) luôn luôn có quan hệ $m \leq n$ có nghĩa là tín hiệu ra phụ thuộc vào *m* tín hiệu vào trong quá khứ. Nếu $m > n$ thì tín hiệu ra phụ thuộc cả vào tín hiệu vào trong tương lai là điều không xảy ra trong thực tế được.

Vì tín hiệu ra $[y_k]$ và tín hiệu vào $[x_k]$ đều có cùng bước gián đoạn *T* nên để cho gọn phương trình (4 - 3) có thể viết lại:

$$y(k) = \sum_{i=o}^{m} b_{m-i} x(k - i) - \sum_{j=1}^{n} a_{n-i} y(k - j); \qquad (4 - 4)$$

Phương trình (4 - 4) có thể triển khai thành:

$$y(k) + a_{n-1}\, y(k-1) + a_{n-2}\, y(k-2) + \ldots + a_1\, y(k- n +1) + a_o y(k- n) =$$

$$= b_m x(k) + b_{m-1} x(k-1) + b_{m-2} x(k-2) + \ldots + b_o x(k - m) \qquad (4 - 5)$$

Phương trình (4 -5) có dạng phương trình sai phân bậc *n*.

Các hệ số $a_{n-1}, \ldots a_o; b_m \ldots b_o$ đặc trưng đặc tính động của hệ thống. Nếu các hệ số là hằng ta có phương trình sai phân tuyến tính phản ảnh hệ dừng (đặc tính không biến đổi theo thời gian), trong trường hợp ngược lại *a(t), b(t)* - hệ không dừng. Ở đây ta chỉ khảo sát các hệ tuyến tính dừng mà thôi.

Bậc của phương trình sai phân là sai biệt giữa bậc của số hạng tín hiệu ra lớn nhất và bé nhất. Trong trường hợp phương trình (4 - 5) bậc của phương trình là *(k) - (k-n) = n*.

Vậy ta có thể kết luận rằng phương trình máy tính có dạng của phương trình sai phân tuyến tính.

Từ phương trình (4 - 5) ta có thể viết:

$$y(k) = - a_{n-1}\, y(k-1) - a_{n-2}\, y(k-2) -...- a_0 y(0) + b_m x(k) + \\ + b_{m-1}\, x(k-1)...+ b_0\, x(0) \quad (4 - 6)$$

Như vậy nếu biết điều kiện đầu *x(0), y(0),* bằng cách tăng dần bước *k* ta có thể tính được *y(k)* ở các thời điểm khác nhau. Các kết quả tính toán được lưu trữ trong bộ nhớ và giá trị tín hiệu ra của bước tiếp theo phụ thuộc vào giá trị của tín hiệu vào và tín hiệu ra của các bước trong quá khứ.

4.3.2. Phương pháp mô phỏng hệ liên tục bằng máy tính số

Từ các phân tích ở trên ta thấy rằng muốn dùng máy tính số để mô phỏng hệ liên tục thì phải mô tả hệ liên tục dưới dạng phương trình sai phân tuyến tính, sau đó đưa phương trình sai phân tuyến tính đó vào máy tính để tìm các đặc tính của hệ liên tục.

Chú ý rằng hệ liên tục thường được biểu diễn bằng phương trình vi tích phân. Để biến đổi phương trình vi tích phân thành phương trình sai phân tương ứng có thể dùng phương pháp số Runge - Kutta. Tuy nhiên phương pháp này có khối lượng tính toán lớn, đặc biệt đối với phương trình có bậc từ 3 trở lên thì tính toán rất phức tạp nhiều khi không thực hiện được. Vì vậy ở phần tiếp theo sẽ trình bày phương pháp tiện dụng để tìm phương trình sai phân của hệ liên tục.

Từ phương trình Laplace *W(s)* của hệ liên tục, bằng cách đổi biến số *s* thành *z*, người ta có thể tìm được phương trình biến đổi *Z* tương ứng *W(z),* rồi tìm ngược lại phương trình sai phân của hệ để giải trên máy tính số.

4.4. Biến đổi Z và các tính chất

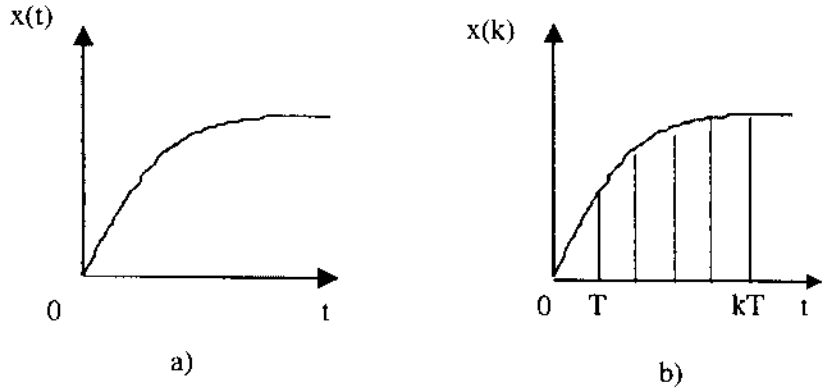
• Mục đích của phép biến đổi *Z*

Khi giải phương trình sai phân bậc cao người ta gặp nhiều khó khăn, vì vậy người ta thường dùng biến đổi *Z* để biến phương trình sai phân tuyến tính của hệ gián đoạn thành phương trình đại số. Điều này hoàn toàn tương tự như trong trường hợp hệ liên tục dùng biến đổi Laplace để biến phương trình vi tích phân

thành phương trình đại số.

• Một số định nghĩa trong phép biến đổi Z.

Giả thiết rằng không có tín hiệu ở phía âm của trục thời gian (hình 4.3).



Hình 4.3. Các dạng của tín hiệu :

a) Liên tục; b) Gián đoạn

Đối với tín hiệu liên tục *x(t)* ta có định nghĩa về biến đổi Laplace như sau:

$$L\,[x(t)] = X(s) = \int_0^{\infty} x(t)\, e^{-st}\, dt \qquad (4 - 7)$$

Đối với tín hiệu gián đoạn *x[k]* ta có định nghĩa về biến đổi Z như sau:

$$Z\,[x\,(k)] = X(Z) = x(0)\, Z^{o} + x(1)\, Z^{-1} + x(k)\, Z^{-k} + \ldots$$

$$= \sum_{k=0}^{\infty} x(k) Z^{-k} \qquad (4 - 8)$$

trong đó: Z - là biến phức.

Nếu hàm *x(t)* tồn tại biến đổi Laplace có dạng:

$$X(s) = \frac{B(s)}{A(s)}$$

thì chuỗi (4-8) là biến đổi Z của hàm gián đoạn *x[k]* tương ứng.

Bảng 4-1 liệt kê một số hàm thời gian thông dụng và các biến đổi Laplace *F(s)*, biến đổi Z *F(z)* tương ứng.

Bảng 4-1

Hàm thời gian	*F(s)*	*F(z)*
$\delta(k)$	1	1
1	$\frac{1}{s}$	$\frac{Z}{Z-1}$
t	$\frac{1}{s^2}$	$\frac{TZ}{(Z-1)^2}$
t^2	$\frac{2!}{s^3}$	$\frac{T^2Z(Z+1)}{(Z-1)^3}$
t^3	$\frac{3!}{s^4}$	$\frac{T^3Z(Z^2+4Z+1)}{(Z-1)^4}$
e^{-at}	$\frac{1}{s+a}$	$\frac{Z}{Z-e^{-aT}}$
te^{-at}	$\frac{1}{(s+a)^2}$	$\frac{TZe^{-aT}}{(Z-e^{-aT})^2}$
$\sin(\omega t)$	$\frac{\omega}{s^2+\omega^2}$	$\frac{Z\sin(\varpi T)}{Z^2-2Z\cos(\varpi T)+1}$
$\cos(\omega t)$	$\frac{s}{s^2+\omega^2}$	$\frac{Z^2-Z\cos(\varpi T)}{Z^2-2Z\cos(\varpi T)+1}$

• *Các tính chất của biến đổi Z*

1. Tính chất tuyến tính

$$Z\{a\,[v(k)] + b[w(k)]\} = aV(z) + bW(z) \qquad (4 - 9)$$

2. Dịch hàm gốc f(k) về phía trước m bước

Theo (4 - 8) ta có biến đổi Z của *f(k) là:*

$$Z[f(k)] = F(z) = \sum_{k=0}^{\infty} f(k)Z^{-k}.$$

Có thể chứng minh được:

$$Z[f(k+1)] = ZF(z) - Zf(0):$$

$$Z[f(k+2)] = Z^2 F(z) - Z^2 f(0) - Zf(1)$$

Tổng quát dịch m bước

$$Z[f(k+m)] = Z^m F(Z) - \sum_{j=0}^{m-j} f(k) Z^{m-j} \qquad (4 - 10)$$

Với điều kiện đầu bằng không ta có

$$Z[f(k+1)] = ZF(Z)$$

$$Z[f(k+m)] = Z^m F(Z) \qquad (4 - 11)$$

3. Dịch hàm gốc f(k) về phía sau m bước

Tương tự như công thức (4 - 11) ta có

$$Z[f(k-m)] = Z^{-m} F(Z) \qquad (4 - 12)$$

4. *Biến đổi Z của sai phân tiến*

$$\Delta f(k) = f(k+1) - f(k)$$

$$Z[\Delta f(k)] = (Z-1)F(Z) \qquad (4 - 13)$$

5. *Biến đổi Z của sai phân lùi*

$$\Delta f(k) = f(k) - f(k-1)$$

$$Z[\Delta f(k)] = (1 - Z^{-1})F(Z) \qquad (4 - 14)$$

6. *Giá trị đầu của hàm gốc rời rạc f(0)*

$$f(0) = \lim_{z \to \infty} F(Z) \qquad (4 - 15)$$

7. Giá trị cuối của hàm gốc rời rạc f(0)

$$f(\infty) = \lim_{z \to 1} (Z-1)F(Z) \qquad (4 - 16)$$

4.5. Hàm truyền số của hệ gián đoạn

Hàm truyền số của hệ gián đoạn tuyến tính là tỷ số giữa biến đổi Z của dãy tín hiệu ra gián đoạn với biến đổi Z của dãy tín hiệu gián đoạn vào với điều kiện đầu bằng không.

Giả sử một hệ gián đoạn được mô tả bằng phương trình sai phân tuyến tính sau:

$$a_n y(k+n) + a_{n-1} y(k+n-1) + \ldots + a_o y(k) = b_m x(k+m) + \ldots + b_o x(k) \qquad (4\text{-}17)$$

Trong đó $m \leq n$, điều kiện này đảm bảo khả năng giải phương trình (4-17) trên máy tính.

Thực hiện biến đổi Z các phần tử của phương trình (4-17) với điều kiện đầu bằng *0* và tính chất dịch hàm gốc đi *n* và *m* bước như đã nêu ở phương trình (4-11) ta có hàm truyền số sau:

$$W(Z) = \frac{Y(Z)}{X(Z)} = \frac{b_m Z^m + b_{m-1} Z^{m-1} + ... + b_o}{a_n Z^n + a_{n-1} Z^{n-1} + ... + a_o} \qquad (4 - 18)$$

Nếu tín hiệu vào *x(k)* là xung đơn vị -> hàm Dirac δ thì ta có:

$$X(Z) = Z[\delta(k)] = 1$$

Lúc này $W(Z) = Y(Z)$ (4 - 19)

Như vậy cũng giống như trong trường hợp biến đổi Laplace, hàm truyền *W(s)* của hệ liên tục là phản ứng của hệ đối với hàm đơn vị *1(t)*, hàm truyền số *W(z)* là phản ứng của hệ thống gián đoạn đối với tín hiệu vào là xung Dirac *δ(k)*.

4.6. Hàm truyền số của hệ liên tục

Đối với hệ liên tục người ta dùng biến đổi Laplace gián đoạn để tìm hàm truyền số của hệ liên tục, nhưng phép biến đổi này thường dẫn đến hàm siêu việt đối với biến *s*, do đó rất khó tính toán nên không được dùng trong thực tế. Trong thực tế người ta dùng phương pháp chuyển đổi từ hàm truyền Laplace *W(s)* sang hàm truyền số biến đổi *Z* là *W(Z)* bằng cách thay biến số.

$$Z = e^{st} \qquad (4 - 20)$$

Từ biểu thức (4 - 20) ta có thể giải được

$$s = \frac{1}{T} \ln Z \qquad (4 - 21)$$

trong đó *lnZ* có thể triển khai thành chuỗi

$$\ln Z = 2\left(U + \frac{U^3}{3} + \frac{U^5}{5} + ...\right) \qquad (4 - 22)$$

ở đây:

$$U = \frac{1 - Z^{-1}}{1 + Z^{-1}} = \frac{Z - 1}{Z + 1}$$

Bỏ qua các số hạng bậc cao trong (4 - 22) ta có:

$$\ln Z = 2U = 2\,\frac{Z - 1}{Z + 1}$$

Vậy phép biến đổi số tương đương (4 - 21) có thể viết thành:

$$s = \frac{2}{T}\frac{Z-1}{Z+1} \qquad (4 - 23)$$

trong đó: T - thời gian cắt mẫu.

Phương pháp chọn thời gian cắt mẫu T sẽ được trình bày ở mục 4.8 dưới đây.

4.7. Trình tự tìm hàm truyền số

Trình tự tìm hàm truyền số như sau:

1. Từ hàm $W(s)$ ta phân tích thành các biểu thức đơn giản $W_1(s)$, $W_2(s)$ v.v..

2. Tìm biến đổi Z tương ứng của các biểu thức đơn giản kể trên bằng cách đổi biến số theo (4- 23) ta được các hàm tương ứng $W_1(Z)$, $W_2(Z)$ v.v...

Rút gọn lại ta được hàm truyền số của hệ liên tục tuyến tính.

Khi sử dụng phương pháp này người ta phải công nhận những điều kiện sau đây (có thể chứng minh được).

• Nếu hệ liên tục có hàm truyền đạt $W(s)$ là ổn định thì hàm truyền đạt số tương đương $W(Z)$ cũng sẽ ổn định.

• Nếu hàm $W(s)$ có thể phân tích thành $W(s) = W_1(s).\ W_2(s)$ v.v... thì khi chuyển sang hàm truyền số vẫn giữ tính nhân như trước, có nghĩa là ta có thể viết:

$$W(Z) = W_1(Z).\ W_2(Z)...$$

trong đó: $W_1(Z)$ chuyển từ $W_1(s)$ bằng cách đổi biến số theo (4 - 23).

• Khi chuyển từ $W(s)$ sang $W(Z)$ các hằng số và hệ số khuếch đại vẫn giữ nguyên.

Ví dụ:

Cho hàm $W(s) = \dfrac{K}{s + as + K}$

Hãy tìm hàm truyền Z tương ứng $W(Z)$

Giải:

Thay $s = \dfrac{2}{T}\dfrac{Z-1}{Z+1}$ vào W(s) ta có:

$$W(Z) = \frac{K}{\left(\frac{2}{T}\frac{Z-1}{Z+1}\right)^2 + a\frac{2}{T}\left(\frac{Z-1}{Z+1}\right) + K}$$

$$W(Z) = \frac{KT^2Z^2 + 2KT^2Z + KT^2}{(4 - 2aT + kT^2)Z^2 + (2KT^2 - 8)Z + (KT^2 - 4 - 2aT)}$$

Đặt

$A = kT^2$

$B = 4 - 2aT + kT^2$

$C = 2kT^2 - 8$

$D = kT^2 - 4 - 2aT$

Trong đó T là bước cắt mẫu

Ta có:

$$W(Z) = \frac{AZ^2 + 2AZ + A}{BZ^2 + CZ + D}$$

4.8. Cách chọn bước cắt mẫu T

Theo lý thuyết lấy mẫu của Shannon, để đảm bảo khả năng khôi phục lại tín hiệu liên tục từ dãy tín hiệu gián đoạn thì tần số lấy mẫu thấp nhất f_m phải lớn hơn hoặc bằng $2f_{max}$, trong đó f_{max} là tần số tín hiệu vào cao nhất, có nghĩa là $f_m \geq 2f_{max}$, thoả mãn điều kiện này thì có thể khôi phục lại tín hiệu mà không bị méo dạng. Từ đó suy ra bước cắt mẫu phải thoả mãn biểu thức sau $T = \frac{1}{f_m} \leq \frac{1}{2f_{\max}}$.

Trong thực tế người ta thường chọn tần số lấy mẫu f_m lớn hơn nhiều so với tần số tín hiệu vào cao nhất f_{max}. Tần số lấy mẫu f_m có thể chọn theo quy tắc sau đây [11]:

$$f_m = (6 \div 25) f_{BT}, Hz. \qquad (4 - 24)$$

trong đó: f_{BT} - là băng thông của hệ kín.

Để thuận tiện trong tính toán, người ta chọn giá trị bước cắt mẫu T ($T = 1/f_m$) phụ thuộc vào các hằng số thời gian trong hàm truyền của hệ kín [11].

Sau đây ta xét một số trường hợp cụ thể.

+ Đối với hệ quán tính bậc nhất $W(s) = \dfrac{1}{1+T_0 s}$

ta có băng thông của hệ thống

$$f_{BT} = f_0 = \frac{1}{2\pi T_0}$$

Theo quy tắc (4-24), bước cắt mẫu T có quan hệ sau [11]:

$$\frac{T_0}{4} < T < T_0$$

trong đó : T_0 là hằng số thời gian của hệ.

Thời gian để đường đặc tính quá độ đạt giá trị lớn nhất được gọi là T_{max}, ta có quan hệ giữa bước cắt mẫu T và T_{max} như sau

$T_{max} = (2 \div 9)\ T$

+ Đối với hệ bậc 2 $W(s) = \dfrac{\omega_0^2}{\omega_0^2 + 2\zeta\omega s + s^2}$

Bước cắt mẫu được tính như sau:

$0{,}25 < \omega_0 T < 1{,}5;\ 0{,}7 \le \zeta \le 1$

trong đó: ζ - là hệ số tắt dần; ω_0 - là tần số dao động tự nhiên

Bảng 4-2 dưới đây cho ta những chỉ dẫn tham khảo để chọn bước cắt mẫu đối với các biến hoặc quá trình khác nhau [11]:

Bảng 4-2

Loại biến hoặc quá trình	***Bước cắt mẫu (s)***
Lưu lượng	1 - 3
Mức	5 - 10
Áp suất	1 - 5
Nhiệt độ	10 - 45
Chưng cất	10 - 180
Sấy	20 - 45
Hệ thống điều khiển	0,001 - 0,1

Đối với hệ thống điều khiển, theo kinh nghiệm nên chọn bước cắt mẫu như sau:

$\frac{T_{min}}{4} \le T \le T_{min}$, trong đó T_{min} - là hằng số thời gian nhỏ nhất trong hàm truyền của hệ thống. Chọn bước cắt mẫu theo quy tắc này cũng tương đương như chỉ dẫn đối với mục Hệ thống điều khiển trong Bảng 4 -2.

4.9. Dùng phương pháp toán tử để tìm phương trình sai phân của hệ điều khiển tự động

Mục 4.3 ở trên đã chứng minh rằng phương trình máy tính có dạng phương trình sai phân tuyến tính. Vì vậy để mô phỏng hệ điều khiển tự động (ĐKTĐ) trên máy tính người ta phải tìm cách viết được phương trình sai phân của hệ. Có hai cách tìm phương trình sai phân:

- Từ phương trình vi tích phân của hệ người ta viết thành phương trình sai phân tương ứng - Đó là phương pháp Runge-Kutta. Phương pháp này chính xác nhưng phức tạp.
- Dùng phương pháp toán tử:

Chúng ta có sơ đồ quan hệ giữa *W(s)* và *W(z)* như sau:

(Hệ liên tục)

↓

*Phương trình vi tích phân → biến đổi Laplace → Phương trình đại số **W(p)***

Đổi biến số $s = \frac{2}{T}\frac{Z-1}{Z+1}$ ↓

(Hệ gián đoạn)

↓

*Phương trình sai phân ↔ biến đổi Z ↔ Phương trình đại số **W(z)***

Như vậy theo sơ đồ trên ta có quá trình tìm phương trình sai phân của một hệ ĐKTĐ có hàm truyền *W(s)* như sau:

Từ hàm truyền Laplace *W(s)* đổi biến số $s = \frac{2}{T}\frac{Z-1}{Z+1}$ được hàm truyền *W(z)*.

Từ hàm *W(z)* tìm ngược lại được phương trình sai phân *y(k)* để viết chương

trình mô phỏng hệ ĐKTĐ trên máy tính.

Vì hàm truyền *W(s)* của hệ ĐKTĐ tương đối dễ dàng tìm được nên dùng phương pháp này để tìm phương trình sai phân rất thuận tiện. Phương pháp này gọi là phương pháp toán tử.

4.10. Khái niệm về toán tử tích phân số

Phương pháp toán tử được dùng rất rộng rãi để tìm phương trình sai phân của hệ ĐKTĐ. Sau đây sẽ trình bày về khái niệm toán tử tích phân số.

Tích phân một quá trình liên tục được biểu diễn như sau:

$$y(t) = \int_0^t u(t)dt\,, \qquad (4 - 25)$$

trong đó: $u(t) = \dfrac{dy(t)}{dt}$.

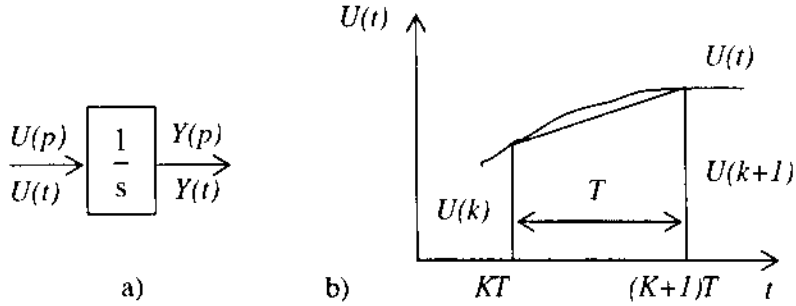
Biến đổi Laplace hai vế của biểu thức (4. 25) với điều kiện đầu bằng không.

$$Y(s) = \frac{1}{s}U(s) \qquad (4 - 26)$$

Vậy ta có hàm truyền của tích phân lý tưởng còn gọi là toán tử tích phân:

$$W(s) = \frac{Y(s)}{U(s)} = \frac{1}{s} \qquad (4 - 27)$$

Sơ đồ cấu trúc của toán tử tích phân như trên hình 4.4 a.



Hình 4.4. Cách tính toán tử Tustin.

Vấn đề đặt ra là hãy tìm toán tử số *I(z)* để khi chuyển từ hệ liên tục sang hệ gián đoạn ta có thể thay thế toán tử tích phân $\frac{1}{s}$ bằng *I(z)*.

Có nhiều phương pháp để tìm toán tử *I(z)*, sau đây sẽ trình bày phương pháp Tustin hay còn gọi là phương pháp hình thang.

Từ (4 - 25) có thể suy ra đạo hàm của tín hiệu ra chính là tín hiệu vào. Trong trường hợp tín hiệu gián đoạn ta có thể viết:

$$y(k+1) - y(k) = \underbrace{\frac{T}{2}\left[u(k+1) + u(k)\right]}_{\text{Diện tích hình thang trên hình 4.4b}} \qquad (4 - 28)$$

trong đó, vế phải của (4 - 28) chính là diện tích hình thang có hai cạnh đáy u(k), u(k+1) và chiều cao là T như ở trên hình 4.4b.

Biến đổi *Z* phương trình sai phân (4 - 28) ta được phương trình sau:

$$ZY(Z) - Y(Z) = \frac{T}{2}\left[ZU(Z) + U(Z)\right] \qquad (4 - 29)$$

$$Y(Z)(Z-1) = \frac{T}{2}(Z+1)U(Z)$$

Vậy hàm truyền số trong trường hợp này chính là toán tử tích phân số:

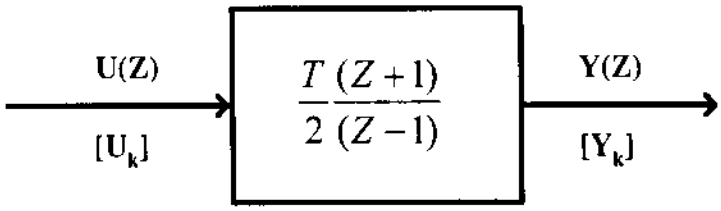
$$I(Z) = \frac{Y(Z)}{U(Z)} = \frac{T}{2}\frac{Z+1}{Z-1} \qquad (4 - 30)$$

Từ (4 - 29) và (4 - 30) ta có tương đương giữa hai toán tử liên tục và số:

$$\frac{1}{s} = \frac{T}{2}\frac{Z+1}{Z-1} \quad \text{hoặc } s = \frac{2}{T}\frac{Z-1}{Z+1} \qquad (4 - 31)$$

Công thức (4 - 31) có tên là công thức Tustin. Công thức này chính là công thức (4 - 23) ta đã tìm thấy ở mục 4.6.

Hình 4.5 là sơ đồ cấu trúc của toán tử tích phân số.



Hình 4.5. Sơ đồ cấu trúc của tích phân số.

Như vậy từ hàm *W(s)* ta thay *s* bằng biểu thức (4 - 31) ta được hàm *W(Z)* Dùng phương pháp biến đổi ảnh và gốc ta tìm được phương trình sai phân và từ đó viết chương trình mô hình hóa hệ liên tục trên máy tính.

Thông thường người ta dùng phương pháp Tustin vì nó đơn giản và cho độ chính xác khá cao

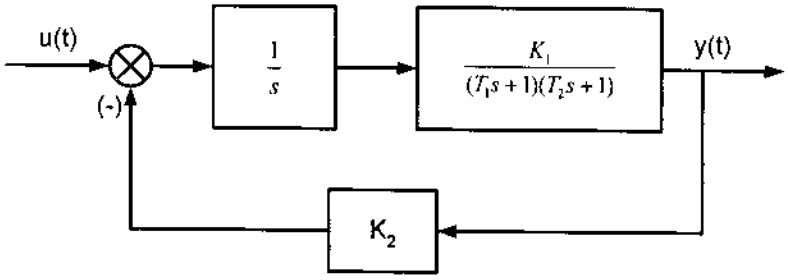
4.11. Ví dụ minh hoạ

Cho hệ điều khiển tự động sau, hãy mô hình hoá trên máy tính, viết chương trình trên máy tính bằng ngôn ngữ Pascal, tìm hàm quá độ. In ra 100 kết quả bằng số.

Vẽ đường cong quá trình quá độ y(t)

Dùng phần mềm MATLAB để kiểm tra kết quả đã tính.

1. Sơ đồ cấu trúc của hệ điều khiển tự động



2. Tham số của hệ thống

u(t) = 1(t) ;

$K_1 = 50$; $K_2 = 0,5$;

$T_1 = 0,02$; $T_2 = 0,03$;

Bước cắt mẫu chọn T = 0,001.

3. Tìm phương trình sai phân

$$W(s) = \frac{\dfrac{K_1}{s(T_1 s+1)(T_2 s+1)}}{1+\dfrac{K_1.K_2}{s(T_1 s+1)(T_2 s+1)}}$$

$$W(s) = \frac{K_1}{T_1T_2s^3 + (T_1 + T_2)s^2 + s + K_1K_2}$$

Thay $s = \frac{2}{T}.\frac{z-1}{z+1}$

$$W(z) = \frac{K_1}{T_1T_2\left(\frac{2(z-1)}{T(z+1)}\right)^3 + (T_1 + T_2)\left(\frac{2(z-1)}{T(z+1)}\right)^2 + \left(\frac{2(z-1)}{T(z+1)}\right) + K_1K_2}$$

Biến đổi hàm truyền trên ta có:

$$W(z) = \frac{KT^3(Z^3 + 3Z^2 + 3Z + 1)}{AZ^3 + BZ^2 + CZ + D}$$

trong đó:

$A = 8T_1 T_2 + 4T_1 T + 4T_2 T + 2T^2 + K_1 K_2 T^3$

$B = 24T_1 T_2 - 4T_1 T - 4T_2 T + 2T^2 + 3K_1 K_2 T^3$

$C = 24T_1 T_2 - 4T_1 T - 4T_2 T - 2T^2 + 3K_1 K_2 T^3$

$D = 8T_1 T_2 + 4T_1 T + 4T_2 T - 2T^2 + K_1 K_2 T^3$

Ta có hàm sai phân như sau:

$$\frac{Y(z)}{U(z)} = \frac{K_1T^3 (z^3 + 3 z^2 + 3 z + 1)}{A z^3 + B z^2 + C z + D}$$

$$Az^3 Y(z) + B z^2 Y(z) + C z Y(z) + D Y(z) = K_1 T^3 (z^3 U(z) + 3 z^2 U(z) + 3 zU(z) + U(z))$$

Dùng tính chất dịch hàm gốc của biến đổi Z ta tìm được phương trình sai phân tương ứng với phương trình trên:

$$AY[k+3]+BY[k+2]+CY[k+1]+DY[k] = K_1T^3(U[k+3] + 3U[k+2]+3U[k+1] + U[k])$$

Vì tín hiệu vào là tín hiệu nhảy cấp U(t) = 1(t) nên ta có

$U[k+3] = U[k+2] = U[k+1] = U[k] = 1$

vậy ta có:

$$AY[k+3]+BY[k+2]+CY[k+1]+DY[k] = 8K_1T^3$$

Cuối cùng ta tìm được phương trình sai phân của hệ điều khiển tự động là:

$Y[k+3] = (-BY[k+2]-CY[k+1]-DY[k] + 8K_1T^3) / A$

Từ phương trình sai phân ta viết chương trình máy tính để tìm đáp ứng ra của hệ *y(t)* khi tín hiệu vào là hàm nhảy cấp *1(t)*

Sau đây là chương trình Pascal mô hình hóa hệ ĐKTĐ đã cho. Chương trình cho kết quả là tín hiệu ra *y(k)* dưới dạng số, cứ cách 10 số in ra một số liệu. Chương trình cũng cho kết quả dưới dạng đường cong quá trình quá độ của hệ ĐKTĐ và tính các đặc tính quá độ như: y_{max}, $y_{ôd}$, σ_{max}, T_{max}, $T_{ôd}$. Sau đó dùng phần mềm Matlab để vẽ đường đặc tính quá độ của hệ ĐKTĐ nhằm đối chứng với kết quả của chương trình mô phỏng. Kết quả cho thấy rằng hai đường cong do chương trình mô hình hoá và phần mềm Matlab vẽ ra trùng nhau, điều đó chứng tỏ thuật toán mô hình hoá là đúng.

4. Chương trình Pascal

```
program MO_HINH_HOA;
uses crt,graph;var
a,b,c,max,k1,k2,t1,t2,t,tm,tod::real;
gd,gm,km,ky,i:integer;
y:array[0..1000] of real;
st:string;
BEGIN
clrscr;
k1:=50; k2:=0.5; t1:=0.02; t2:=0.03;
t:=0.001; y[0]:=0; y[1]:=0; y[2]:=0;
 a:= 8*t1*t2+4*t1*t+4*t2*t+k1*k2*t*t*t;
 b:= 24*t1*t2-4*t1*t-4*t2*t+2*t*t+3*k1*k2*t*t*t;
 c:= 24*t1*t2-4*t1*t-4*t2*t-2*t*t+3*k1*k2*t*t*t;
 d:= -8*t1*t2+4*t18t+4*t2*t-2*t*t+k1*k2*t*t*t;
for k:=0 to 997 do
y[k+3]:=(-b*y[k+2]-c*y[k+1]-d*y[k]+8*k1*t*t*t)/a;
 { in ra 100 gia tri]
writeln('100 gia tri cach nhau');
for k:=1 to 100 do
 begin
 write('y[',k*10:3,']=',y[k*10]:8:5,');
```

```
 if (k mod 4)=0 hten writeln;
 if k=80 then readln;
end;
 { tim gia tri ymax}
 max:=y[0];
 for k:=0to 1000 do
  if y[k] > max then
 begin
  max:=y[k];
  km:=k;
 end;
 {Tim khoang thoi gian on dinh Tod }
 k:=1000;
 write abs((y[k]-(1/k2))/(1/k2)) <= 0.05 do
  k:=k-1;
  tod:=k*t;
 { in cac gia tri ra man hinh }
 writeln('Thoi gian on dinh laTod:'.tod:8:4);
 xicma:=(max-1/k2)*100/(1/k2);
 writeln('Gia tri cuc dai la ymax=',max:8;4);
 writeln('Do qua dieu chinh xicma=',xicma:8:4,'%');
 writeln('Thoi gian dat cuc dai la tm=',km*t:8:4);
  { Ve hinh }
 write(' Hay cho he so gian truc y:ky='); readln(ky);
 gd:=detect:
 initgraph(gd,gm,'C:\TP\BGI');
 setbkcolor(white);
 setcolor(blue);
 outtextxy(53,5,'Y');
 outtextxy(600,360,'t(s)'); { Ve cac truc toa do }
 outtextxy(240,390,'KHAO SAT QTQD HE THONG');
 str(max:8:4,st);
 outtextxy(20,410,'GIA TRI CUC DAI LA: Ymax='+st);str(km*t:8:4,st);
```

```
outtextxy(320,410,'THOI GIAN DE DAT CUC DAI: Tmax='+st);
str(xicma:8:4,st);
outtextxy(20,430,'DO QUA DIEU CHINH LA: xicma %='+st);
str(Tod:8:4,st);
outtextxy(320,430.'THOI GIAN ON DINH LA: Yod='+st);
setcolor(5);
line(50,350,620,350; { Ve truc toa do }
line(50,5,50,350);
line(50,5,47,15); { Ve mui ten truc y }
line(50,5,53,15);
line(620,350,615,347); { Ve mui ten truc t }
line(620,350,615,353);for k:=1 to 6 do { Khac do truc t }
 begin
  str(100*(k-1)*t:2:1,st);
  outtextxy(k*100-60,357,st);
  line(k*100-50,347,k*100-50,353);
 end;
for k:=0 to 10 do { khac do truc y }
 begin
 str(k/(5*k2):4:2,st);
 outtextxy(5,345-round(ky*k/(5*k2)),st);
 line(47,350-round(ky*k/(5*k2)),53,350-round(ky*k/(5*k2)));
 end;
str(max:6:4,st);
outtextxy(round(tm/t)=60,340-round(ky*max),'Ymax='+st);
str(l/k2:2:l,st);
outtextxy(560,335-round(ky/k2),'Yod='+st);
outtextxy(km+50,330,Tmax');
outtextxy(round(tod/t)+50,330,'Tod');
outtextxy(round(tod/t)+70,315-round(ky/k2),'5%Yod');
moveto(50,350);
setlinestyle(Dottedln,0,1);
line(round(tod/t)+49,350-round(ky*y[round(tod/t)]),round(tod/t)+49,350);
```

```
line(50,350-round(ky*max),km+50,350-round(ky*max));
line(km+50,350-round(ky*max),km+50,350);
line(50,350-round(1.05*ky/k2),350-round(1.05*ky/k2)); { duong sai so }
line(50,350-round(0.95*ky/k2),620,350-round(0.95*ky/k2));
setlinestyle(SolidLn,0,1);
line(50,350-round(ky/k2),620,350-round(ky/k2)); { duong on dinh }
line(round(tod/t)+100,350-round(1.05*ky/k2),round(tod/t)+100,330
-round(1.05*ky/k2));
line(round(tod/t)+100,350-round(1.05*ky/k2),round(tod/t)+97,340
-round(1.05*ky/k2));
line(round(tod/t)+100,350-round(1.05*ky/k2),round(tod/t)+103,340
-round(1.05*ky/k2));
line(round(tod/t)+100,350-round(0.95*ky/k2),round(tod/t)+100,370
-round(0.95*ky/k2));
line(round(tod/t)+100,350-round(0.95*ky/k2),round(tod/t)+97,360
-round(0.95*ky/k2));
line(round(tod/t)+100,350-round(0.95*ky/k2),round(tod/t)+103,360
-round(0.95*ky/k2));
 { Ve do thi }
for k=1to 560 do
 begin
 if GraphResult<>grOK then
 halt(1);
setlinestyle(SolidLn,$C3,ThickWidth);
setcolor(Blue);lineto(k+50,350-round(ky*y[k])):
 end;
repeat until keypressed;
closegraph;
END.
```

Sau khi chạy chương trình ta được kết quả sau:

100 gia tri cach nhau

y[10] =0.00832 y[20] =0.06489 y[30] =0.19076 y[40] =0.38468

y[50] =0.63390 y[60] =0.92046 y[70] =1.22483 y[80] =1.52826

y[90] =1.81425 y[100] =2.06938 y[110] =2.28377 y[120] =2.45122

.....

.....

y[930] =2.00113 y[940] =2.00127 y[950] =2.00132 y[960] =2.00129

y[970] =2.00119 y[980] =2.00104 y[990] =2.00085 y[900] =2.00064

Thoi gian on dinh la: Tod=0.3360

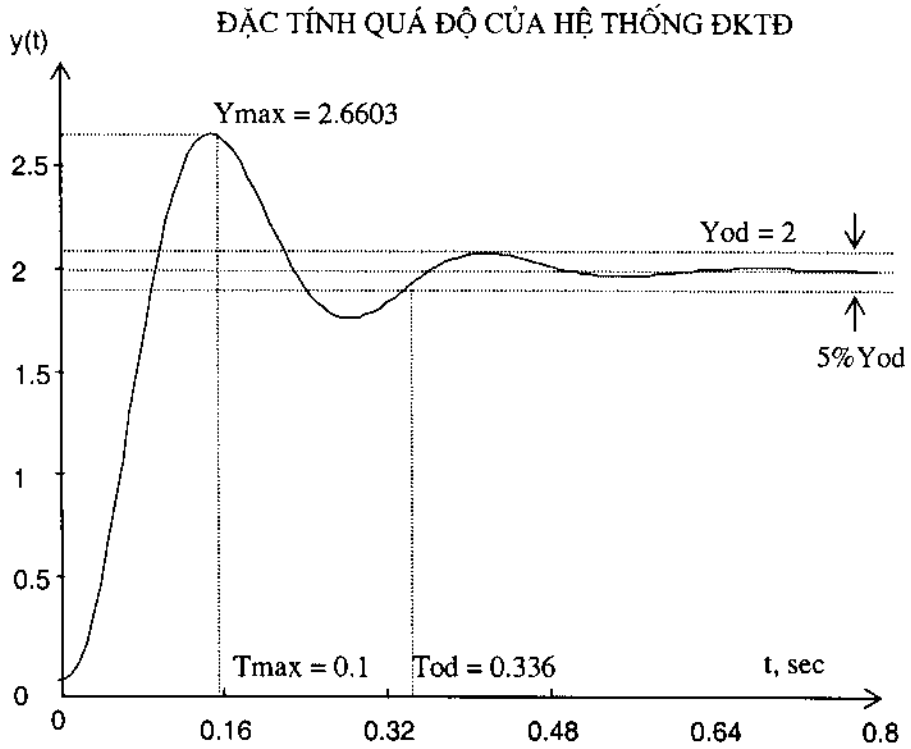
Gia tri cuc dai la: Ymax=2.6603

Do qua dieu chinh:xicma=33.0161%

Thoi gian dat cuc dai: tm=0.1100

Hay cho he so gian truc y: Ky=100

Đường cong quá trình quá độ của hệ thống do chương trình mô phỏng vẽ ra như trên hình 4.4.



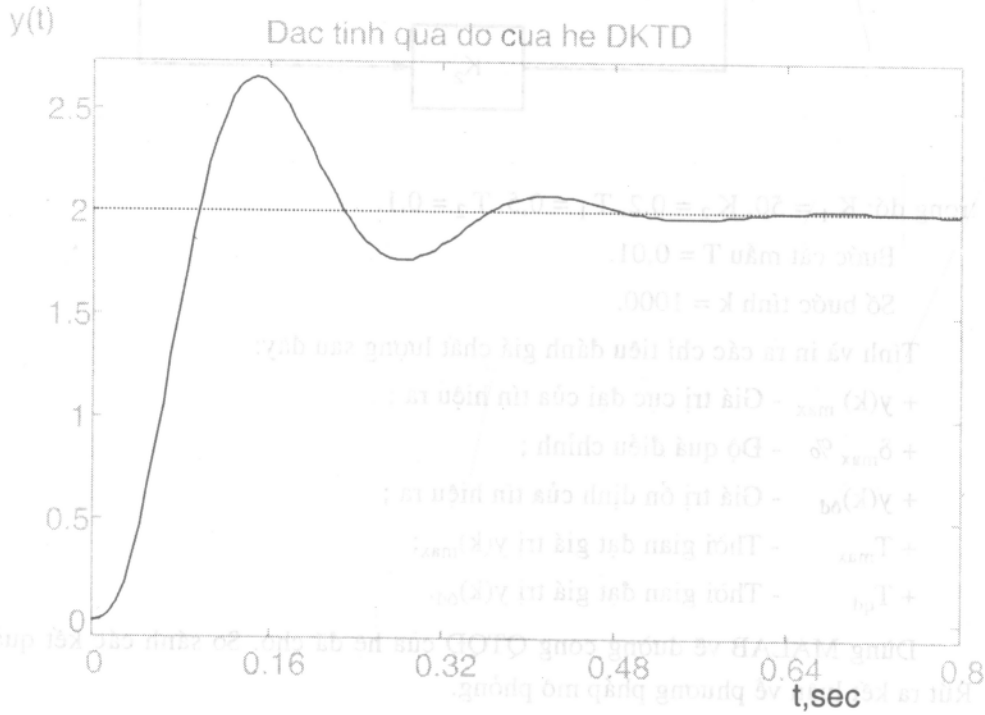
Hình 4.6. Đường cong QTQĐ do mô phỏng vẽ.

7. Chương trình Matlab

```
num=[k1];
den=[t1*t2,t1+t2,1,k1*k2];
step(num,den); title('Dac tinh qua do cua he DKTD');
ylabel('y(t)'); xlabel('t,sec');
```

Đường cong quá trình quá độ của hệ thống do chương trình Matlab vẽ ra như trên hình 4.7.

Chúng ta dễ nhận thấy rằng hai đường cong trên hình 4.6 và 4.7 trùng nhau, điều đó chứng tỏ thuật toán đã dùng là đúng đắn.



Hình 4.7. Đường cong QTQĐ do Matlab vẽ.

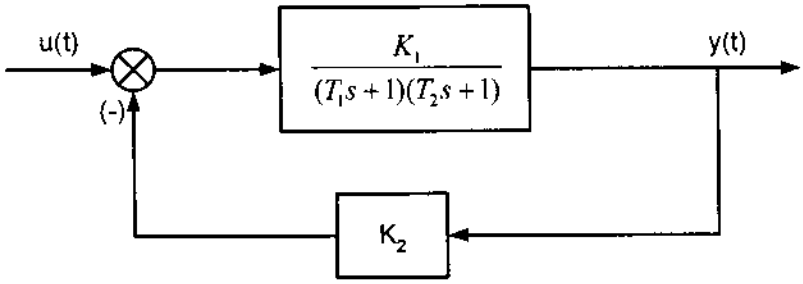
Bài tập

1. Hãy trình bày phương pháp tìm hàm sai phân của hệ ĐKTĐ liên tục.

2. Tìm hàm sai phân của hệ liên tục có hàm truyền sau:

$$W(s) = \frac{K}{s(s^2 + 2a\sqrt{K}\ s + K}$$

3. Hãy dùng máy tính để mô phỏng và khảo sát quá trình quá độ của hệ liên tục có sơ đồ cấu trúc sau đây:



trong đó: $K_1 = 50$, $K_2 = 0.2$, $T_1 = 0,5$, $T_2 = 0,1$.

Bước cắt mẫu T = 0,01.

Số bước tính k = 1000.

Tính và in ra các chỉ tiêu đánh giá chất lượng sau đây:

+ $y(k)_{max}$ - Giá trị cực đại của tín hiệu ra ;

+ δ_{max} % - Độ quá điều chỉnh ;

+ $y(k)_{ổd}$ - Giá trị ổn định của tín hiệu ra ;

+ T_{max} - Thời gian đạt giá trị $y(k)_{max}$;

+ T_{qd} - Thời gian đạt giá trị $y(k)_{ổd}$.

Dùng MALAB vẽ đường cong QTQĐ của hệ đã cho. So sánh các kết quả. Rút ra kết luận về phương pháp mô phỏng.

Chương 5
MÔ HÌNH HOÁ CÁC HỆ NGẪU NHIÊN

5.1. Khái niệm chung về phương pháp mô hình hoá các hệ ngẫu nhiên

Hệ ngẫu nhiên là hệ trong đó có các các biến ngẫu nhiên. Các biến ngẫu nhiên được đặc trưng bởi luật phân phối xác suất. Vì vậy những *mô hình mô phỏng* của các hệ ngẫu nhiên thường được gọi là *mô hình xác suất*.

Thực chất của phương pháp xây dựng mô hình xác suất là xây dựng trên máy tính hệ thống *S* với các quan hệ nội tại của nó trong đó có các biến ngẫu nhiên. Đầu vào của hệ có các tác động mang tính ngẫu nhiên như số lượng các sự kiện xảy ra, thời gian giữa các sự kiện hoặc tác động của môi trường xung quanh *E*. Trên cơ sở đó phân tích các dữ liệu đầu ra người ta nhận được dáng điệu phản ứng của hệ thống. Phương pháp này thường được gọi là phương pháp mô phỏng (*Simulation*). Mỗi một lần thực hiện phép thử người ta thu được một lời giải chứa đựng những thông tin về dáng điệu của hệ thống *S*. Nếu số phép thử *N* đủ lớn thì kết quả thu được bằng cách lấy trung bình theo xác suất sẽ ổn định và đạt độ chính xác cần thiết.

Phương pháp mô phỏng thường được dùng để nghiên cứu các hệ ngẫu nhiên nhưng đồng thời trong một số trường hợp cũng có thể được dùng để giải các bài toán đối với các hệ tiền định.

Mục 5.2 trình bày về phân phối xác suất của các biến ngẫu nhiên. Mục 5.3 và 5.4 trình bày về số ngẫu nhiên phân phối đều *U(0,1)* và phương pháp tạo các biến ngẫu nhiên có phân phối mong muốn. Cuối cùng mục 5.5 đưa ra một số ví dụ mô phỏng một số hệ ngẫu nhiên.

5.2. Phân phối xác suất của các biến ngẫu nhiên

Trong hệ ngẫu nhiên có nhiều biến ngẫu nhiên. Bảng 5-1 liệt kê một số biến ngẫu nhiên trong các hệ khác nhau.

Các đặc trưng quan trọng nhất của biến ngẫu nhiên là hàm mật độ xác suất, hàm phân phối xác suất, các thông số, kỳ vọng toán, phương sai và một số đặc

trưng khác. Sau đây sẽ xem xét một số phân phối liên tục và gián đoạn thường dùng nhất trong mô phỏng các hệ ngẫu nhiên.

Bảng 5-1

Loại hệ thống	***Các biến ngẫu nhiên***
Hệ thống sản xuất	Thời gian vận hành máy, ngừng máy do hỏng hóc, thời gian thao tác, số lần hỏng hóc...
Hệ thống máy tính	Thời gian giữa các lần làm việc, thời gian giải các bài toán...
Hệ thống thông tin liên lạc	Số khách hàng, thời gian giữa các lần liên lạc, thời gian liên lạc, thời gian phục vụ...

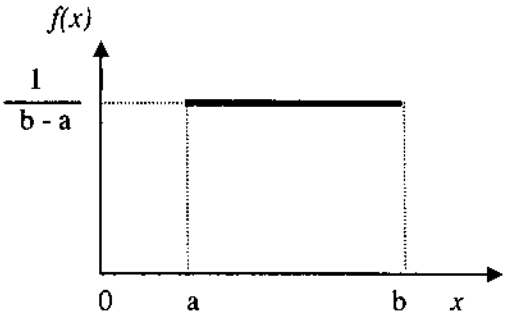
A. Phân phối liên tục (Continuous Distribution)

a) Phân phối đều (Uniform) ***U(a,b)***

Đại lượng ngẫu nhiễn X được gọi là tuân theo luật phân phối đều liên tục trên khoảng [a,b] ký hiệu $X \sim U(a,b)$, nếu X có các đặc trưng sau:

* Hàm mật độ xác suất:

$$f(x) = \begin{cases} \dfrac{1}{b-a} & \text{nếu } a \le x \le b \\ 0 & \text{phần còn lại} \end{cases}$$



* Hàm phân phối xác suất

$$F(x) = \begin{cases} 0 & \text{nếu } a \le x \le b \\ \dfrac{x-a}{b-a} & \text{nếu } x < a \\ 1 & \text{nếu } b \le x \end{cases}$$

* Thông số $a < b$, a,b - số thực

* Kỳ vọng toán: $M(x) = \dfrac{a+b}{2}$

* Phương sai: $S^2(x) = \dfrac{(b-a)^2}{12}$

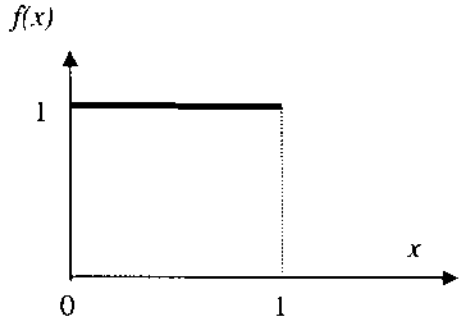
b) Phân phối đều (Uniform) ***U (0,1)***

Phạm vi ứng dụng: Phân phối đều trong khoảng [0,1] ký hiệu là *U(0,1)* là

trường hợp đặc biệt của phân phối đều *U(a,b)* với a = 0, b = 1. Phân phối đều *U(0,1)* được dùng nhiều trong kỹ thuật mô phỏng để tạo nên các đại lượng ngẫu nhiên khác có phân phối mong muốn

* Hàm mật độ phân phối

$$f(x) = \begin{cases} 1 & \text{nếu } 0 \le x \le 1 \\ 0 & \text{phần còn lại} \end{cases}$$



* Hàm phân phối

$$F(x) = \begin{cases} x & \text{nếu } 0 \le x \le 1 \\ 1 & \text{nếu } x > 1 \end{cases}$$

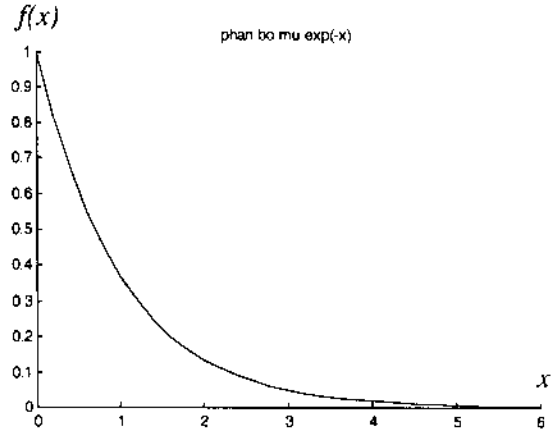
*Kỳ vọng toán: $M(x) = \frac{1}{2}$

*Phương sai: $S^2(x) = \frac{1}{12}$

c) Phân phối mũ (Exponential) ***expo (β)***

* Hàm mật độ phân phối

$$f(x) = \begin{cases} \frac{1}{\beta} e^{-x/\beta} & \text{nếu } x \ge 0 \\ 0 & \text{phần còn lại} \end{cases}$$



* Hàm phân phối

$$F(x) = \begin{cases} 1 - e^{-x/\beta} & \text{nếu } x \ge 0 \\ 0 & \text{phần còn lại} \end{cases}$$

* Thông số β

* Kỳ vọng toán: M(x) =

* Phương sai: $S^2(x) = \beta^2$

Phạm vi ứng dụng: Thường dùng để biểu diễn thời gian giữa hai sự kiện trong dòng sự kiện tối giản.

Chú ý: Hàm *expo (β)* là trường hợp đặc biệt của hai hàm phân bố Gamma và Veibull

d) Phân phối chuẩn (Normal) N(μ,σ²)

* Hàm mật độ phân phối

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$ cho mọi x là số thực

* Hàm phân phối F(x): không có công thức biểu diễn

* Thông số: $\mu > 0$, $\sigma \in (-\infty, \infty)$

* Kỳ vọng toán $M(x) = \mu$

* Phương sai $S^2(x) = \sigma^2$

Phạm vi ứng dụng: Phân phối chuẩn còn có tên là *phân phối Gaoxơ (Gauss)* là phân phối có ứng dụng rất rộng rãi trong việc đánh giá các đại lượng ngẫu nhiên.

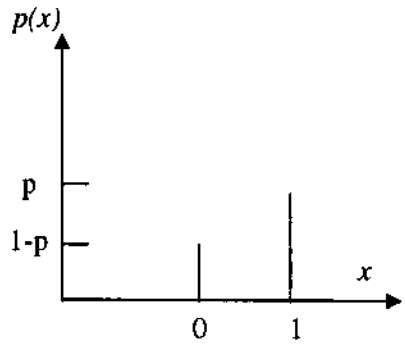
B. Phân phối gián đoạn (Discrete Distribution)

a) Phân phối Bernoulli ***Bernoulli (p)***

* Phạm vi ứng dụng: là số ngẫu nhiên xảy ra với hai khả năng khác nhau, thường được dùng để tạo ra các biến ngẫu nhiên gián đoạn như: nhị phân, hình học và âm nhị phân

* Hàm khối lượng

$$p(x) = \begin{cases} 1-p & \text{nếu } x = 0 \\ p & \text{nếu } x = 1 \\ 0 & \text{phần còn lại} \end{cases}$$



* Hàm phân phối

$$F(x) = \begin{cases} 0 & \text{nếu } x < 0 \\ 1-p & \text{nếu } 0 \le x < 1 \\ 1 & \text{nếu } 1 \le x \end{cases}$$

* Thông số: $p \in (0,1)$

* Kỳ vọng toán $M(x) = p$

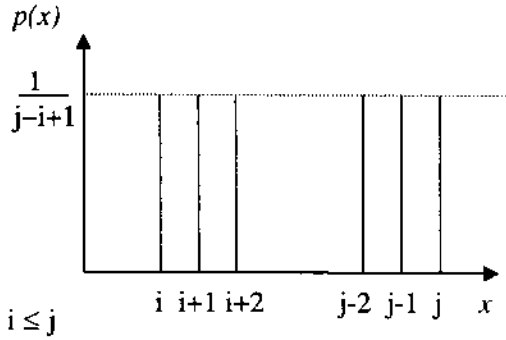
* Phương sai $S^2(x) = p(1-p)$

b) Phân phối đều gián đoạn (Discrete Uniform) ***DU(i,j)***

Phạm vi ứng dụng: Dùng để mô tả các số ngẫu nhiên xảy ra gián đoạn như nhau trên khoảng từ i đến j (j > i).

*Hàm khối lượng

$$p(x)=\begin{cases}\dfrac{1}{j-i+1} & \text{nếu } x\in i, i+1...j\\ 0 & \text{phần còn lại}\end{cases}$$



*Hàm phân phối

$$F(x)=\begin{cases}0 & x<i\\ \dfrac{|x|-i+1}{j-i+1} & \text{nếu } i\le x\le j\\ 1 & \text{nếu } j<x\end{cases}$$

* Thông số: i và j - số nguyên , $i \le j$

i - thông số đầu

j - thông số cuối

*Kỳ vọng toán: $M(x) = \dfrac{i+j}{2}$

*Phương sai: $S^2(x) = \dfrac{(j-i+1)^2-1}{12}$

Chú ý phân phối Bernoulli ($\frac{1}{2}$) chính là DU (0,1)

c) Phân phối Poatxông (Poisson) ***Poisson (λ)***

* Phạm vi ứng dụng: mô tả các sự kiện độc lập xảy ra với cường độ là hằng số

* Hàm khối lượng

$$p(x) = \begin{cases}\dfrac{e^{-\lambda}\lambda^x}{x!} & \text{nếu } x\in(0,1)\\ 0 & \text{phần còn lại}\end{cases}$$

* Hàm phân phối

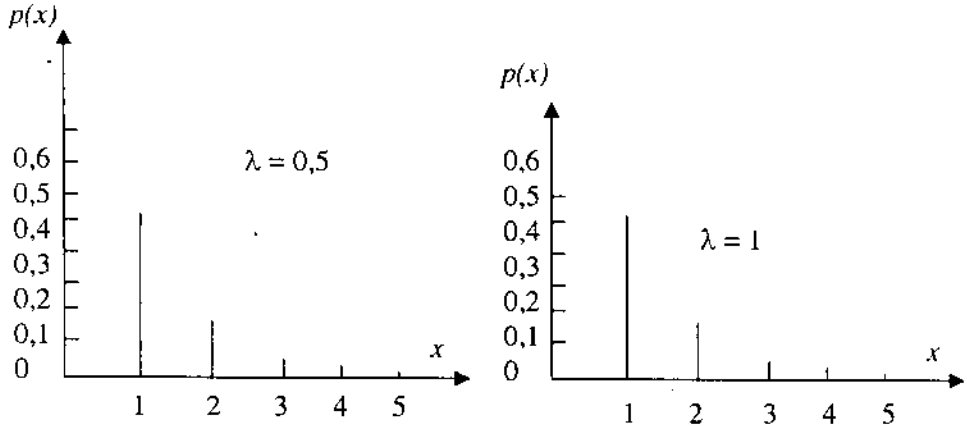
$$F(x)=\begin{cases}0 & \text{nếu } x<0\\ e^{-\lambda}\displaystyle\sum_{i=0}^{[x]}\frac{\lambda^i}{i!} & \text{nếu } 0\ge x\end{cases}$$

* Thông số $\lambda > 0$

* Kỳ vọng toán $M(x) = \lambda$

* Phương sai $S^2(x) = \lambda$

Sau đây là hàm khối lượng Poisson với thông số $\lambda = 0,5$ và $\lambda = 1$



5.3. Số ngẫu nhiên (Random numbers) phân phối đều U (0,1)

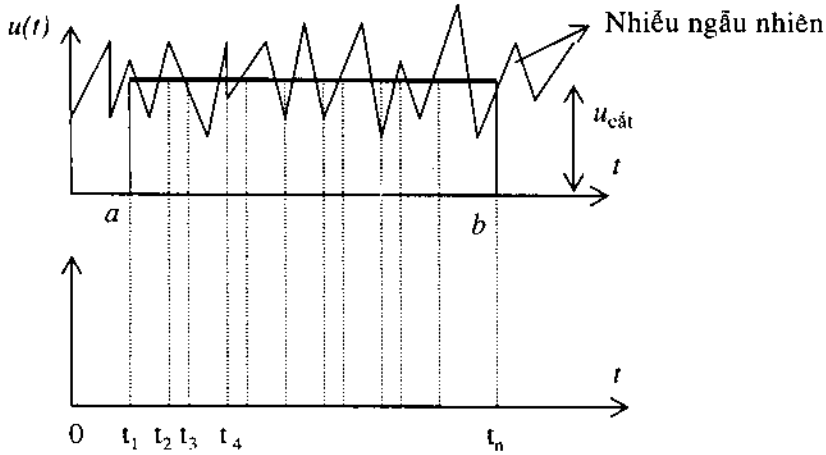
Khi mô phỏng hệ thống người ta thường cần có các số ngẫu nhiên phân phối theo những luật phân phối nhất định để mô phỏng các sự kiện ngẫu nhiên xảy ra trong hệ. Số ngẫu nhiên phân phối đều trong khoảng [0,1] thường được dùng làm cơ sở để sản sinh ra số ngẫu nhiên có các phân phối khác nhau. Phân phối đều *U(0,1)* đóng vai trò quan trọng trong kỹ thuật mô phỏng, vì vậy đã được nhiều người nghiên cứu và có nhiều phương pháp tạo ra nó. Sau đây chúng ta nghiên cứu một số phương pháp thông dụng tạo ra phân phối đều *U (0, 1)*.

a) Dùng máy phát ngẫu nhiên

Máy phát số ngẫu nhiên dựa trên nguyên tắc sử dụng nhiễu do các thiết bị điện tử gây ra. Trên hình 5.1, điện áp *u(t)* lấy trên điện trở trong một mạch khuếch đại điện tử đóng vai trò của nhiễu ngẫu nhiên. Người ta chọn quãng thời gian lấy mẫu *T(a,b)* và biên độ điện áp cắt $U_{cắt}$ tuỳ ý. Giao điểm giữa *u(t)* và $U_{cắt}$ tạo thành dãy số ngẫu nhiên $t_1, t_2, t_3, t_4, ..., t_n$. Đây là dãy số ngẫu nhiên phân phối đều *U(0,1)*

Ưu điểm: Nhận được dãy số hoàn toàn ngẫu nhiên với số lượng không hạn chế (bằng cách thay đổi thời gian lấy mẫu *T(a,b)* và điện áp cắt $U_{cắt}$).

Nhược điểm: Phải lắp thêm vào máy tính máy phát số ngẫu nhiên. Khi cần làm lại quá trình mô phỏng thì không tạo được dãy số ngẫu nhiên giống lần trước nên không thể so sánh chính xác kết quả của hai lần thử nghiệm.



Hình 5.1. Phương pháp tạo số ngẫu nhiên.

b) Dùng bảng số ngẫu nhiên

Bằng nhiều cách người ta lập bảng các số ngẫu nhiên, xem bảng 5-2 [2].

Bảng 5-2. Các số ngẫu nhiên có phân bố đều U(0,1)

10097	32533	76520	13586	34673	54876	80959	09117	39292	74945
37542	04805	64894	74296	24805	24037	20616	10402	00822	91655
08422	68953	19645	09303	23209	02560	15 953	34764	35808	33606
99019	02529	09376	70715	38311	31165	88676	74397	04436	27659
12087	99910	80157	36147	64032	36653	98951	16877	12171	76833
66065	74717	34072	76850	36697	36170	65813	39885	11199	29170
31060	10805	45571	82406	35303	42614	86799	07439	23043	09732
85269	77602	02051	65692	68665	74818	73053	85247	18623	88579
63573	32135	05325	47048	90533	57548	28468	28709	83491	25624
73796	45753	03529	64778	35808	34282	60935	20344	35237	88435
98520	17767	14905	68607	22109	40558	60970	93433	50500	73998
11805	05431	39808	27732	50725	68248	29405	24201	52775	67851
83452	99614	06288	98083	13746	70078	18475	40610	68711	77817
88685	40200	86507	58401	36766	67951	90364	76493	29609	11062
99594	67348	87517	64969	91825	08928	93785	61368	23478	34113
65481	17674	17468	50950	58047	76974	73039	57186	40218	16544
80124	35635	17727	08015	45318	22374	21115	78253	14385	53763
74350	99817	77402	77214	43236	00210	45521	64237	96286	02655
69916	26803	66252	29148	36936	87203	76621	13990	94400	56418
09893	14523	14225	68514	46427	56788	96297	78822	34382	14398

91499	14523	68479	27686	46162	83554	94750	89923	37089	20048
80336	94598	26940	36858	70297	34135	53140	33340	42050	82341
44104	81949	85157	47954	32979	26575	57600	40881	22222	06413
12550	73742	11100	02040	12860	74697	96694	89419	23707	25815
63606	49329	16505	34484	20419	52563	43651	77082	07207	31790
61196	90446	26457	47774	51924	33792	65394	59593	42582	60527
15474	45266	95270	79953	59367	83848	82396	10118	33211	59446
94557	28573	67897	54187	54622	44431	91190	42592	92927	45973
42418	16213	97344	09721	16868	48767	03071	12059	23701	46670
23523	78317	73208	89837	68935	91416	26252	29633	05522	82562
04493	52494	75248	33824	45862	51025	61962	79335	65337	12472
00549	97654	64051	88159	96119	63896	54692	82391	23287	29529
35963	15307	26898	09354	33351	35462	77974	50024	90103	39333
59808	08391	45427	26842	83609	49700	13021	24892	78565	20160
46058	85236	01390	92286	77281	44077	93910	83647	70617	42941
32197	00597	37397	25241	05567	07007	86743	17157	85394	11838
69234	61406	20117	45204	15956	60000	18743	92423	97118	96338
19565	41430	01758	75379	40419	21585	66674	36806	84962	85207
45155	14938	19476	07246	43667	94543	59047	90033	20826	69541
94864	31994	36168	10851	34888	81553	01540	35456	05014	51176
98086	24826	45240	28044	44999	08896	39094	73407	35441	31880
33185	16232	41941	50949	89435	48481	88695	41994	37548	73043
80951	00406	96382	70774	20151	23387	25016	25298	94624	61171
79752	49140	71961	28296	69861	02591	74852	20539	00187	59579
18633	32537	98145	06571	31010	24674	05455	61247	77938	91936
74029	43902	77557	32270	97790	17119	52527	58021	80814	51748
34178	45611	80993	37143	05335	12969	56127	19225	26040	90234
11644	49883	52079	84827	59381	71539	09973	33440	15020	09994
69074	94138	87637	91976	35584	04401	10518	21615	01848	76918

Khi mô phỏng có thể lấy các số ngẫu nhiên trong bảng ra theo một thứ tự nào đó: lấy lần lượt, lấy cách quãng v.v...

Ưu điểm: có thể lặp lại dãy số ngẫu nhiên để dùng cho các lần mô phỏng khác nhau.

Nhược điểm: tốn bộ nhớ để nhớ bảng số ngẫu nhiên.

c) Dùng algorithm tạo số giả ngẫu nhiên (Pseudorandom Numbers)

Ngày nay người ta thường dùng algorithm tạo số ngẫu nhiên. Như vậy rất thuận tiện vì khi lập trình chỉ cần lập chương trình con tạo số ngẫu nhiên mà không cần phải ghi dãy số ngẫu nhiên vào bộ nhớ của máy tính. Tuy nhiên người ta chứng minh được rằng bất kỳ thuật toán nào cũng tạo ra số ngẫu nhiên

có chu kỳ nên nó không hoàn toàn là số ngẫu nhiên mà chỉ là *số giả ngẫu nhiên (Pseudorandom Numbers)*. Tuy nhiên nếu chu kỳ của số giả ngẫu nhiên đủ lớn khoảng $(1 \div 5)\ 10^6$ số thì số giả ngẫu nhiên đó có thể được coi là ngẫu nhiên đối với các bài toán mô phỏng thông thường. Có nhiều thuật toán tạo ra số giả ngẫu nhiên.

- *Thuật toán lấy phần giữa của bình phương*

Cho số khởi đầu $x_0 = 0{,}2152$, vậy số ngẫu nhiên x_1, x_2... tiếp theo sẽ được tính như sau:

$$(x_0)^2 = 0{,}04631104 \rightarrow x_1 = 0{,}6311$$

$$(x_1)^2 = 0{,}39828721 \rightarrow x_2 = 0{,}8287$$

Nhược điểm của phương pháp này là rất dễ xảy ra trường hợp chu kỳ lặp lại của số ngẫu nhiên quá ngắn.

Ví dụ chọn $x_0 = 0{,}4500$, vậy ta có

$$(x_0)^2 = 0{,}20250000 \rightarrow x_1 = 0{,}2500$$

$$(x_1)^2 = 0{,}06250000 \rightarrow x_2 = 0{,}2500$$

$$(x_2)^2 = 0{,}06250000 \rightarrow x_3 = 0{,}2500$$

- *Thuật toán nhân*

Thuật toán

$$Z_{i+1} = \lambda\, x_i$$

$$x_{i+1} = \,]\, Z_{i+1} \,[$$

trong đó:

xi - số ngẫu nhiên phân bố đều U (0, 1);

$]\, Z_{i+1} \,[$ - là phần lẻ của số Z_{i+1};

λ - hệ số ; $\lambda = 8\,t \pm 3$;

t - số nguyên dương bất kỳ.

Ví dụ chọn $t = 5 \rightarrow \lambda = 8 \times 5 \pm 3$ lấy $\lambda = 37$

Cho trước $x_0 = 0{,}37843$

Ta có dãy số ngẫu nhiên sau

$$Z_1 = \lambda\, x_0 \rightarrow x_1 = \,]\, Z_1 [$$

$$Z_2 = \lambda\, x_1 \rightarrow x_2 = \,]\, Z_2 [$$

Cụ thể tính được như sau

$$Z_1 = 37 \times 0{,}37843 = 14{,}00191 \rightarrow x_1 = 0{,}00191$$

$$Z_2 = 37 \times 0{,}00191 = 0{,}07067 \rightarrow x_2 = 0{,}07067$$

$$Z_3 = 37 \times 0{,}07067 = 2{,}61497 \rightarrow x_3 = 0{,}61479$$

$$Z_4 = 37 \times 0{,}61479 = 22{,}74723 \rightarrow x_4 = 0{,}74723$$

Dãy số ngẫu nhiên thu được sẽ phân phối đều trong khoảng [0,1]. Người ta chứng minh được chu kỳ lặp lại của dãy số ngẫu nhiên này đủ lớn nên có thể dùng để dùng trong phép mô phỏng.

5.4. Phương pháp tạo các biến ngẫu nhiên có phân phối mong muốn

Khi mô hình hoá hệ thống thường phải mô phỏng các sự kiện ngẫu nhiên có phân phối khác nhau. Để tạo ra các số ngẫu nhiên như vậy người ta thường dùng các số ngẫu nhiên phân phối đều *U(0,1)* để tạo ra các số ngẫu nhiên có phân phối mong muốn. Sau đây chúng ta nghiên cứu phương pháp thường dùng nhất được gọi là *phương pháp biến đổi nghịch đảo*

a) Thuật toán biến đổi nghịch đảo

Giả thiết rằng chúng ta muốn tạo ra số ngẫu nhiên liên tục X có hàm phân phối liên tục tăng trong khoảng $0 < F(x) < 1$ có nghĩa là nếu $x_1 < x_2$ và $0 < F(x_1) \leq F(x_2) < 1$ thì $F(x_1) < F(x_2)$. Gọi F^{-1} là nghịch đảo của F. Vậy thuật toán để tạo ra biến ngẫu nhiên X có hàm phân phối $F(x)$ như sau:

a. Lấy $U \sim U(0,1)$

b. Vậy $X = F^{-1}(U)$

Trong đó ký hiệu ~ được đọc là "có phân phối". Hàm $F^{-1}(U)$ luôn luôn xác định trong khoảng [0,1]. Sau đây sẽ xem xét một số thuật toán tạo các biến ngẫu nhiên có phân phối mong muốn.

b) Thuật toán tạo biến ngẫu nhiên có phân phối mũ ***expo*** *(β)*

Thuật toán

a. Lấy $U \sim U(0,1)$

b. Vậy $X = -\beta \ln U$

Trong đó β là thông số của phân phối mũ *expo (β)*

Chứng minh thuật toán nêu trên

Ta có hàm mật độ phân phối mũ (xem mục 5-2)

$$f(x) = \frac{1}{\beta} e^{-\frac{1}{\beta}x} \qquad x \geq 0$$

Gọi X_i là số ngẫu nhiên có phân phối mũ

U_i là số ngẫu nhiên có phân phối đều trong khoảng [0,1]

$$U_i = \int_0^{xi} f(x)\ dx = \int_0^{xi} \frac{1}{\beta}\ e^{-\frac{1}{\beta}x} dx$$

Đặt

$$U_i = -\ e^{-\frac{1}{\beta}x} \Big|_0^{xi} = 1 - e^{-\frac{1}{\beta}xi}$$

Vậy $X_i = -\beta \ln(1-U_i)$

Chú ý rằng nếu U_i phân phối đều trong [0 ,1] thì $(1-U_i)$ cũng phân phối đều trong khoảng [0,1] nên ta có thể viết

$X_i = -\beta \ln U_i$

c) Thuật toán tạo biến ngẫu nhiên có phân phối đều ***U(a,b)***

Thuật toán

a. Lấy $U \sim U(0,1)$

b. Vậy $X = a + (b-a)\,U$

d) Thuật toán tạo biến ngẫu nhiên có phân phối chuẩn $N(\mu, \sigma^2)$

Thuật toán tìm phân phối chuẩn khá phức tạp, tuy nhiên có thể áp dụng định lý giới hạn trung tâm sau đây: Phân phối chuẩn có thể được coi là tổng một số N khá lớn các ngẫu nhiên U_i có phân phối đều trong [0,1].

Nếu có $U_1, U_2, \ldots, U_N$

Kỳ vọng toán $M_1, M_2, \ldots, M_N$

Độ lệch chuẩn $\sigma_1, \sigma_2, \ldots, \sigma_N$

Vậy ta có kỳ vọng toán của phân phối chuẩn $\mu = \mu_1 N$

Độ lệch chuẩn của phân phối chuẩn $\sigma = \sigma_1 \sqrt{N}$

Tóm lại, tổng $\sum_{i=1}^{N} Ui$ có phân phối gần với phân phối chuẩn

Trong thực tế lấy N = 8 ~ 12 là đủ.

e) Thuật toán tạo biến ngẫu nhiên có phân phối Poisson gián đoạn, **Poisson** (λ)

Thuật toán

1. Lấy $a = e^{-\lambda}$, $b = 1$ và $i = 0$

2. Lấy $U_{i+1} \sim U(0, 1)$ và thay b bằng bU_{i+1}. Nếu $b < a$ thì lấy $X = i$ nếu không đến bước 3.

3. Thay i bằng $i+1$ và trở về bước 2.

Chú ý rằng điều kiện $X = i$ chỉ xảy ra khi và chỉ khi có

$$\sum_{j=1}^{i} Y_j \le 1 \le \sum_{j=1}^{i+1} Y_j$$

trong đó: $Y_j = -\frac{1}{\lambda}\ln U_j \sim \text{expo}\left(\frac{1}{\lambda}\right)$ và là các số ngẫu nhiên độc lập.

g) Thuật toán tạo biến ngẫu nhiên có phân phối đều gián đoạn ***DU(i, j)***

Thuật toán

1. Lấy $U \sim U(0, 1)$
2. Vậy $X = i + [(j - i + 1)U]$

h) Thuật toán tạo biến ngẫu nhiên có phân phối ***Bernoulli (p)***

Thuật toán

1. Lấy $U \sim U(0, 1)$
2. Nếu $U \le p$, vậy $X = 1$ nếu không $X = 0$.

5.5. Một số ví dụ về mô phỏng các hệ ngẫu nhiên

Phần này chúng ta nghiên cứu phương pháp mô phỏng một số hệ ngẫu nhiên.

Ví dụ 1: Các dòng sự kiện có phân bố mũ expo (λ).

Xét một số trường hợp sau:

- Dòng các cuộc gọi của một trạm điện thoại.
- Dòng các khách hàng đi vào một cơ sở dịch vụ như siêu thị, khu vui chơi giải trí, quán ăn, hiệu cắt tóc v.v...
- Dòng các khách hàng đi ra khỏi các cơ sở dịch vụ sau khi đã được phục vụ.
- Dòng các nhiễu ngẫu nhiên tác động vào hệ truyền tin gây ra các sai số

- Dòng các hỏng hóc xảy ra trong hệ thống kỹ thuật do tác động của các yếu tố bên ngoài như: nhiệt độ, độ ẩm, rung động v.v...

Các dòng sự kiện nói trên thường có tính chất sau đây:

- Dòng dừng vì cường độ xảy ra sự kiện $\lambda = const$.
- Các sự kiện xảy ra hoàn toàn độc lập với nhau (không hậu quả).
- Tại một thời điểm chỉ có một sự kiện xảy ra.

Một dòng có ba tính chất trên là dòng Poisson dừng hay còn gọi là dòng tối giản. Như chúng ta đã biết *khoảng cách giữa các sự kiện xảy ra trong dòng tối giản tuân theo luật phân phối mũ expo (λ).*

Ví dụ 2: Mô phỏng độ tin cậy của thiết bị điện tử

Giả thiết rằng cường độ xảy ra hỏng hóc của một thiết bị điện tử là hằng số λ lần /giờ = const. Hãy xác định độ tin cậy $P(t > T)$ và tuổi thọ trung bình T của thiết bị điện tử.

Do $\lambda = const$ nên dòng các hỏng hóc là dòng tối giản. Như vậy khoảng cách giữa các hỏng hóc t_i tuân theo luật phân phối mũ *expo (λ).* Gọi T là thời gian khảo sát, vậy thiết bị được coi là làm việc tin cậy nếu $t_i \geq T$. Khi khảo sát một lượng đủ lớn các thiết bị thì độ tin cậy của nó được đánh giá bằng xác suất xảy ra sự kiện $t_i \geq T$. Như vậy độ tin cậy được định nghĩa là xác suất để thời gian làm việc không hỏng hóc lớn hơn hoặc bằng thời gian khảo sát $P(t_i \geq T)$.

- Thuật toán mô phỏng

a. Lấy Ui ~ U(0, 1)

b. Vậy $t_i = -\frac{1}{\lambda}\ln U_i$

c. So sánh t_i với T

nếu $t_i \geq T$ - thiết bị làm việc tin cậy

nếu $t_i < T$ - thiết bị làm việc không tin cậy

d. Thực hiện N "thử nghiệm" như vậy từ bước a đến bước c

Độ tin cậy của thiết bị được đánh giá như sau:

$$\text{Độ tin cậy} = P(t_i > T) \approx \frac{\textit{Số thiết bị làm việc tin cậy}}{\textit{Số thiết bị thử nghiệm } N}$$

Nếu N càng lớn thì đánh giá sẽ càng chính xác.

Ví dụ 3: Mô phỏng hệ truyền tin

- *Mô phỏng nguồn sai:*

Xét trường hợp kênh truyền tin là nhị phân đối xứng. Nhiễu trong kênh liên

lạc là loại nhiễu xung ngẫu nhiên. Các xung nhiễu (dương hoặc âm) có tham số gần bằng xung tín hiệu sẽ gây ra các sai là biến tín hiệu 1 thành 0 hoặc ngược lại biến tín hiệu 0 thành 1. Do kênh nhị phân đối xứng nên có $p_{0\to1} = p_{1\to0} = p$. Các sai xảy ra trong kênh liên lạc là các sự kiện ngẫu nhiên độc lập.

Như vậy dòng sai trong kênh liên lạc có thể được mô phỏng bằng dòng tối giản có cường độ $\lambda = const$.

Thông thường cường độ dòng sai trong kênh liên lạc nằm trong khoảng

$\lambda = 10^{-1} \div 10^{-4}$ 1 / giờ

Như đã phân tích ở ví dụ 2, khoảng cách giữa các sai tuân theo luật phân phối mũ *expo(λ)*, khoảng cách giữa các sai được tính theo công thức sau:

$$t_i = -\frac{1}{\lambda}\ln U_i$$

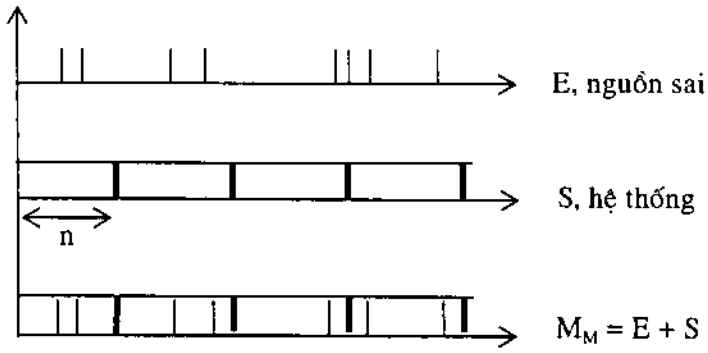
trong đó: U_i - số ngẫu nhiên ~ U(0, 1);

t_i - khoảng cách giữa các sai i-1 và i.

Vậy *mô hình nguồn sai là dãy các sai có cường độ λ, khoảng cách giữa các sai tuân theo luật phân phối mũ expo(λ).*

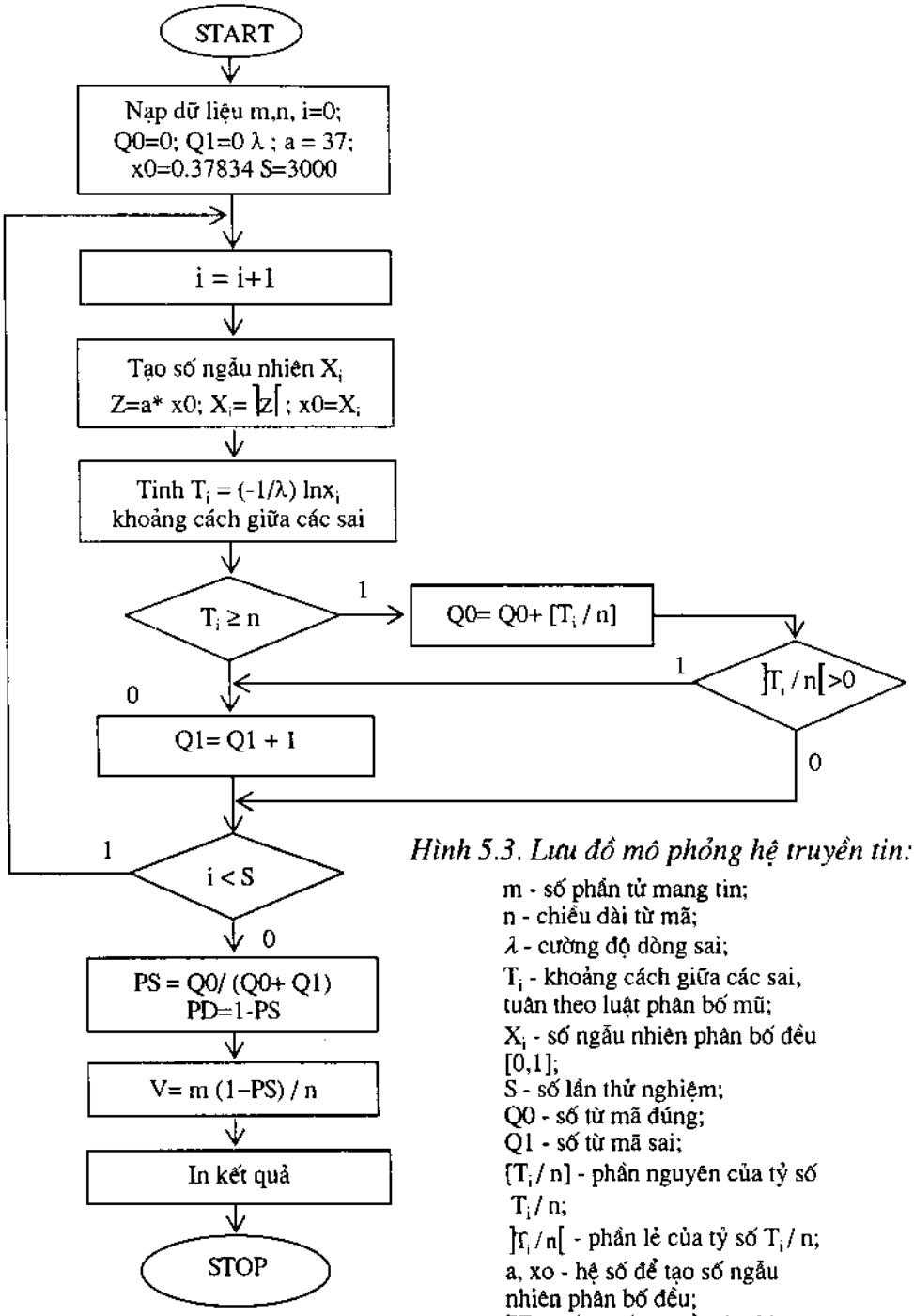
- *Mô phỏng quá trình truyền tin*

Giả sử truyền đi liên tục các từ mã có chiều dài *n*, trong đó số phần tử mang tin là *m (m < n)*. Vậy mô hình hệ thống truyền tin *S* là một dãy liên tục các từ mã có chiều dài *n*. Mô hình mô phỏng hệ truyền tin M_M là sự xếp chồng của mô hình hệ thống *S* và mô hình môi trường *E* - chính là mô hình nguồn sai như trên hình 5.2.



Hình 5.2. Mô hình mô phỏng quá trình truyền tin.

- Thuật toán mô phỏng: Xem lưu đồ mô phỏng hình 5.3



Hình 5.3. Lưu đồ mô phỏng hệ truyền tin:

m - số phần tử mang tin;
n - chiều dài từ mã;
λ - cường độ dòng sai;
T_i - khoảng cách giữa các sai, tuân theo luật phân bố mũ;
X_i - số ngẫu nhiên phân bố đều [0,1];
S - số lần thử nghiệm;
Q0 - số từ mã đúng;
Q1 - số từ mã sai;
[T_i / n] - phần nguyên của tỷ số T_i / n;
]T_i / n[- phần lẻ của tỷ số T_i / n;
a, xo - hệ số để tạo số ngẫu nhiên phân bố đều;
PD - xác suất truyền tin đúng;
PS - xác suất truyền tin sai;
V - tốc độ truyền tin.

a. Lấy $U_i \sim U(0,1)$

b. Tính khoảng cách giữa các sai

$$t_i = -\frac{1}{\lambda}\ln U_i$$

c. So sánh giữa t_i và chiều dài từ mã n

+ Nếu $t_i > n \rightarrow$ số từ mã đúng Q0 (không có sai) sẽ được tăng lên

$$Q0 = Q0 + \left[\frac{T}{N}\right]$$

+ Kiểm tra phần lẻ của tỷ số $]T/N[$, nếu $]T/N[> 0$: số từ mã sai Q1 (có sai) sẽ được tăng lên một đơn vị, Q1 = Q1 + 1

+ Nếu $t_i < n$: số từ mã sai Q1 sẽ được tăng lên một đơn vị Q1 = Q1 + 1

d. Lặp lại từ bước *a* đến bước *c* khi số lần thử nghiệm hãy còn nhỏ hơn số *S* đã định trước.

Ví dụ 4: Ví dụ minh hoạ - Mô phỏng trạm xe buýt sinh viên

Sinh viên đi từ ký túc xá đến trường bằng xe buýt, mỗi xe chứa được 60 sinh viên. Thời gian đi đến trường bắt đầu từ 6 giờ 30 đến 7 giờ 30. Sinh viên đi đến trạm xe buýt được mô tả bằng một dòng tối giản với cường độ $\lambda = 0.8$ *sv/ s*. Cứ sau $T_{xe} = 15$ *phút* có một chuyến xe buýt đi đến trường. Nếu số sinh viên chờ xe < 60 sinh viên, xe vẫn chạy đúng giờ. Nếu số sinh viên chờ xe > 60 sinh viên thì số sinh viên thừa ra sẽ chờ chuyến xe tiếp theo.

Sinh viên than phiền rằng không đủ xe để đi đến trường nên sau 7 giờ 30 vẫn còn nhiều sinh viên bị kẹt lại không kịp đến trường đúng giờ.

Hãy cho biết sau 7 giờ 30 trung bình có bao nhiêu sinh viên bị kẹt lại tại trạm xe buýt?

Để đảm bảo không có sinh viên bị chậm giờ học thì thời gian giữa các chuyến xe buýt lớn nhất $T_{xe\,max}$ phải là bao nhiêu?

Viết chương trình mô phỏng. Xây dựng giao diện để nạp thông số của đầu bài và lấy kết quả ra từ màn hình máy tính.

- *Thuật toán mô phỏng*

a. Xây dựng mô hình dòng sinh viên đi đến trạm xe buýt. Khoảng cách giữa các sinh viên đi đến trạm xe buýt là $t_i = -\frac{1}{\lambda}\ln(U_i)$, $U_i \sim U(0,1)$.

b. Xây dựng mô hình vận chuyển xe buýt. Số xe phân bố đều đặn, sau T_{xe} = 15 phút có một chuyến xe buýt.

c. Xếp chồng hai mô hình trên với nhau.

d. Đếm số sinh viên chờ ở trạm xe buýt.

e. Đếm số sinh viên còn lại tại trạm sau mỗi chuyến xe.

g. Mô phỏng quá trình vận chuyển sinh viên sau 1 giờ (từ 6^{30} đến 7^{30})... Đếm số sinh viên bị kẹt lại sau 7 giờ 30. Lặp lại với T_{xe} khác nhau để tìm $T_{xe\ max}$. Nhận xét?.

- *Lưu đồ mô phỏng*

Lưu đồ mô phỏng như trên hình 5.4. Trong đó đặt các biến như sau:

+ tx = T_{xe} - thời gian giữa các chuyến xe buýt;

+ *t* - là thời gian mô phỏng. Đây là thời gian sự kiện, *t* sẽ được tăng lên khi có sự kiện sinh viên đến trạm xe buýt. Quá trình mô phỏng thực hiện khi

t = 0 (s) (lúc 6 giờ 30) đến *t = 3600 (s)* (lúc 7 giờ 30). Dừng mô phỏng trong trường hợp ngược lại;

+ *c* - là số sinh viên chờ ở trạm xe buýt.

Như vậy khi có một sinh viên di đến trạm xe buýt thì

c = c+1

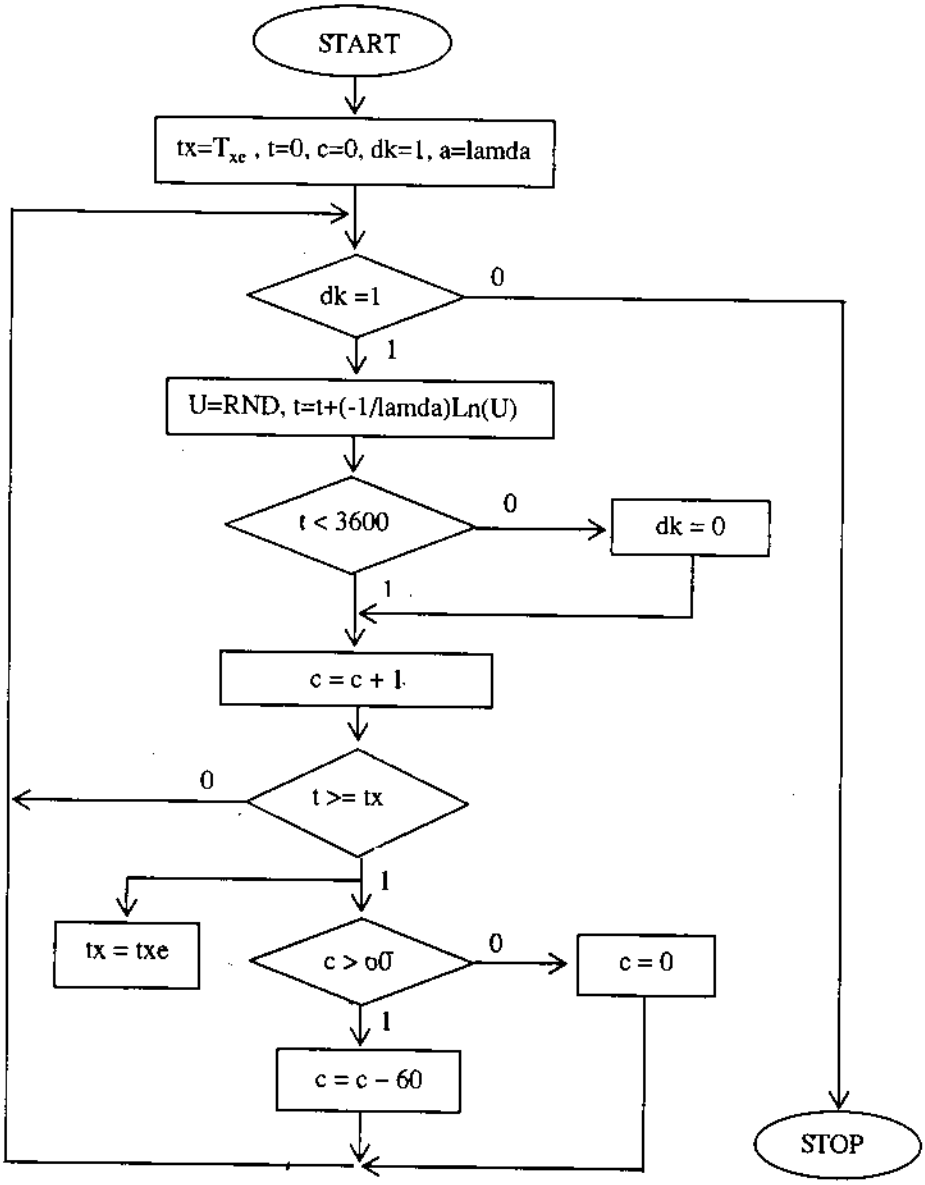
t = t + ti, trong đó ti là khoảng cách giữa các sinh viên đi đến trạm xe buýt

+ *dk* - là biến trung gian.

- *Chương trình mô phỏng viết bằng ngôn ngữ Visual Basic*

```
//*****************************************************
//****** chương trình viết cho Form1 *****
//****** khai báo các biến *************
Dim i, k, tx, txe, times, c, s, dk As Integer
Dim t As Double
Dim ast, bst, at, bt As String
Dim x(160), y(160) As Integer
//***** thủ tục tính *****
Public Sub tinh()
  Const e=2.7182
  a=lamda
  U=RND
```

```
  t = t + (- Log(U) / a*Log(e)
  If t< 3600 Then
  c=c+1
Else: t=3600
End If
End Sub
```



Hình 5.4. Lưu đồ mô phỏng trạm xe buýt sinh viên.

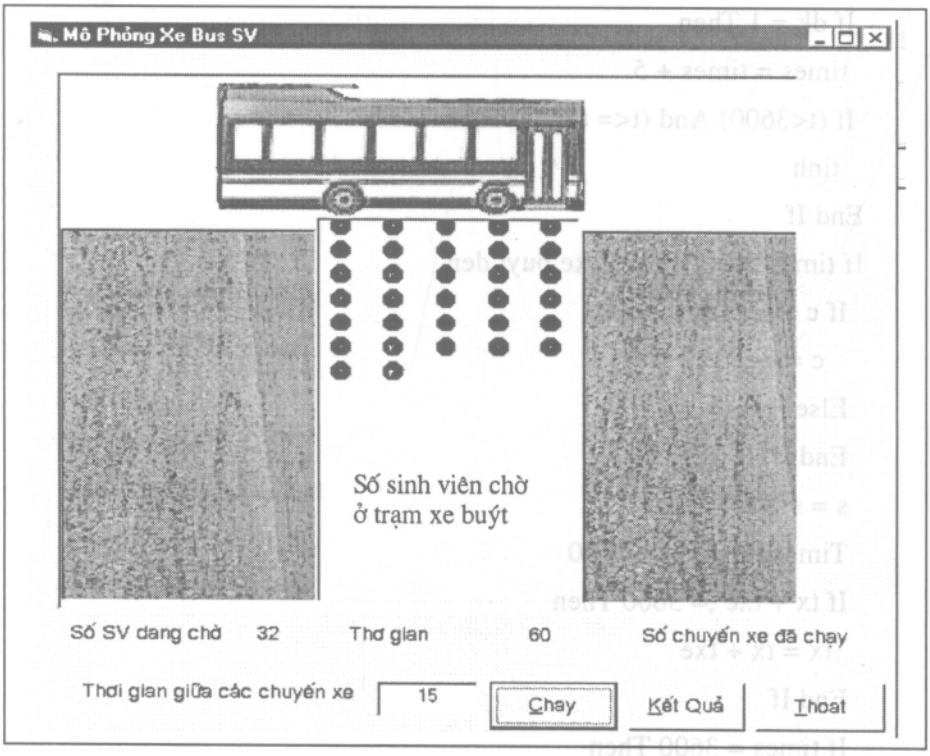
```
//***** thủ tục vẽ đường đi đến trạm xe của sinh viên*****
Public Sub veduong()
 Const r=50
 pictD.Cls
 For j=1 To c
 pictD.DrawWidth = 6
 pictD.Circle (x(j), y(j), r, RGB (0, 0, 0)
 Next
End Sub
//***** thủ tục vẽ ôtô chạy*****
Public Sub vecto()
  im.Left =i
  imI.Left =k
  If i> 14000 Then
  i = - 4000
  End If
  k = k+45000/ txe
End Sub
//*****thủ tục vẽ Form2*****
Public Sub hienthi()
  Form2.lbSVCC.Caption = c
  Form2.lbSCX2.Caption = s
  lbSCX2.Caption = Int(t/60)
  Form2.pt2.Cls
  Form2.pt1.Cls
  For j=1 To c
    Form2.pt2.Line((40*j-30), 10)-((40*j), 365), RGB(0, 250, 50), BF
    Next
  For j=1 To s
    Form2.pt1.Line((200*j-150), 10)-((200*j), 365), RGB(0, 250, 50), BF
    Next
```

```
End Sub
//*****nút lệnh hiển thị Form2*****
Private Sub cmdBD_Click()
  Form2.Left = 6600
  Form2.Top = 3000
  Form2.Show
End Sub
//*****nút lệnh chạy chương trình*****
Private Sub cmdChay_Click()
 t = 0
 s = 0
 c = 0i= =-7500
 k = 1500
 times = 0
 ast = lbSVC.Caption
 bst = lbSCX.Caption
 If txtTxe.Text = ""Then
   MsgBox " Nhap thoi gian giua cac chuyen xe",MB_OK
 Else
   txe = CDbl(txtTxe.Text)*60
   tx=txe
  dk=1
  ' tao vi tri chi SV choxe
  For j=1 To 160
    x(j) = 200
    x(j + 1) = 700
    x(j + 2) = 1200
    x(j + 3) = 1700
  x(j + 4) = 2200
  y(j) = 50*j
  y(j + 1) = 50*j
```

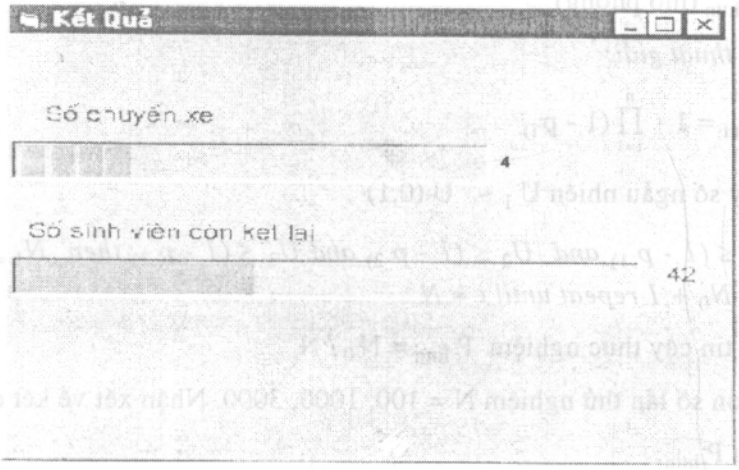
```
    y(j + 2) = 50*j
    y(j + 3) = 50*j
    y(j + 4) = 50*j
    j = j+4
   Next
End If
End Sub
//*****nút lệnh kết túc chương trình*****
Private Sub Command l_Click
Beep
End
End Sub
//*****thủ tục điều khiển chương trình bằng bộ Timerl*****
Private Sub Timerl_Timerl()
Timerl.Interval = 5
  If dk = 1 Then
   times = times + 5
  If (t<3600) And (t<= times) Then
    tinh
  End If
  If times >= tx The 'co xe buyt den
   If c > 60 Then
     c = c- 60
   Else: c = 0
   End If
   s = s + 1
   Timerl.Interval = 1000
   If tx + txe ,= 3600 Then
     tx = tx + txe
   End If
   If times = 3600 Then
```

```
    dk = 0
    End If
  End If
   veduong
   vecto
  hienthi
End If
End Sub
//*****thủ tục điều khiển chương trình bằng bộ Timer2*****
Private Sub Timer2_Timer2()
  a = Left(14.Caption,1)
  b = Right(14.Caption, Len(14.Caption) - 1)
  14.Caption = b + a
End Sub
```

- *Giao diện cho Form1 - Menu chính*



• *Giao diện cho Form2 - Hiển thị kết quả*



Kết quả mô phỏng

+ Nếu T_{xe} = *15 phút*, sau 1 giờ có 4 chuyến xe , số sinh viên kẹt lại là 42 sv.

+ Nếu T_{xe} = *12 phút*, sau 1 giờ có 5 chuyến xe. Các chuyến xe đều chở đầy sinh viên và sau mỗi chuyến xe không có sinh viên kẹt lại.

Như vậy nên chọn $T_{xe\ max}$ = *12 phút*

Bài tập

1. Mô phỏng độ tin cậy của hệ thống kỹ thuật

Dùng phương pháp mô phỏng để tính độ tin cậy của thiết bị kỹ thuật, biết rằng cường độ hỏng hóc $\lambda = 2.10^{-3}$ 1/ giờ.

Hãy vẽ đường cong biểu diễn độ tin cậy *P(t)* lý thuyết và *P(t)* mô phỏng khi số lần thực nghiệm lần lượt là S = 300, 1000, 3000. Rút ra kết luận về phương pháp mô phỏng.

Gợi ý: Độ tin cậy lý thuyết được tính bằng xác suất $P(t) = expo\ (-\lambda t)$, trong đó *t* - là thời gian khảo sát.

2. Đánh giá độ tin cậy của hệ thống kỹ thuật có các phần tử nối song song.

Cho hệ thống kỹ thuật gồm 3 phần tử nối song song có độ tin cậy lần lượt như sau: $p_1 = 0,7;\ p_2 = 0,75;\ p_3 = 0,8$

Hãy mô hình hoá để tính độ tin cậy của hệ thống.

So sánh độ tin cậy tính theo công thức lý thuyết P_{htlt} với độ tin cậy thực nghiệm P_{httn} (mô phỏng)

Gợi ý thuật giải:

a. $P_{htlt} = 1 - \prod_{i=1}^{n}(1 - p_i)$

b. Lấy số ngẫu nhiên $U_1 \sim U(0,1)$

if $U_1 \le (1 - p_1)$ *and* $U_2 \le (1 - p_2)$ *and* $U_3 \le (1 - p_3)$ *then* $N_1 = N_1 + 1$ *else* $N_0 = N_0 + 1$ *repeat until* $i = N$

c. Độ tin cậy thực nghiệm $P_{httn} = N_0 / N$

d. Chọn số lần thử nghiệm N = 100, 1000, 3000. Nhận xét về kết quả mô phỏng P_{httn}.

3. Đánh giá độ tin cậy của hệ thống kỹ thuật có các phần tử nối tiếp

Cho một hệ thống kỹ thuật gồm 3 phần tử nối tiếp có độ tin cậy lần lượt là $p_1 = 0,8$; $p_2 = 0,85$; $p_3 = 0,9$.

Hãy mô hình hoá để tính độ tin cậy thực nghiệm của hệ thống P_{httn}.So sánh với công thức lý thuyết P_{htlt}.

Gợi ý thuật giải:

a. $P_{htlt} = \prod_{i=1}^{n} p_i$

b. Lấy số ngẫu nhiên $U_1 \sim U(0,1)$

if $U_1 \le p_1$ *and* $U_2 \le p_2$ *and* $U_3 \le p$ *then* $N_0 = N_0 + 1$ *else*

$N_1 = N_1 + 1$ *repeat until* $i = N$

c. Độ tin cậy thực nghiệm $P_{httn} = N_0 / N$

d. Chọn số lần thử nghiệm N = 100, 1000, 3000. Nhận xét về kết quả mô phỏng P_{httn}

4. *Rùa và Thỏ chạy thi*

Thỏ chạy nhanh nhưng kiêu ngạo, chủ quan. Hệ số sẵn sàng của thỏ là *0,2* (hoặc tự chọn). Tốc độ chạy thi của thỏ là *1m / phút* (hoặc tự chọn).

Rùa chạy chậm, tốc độ khoảng *0,2 m / phút* (hoặc tự chọn). Nhưng rùa khiêm tốn, cần mẫn nên hệ số sẵn sàng của rùa là 0,9 (hoặc tự chọn).

Khoảng cách chạy thi là AB = 200 *m*

Rùa và thỏ cùng xuất phát tại điểm gốc A, tại thời điểm $t = 0$.

Luật chơi (hoặc tự chọn): chạy từng nhịp một, mỗi nhịp 1 phút. Sau đó xuất phát lại có tính đến hệ số sẵn sàng của từng đấu thủ. Quãng đường chạy được sẽ được cộng dồn. Ai đến điểm đích B trước là người ấy thắng cuộc.

Hãy thiết lập bài toán và tiến hành mô phỏng để rùa thắng thỏ.

5. *Máy phân loại gạch ốp lát*

Gạch ốp lát có 4 loại hỏng hóc chính là chiều dài, chiều rộng, độ cong vênh và hoa văn với xác suất tương ứng là p_1, p_2, p_3, p_4

Máy phân loại gạch ốp lát có nhiệm vụ phát hiện các hỏng hóc và phân loại gạch theo tiêu chuẩn sau:

- Gạch loại một: Không có hỏng hóc nào.
- Gạch loại hai: Có một loại hỏng hóc xuất hiện.
- Gạch loại ba: Có hai loại hỏng hóc đồng thời xuất hiện.
- Gạch loại bốn: Có ba loại hỏng hóc đồng thời xuất hiện.
- Gạch phế phẩm: Có bốn loại hỏng hóc đồng thời xuất hiện.

Hãy mô phỏng quá trình làm việc của máy phân loại gạch ốp lát. Số sản phẩm cần kiểm tra và phân loại là *N*.

Số liệu (gợi ý): N = 50, 100, 150, 200 viên gạch

$$p_1 = 0,1;\ p_2 = 0,1;\ p_3 = 0.05;\ p_4 = 0,01.$$

Chương 6
THU THẬP VÀ PHÂN TÍCH DỮ LIỆU ĐẦU VÀO

6.1. Khái niệm chung về thu thập và phân tích dữ liệu đầu vào

Thu thập và phân tích dữ liệu đầu vào có vai trò đặc biệt quan trọng trong quá trình mô hình hoá và mô phỏng hệ thống. Bởi vì nếu dữ liệu đầu vào không chính xác thì mô hình sẽ không phản ánh đúng hệ thống thực mà chúng ta muốn mô phỏng. Đối với hệ ngẫu nhiên, dữ liệu đầu vào thường là các biến ngẫu nhiên (xem bảng 5-1) như là khoảng thời gian giữa các khách hàng đến hệ thống, khoảng thời gian phục vụ v.v... Vấn đề đặt ra ở đây là thu thập các dữ liệu đầu vào, phân tích và tìm các phân phối xác suất đặc trưng cho các biến ngẫu nhiên đó. Có hai phương pháp để tìm phân phối xác suất của biến ngẫu nhiên:

- Dựa vào các quan sát trực tiếp trên hệ thống thực để lựa chọn "phân phối kinh nghiệm" đặc trưng cho biến ngẫu nhiên trong hệ thống.

- Dùng các dữ liệu thu thập được để phân tích và lựa chọn "phân phối lý thuyết" đặc trưng cho biến ngẫu nhiên trong hệ thống.

Các mục tiếp theo dưới đây chủ yếu trình bày phương pháp thứ hai là phương pháp được dùng phổ biến hiện nay trong mô phỏng hệ thống.

Mục 6.2 trình bày các phương pháp để thu thập dữ liệu đầu vào. Mục 6.3 trình bày cách lựa chọn phân phối xác suất của các dữ liệu đầu vào. Mục 6.4 trình bày phương pháp kiểm chứng tính chính xác của phân phối xác suất của các dữ liệu đầu vào đã được lựa chọn. Cuối cùng mục 6.5 trình bày về mô hình dòng đầu vào.

6.2. Các phương pháp thu thập dữ liệu đầu vào

Thu thập dữ liệu là một công việc đơn điệu, buồn tẻ nhưng rất quan trọng vì nếu dữ liệu thu thập được không chính xác thì các biến ngẫu nhiên trong mô hình mô phỏng sẽ không phản ánh đúng các quá trình trong hệ thực. Thu thập dữ liệu khi nào và bằng cách nào sẽ cho ta các dữ liệu chính xác đó là nghệ thuật của việc thu thập dữ liệu.

Để làm ví dụ chúng ta xem xét việc mô phỏng một siêu thị. Để tiến hành mô phỏng chúng ta phải thu thập các dữ liệu chính sau :

- Số lượng khách hàng đến siêu thị trên một đơn vị thời gian;
- Số lượng các mặt hàng khách hàng chọn mua;
- Thời gian chờ ở quầy tính tiền của các khách hàng;
- Số lượng và thời gian làm việc của các quầy tính tiền;
- Thời gian trung bình để kiểm tra và tính tiền một mặt hàng v.v...

Số lượng các dữ liệu thu thập được phải đủ lớn, có nghĩa là thời gian quan sát phải đủ dài để có thể mô phỏng một cách chính xác các biến ngẫu nhiên trong mô hình.

Có ba phương pháp thu thập dữ liệu đầu vào được dùng với các mục đích khác nhau.

- Phương pháp dữ liệu trực tiếp

Đối với phương pháp này, dữ liệu thu thập được dùng trực tiếp cho mô phỏng. Ví dụ dữ liệu là thời gian sửa chữa máy thì dữ liệu đó sẽ cần đến mỗi khi mô phỏng đến thời gian sửa chữa máy. Đây là phương pháp mô phỏng trực tiếp. Do không có tính tổng quát hoá cao nên phương pháp mô phỏng này ít được sử dụng.

- Phương pháp phân phối xác suất kinh nghiệm (*Empirical Distributions*)

Đối với phương pháp này, dữ liệu thu thập được dùng để xác định phân phối xác suất theo kinh nghiệm của biến ngẫu nhiên. Do phân phối được chọn là theo kinh nghiệm của người nghiên cứu nên có thể xảy ra trường hợp phân phối được chọn không chính xác, đặc biệt là khi dữ liệu thu thập được không đầy đủ.

- Phương pháp phân phối xác suất lý thuyết (*Theoretical Distributions*)

Đối với phương pháp này, sau khi thu thập được các dữ liệu đầu vào người ta tìm phân phối xác suất lý thuyết phù hợp nhất với dữ liệu đã có. Phương pháp này tránh được các hạn chế ở phương pháp phân phối xác suất kinh nghiệm đồng thời nó cho phép suy diễn các dữ liệu vượt ra ngoài tập các dữ liệu quan sát được. Chính vì vậy phương pháp này được dùng rộng rãi. Mục 6.3 dưới đây sẽ trình bày chi tiết các bước tìm phân phối xác suất của biến ngẫu nhiên theo phương pháp phân phối xác suất lý thuyết.

Có hai phương pháp phân tích dữ liệu thường dùng : phương pháp thống kê mô tả *(descriptive statistics)* và phương pháp thống kê suy diễn *(inferential statisstics)*.

Thế mạnh của phương pháp thống kê mô tả là từ một số lớn các dữ liệu thu thập được có thể quy tụ lại thành một số ít các mô tả rõ ràng dưới dạng các phân bố xác suất, do đó dễ sử dụng trong mô phỏng. Phương pháp thống kê mô tả được dụng rộng rãi trong mô phỏng.

Mục đích của phương pháp thống kê suy diễn là đánh giá đặc tính thống kê của các dữ liệu. Trong trường hợp các dữ liệu thu thập được có nhiều giá trị thì người ta cần làm phép thử để đánh giá xem các dữ liệu đó có mang tính ngẫu nhiên và độc lập với nhau và không. Trên cơ sở các đánh giá đó người ta có thể loại bỏ các dữ liệu " thừa " là các dữ liệu không mang tính độc lập với nhau.

Về nguyên tắc, số lượng các dữ liệu thu thập được phải càng nhiều càng tốt để có thể đánh giá đặc tính thống kê của các dữ liệu một cách khách quan. Trong thực tế, căn cứ vào đặc điểm của dòng đầu vào mà ta lựa chọn số lượng đủ lớn, thông thường phải có hàng trăm dữ liệu trở lên.

6.3. Phương pháp tìm phân phối xác suất của dữ liệu đầu vào

Phương pháp tìm phân phối xác suất của dữ liệu đầu vào thuộc loại phương pháp thống kê mô tả. Đặc điểm của phương pháp này là từ một số lớn các dữ liệu thu thập được người ta quy tụ lại thành phân phối xác suất. Như vậy rất thuận tiện trong việc mô tả biến ngẫu nhiên là các dữ liệu đầu vào.

Các bước tìm hàm phân phối xác suất của dữ liệu đầu vào có thể tóm tắt như sau:

Bước 1 - Thu thập dữ liệu đầu vào

Bước 2 - Xây dựng biểu đồ tần số (*Histogram*) từ biểu đồ đó chọn một loại phân phối xác suất lý thuyết tương đối phù hợp biểu đồ tần số.

Bước 3 - Xác định thông số của phân phối xác suất đã chọn

Bước 4 - Kiểm tra tính phù hợp giữa phân phối xác suất lý thuyết với các dữ liệu thực tế trên biểu đồ tần số bằng cách dùng phép thử Chi - bình phương χ^2 hoặc phép thử Kônmôgôrôp - Smirnôp (*Kolmogorov - Smirnov*).

Sau đây sẽ trình bày một ví dụ để minh hoạ các bước xử lý dữ liệu đầu vào của một trạm sửa chữa ôtô:

Bước 1: Thu thập dữ liệu

Để mô phỏng dòng dữ liệu đầu vào của một trạm sửa chữa ôtô, người ta ghi lại các khoảng thời gian, tính bằng phút, giữa các ôtô đến trạm sửa chữa tức là khoảng cách giữa các khách hàng của trạm. Các dữ liệu này phải được ghi lại hoàn toàn ngẫu nhiên và độc lập với nhau trong khoảng thời gian đủ dài để có số lượng dữ liệu đủ lớn.

Kết quả ghi lại được 120 dữ liệu như trên Bảng 6-1 n = 120 khoảng thời gian (phút) giữa các khách hàng.

Bảng 6-1

1,8	1	2,3	4	2,6	9,5
1	1,8	2,5	9,1	6,2	1
1	1,9	14	9,8	6,3	1,2
1,9	3	15,1	3	4,2	1,3
1,8	6,7	3	3	9,2	2,1
3	4,2	6,7	1	3	6,6
3,1	4	4,8	1,9	6,4	2,2
8,2	10,1	5	19	1	2,5
8,8	12,1	19,6	7,2	1,5	3,1
1	1,9	9,1	5,2	1,5	1,6
1,7	1,9	5,4	4	1,9	1,7
1,5	2,1	5,4	10,5	6,6	7,8
2,1	1,9	12,6	5,3	7,8	1,5
2,3	3,5	1	13	8,2	7,8
3,8	3,6	1	1	1,9	1,3
5,6	5,6	3	6,2	2,1	1,9
19	16,8	6,8	4,1	16,1	8,2
4,3	3,6	5	10,8	4,2	10,8
4,2	1,8	4	3	1,9	16,7
6,7	6	3,8	2,6	3,5	16,7

Bước 2: Xây dựng biểu đồ tần số

Biểu đồ tần số được dùng để đánh giá một cách trực quan dạng của hàm mật độ của phân phối liên tục. Căn cứ vào hình dáng của biểu đồ tần số, chúng ta có thể suy đoán lựa chọn một phân phối lý thuyết phù hợp với các dữ liệu thực tế thu thập được.

Để xây dựng biểu đồ, trước tiên lập bảng dữ liệu theo thứ tự tăng dần như ở bảng 6.2; n = 120 quãng thời gian (phút) giữa các khách hàng theo thứ tự tăng dần.

Bảng 6-2

1	1,8	2,3	3,8	6	9,1
1	1,8	2,5	4	6,2	9,2
1	1,8	2,5	4	6,2	9,5
1	1,8	2,6	4	6,3	9,8
1	1,9	2,6	4	6,4	10,1
1	1,9	3	4,1	6,6	10,5
1	1,9	3	4,2	6,6	10,8
1	1,9	3	4,2	6,7	10,8
1	1,9	3	4,2	6,7	12,1
1	1,9	3	4,2	6,7	12,6
1,2	1,9	3	4,3	6,8	13
1,3	1,9	3	4,8	7,2	14
1,3	1,9	3	5	7,8	15,1
1,5	1,9	3,1	5	7,8	16,1
1,5	2,1	3,1	5,2	7,8	16,7
1,5	2,1	3,5	5,3	8,2	16,7
1,5	2,1	3,5	5,4	8,2	16,8
1,6	2,1	3,6	5,4	8,2	19
1,7	2,2	3,6	5,6	8,8	19
1,7	2,3	3,8	5,6	9,1	19,6

Sau khi có bảng 6-2 chúng ta sẽ xây dựng bảng tần số các dữ liệu.

Trước tiên xác định giá trị min, max của các dữ liệu trong bảng 6-2, giá trị min = 1, max = 19,6.

Chúng ta chia khoảng (min, max) ra làm *k* khoảng khảo sát bằng nhau. Độ rộng của khoảng khảo sát không nên bé quá vì như vậy đồ thị tần số sẽ quá nhấp nhô, nếu chọn của khoảng khảo sát lớn quá sẽ bỏ qua nhiều thông tin về dữ liệu, kết quả sẽ khó đánh giá chính xác về hàm mật độ của phân phối.Theo kinh nghiệm, giá trị *k* nên lấy trong khoảng $k = 10 \div 20$. Trong ví dụ này chúng ta chọn khoảng khảo sát bằng 2 phút, có nghĩa là $k = 10$. Tiếp đó xác định số dữ liệu xuất hiện trong các khoảng khảo sát $(i, i+1)$ và tần suất tương ứng của nó.

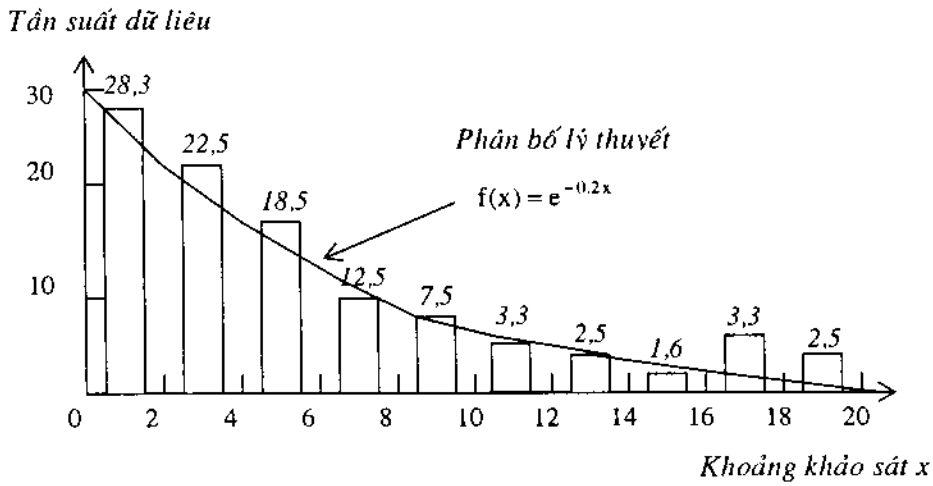
Bằng cách đó chúng ta xây dựng được bảng tần suất các dữ kiện như trên bảng 6-3

Bảng 6-3. Tần suất các dữ liệu

Khoảng khảo sát	***Số dữ liệu quan sát được, x_I***	***Tần suất, p_I %***
0-2	34	28,3
2-4	27	22,5
4-6	19	15,8
6-8	15	12,5
8-10	9	7,5
10-12	4	3,3
12-14	3	2,5
14-16	2	1,6
16-18	4	3,3
18-20	3	2,5
	Tổng :	*100%*

Từ bảng 6-3 chúng ta vẽ được biểu đồ tần số (*Histogram)* như ở hình 6.1.

Quan sát biểu đồ tần số trên hình 6.1 chúng ta thấy rằng tần suất dữ liệu giảm dần về phía phải, như vậy phân phối mũ có thể phù hợp với biểu đồ tần số. Vì vậy phân phối mũ được chọn là phân phối lý thuyết để mô tả dữ liệu đầu vào đã thu thập được.



Hình 6.1. Biểu đồ tần số và phân bố lý thuyết expo (- 0,2 x).

Bước 3: Xác định thông số của phân phối mũ lý thuyết *expo (λ)*

Trước tiên cần tính thông số quan trọng của phân phối xác suất là kỳ vọng toán.

Từ bảng 6-3, ta có thể tính được kỳ vọng toán (trung bình theo xác suất) của các khoảng cách giữa các khách hàng

$$M = \sum_{i=1}^{10} x_i p_i = 5{,}02 \text{ phút}$$

trong đó x_i , p_i lấy trong bảng 6-3.

Vậy cường độ dòng khách hàng (số khách hàng trong 1 phút) sẽ là

$$\lambda = \frac{1}{M} = \frac{1}{5{,}02} = 0{,}2 \text{ khách hàng / phút}$$

Từ đó ta có thể chọn thông số của phân bố mũ λ bằng cường độ dòng khách hàng . Vậy phân phối mũ được chọn có hàm mật độ phân phối như sau:

$$f(x) = e^{-0{,}2x}$$

Bước 4: Kiểm tra tính phù hợp giữa phân phối xác suất lý thuyết với các dữ liệu thực tế (*Goodness of Fit Test*)

Sau khi đã chọn được phân phối xác suất lý thuyết, chúng ta phải kiểm tra

mức độ phù hợp giữa phân phối lý thuyết với các dữ liệu thực tế. Việc kiểm tra được thực hiện theo những tiêu chuẩn nhất định. Nếu không đạt phải chọn lại thông số của phân phối xác suất hoặc khi cần thiết phải chọn lại dạng phân phối khác. Cách thực hiện bước 4 được trình bày trong mục 6.4 sau đây.

6.4. Kiểm tra tính phù hợp giữa phân phối xác suất lý thuyết với các dữ liệu thực tế

Kiểm tra tính phù hợp (*Goodness of Fit Test*) của các dữ liệu thực tế thu thập được (biểu đồ tần số) với phân phối mũ lý thuyết đã chọn là bước tiếp theo của việc xử lý dữ liệu đầu vào. Kiểm tra tính phù hợp là phép kiểm định giả thuyết thống kê nhằm xác định rằng các dữ liệu quan sát được $X_1, X_2, \ldots X_i, \ldots X_N$ có phải là các dữ liệu độc lập và là các mẫu của một phân phối xác suất hay không .

Như vậy kiểm tra tính phù hợp có thể được dùng để kiểm định giả thuyết không sau đây :

H_0 : Tập dữ liệu X_i là các dữ liệu ngẫu nhiên độc lập có hàm phân phối F.

Giả thuyết H_0 là đúng nếu phân phối lý thuyết phù hợp với các dữ liệu đầu vào quan sát được.

Có hai phương pháp thường dùng để kiểm tra tính phù hợp là phương pháp Pearson Chi- bình phương χ^2 (*Chi -Square Test*) và phương pháp Kolmogorov-Smirnov. Hạn chế của phương pháp Kolmogorov-Smirnov là chỉ dùng cho phân phối liên tục vì vậy sau đây chúng ta sẽ nghiên cứu phương pháp Chi - bình phương χ^2

Phương pháp Chi - bình phương χ^2

Để tính χ^2 cho phân phối liên tục cũng như gián đoạn người ta chia tập dữ liệu ra thành từng khoảng $(a_0,a_1), (a_1,a_2),\ldots(a_{K-1}, a_K)$. Gọi N_j = số sự kiện xảy ra trong khoảng thứ j (a_{j-1}, a_j) với $j = 1, 2\ldots k$ (chú ý rằng $\sum_{j=1}^{k} N_j = n$, trong đó n là tổng các dữ liệu quan sát được).

Trên bảng 6-3 ta có $n = 120$, số khoảng chia k =10, N_j là số các dữ liệu quan sát được trong khoảng thứ j tức là các giá trị x_i trong cột " số dữ liệu quan sát được" trong bảng 6-3. Tiếp đến xác định xác suất xuất hiện dữ liệu p_j của phân phối lý thuyết trong khoảng thứ j.

Đối với trường hợp liên tục

$$p_J = \int_{a_{j-1}}^{a_i} f(x)dx$$

trong đó: $f(x)$ là hàm mật độ phân phối lý thuyết.

Đối với trường hợp gián đoạn

$$p_J = \sum \hat{p}(x_i)$$

$$a_{j-1} \le x_i < a_j$$

trong đó: $\hat{p}(x_i)$ là hàm trọng lượng của phân phối lý thuyết.

Vậy Chi - bình phương sẽ được tính như sau

$$\chi^2 = \sum_{j=1}^{k} \frac{(N_j - np_j)^2}{np_j}$$

trong đó: N_j là số các dữ liệu quan sát được trong khoảng thứ j;

np_j là số các dữ liệu nằm trong khoảng thứ j với giả thiết rằng các dữ liệu có phân phối theo phân phối lý thuyết, có nghĩa là giả thuyết H_0 là đúng.

Về thực chất χ^2 là tổng bình phương sai số giữa các dữ liệu quan sát được và dữ liệu của phân phối lý thuyết. Vì vậy nếu χ^2 càng nhỏ thì phân phối lý thuyết đã chọn càng gần với các dữ liệu đầu vào quan sát được.

Để thuận tiện cho việc kiểm định người ta đưa ra giá trị tới hạn $\chi^2_{\text{tới hạn}}$. Nếu các dữ liệu đầu vào (ví dụ bảng 6.1) là các số ngẫu nhiên độc lập thì χ^2 sẽ có giá trị nhỏ hơn một giá trị tới hạn $\chi^2_{\text{tới hạn}}$ xác định trước (giá trị $\chi^2_{\text{tới hạn}}$ cho trước trong bảng, ví dụ bảng T2 của [1]). Nếu ngược lại, $\chi^2 > \chi^2_{\text{tới hạn}}$ thì phải chọn lại thông số của hàm mật độ phân phối hoặc thậm chí phải chọn lại dạng của hàm phân phối lý thuyết. Nếu việc chọn lại cũng không đạt yêu cầu, điều đó chứng tỏ dữ liệu đầu vào không phải là những số ngẫu nhiên độc lập. Các dữ liệu này phải vứt bỏ, và người ta thu thập dữ liệu đầu vào mới. Bảng 6-4 ghi các số liệu tính toán về χ^2 cho ví dụ nêu trên.

Để tìm giá trị tới hạn của χ^2 cần xác định hai thông số là số bậc tự do ν của χ^2 và xác suất để χ^2 nhỏ hơn hoặc bằng giá trị χ^2 tới hạn, $\gamma = P(\chi^2 \le \chi^2_{\text{tới hạn}})$. Số bậc tự do được tính như sau $\nu = k - r - 1$, trong đó k- số khoảng khảo sát, r- số thông số của phân phối lý thuyết. Trong ví dụ trên số bậc tự do bằng

$\nu = 10 - 1 - 1 = 8$ và với $\gamma = 0{,}950$ tra bảng T2 của [1] ta có giá trị tới hạn $\chi^2_{8,\,0{,}950} = 15{,}507$. Giá trị này lớn hơn giá trị $\chi^2 = 11{,}18$ tính ở bảng trên. Từ đó rút ra kết luận hàm mũ $f(x) = e^{-0{,}2\,X}$ phù hợp với dữ liệu đầu vào đã thu thập được.

Bảng 6-4. Tính χ^2

Khoảng dữ liệu (a_{J-1}, a_J)	$p_J = e^{-0{,}2\,(AJ-1-AJ)}$	np_J	N_J	$\frac{(N_j - np_j)^2}{np_j}$
0 - 2	0,33	39,6	34	0,79
2 - 4	0,22	26,4	27	0,01
4 - 6	0,15	18	19	0,06
6 - 8	0,10	12,0	15	0,75
8 - 10	0,07	8,4	9	0,04
10 - 12	0,04	4,8	4	0,13
12 - 14	0,03	3,6	3	0,10
14 - 16	0,02	2,4	2	0,07
16 - 18	0,01	1,2	4	6,53
18 - 20	0,01	1,2	3	2,7
$\chi^2 = \sum_{j=1}^{10} \frac{(N_j - np_j)^2}{np_j} = 11{,}18$				

6.5. Mô hình dòng đầu vào (Models of Arrival Processes)

Trong nhiều bài toán mô phỏng chúng ta cần tạo ra dãy dữ liệu ngẫu nhiên theo thời gian $0 = t_1 \le t_2 \le t_3$ ... sao cho sự kiện thứ i xảy ra tại thời điểm t_i ($i = 1, 2, ...$) có phân bố xác định nào đó. Người ta gọi quá trình ngẫu nhiên $\{N(t), t >= 0\}$ là *quá trình đầu vào (arrival process)* và $A_i = t_i - t_{i-1}$ là quảng thời gian giữa sự kiện thứ i và thứ $i-1$ (*interarrival time*). Nếu các sự kiện là ngẫu nhiên và độc lập thì dòng đầu vào là một *dòng Poatxông* (*Poisson process*). Dòng Poatxông thường được dùng để mô tả dòng khách hàng đi đến cơ sở dịch vụ trong mô hình hàng đợi (xem Chương 10).

Một dòng khách hàng (sự kiện ngẫu nhiên) $\{N(t), t \ge 0\}$ được gọi là một dòng Poatxông nếu có các tính chất sau:

- Tại một thời điểm chỉ có một khách hàng xuất hiện (sự kiện đơn)

- $N(t+s) - N(t)$ (là số khách hàng trong quãng thời gian $(t, t+s)$) là độc lập đối với $\{N(u), 0 \leq u \leq t\}$ (sự kiện độc lập)

- Phân bố của $N(t+s) - N(t)$ là độc lập đối với t với mọi $t, s \geq 0$ (dòng dừng)

Đối với dòng *Poatxông*, khoảng thời gian giữa các sự kiện tuân theo luật phân phối mũ với thông số λ (λ là cường độ dòng khách hàng đi đến hệ thống).

Tính chất 1 và 2 là đặc tính của nhiều dòng đầu vào thường gặp trong mô hình hàng đợi. Trong khi đó tính chất thứ 3 không phải lúc nào cũng đạt được vì số khách hàng đến hệ thống trên một đơn vị thời gian có thể thay đổi theo thời gian. Lúc này chúng ta có dòng Poatxông không dừng (*Nonstationary Poisson Processes)* với đặc điểm của dòng này là cường độ dòng khách hàng $\lambda(t) \neq const$. Thông thường trong mô phỏng để đơn giản hoá người ta thường coi dòng đầu vào là dòng dừng, tức là đảm bảo tính chất thứ 3 và $\lambda(t) = const$.

Bài tập

1. Hãy trình bày vai trò của biểu đồ tần số trong việc tìm phân phối của dữ liệu đầu vào.

2. Tại sao các dữ liệu đầu vào (các quan sát) phải là những dữ liệu độc lập.

3. Hãy trình bày các bước cơ bản để tìm phân phối xác suất của dữ liệu đầu vào.

Chương 7
KIỂM CHỨNG VÀ HỢP THỨC HOÁ MÔ HÌNH

7.1. Khái niệm chung

Một trong những vấn đề khó khăn nhất mà mô phỏng phải đối mặt là phải xác định xem mô hình mô phỏng có phản ánh đúng hệ thống thực hay không? Nói một cách khác đó là vấn đề kiểm chứng và hợp thức hoá mô hình *(Verification and Validation of Simulation Model).*

Kiểm chứng là xác định xem liệu mô hình mô phỏng tức mô hình trên máy tính có phù hợp với logic của thuật toán không, trong chương trình có sai sót và chương trình có chạy được không?

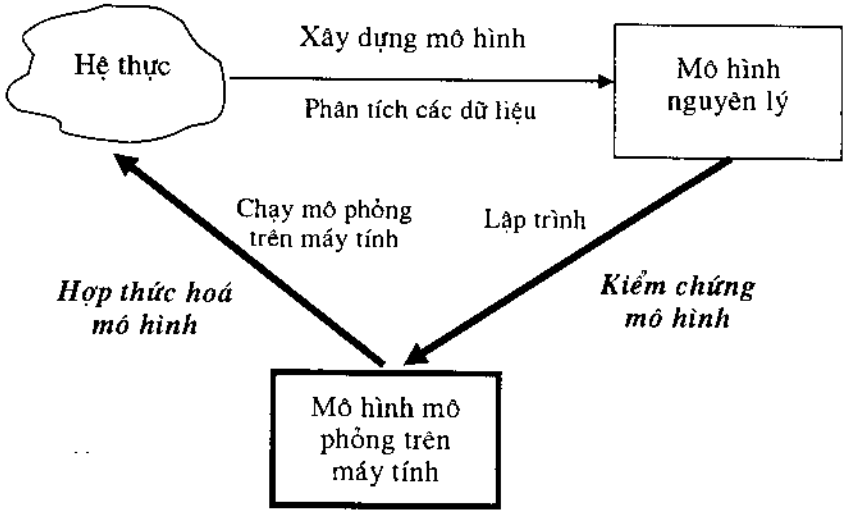
Hợp thức hoá mô hình là xác định xem liệu mô hình nguyên lý *(Conceptual model)* hay còn gọi là mô hình trừu tượng *(Abstraction model)* có phản ánh đúng bản chất của hệ thống thực hay không?

Chương này được dành để trình bày các vấn đề về kiểm chứng và hợp thức hoá mô hình. Mục 7.2 trình bày vai trò của kiểm chứng và hợp thức hoá mô hình trong mô phỏng. Mục 7.3 trình bày phương pháp kiểm chứng. Mục 7.4 trình bày phương pháp hợp thức hoá mô hình .

7.2. Vai trò của kiểm chứng và hợp thức hoá mô hình trong mô phỏng

Quá trình mô phỏng phải qua nhiều bước như đã trình bày ở mục 3.3 và 3.8 Chương 3. Trước tiên, người ta phân tích các dữ liệu thu thập được từ hệ thống thực để xây dựng mô hình nguyên lý. Mô hình nguyên lý là mô hình toán học phản ánh bản chất của hệ thống thực (xem mục 3.4). Bước thứ hai, dựa trên mô hình nguyên lý người ta xây dựng mô hình mô phỏng tức mô hình trên máy tính. Chạy mô hình mô phỏng người ta nhận được các đặc tính phản ánh hệ thống thực. Qua mỗi bước người ta phải kiểm chứng xem mô hình có được xây dựng đúng không. Cuối cùng phải xác định mô hình có hợp thức không, tức là mô hình có phản ánh đúng bản chất của hệ thống thực hay không.

Hình 7.1 trình bày quan hệ giữa các quá trình xây dựng mô hình nguyên lý và mô hình mô phỏng.



Hình 7.1. Quan hệ giữa các quá trình xây dựng mô hình mô phỏng.

Kiểm chứng và hợp thức là hai thủ tục quan trọng của quá trình mô phỏng. Bảng so sánh dưới đây phân tích bản chất của hai thủ tục nói trên.

Bảng 7-1

Kiểm chứng	***Hợp thức***
• Mô hình có làm việc đúng không	• Mô hình về nguyên lý có đúng không
• Phát hiện các sai sót trong khi lập trình	• Phát hiện chỗ không phù hợp giữa mô hình và hệ thống thực
• Sửa lỗi, gỡ rối	• Thảo luận về bản chất của mô hình
• Hạn chế trong phạm vi của mô hình	• Quan hệ với hệ thống thực
• Chỉ liên quan tới người xây dựng mô hình	• Quan hệ với toàn bộ quá trình mô phỏng
• Dễ thực hiện nhưng đòi hỏi phải có nhiều thời gian	• Rất khó thực hiện và phụ thuộc vào chủ quan của người nghiên cứu

Để kiểm chứng và hợp thức hoá mô hình chúng ta phải xây dựng các chỉ tiêu đánh giá. Không có chỉ tiêu vạn năng cho mọi bài toán mô phỏng nhưng các chỉ tiêu đánh giá phải phản ánh mục tiêu mô phỏng, phản ánh các câu hỏi

đặt ra khi tiến hành mô phỏng. Nói chung có ba câu hỏi có thể đặt ra để kiểm chứng mô hình mô phỏng như sau:

+ Quá trình xảy ra các sự kiện trong mô hình có đúng không?

+ Các công thức toán học và quan hệ giữa chúng có đúng không?

+ Các số liệu thống kê về đặc tính của hệ thống có đúng không?

Điều quan trọng là phải sớm xây dựng các chỉ tiêu đánh giá để kiểm chứng và hợp thức hoá mô hình ngay từ khi mới bắt đầu quá trình mô phỏng. Đồng thời việc kiểm chứng và hợp thức hoá mô hình được thực hiện trong suốt các giai đoạn của quá trình phát triển mô hình mô phỏng.

Trong giai đoạn xây dựng mô hình nguyên lý người ta phải xác định các phần tử chủ yếu của hệ thống, các sự kiện, các biến trong và ngoài hệ thống. Những yếu tố này phải được đưa vào mô hình và phải kiểm chứng xem liệu mô hình có thể trả lời các câu hỏi đề ra khi xác định mục tiêu mô phỏng hay không. Cần phải xác định xem mô hình có phản ánh đúng bản chất của hệ thực có nghĩa là mô hình có hợp thức hay không.

Sau khi mô hình nguyên lý đã được hợp thức người ta bước sang giai đoạn xây dựng mô hình logic (lưu đồ thuật toán) và mô hình mô phỏng (chương trình tính trên máy tính). Trong lưu đồ phải phản ánh logic của mô hình nguyên lý, các biến, các sự kiện xảy ra trong hệ thống đều phải được đưa vào lưu đồ. Sau khi có lưu đồ hợp lý người ta chuyển sang xây dựng mô hình mô phỏng có nghĩa là lập trình bằng ngôn ngữ đã chọn. Đây là giai đoạn đòi hỏi có kỹ năng lập trình tốt. Chương trình mô phỏng trên máy tính không những phải đúng về cú pháp để có thể chạy được mà chương trình phải được tổ chức tốt, sử dụng bộ nhớ hợp lý, tiết kiệm, tổ chức vào ra dữ liệu thuận tiện và có thời gian tính càng ngắn càng tốt. Nói chung chương trình phải thân thiện với người sử dụng. Trong giai đoạn này phải kiểm tra từ các câu câu lệnh, các chương trình con đến toàn bộ chương trình. Đây chính là thực hiện việc kiểm chứng chương trình. sau khi chương trình chạy tốt phải xác định xem kết quả mô phỏng có phản ánh đúng bản chất của hệ thực hay không có nghĩa là mô hình mô phỏng có hợp thức hay không. Như vậy kiểm chứng và hợp thức là hai thủ tục quan trọng phải được thực hiện xuyên suốt quá trình xây dựng và phát triển mô hình mô phỏng.

Quá trình thực hiện thủ tục kiểm chứng và hợp thức mô hình có thể tóm tắt trong bảng 7-2 sau đây:

Bảng 7-2. Các câu hỏi cho kiểm chứng và hợp thức mô hình

Loại mô hình	*Kiểm chứng mô hình*	*Hợp thức mô hình*
Mô hình nguyên lý		• Trong mô hình có bao gồm các phần tử, sự kiện và các mối liên hệ chủ yếu không? • Kết quả mô phỏng có trả lời được các câu hỏi đặt ra khi xác định mục tiêu mô phỏng hay không?
Mô hình logic (lưu đồ thuật toán)	• Các sự kiện có được biểu diễn đúng không? • Các quan hệ và công thức toán học có đúng không?	• Trong mô hình có bao gồm các sự kiện có trong mô hình nguyên lý không? • Trong mô hình có bao gồm các quan hệ có trong mô hình nguyên lý không?
Mô hình mô phỏng (Chương trình tính)	• Trong chương trình mô phỏng có bao gồm các thành phần chính của mô hình logic hay không? • Các công thức toán học có đúng không? • Trong chương trình có lỗi về lập trình hay không?	• Chương trình mô phỏng có phản ánh bản chất của hệ thực hay không? • Kết quả của chương trình mô phỏng có trùng với đặc tính của hệ thực hay không? • Các chuyên gia có đánh giá rằng mô hình mô phỏng phản ánh đúng bản chất của hệ thực

7.3. Phương pháp kiểm chứng mô hình

Trước khi viết chương trình mô phỏng trên máy tính người ta thường phải xây dựng lưu đồ tính toán hay còn gọi là mô hình logic. Lưu đồ là cầu nối giữa mô hình nguyên lý và mô hình mô phỏng. Lưu đồ phản ánh logic giữa các bước tính tức phản ánh thuật toán mô phỏng. Dựa vào lưu đồ người ta xây dựng chương trình mô phỏng.

Khi xây dựng mô hình mô phỏng trên máy tính thường gặp các dạng sai sót sau.

• Sai về dữ liệu: về mã hoá, về cú pháp trong lập trình;

- Sai về điều kiện đầu : điều kiện khởi động, giá trị của các biến ở t = 0;
- Sai về đơn vị đo : thời gian (phút, giây) khoảng cách (mét);
- Sai về lưu đồ hoạt động của mô hình;
- Tắc nghẽn, đình trệ quá trình mô phỏng;
- Sai về thuật toán số học: sai về cú pháp, sai về thuật toán;
- Sai về các biến và tính chất;
- Sai về nguyên lý của mô hình.

Kiểm chứng là quá trình kiểm tra và phát hiện các sai sót nêu trên để hiệu chỉnh lại chương trình mô phỏng.

Sau đây trình bày một số phương pháp để kiểm chứng mô hình.

7.3.1. Kiểm chứng các công thức và quan hệ lôgic trong mô hình

Chúng ta biết rằng trong chương trình mô phỏng (chương trình máy tính) luôn chứa các công thức, các hàm toán học và các quan hệ lôgic dưới dạng tường minh hoặc dạng ẩn. Cần phải tiến hành kiểm tra cẩn thận và phát hiện các sai sót trong cách biểu diễn các công thức và các hàm toán học cũng như trong việc lập trình. Việc kiểm tra này không nên chỉ do một người làm mà nên giao cho nhiều người cùng kiểm tra để dễ dàng phát hiện sai sót trong chương trình máy tính.

7.3.2. Chương trình gỡ rối

Khi viết chương trình mô phỏng trên máy tính người ta thường viết chương trình con dùng để gỡ rối, tìm sai sót trong quá trình chạy chương trình. Chương trình gỡ rối thường được dùng cho các chương trình lớn có hàng nghìn câu lệnh trở lên, độ phức tạp tương đối cao. Các ngôn ngữ mô phỏng chuyên dụng thường có sẵn chương trình gỡ rối.

7.3.3. Chạy thử chương trình mô phỏng

Chạy chương trình mô phỏng với các dữ liệu đầu vào khác nhau và kiểm tra xem kết quả đầu ra có hợp lý không. Nếu kết quả đầu ra là hợp lý hoặc có xu hướng tiệm cận tới giá trị hợp lý (do số lần chạy thử bị hạn chế) thì có thể coi chương trình mô phỏng là đúng. Trong một số trường hợp đơn giản, giá trị dữ liệu đầu ra có thể tính chính xác trước được, có thể dùng dữ liệu này để và so sánh với kết quả mô phỏng nhằm kết luận về tính đúng đắn của chương trình mô phỏng.

7.3.4. Phương pháp tìm "vết mô phỏng"

Một trong những phương pháp hữu hiệu để kiểm chứng chương trình mô phỏng các sự kiện gián đoạn là phương pháp tìm "*vết mô phỏng*" (*Simulation Trace*).

"*Vết mô phỏng*" thể hiện trạng thái của hệ thống được mô phỏng như: danh sách các sự kiện, giá trị các biến, các số liệu thống kê v.v... Cứ sau mỗi lần xuất hiện một sự kiện (*Event*) thì "*vết mô phỏng*" được in ra. Từ đó ta có thể so sánh với các dữ liệu được tính trước bằng tay, nếu kết quả khác với mong đợi người ta tìm các sai sót trong chương trình, sửa chữa, bổ sung để chương trình mô phỏng chạy đúng hướng. Chú ý rằng có thể có những sai sót không thể hiện ra khi chỉ chạy thử mô hình với chiều dài mô phỏng tương đối ngắn. Các ngôn ngữ mô phỏng chuyên dụng đều có chương trình con để in ra "*vết mô phỏng*" giúp cho người sử dụng dễ dàng kiểm chứng tính đúng đắn của mô hình. Phương pháp đơn giản thường dùng là in ra trạng thái ban đầu của các hàm hoặc chương trình con quan trọng. Bằng cách đó người ta có thể theo dõi diễn biến của các hàm và chương trình con trong quá trình mô phỏng.

7.3.5. So sánh các đặc trưng thống kê

Chạy một số lần mô phỏng với dữ liệu đầu vào ngẫu nhiên. Ứng với mỗi lần như vậy tính trung bình mẫu và phương sai mẫu của dữ liệu đầu ra và so sánh với giá trị trung bình và phương sai mong muốn.

7.3.6. Phương pháp hoạt hình

Một số mô hình mô phỏng có thể chạy dưới dạng hoạt hình nên có thể dễ dàng kiểm tra logic hoạt động của một số phần tử trong cũng như toàn bộ mô hình.

7.4. Phương pháp hợp thức hoá mô hình mô phỏng

Mô hình mô phỏng của một hệ thống thực luôn luôn chỉ là một mô hình gần đúng, phản ánh những đặc tính cơ bản của hệ thống thực. Vì vậy không cần thiết phải có sự tương ứng một - một giữa các phần tử của hệ thống thực và mô hình, do đó không có khái niệm hợp thức tuyệt đối.

Một mô hình mô phỏng được dùng cho một mục đích nghiên cứu nhất định. Vì vậy mô hình hợp thức đối với mục đích mô phỏng này nhưng có thể không hợp thức với mục đích mô phỏng khác. Có nghĩa là thủ tục hợp thức phụ thuộc

vào hệ thực được mô phỏng và môi trường mô hình hoá. Ví dụ thủ tục hợp thức mô hình của hệ máy tính có thể rất khác với thủ tục hợp thức mô hình của hệ thống sản xuất.

Quá trình hợp thức khó hay dễ phụ thuộc rất nhiều vào độ phức tạp của hệ thực.

Các vấn đề cần được hợp thức là:

- Hợp thức về nguyên lý của mô hình.

Cần phải xác định liệu mô hình nguyên lý có phản ánh đúng bản chất của hệ thống thực không?

- Hợp thức về thuật toán

Thuật toán dùng trong mô hình có phản ánh lôgic trong hệ thống thực không?

- Hợp thức về độ tin cậy

Liệu kết quả của mô hình có đáng tin cậy đối với người sử dụng mô hình hay không?

Mô hình có khả năng hợp thức cao là mô hình sử dụng được các thông tin sẵn có, bao gồm:

- Đối thoại với chuyên gia của hệ thống. Mô hình mô phỏng được xây dựng bởi các thông tin gắn liền với hệ thống thực. Vì vậy khi xây dựng mô hình mô phỏng cần phải đối thoại với các chuyên gia hiểu biết rõ về hệ thống thực để thu nhận được các thông tin cần thiết cho việc xây dựng mô hình.

- Quan sát hệ thống hiện có

Nếu hệ thống cần mô phỏng giống một hệ thống thực hiện có thì những dữ liệu của hệ thống thực được dùng để xây dựng mô hình mô phỏng.

- Dựa vào lý thuyết hiện có

Có thể sử dụng những lý thuyết hiện có về các hệ thống mô phỏng để làm căn cứ cho việc xây dựng mô hình mô phỏng. Ví dụ: theo lý thuyết khoảng cách giữa các sự kiện gián đoạn, độc lập và ngẫu nhiên tuân theo luật phân bố mũ. Vì vậy khi mô phỏng các dòng đầu vào có đặc tính như trên thì ta có thể dùng luật phân bố mũ để mô tả khoảng cách giữa các sự kiện - cách làm như vậy là hợp thức vì mô hình phù hợp với hệ thống thực.

Hiện nay có nhiều phương pháp hợp thức hoá mô hình như [1, 3]:

Sau đây sẽ trình bày một số phương pháp thông dụng.

7.4.1. Kiểm tra giả thiết của mô hình

Mục đích của phương pháp này là kiểm tra các giả thiết khi xây dựng mô hình nguyên lý

- Kiểm tra tính phù hợp giữa phân bố xác suất lý thuyết với dữ liệu đầu vào
- Phân tích độ nhạy của dữ liệu mô phỏng đầu ra khi

+ Thay đổi giá trị của các thông số

+ Thay đổi phân bố xác suất của dữ liệu đầu vào

+ Thay đổi mức độ chi tiết của các hệ thống con trong mô hình

Phân tích độ nhạy là phương pháp hữu hiệu để đánh giá mức độ hợp thức của mô hình mô phỏng.

Nếu dữ liệu mô phỏng đầu ra nhạy cảm với một yếu tố nào của mô hình thì phải thay đổi yếu tố đó một cách thận trọng.

Ví dụ : khi tiến hành mô phỏng một hệ thống mới, qua đối thoại với "chuyên gia" hệ thống (xem bước 1) giá trị một thông số quan trọng của mô hình được xác định bằng 0,75. Để xác định độ nhạy của dữ liệu đầu ra đối với thông số này người ta chạy ba lần mô phỏng với giá trị của thông số là 0,75, 0,7 và 0,8. Nếu kết quả của ba lần mô phỏng đều như nhau chứng tỏ dữ liệu đầu ra không nhạy cảm đối với thông số đã chọn trong khoảng 0,7 đến 0,8. Như vậy giá trị đã chọn của thông số trên là đúng và được hợp thức hoá.

7.4.2. Phương pháp đánh giá dữ liệu đầu ra của mô phỏng

Một trong những phương pháp tốt nhất để hợp thức hoá mô hình là đánh giá mức độ phù hợp giữa dữ liệu mô phỏng với dữ liệu đầu ra của hệ thống thực. Nếu hai dữ liệu này là phù hợp với nhau thì mô hình mô phỏng được coi là đúng, tức được hợp thức hoá. Trong trường hợp mô phỏng một hệ chưa có thực thì có thể lấy dữ liệu đầu ra của hệ thống thực tương tự để so sánh với dữ liệu đầu ra của mô hình mô phỏng.

So sánh giữa mô hình và hệ thực là một phép so sánh thống kê nên sự khác nhau giữa các dữ liệu thu được từ mô hình và từ hệ thực phải được xử lý theo phương pháp thống kê xác suất. Các dữ liệu thu được từ mô hình thường do mô phỏng tương ứng thời gian dài để có kết quả ổn định, chẳng hạn hàng tháng

hàng năm. Trong lúc đó dữ liệu thu được từ hệ thực lại thu nhận trong khoảng thời gian ngắn như hàng ngày hàng giờ. Vì vậy cần phải xử lý khi dung hai loại dữ liệu này để so sánh.

Một vấn đề khác cũng cần được lưu ý là trong hệ thực có nhiều yếu tố ngẫu nhiên ảnh hưởng đến dữ liệu mà chúng ta quan tâm, tuy nhiên nhận dạng các yếu tố ngẫu nhiên đó thường rất khó khăn.

Trường hợp hệ thực chưa có thì việc thu nhận dữ liệu từ hệ thực là điều không thực hiện được. Trong trường hợp này người ta có thể dùng một hệ thực tương tự để lấy dữ liệu so sánh với dữ liệu mô phỏng. Tuy nhiên việc so sánh này chỉ có tính chất tham khảo, người ta cần thực hiện các phương pháp đánh giá khác để có thêm căn cứ cho việc hợp thức mô hình nguyên lý.

7.4.3. Phương pháp chuyên gia

Ý kiến của các chuyên gia có kinh nghiệm làm việc trong lĩnh vực mô phỏng hệ thống luôn đóng vai trò quan trọng trong quá trình kiểm chứng và hợp thức hoá mô hình. Theo phương pháp chuyên gia Delphi [3] người ta tổ chức nhóm làm việc để kiểm chứng và hợp thức hoá mô hình bao gồm các chuyên gia về sử dụng và điều hành hệ thống thực. Người ta gửi cho nhóm chuyên gia bảng điều tra gồm các câu hỏi về hành vi và khả năng của hệ thống thực trong các điều kiện làm việc đặc biệt.

Thông thường có ba bước điều tra.

- Đầu tiên người ta gửi cho các chuyên gia bảng điều tra gồm các câu hỏi về phản ứng của hệ thực khi dữ liệu đầu vào hoặc cấu trúc của hệ có thay đổi đặc biệt.

- Căn cứ vào trả lời ở bước 1, người ta xây dựng bảng điều tra thứ hai với các câu hỏi nhằm làm rõ hơn các phản ứng nói trên của hệ thực.

- Bảng điều tra thứ hai được gửi lại cho các chuyên gia để nhận được các dự đoán về hành vi của hệ khi dữ liệu đầu vào hoặc cấu trúc của hệ có thay đổi.

Ba bước trên được lặp lại vài lần cho đến khi nhận được các đánh giá tương đối chính xác về hành vi của hệ thực. Người ta dùng các dữ liệu này để so sánh với các dữ liệu thu được từ kết quả mô phỏng. Căn cứ vào kết luận của các chuyên gia người ta có thể kiểm chứng tính chính xác và mức độ hợp thức của mô hình.

Nhược điểm của phương pháp chuyên gia Delphi là đòi hỏi nhiều thời gian.

Bài tập

1. Để đánh giá tình trạng tắc nghẽn giao thông trong thành phố người ta xây dựng mô hình mô phỏng một số đường phố chính với hệ thống đèn tín hiệu giao thông.

a) Hãy xây dựng mô hình nguyên lý của hệ thống đèn tín hiệu giao thông.

b) Xây dựng lưu đồ điều khiển hệ thống đèn tín hiệu giao thông.

c) Kiểm chứng mô hình mô phỏng bằng cách nào?

d) Hợp thức hoá mô hình mô phỏng bằng cách nào?

2. Có thể phân loại mô hình hệ thống thành mô hình nguyên lý, mô hình vật lý, mô hình giải tích và mô hình mô phỏng. Hãy phân tích sự giống nhau và khác nhau khi kiểm chứng và hợp thức hoá các loại mô hình nêu trên.

3. Ngôn ngữ dùng để viết chương trình mô phỏng ảnh hưởng như thế nào đến quá trình kiểm chứng và hợp thức hoá mô hình?

Chương 8
XỬ LÝ VÀ PHÂN TÍCH CÁC DỮ LIỆU ĐẦU RA CỦA MÔ PHỎNG

8.1. Khái niệm chung

Như chúng ta đã biết trong mô hình mô phỏng thường có các biến ngẫu nhiên. Vì vậy, phần lớn nếu như không muốn nói là tất cả, các dữ liệu đầu ra của mô hình đều có tính ngẫu nhiên. Xử lý các dữ liệu đầu ra đó để đánh giá hành vi của hệ thống là một nhiệm vụ khó khăn. Có nhiều lý do khiến cho việc xử lý dữ liệu đầu ra của mô phỏng gặp nhiều khó khăn. Thứ nhất là do bước tính mô phỏng thường phải đủ lớn nên dữ liệu đầu ra nhiều. Thứ hai là các dữ liệu đầu ra mang tính ngẫu nhiên nên phải dùng phương pháp xác suất thống kê để xử lý. Do tính phức tạp của phương pháp xử lý nên rất dễ xảy ra sai sót và do đó các kết luận rút ra được không phản ánh đúng bản chất của hệ thực mà được mô tả bởi mô hình đang khảo sát. Xử lý và phân tích các dữ liệu đầu ra là bước cuối cùng của mô phỏng nhằm đạt được mục đích mô phỏng. Thông thường người ta dùng phương pháp thống kê để thu thập và xử lý dữ liệu đầu ra của mô phỏng. Việc phân tích cẩn thận và hợp lý dữ liệu đầu ra sẽ quyết định thành công của mô phỏng.

Mục 8.2 trình bày mục đích việc xử lý dữ liệu đầu ra mô phỏng. Mục 8.3 trình bày phương pháp đánh giá dữ liệu đầu ra. Do tính ngẫu nhiên của dữ liệu nên phương pháp thống kê được dùng để đánh giá dữ liệu đầu ra mô phỏng. Mục 8.4 được dành cho phân tích dữ liệu đầu ra mô phỏng của hệ giới hạn. Mục 8.5 được dành cho phân tích dữ liệu đầu ra mô phỏng của hệ không giới hạn. Cuối cùng mục 8.6 nói về sử dụng kết quả mô phỏng.

8.2. Mục đích của việc xử lý các dữ liệu đầu ra của mô phỏng

Khi xây dựng mô hình mô phỏng việc quan trọng đầu tiên là xác định mục đích mô phỏng, hay nói cách khác là xác định tập các câu hỏi mà mô phỏng phải trả lời. Mục đích của việc phân tích và xử lý dữ liệu đầu ra của mô phỏng

là nhằm trả lời một cách rõ ràng các câu hỏi đã được đề ra từ đầu. Để minh hoạ điều này ta lấy ví dụ về mô phỏng hoạt động của phòng giao dịch ngân hàng. Giả sử phòng giao dịch có nhiều nhân viên phục vụ và mở cửa từ 8 giờ sáng đến 4 giờ chiều hàng ngày. Vậy có thể có các câu hỏi sau đây được đặt ra khi mô phỏng hệ thống:

- Thời gian chờ đợi trung bình của khách hàng?
- Số phần trăm khách hàng phải chờ đợi khi đến giao dịch?
- Xác suất số khách hàng phải chờ đợi lớn hơn 10 phút?
- Trị số trung bình khách hàng đứng trong hàng đợi?
- Số nhân viên phục vụ tăng hay giảm ảnh hưởng như thế nào đến thời gian chờ đợi của khách hàng?
- Có bao nhiêu khách hàng tồn lại khi phòng giao dịch đóng cửa?

Nếu mô hình được xây dựng đúng thì các câu hỏi sẽ được trả lời đầy đủ. Rõ ràng rằng mỗi một câu hỏi cần có các dữ liệu khác nhau mà chúng ta phải thu thập khi mô phỏng và mỗi một dữ liệu có thể được xử lý bằng nhiều cách khác nhau. Hay nói một cách khác kết quả mô phỏng, tức câu trả lời, phụ thuộc vào phương pháp phân tích và xử lý dữ liệu đầu ra của mô phỏng.

Mục đích cuối cùng của việc phân tích và xử lý dữ liệu đầu ra của mô phỏng là thành lập báo cáo mô phỏng (*Simulation Report*).

Báo cáo mô phỏng gồm có:

- Các đặc trưng của biến ngẫu nhiên trong mô hình như kỳ vọng toán, giá trị max, min và phương sai của chúng.
- Các đánh giá về hành vi của hệ thực, tức là trả lời của các câu hỏi được đặt ra khi tiến hành mô phỏng.

Sau đây chúng ta sẽ phân tích kỹ hơn về bản chất ngẫu nhiên của dữ liệu đầu ra mô phỏng. Giả thiết $Y_1, Y_2 \ldots Y_i$ là dãy dữ liệu đầu ra của một lần chạy mô phỏng. Ví dụ Y_i có thể là số khách hàng có mặt tại ngân hàng ở giờ thứ i trong ví dụ phòng giao dịch ngân hàng nêu trên. Gọi $y_{11}, y_{12}, \ldots y_{1m}$ là thể hiện của biến ngẫu nhiên $Y_1, Y_2 \ldots Y_m$ khi chạy mô phỏng dùng số ngẫu nhiên u_1. Nếu ta dùng n số ngẫu nhiên $u_1, u_2 \ldots u_n$ để thực hiện n lần chạy mô phỏng có cùng điều kiện đầu thì sẽ thu được bộ giá trị quan sát được (đầu ra) của mô phỏng có chiều dài m là:

$y_{11}, y_{12}, \ldots y_{1m}$

$y_{21}, y_{22}, \ldots y_{2m}$

. . . . . .

$y_{n1}, y_{n2}, \ldots y_{nm}$

Nhiệm vụ chính của xử lý đầu ra mô phỏng là dùng phương pháp thống kê ước lượng điểm và ước lượng khoảng để xác định xem các quan sát y_{ij} $(i = 1, 2, \ldots n; j = 1, 2, \ldots m)$ có phải là độc lập với nhau và dùng các quan sát y_{ij} để tìm các đặc trưng của biến ngẫu nhiên $Y_1, Y_2 \ldots Y_m$ như kỳ vọng toán, phương sai, độ lệch chuẩn, khoảng tin cậy v.v...

8.3. Phương pháp đánh giá dữ liệu đầu ra

Các dữ liệu đầu ra của mô phỏng không hoàn toàn độc lập với nhau và mang tính ngẫu nhiên. Sau mỗi một lần thử nghiệm, tức là một lần chạy mô phỏng, ta thu được các giá trị "quan sát được" của một biến. Các giá trị này ở thời điểm quan sát được lại phụ thuộc vào giá trị của nó ở các thời điểm trước đó. Chính vì vậy phương pháp đánh giá thích hợp các dữ liệu đầu ra của mô phỏng là phương pháp xác suất thống kê. Phương pháp đánh giá phải đảm bảo độ tin cậy theo nghĩa là nếu bước mô phỏng được tăng lên thì các đánh giá sẽ tiến đến giá trị thực của biến.

Trên quan điểm của lý thuyết thống kê, dữ liệu đầu ra của mô phỏng là những biến ngẫu nhiên và được gọi là *mẫu (Sample)*. Mẫu được lấy từ một tập lớn hơn được gọi là *tập đám đông (Population)* hay còn gọi là *tập nền*. Như vậy tập mẫu và tập nền có cùng bản chất, có cùng phân phối xác suất. Để có đủ thông tin về mô hình và về hệ thực chúng ta phải nghiên cứu trên tập nền. Tuy nhiên vì tập nền thường tương đối lớn nên đòi hỏi nhiều chi phí vật chất và thời gian. Trong thực tế người ta chọn mẫu ngẫu nhiên với số lượng đủ lớn thì nghiên cứu trên tập mẫu cũng có thể cho ta kết quả đạt độ chính xác và tin cậy cần thiết.

Trong trường hợp chung, các dữ liệu đầu ra của mô phỏng (mẫu) không hoàn toàn độc lập với nhau. Tuy nhiên để đơn giản trong tính toán mà vẫn đảm bảo độ chính xác cần thiết chúng ta có thể chấp nhận các giả thiết sau đây [3, 17]:

- Các mẫu (dữ liệu) là biến ngẫu nhiên và độc lập thống kê với nhau.

- Quá trình khảo sát là quá trình dừng.

- Khi số lượng mẫu tương đối lớn (30 trở lên), trung bình mẫu (kỳ vọng toán) là biến ngẫu nhiên có phân phối chuẩn.

Khi xử lý dữ liệu đầu ra của mô phỏng người ta phân biệt hai loại hệ thống: *hệ thống giới hạn và không giới hạn (Terminating and Nonterminating Systems)* phụ thuộc vào việc có thể xác định một cách rõ ràng chiều dài mô phỏng hay không. Trong hệ thống giới hạn, có một sự kiện E mà khi sự kiện này xuất hiện thì làm cho hệ thống ngừng hoạt động. Điều đó có nghĩa là chiều dài mô phỏng được xác định bởi thời gian để sự kiện E xuất hiện, nói chung thời gian đó là một biến ngẫu nhiên. Ngược lại, trong hệ thống không giới hạn không có sự kiện E nào để xác định chiều dài mô phỏng. Việc phân biệt hai loại hệ thống rất quan trọng vì ứng với mỗi loại hệ thống khác nhau, phương pháp xử lý dữ liệu đầu ra của mô phỏng sẽ khác nhau.

Sau đây là một số ví dụ minh hoạ về hệ thống giới hạn và không giới hạn:

Hệ thống giới hạn:

Ví dụ 1: Giả sử một nhà máy có hợp đồng sản xuất 100 chiếc máy. Thời hạn giao hàng trong vòng 12 tháng. Lãnh đạo nhà máy muốn mô phỏng các phương án sản xuất khác nhau để tìm phương án giao hàng đúng hạn với giá thành thấp nhất. Đây là hệ thống giới hạn, sự kiện cho phép hệ thống ngừng hoạt động, hay nói cách khác điều kiện ngừng mô phỏng là $E = 100$ máy hoàn thành. Trong hệ thống này trạng thái của chu kỳ làm việc trước không ảnh hưởng đến trạnh thái của chu kỳ làm việc sau.

Ví dụ 2: Một ngân hàng làm việc từ 9 giờ sáng đến 5 giờ chiều. Người ta thực hiện mô phỏng một ngày làm việc của ngân hàng để đánh giá khả năng phục vụ khách hàng của ngân hàng bắt đầu từ 9 giờ sáng cho đến khi khách hàng cuối cùng được phục vụ xong trước giờ đóng cửa là 5 giờ chiều. Trong trường hợp này sự kiện E = (kết thúc 8 giờ mô phỏng và hệ thống rỗng). Điều kiện đầu của mô phỏng là số khách hàng tại thời điểm $t = 0$.

Hệ thống không giới hạn:

Ví dụ 3: Một phân xưởng có một số máy gia công. Chi tiết máy được đưa đến phân xưởng phải đi qua tất cả các máy gia công mới hoàn thành và đi ra khỏi phân xưởng. Giả sử phân xưởng làm việc hai ca và 6 ngày trong một tuần. Đây là hệ thống thuộc loại không giới hạn. Bởi vì trạng thái của phân xưởng khi

ngừng làm việc, tức cuối ca thứ hai của ngày thứ bảy, sẽ là trạng thái bắt đầu làm việc của ca thứ nhất của ngày thứ hai tuần kế tiếp. Có nghĩa là trạng thái của chu kỳ làm việc trước ảnh hưởng trực tiếp đến trạng thái của chu kỳ làm việc sau.Trong hệ thống này không có sự kiện E để xác định thời điểm ngừng làm việc của hệ thống, tức là không có sự kiện để xác định thời gian ngừng mô phỏng. Đối với hệ thống này người ta mô phỏng cho đến khi dữ liệu đầu ra đạt tới giá trị ổn định nào đó.

Ví dụ 4: Một công ty chuẩn bị xây dựng nhà máy mới và muốn xác định năng suất trung bình tính trên giờ khi nhà máy làm việc ổn định. Giả thiết rằng nhà máy làm việc liên tục 16 giờ một ngày (làm việc hai ca) và 5 ngày một tuần đồng thời bỏ qua phế phẩm có thể sinh ra khi bắt đầu và kết thúc một ca sản xuất. Gọi N_i là số sản phẩm được sản xuất ra trong giờ thứ i. Nếu qúa trình ngẫu nhiên N_1, N_2, N_3 ... có phân bố ổn định ứng với biến ngẫu nhiên N thì chúng ta có thể tìm được giá trị trung bình của nó $E(N)$. Trong hệ thống này không có sự kiện E để xác định thời điểm ngừng mô phỏng mà người ta mô phỏng cho đến khi năng suất trung bình của nhà máy đạt được giá trị mong muốn $E(N)$ nào đó.

8.4. Phân tích dữ liệu đầu ra mô phỏng của hệ giới hạn

Giả thiết rằng chúng ta thực hiện N lần chạy mô phỏng (*Replication*) tức là thực hiện N phép thử độc lập để xác định các giá trị mẫu. Gọi x_i. là biến ngẫu nhiên chỉ giá trị của mẫu thu được ở phép thử thứ i, $i = \overline{1, N}$. Như vậy x_i. là giá trị quan sát hay *thể hiện* của mẫu. Tập mẫu ngẫu nhiên $\{x_1, x_2, x_3, ... x_i, ... x_N\}$ đại diện cho tập nền là toàn bộ dữ liệu đầu ra của mô phỏng. Vì vậy nghiên cứu trên tập mẫu đủ đánh giá về tập nền.

Đặc trưng quan trọng nhất của dữ liệu đầu ra mô phỏng là:

- *Trung bình (kỳ vọng) mẫu E (Sample Mean)*
- *Phương sai mẫu S (Sample Variance).*
- *Độ lệch chuẩn mẫu σ (Standard Deviation)*
- *Hệ số tương quan k_{xy} (Correlation coefficient)*
- *Khoảng tin cậy (Confidence interval)* của trung bình mẫu và phương sai mẫu.

Sau đây lần lượt trình bày phương pháp tính các đặc trưng nêu trên.

8.4.1. *Trung bình (kỳ vọng) mẫu EX*

Giả thiết rằng chúng ta thực hiện N lần chạy mô phỏng. Mỗi lần chạy mô phỏng đều có cùng điều kiện đầu và cùng kết thúc khi sự kiện E xuất hiện. Dữ liệu đầu ra của mô phỏng chính là tập mẫu ngẫu nhiên $\{ x_1, x_2, x_3, \ldots x_i \ldots x_N \}$. Khi tiến hành với số phép thử hữu hạn N ta chỉ có thể thu được đánh giá của kỳ vọng toán mà thôi.

Định nghĩa: Kỳ vọng của biến ngẫu nhiễn x, ký hiệu là EX và được xác định như sau:

$$EX = \sum_{i=1}^{N} x_i \, p_i \, ,$$

trong đó – $p(x_i) = p_i$ là xác suất xuất hiện biến ngẫu nhiên x_i.

Trong trường hợp đang khảo sát ở đây, xác suất xuất hiện biến ngẫu nhiên (mẫu) x_i là như nhau và bằng $1 / N$. Vì vậy kỳ vọng toán có thể viết [7,12]:

$$EX = \bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i \qquad (8 - 1)$$

Khi trung bình (kỳ vọng) tập nền θ không biết trước, chúng ta có thể dùng trung bình mẫu để đánh giá trung bình tập nền.

$$E[\bar{x}] = E\left[\frac{1}{N}\sum_{i=1}^{N} x_i\right]$$

$$= \sum_{i=1}^{N} \frac{E[x_i]}{N}$$

$$= \frac{N\theta}{N} = \theta$$

Như vậy kỳ vọng toán của trung bình mẫu chính là trung bình tập nền.

8.4.2. *Phương sai mẫu S^2*

Định nghĩa: Phương sai của biến ngẫu nhiên x, được ký hiệu là S^2, được tính như sau:

$$S^2 = E(x - EX)^2$$

Cũng như trường hợp trên, xác suất xuất hiện biến ngẫu nhiên (mẫu) x_i là như nhau và bằng $1 / N$. Vì vậy phương sai mẫu được tính theo công thức sau [7, 12]:

$$S^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2 , \qquad (8 - 2)$$

Từ (8 - 2) ta thấy rằng phương sai chính là trung bình của bình phương độ lệch của mẫu so với kỳ vọng của nó.

Tính toán theo (8 - 2) là khá phức tạp, vì vậy có thể dùng tính chất của kỳ vọng để biến đổi về dạng tương đương dễ tính toán hơn

$$S^2 = E\left(x^2\right) - \left(EX\right)^2$$

ứng với trường hợp ta đang khảo sát, có thể viết:

$$S^2 = \frac{1}{N}\left(\sum_{i=1}^{N} x_i^2 - N\bar{x}^2\right) \qquad (8 - 3)$$

Tính phương sai theo công thức (8-3) trên là không hợp lý bởi vì giá trị $\bar{x}$ thay đổi theo quá trình tích luỹ của x_i như vậy phải dùng nhiều ô nhớ của máy tính.

Thông thường có thể tính theo công thức sau [12]:

$$S_x^{\ 2} = \frac{1}{N}\left[\sum_{i=1}^{N} x_i^2 - \frac{1}{N}\left(\sum_{i=1}^{N} x_i\right)^2\right] \qquad (8.4)$$

Theo công thức (8-4) chỉ cần nhớ tổng của hai giá trị x_i và $x_i^{\ 2}$ mà thôi.

8.4.3. *Độ lệch chuẩn mẫu σ*

Để ý rằng phương sai S^2 luôn là số không âm và về mặt vật lý nó không cùng thứ nguyên (cùng đơn vị đo) đối với x, vì vậy người ta đưa vào khái niệm *độ lệch chuẩn (Standard Deviation).*

Định nghĩa: Độ lệch chuẩn của biến ngẫu nhiên x, ký hiệu là $\sigma(x)$, được tính như sau:

$$\sigma(x) = \sqrt{S^2} \qquad (8\text{-}5)$$

Độ lệch chuẩn được dùng thường xuyên hơn phương sai vì có cùng đơn vị đo với chính biến x.

8.4.4. *Hệ số tương quan k_{xy} (Correlation coefficient)*

Trong trường hợp cần đánh giá quan hệ giữa hai số ngẫu nhiên x_i, y_i thì người ta dùng hệ số tương quan và được tính như sau:

$$k_{xy} = \frac{1}{N}\left(\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})\right) \qquad (8 - 6)$$

hay thuận tiện hơn ta có thể dùng dạng sau

$$k_{xy} = \frac{1}{(N-1)}\left(\sum_{i=1}^{N} x_i y_i - \frac{1}{N}\sum_{i=1}^{N} x_i \sum_{i=1}^{N} y_i\right) \qquad (8 - 7)$$

Biểu thức (8 - 7) được tính khi chỉ cần nhớ hai giá trị x_i , y_i trong quá trình mô phỏng mà thôi nên rất thuận tiện và tốn ít bộ nhớ. Giá trị k_{xy} càng nhỏ thì các biến ngẫu nhiên x và y càng ít phụ thuộc vào nhau và ngược lại.

8.4.5. *Khoảng tin cậy (Confidence interval)*

Mỗi một mẫu x_i là một ước lượng điểm của dữ liệu đầu ra. Ước lượng điểm cho ta thông tin quan trọng về dữ liệu đầu ra, nhưng nó có nhược điểm là không thể biết được độ chính xác và xác suất để ước lượng độ chính xác đó. Nhất là khi kích thước mẫu nhỏ thì sự sai lệch của ước lượng so với giá trị thật khá lớn và khó đánh giá được khả năng mắc sai lầm khi ước lượng là bao nhiêu. Để khắc phục hạn chế đó người ta đưa ra khái niệm *khoảng tin cậy (Confidence interval)*. Khoảng tin cậy (h_1 , h_2) là một khoảng giá trị sao cho với xác suất cho trước ước lượng điểm sẽ rơi vào khoảng đó. Như vậy ước lượng khoảng được xây dựng xung quanh ước lượng điểm và cho phép xác định khách quan sai số ước lượng.

Định nghĩa: Cho tập mẫu $\{x_1, x_2, x_3, \ldots x_i, \ldots x_N\}$ với N đủ lớn, giả thiết có trung bình mẫu $\bar{x}$ và độ lệch chuẩn mẫu σ, với độ tin cậy 100 (1 - α) phần trăm ta có *khoảng tin cậy đối xứng* sau:

$$\bar{x} \pm z_{1-\alpha/2}\frac{\sigma}{\sqrt{N}}, \qquad (8 - 8)$$

trong đó: $z_{1-\alpha/2}$ là điểm tới hạn của biến ngẫu nhiên phân bố chuẩn tìm được bằng cách tra bảng T.1 của [1]

Ví dụ 5: Người ta muốn ước lượng thời gian trung bình *EX* để sản xuất một chi tiết máy. Giả sử lượng thời gian đó tuân theo luật phân bố chuẩn với độ lệch chuẩn là $\sigma = 0,3$ *phút*. Trên một tập mẫu gồm thời gian để sản xuất 36 chi tiết máy ta có trung bình mẫu $\bar{x} = 1,2$ phút / chi tiết máy. Hãy xác định khoảng tin cậy 95% cho thời gian trung bình *EX* nói trên.

Giải: Ta có $\bar{x} = 1,2$; $\sigma = 0,3$; $N = 36$; $\alpha = 1 - 95\% = 5\%$. Tra bảng T.1 của [1] tìm được $z_{1-\alpha/2} = 1,96$. Từ đó khoảng tin cậy đối xứng (8.8) được xác định:

$$\bar{x} - z_{1-\alpha/2}\frac{\sigma}{\sqrt{N}} < EX < \bar{x} + z_{1-\alpha/2}\frac{\sigma}{\sqrt{N}}$$

$$\left(1,2 - 1,96\frac{0,3}{\sqrt{36}}\,;\,1,2 + 1,96\frac{0,3}{\sqrt{36}}\right) = (1,102\,;1,298)$$

Kết luận: thời gian trung bình *EX* để sản xuất một chi tiết máy là 1,2 phút sẽ rơi vào khoảng (1,102 ; 1,298) với xác suất *95%*.

Ví dụ 6: Để đánh giá năng suất của một phân xưởng người ta thực hiện N = 10 lần chạy mô phỏng và thu được dữ liệu đầu ra như sau:

N	1	2	3	4	5	6	7	8	9	10
Năng suất, x_i	93	113	107	103	112	103	112	100	98	105

Hãy xác định khoảng tin cậy 95% cho năng suất trung bình *EX* nói trên.

Giải: Sử dụng bảng số liệu trên và ứng dụng công thức (8-1), (8-3), (8-5) ta tính được trung bình mẫu, phương sai mẫu và độ lệch chuẩn như sau

$$\bar{x} = 104,6\,; \qquad S^2 = 6,6\,; \qquad \sigma = 2,56$$

Dùng công thức (8-8) ta tính được khoảng tin cậy đối xứng 95% đối với trung bình mẫu tức là năng suất trung bình *EX* của phân xưởng.

$$\bar{x} - z_{1-\alpha/2}\frac{\sigma}{\sqrt{N}} < EX < \bar{x} + z_{1-\alpha/2}\frac{\sigma}{\sqrt{N}}$$

Tra bảng được $z_{1-\alpha/2} = 2,26$, từ đó tính được khoảng tin cậy đối xứng sau

$$104,6 - 2,26\frac{2,56}{\sqrt{10}} < EX < 104,6 + 2,26\frac{2,56}{\sqrt{10}} = (102,74\,;106,46)$$

Như vậy với xác suất 95%, năng suất trung bình *EX* của phân xưởng nằm trong khoảng ($102,47 < EX < 106,46$).

Đối với hệ giới hạn, *điều kiện đầu (initial conditions)* ảnh hưởng mạnh đến kết quả mô phỏng. Vì vậy cần phải chọn điều kiện đầu chính xác và thích hợp với mục đích mô phỏng. Nếu chúng ta quan tâm đến toàn bộ hành vi của hệ thống thì điều kiện đầu được chọn là tham số trạng thái của hệ thống tại thời

điểm $t = 0$. Nếu chúng ta chỉ quan tâm đến trạng thái ổn định của hệ thống thì điều kiện đầu được chọn là tham số trạng thái của hệ thống tại thời điểm kết thúc thời kỳ quá độ.

Ví dụ: Mô phỏng hoạt động của một ngân hàng. Hoạt động của ngân hàng có thể được mô hình hoá bằng mô hình hàng đợi *M/M/1*. Giả sử cường độ dòng khách hàng là $\lambda = 10$ *người / giờ*. Cường độ phục vụ $\mu = 12,5$ *người / giờ* (có nghĩa là thời gian phục vụ trunh bình là 4,8 phút / người). Ngân hàng làm việc 8 giờ một ngày. Hãy đánh giá thời gian đợi của khách hàng khi đến giao dịch tại ngân hàng.

Sau khi thực hiện 10 lần chạy mô phỏng (*Replications*) ta thu được các dữ liệu về thời gian đợi của khách hàng như trong bảng 8-1 [3].

Bảng 8-1. Thời gian đợi của khách hàng (phút)

Lần mô phỏng, i	x_{i1} Giờ thứ 1	x_{i2} Giờ thứ 2	x_{i3} Giờ thứ 3	x_{i4} Giờ thứ 4	x_{i5} Giờ thứ 5	x_{i6} Giờ thứ 6	x_{i7} Giờ thứ 7	x_{i8} Giờ thứ 8
1	1,27	8,65	8,43	0	0	3,54	1,27	27,16
2	7,06	0,82	0	0	24,06	0	15,81	35,3
3	0,09	0	19,01	4,74	44,12	40,69	68,65	47,65
4	7,21	2,35	0	0	2,36	0	0	2,51
5	7,84	0	23,87	0,85	0	2,61	6,58	0
6	7,84	3,23	26,02	29,96	35,58	0	0	0
7	0	0	6,51	0	0	3,54	1,27	27,16
8	22,84	12,75	16,78	0	12,75	0	0	4,49
9	34,07	38,26	0,58	13,02	0	0	0	1,37
10	0	0	13,25	0	19,46	2,89	0	0

Trong bảng 8-1 x_{ij} là trung bình mẫu tức là thời gian đợi trung bình của khách hàng ở giờ thứ $j = 1, 2,..., 8$ của lần chạy mô phỏng thứ $i = 1, 2,..., 10$.

Sử dụng dữ liệu của bảng 8-1 và các công thức (8-1), (8-5), (8-8) chúng ta có thể tính được ước lượng điểm là thời gian đợi trung bình $\overline{X}_j$, độ lệch chuẩn s_j và ước lượng khoảng tin cậy 95% của thời gian đợi của khách hàng, các ước lượng này cho trong bảng 8-2.

Bảng 8-2

	Giờ thứ 1	*Giờ thứ 2*	*Giờ thứ 3*	*Giờ thứ 4*	*Giờ thứ 5*	*Giờ thứ 6*	*Giờ thứ 7*	*Giờ thứ 8*
$\overline{X}_i$	7,34	4,95	8,57	12,33	10,85	7,72	14,65	17,18
s_i	8,31	9,05	10,26	17,09	15,4	18,14	24,46	22,26
Khoảng tin cậy	3,91 - 10,77	1,22 - 8,68	4,34 - 12,8	5,29 - 19,37	4,5 - 17,19	0,25 - 15,2	4,57 - 24,73	8,01 - 26,36

Từ bảng 8-2 chúng ta thấy rằng ở giờ thứ nhất thời gian đợi trung bình của khách hàng là *7,43 phút*, độ lệch chuẩn là *8,31*. Trong giờ thứ nhất có khoảng *95%* khách hàng phải đợi trong khoảng *3,91 - 10,77 phút*. Hoặc nói cách khác, với độ tin cậy *95%* thời gian đợi của các khách hàng trong giờ thứ nhất nằm trong khoảng *7,34 ± 3,42 phút* (3,42 là 1/2 khoảng tin cậy). Tương tự như vậy ta có các dữ liệu tương ứng với các giờ làm việc tiếp theo của ngân hàng như trong bảng 8-2. Độ rộng của khoảng tin cậy nói lên mức độ tập trung của thời gian đợi của khách hàng. Lẽ dĩ nhiên độ rộng của khoảng tin cậy sẽ thu hẹp lại khi tăng số lần chạy mô phỏng. Vì vậy tăng số lần chạy mô phỏng là biện pháp hữu hiệu để tăng độ chính xác của mô phỏng.

8.5. Phân tích dữ liệu đầu ra mô phỏng của hệ không giới hạn

Khi phân tích dữ liệu đầu ra mô phỏng của hệ không giới hạn chúng ta gặp phải một số vấn đề mà trong hệ giới hạn không xảy ra. Những vấn đề đó là:

+ *Ảnh hưởng của điều kiện đầu*. Điều kiện đầu có thể gây ra giai đoạn quá độ không phản ánh đúng hành vi của hệ thống ở trạng thái ổn định. Dữ liệu trong giai đoạn này không được dùng để phân tích và đánh giá hệ thống. Vì vậy cần phải xác định thời điểm kết thúc giai đoạn quá độ và chỉ thu thập dữ liệu đầu ra từ thời điểm đó trở đi mà thôi.

+ *Tính không độc lập giữa các mẫu*. Nếu các mẫu thu thập được không độc lập với nhau thì đánh giá phương sai sẽ bị sai lệch, do đó không phản ánh được bản chất của hệ thống.

+ *Chiều dài mô phỏng*. Mặc dầu hệ thống thuộc loại không giới hạn nhưng chiều dài mô phỏng là có giới hạn. Vì vậy phải chọn chiều dài mô phỏng hợp lý và nằm trong giai đoạn ổn định của hệ thống.

Có nhiều phương pháp xử lý dữ liệu đầu ra mô phỏng của hệ không giới hạn như *phương pháp trung bình bó (The Method of Batch Means), phương pháp tự tương quan (Autocorrelation Methods), phương pháp tái sinh (Regenerative Methods).* Điểm chung của các phương pháp trên là tìm cách tránh ảnh hưởng của giai đoạn quá độ, loại trừ tính phụ thuộc giữa các mẫu và xác định chiều dài mô phỏng hợp lý.

8. 6. Sử dụng kết quả mô phỏng

Sử dụng kết quả mô phỏng là bước tiếp theo của việc phân tích và xử lý dữ liệu đầu ra mô phỏng. Các kết quả mô phỏng không cho ta kết luận hoặc quyết định về hệ thống mà chỉ cung cấp thông tin để dễ dàng đi đến các quyết định đó.

Rõ ràng ưu điểm của mô phỏng là có thể cho ta nhiều phương án để lựa chọn và do đó có thể rút ra kết luận và đi đến các quyết định chính xác.

Các kết quả mô phỏng phải được trình bày một cách khoa học, hợp lý và thuận tiện cho việc sử dụng và lưu trữ. Các kết quả mô phỏng thường được trình bày dưới dạng *báo cáo mô phỏng* (*Simulation Reports*) và *tài liệu mô phỏng* (*Simulation Documentations*).

a) Báo cáo mô phỏng

Báo cáo mô phỏng thường bao gồm các mục sau:

- Giá trị Max, Min, kỳ vọng toán, phương sai của các thông số chủ yếu
- Phân loại các thông tin phù hợp với nguồn lực và sản phẩm của hệ thống
- Báo cáo theo ca, tuần, tháng, năm theo yêu cầu của mô phỏng .

b) Tài liệu về mô phỏng

Tài liệu về mô phỏng bao gồm những phần chính sau đây

- Mô tả hệ thống, mục đích mô phỏng
- Dữ liệu đầu vào
- Mô hình: Bao gồm mô hình nguyên lý, mô hình mô phỏng
- Phân tích và thiết kế thực nghiệm mô phỏng
- Báo cáo phân tích dữ liệu đầu ra
- Kết luận - Quyết định
- Kiến nghị - Tư vấn

Tuỳ thuộc vào mục đích mô phỏng và độ phức tạp của hệ thống mà các loại tài liệu kể trên được phân ra tỉ mỉ hoặc nhập lại cho gọn hơn.

Bài tập

1. Hãy cho biết các hệ thống sau đây thuộc hệ giới hạn hay không giới hạn. Nếu là hệ giới hạn thì sự kiện nào là sự kiện E?. Chú ý rằng các ví dụ sau đây chỉ nhằm xác định loại của hệ thống. Độc giả có thể thêm các dữ liệu cần thiết để xây dựng thành một bài toán mô phỏng.

a) *Ngân hàng:* Một ngân hàng mở cửa vào 9 giờ sáng và đóng cửa vào 4 giờ chiều. Lúc mở cửa trong ngân hàng chưa có khách hàng. Khách hàng đến giao dịch với cường độ 2 khách hàng / giờ và được phục vụ với cường độ 1 khách hàng / giờ. Hãy đánh giá trị số trung bình khách hàng được phục vụ và số khách hàng còn lại tại thời điểm ngân hàng đóng cửa?

b) *Sân bay:* Sân bay làm việc 24 / 24. Máy bay hạ cánh và khởi hành theo lịch. Nhưng do nhiều nguyên nhân ngẫu nhiên thời điểm hạ cánh và khởi hành của máy bay phải thay đổi. Vì vậy trạng thái của sân bay vào đầu giờ buổi sáng phụ thuộc vào đêm hôm trước đã có bao nhiêu chuyến máy bay hạ cánh và khởi hành an toàn. Hãy xác định giá trị trung bình số máy bay phải chờ đợi để hạ cánh và cất cánh.

c) *Bệnh viện:* Bệnh nhân được nhập viện khi có giường trống, nếu không bệnh nhân phải sắp hàng chờ. Số bệnh nhân được nhập viện vào mỗi buổi sáng phụ thuộc vào số bệnh nhân xuất viện ngày hôm trước và số bệnh nhân có trong danh sách chờ nhập viện. Hãy xác định thời gian chờ trung bình của bệnh nhân để nhập viện.

d) *Kho phụ tùng:* Kho phụ tùng của nhà máy có nhiệm vụ đáp ứng phụ tùng thay thế. Mức độ dự trữ phụ tùng trong kho phụ thuộc vào xác suất hỏng hóc của từng loại phụ tùng và thời gian đặt hàng mua sắm của nó. Hãy xác định lượng phụ tùng cần thiết trong thời gian vận hành của nhà máy là 5 năm. Xác định lượng dự trữ tối thiểu của từng loại phụ tùng để đảm bảo nhà máy vận hành an toàn không khi nào phải ngừng sản xuất vì thiếu phụ tùng thay thế.

2. Hãy cho biết đặc trưng quan trọng nhất của dữ liệu đầu ra mô phỏng là gì? Công thức tính các đặc trưng đó. Điều kiện đầu ảnh hưởng như thế nào tới kết quả mô phỏng?

Chương 9
MÔ PHỎNG CÁC HỆ THỐNG SẢN XUẤT

9.1. Khái niệm chung

Hệ thống sản xuất được chia làm hai loại chính: hệ thống liên tục và hệ thống gián đoạn. Trong hệ thống liên tục quá trình sản xuất xảy ra liên tục như sản xuất điện năng, sản xuất hoá chất, vận chuyển dầu khí v.v... Trong hệ thống gián đoạn quá trình sản xuất xảy ra với các sự kiện gián đoạn như sản xuất cơ khí, sửa chữa, lắp ráp, vận tải hành khách, cơ sở dịch vụ v.v...

Hệ thống gián đoạn rất phổ biến và chiếm tỷ trọng lớn trong các hệ thống sản xuất. Vì vậy mô phỏng hệ thống sản xuất gián đoạn có ý nghĩa to lớn trong thực tế. Trong hệ thống gián đoạn các sự kiện gián đoạn có vai trò đặc biệt quan trọng. Nghiên cứu mô phỏng các hệ thống sản xuất gián đoạn đó là nội dung chính của chương này.

Sự kiện gián đoạn (*Discrete Event*) được định nghĩa là sự kiện xảy ra đột ngột tại các thời điểm và làm thay đổi trạng thái của hệ thống. Trong hệ thống sản xuất có nhiều sự kiện được coi là sự kiện gián đoạn. *Ví dụ:* ôtô đến trạm sửa chữa, ôtô rời khỏi trạm khi được sửa chữa xong, khách hàng đến giao dịch tại ngân hàng, các chi tiết đi đến trạm lắp ráp v.v... đều là những sự kiện gián đoạn.

Tóm lại trong các hệ thống sản xuất (*Manufacturing Systems*) thường có nhiều sự kiện gián đoạn. Vì vậy các hệ thống sản xuất gián đoạn được mô hình hoá bằng mô hình gián đoạn (*Discrete Models*) và thường dùng phương pháp mô phỏng gián đoạn (*Discrete Simulation*) để nghiên cứu hệ thống.

Mục 9.2 trình bày về những lợi ích đem lại của mô phỏng hệ thống sản xuất.

Mục 9.3 nói về phương pháp xây dựng mô hình mô phỏng các sự kiện gián đoạn. Mục 9.4 phân tích dòng sự kiện đầu vào và thời gian phục vụ. Mục 9.5 trình bày về phương pháp thiết kế và phân tích thực nghiệm mô phỏng. Mục 9.6 nói về số lần chạy mô phỏng và chiều dài mô phỏng. Mục 9.7 phân tích điều kiện khởi động và ngừng mô phỏng. Cuối cùng mục 9.8 nói về cách tạo dòng thời gian mô phỏng.

9.2. Những lợi ích đem lại của mô phỏng hệ thống sản xuất

Mô phỏng hệ thống sản xuất đưa lại nhiều lợi ích, đó là lý do mà ngày nay mô phỏng được ứng dụng nhiều trong nghiên cứu các hệ thống sản xuất. Nhờ có mô phỏng, người ta có thể nghiên cứu nhiều kịch bản của quá trình sản xuất, đánh giá năng lực sản xuất của từng kịch bản, từ đó tìm ra kịch bản tối ưu. Do đó mô phỏng hệ thống sản xuất có thể đem lại những lời khuyên có ích để :

- Tăng năng suất lao động
- Giảm lượng vật tư phải dự trữ trong kho
- Tăng hệ số sử dụng máy
- Giảm vốn đầu tư (đất đai, nhà xưởng, máy móc...)
- Tăng khả năng cung cấp trực tuyến các sản phẩm cho khách hàng

Mô phỏng cho ta khả năng để đánh giá các vấn đề sau :

- Kế hoạch sản xuất
- Mức dự trữ nguyên vật liệu và phụ tùng thay thế trong kho
- Phân tích độ tin cậy của hệ thống sản xuất
- Chính sách kiểm tra chất lượng

Nói chung mô phỏng đưa lại nhiều lợi ích khi thiết kế cung như khi vận hành các hệ thống sản xuất. Các mục tiếp theo dưới đây trình bày kỹ hơn về các kỹ năng khi mô phỏng các hệ thống sản xuất có các sự kiện gián đoạn.

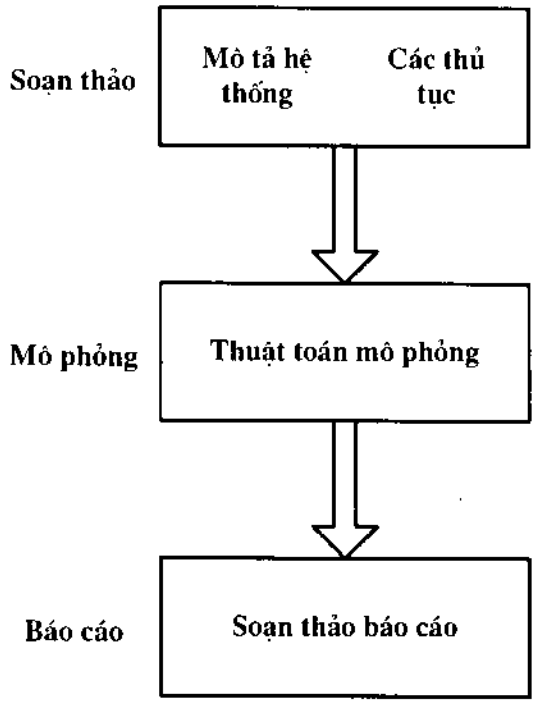
9.3. Phương pháp xây dựng mô hình mô phỏng các sự kiện gián đoạn (Discrete Event Simulation Model)

Khi xây dựng mô hình mô phỏng các hệ thống sản xuất có sự kiện gián đoạn, người ta phải thực hiện ba nhiệm vụ sau đây, xem hình 9.1.

Nhiệm vụ đầu tiên là soạn thảo mô hình và xác định điều kiện đầu, xác định các trạng thái của hệ thống thường được gọi là hình ảnh của hệ thống, ở phần này còn xác định các thủ tục quy định việc thay đổi trạng thái, hình ảnh hệ thống.

Nhiệm vụ thứ hai là xác định thuật toán mô phỏng tức xác định bằng cách nào tiến hành việc mô phỏng.

Nhiệm vụ thứ ba là thu thập các dữ liệu, kết quả mô phỏng đã được xử lý và soạn thảo các báo cáo.



Hình 9.1. Các nhiệm vụ của chương trình mô phỏng.

Hình 9.2 trình bày quan hệ logic giữa các bộ phận trong mô hình mô phỏng.

Trong sơ đồ 9.2:

Trạng thái hệ thống (System state): là tập hợp các dữ liệu mô tả hệ thống ở thời điểm nhất định

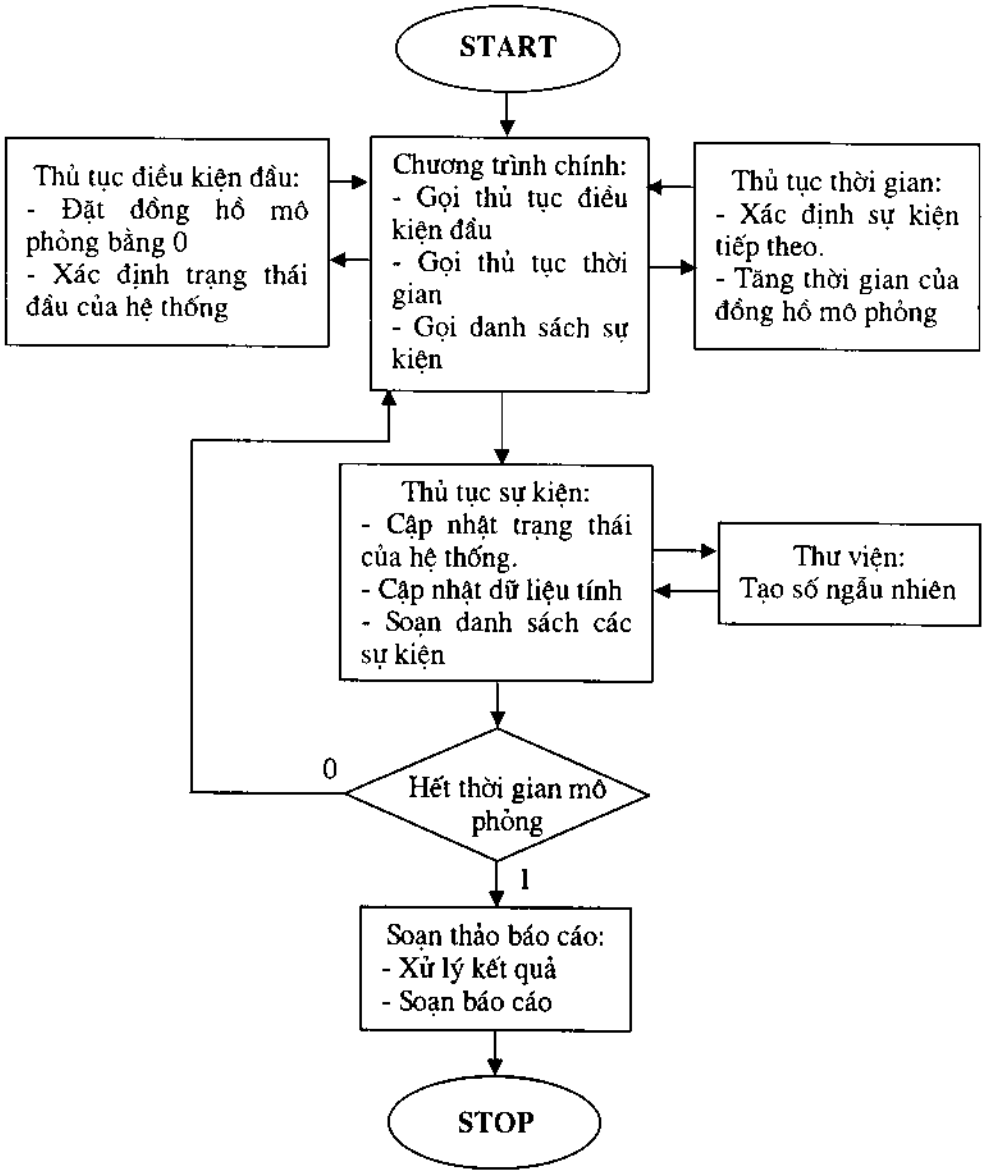
Đồng hồ mô phỏng (Simulation Clock): là biến cho phép mô tả thời gian của quá trình mô phỏng.

Danh sách các sự kiện (Event List): Danh sách các sự kiện xảy ra trong quá trình mô phỏng.

Bộ đếm thống kê (Statistical Counter): là biến để lưu trữ các thông tin thống kê về đặc tính của hệ thống

Thủ tục điều kiện đầu (Initialization Routine): là chương trình con để xác định điều kiện đầu của mô hình và thời gian t = 0.

Thủ tục thời gian (Timing Routine): là chương trình con xác định sự kiện kế tiếp trong danh sách các sự kiện và tăng thời gian của đồng hồ mô phỏng khi sự kiện xảy ra.



Hình 9.2. Sơ đồ logic của mô hình mô phỏng các sự kiện gián đoạn.

Thủ tục sự kiện (Event Routine): là chương trình con để cập nhật trạng thái hệ thống khi các sự kiện xảy ra.

Thủ tục thư viện (Library Routine): là chương trình con dùng để tạo ra các biến ngẫu nhiên có phân bố mong muốn.

Soạn thảo, báo cáo (Report Generator): là chương trình con xử lý, đánh giá các sự kiện trong quá trình mô phỏng đồng thời soạn thảo các báo cáo về kết quả mô phỏng.

Chương trình chính (Main Program): là chương trình để gọi các chương trình con khác để thực hiện việc mô phỏng, đồng thời xác định thời điểm kết thúc mô phỏng và soạn thảo các báo cáo cần thiết.

Liên hệ logic dòng điều khiển giữa các bộ phận trong mô hình và quá trình mô phỏng xảy ra như sau:

• Ở thời điểm bắt đầu mô phỏng $t = 0$, chương trình chính gọi thủ tục điều kiện đầu (*Initialization Routine*). Ở thủ tục này đồng hồ mô phỏng lấy giá trị $t = 0$ và đặt giá trị ban đầu cho trạng thái hệ thống (*System State*), bộ đếm thống kê (*Statistical Counter*) và danh sách các sự kiện (*Event list*).

• Sau đó chương trình chính gọi thủ tục thời gian (*Timing Routine*) để xác định loại sự kiện sắp xảy ra. Nếu sự kiện loại i sẽ xảy ra thì đồng hồ mô phỏng sẽ tăng lên quãng thời gian theo thời gian của sự kiện loại i xảy ra.

• Sau đó chương trình chính gọi thủ tục sự kiện i (*Event Routine*) để xác định thời gian sự kiện kế tiếp sẽ xảy ra. Thông thường ở đây cần phải tạo nên biến ngẫu nhiên có hàm phân bố định trước để tìm thời gian xảy ra sự kiện kế tiếp.

• Sau một chu kỳ tính, chương trình chính sẽ kiểm tra điều kiện ngừng mô phỏng. Nếu đã đạt được điều kiện đó quá trình mô phỏng sẽ ngừng lại và chương trình chính gọi thủ tục báo cáo (*Report Generator*), cùng với bộ đếm thống kê (*Statistical Counter*) để soạn thảo báo cáo về các kết quả mô phỏng.

9.4. Dòng sự kiện đầu vào và thời gian phục vụ

9.4.1. Dòng sự kiện đầu vào

Trong thực tế chúng ta thường cần mô phỏng các hệ thống trong đó có dòng các sự kiện đầu vào (*Arrivals*). Ví dụ: dòng khách hàng chờ ở quầy tính tiền của siêu thị, dòng các sản phẩm đi đến gia công ở các máy công cụ, dòng tàu biển chở vào bến cảng, dòng ôtô đến trạm sửa chữa v.v... Những sự kiện đầu vào như vậy là những sự kiện gián đoạn, thời gian xuất hiện và khoảng cách giữa các sự kiện mang tính ngẫu nhiên. Trong nhiều trường hợp người ta gọi các sự kiện đầu vào là các "khách hàng" (*Customer*)

9.4.2. Thời gian phục vụ

Khi có khách hàng đến, hệ thống sẽ phục vụ khách hàng, thời gian phục vụ (*Service Times*) cũng là một đại lượng ngẫu nhiên tuỳ thuộc vào đặc điểm của khách hàng. Ví dụ: tuỳ theo khách hàng mua nhiều hay ít hàng mà thời gian tính tiền dài hay ngắn, tuỳ theo ô tô hỏng nhiều hay ít mà thời gian sửa chữa tại trạm sửa chữa dài hay ngắn.

Trong trường hợp khách hàng đến mà hệ thống đang bận phục vụ khách hàng đến trước đó thì khách hàng mới sẽ sắp hàng để chờ đến lượt mình. Như vậy sẽ xuất hiện hàng đợi (*Queue)* trước điểm phục vụ. Hệ thống như vậy thường có tên gọi là *hệ thống hàng đợi* (*Queueing System*) mà sẽ được nghiên cứu kỹ hơn ở các mục sau đây.

9.4.3. Dòng sự kiện đầu vào Poatxông (Poisson Arrivals)

Trong trường hợp chung, dòng sự kiện đầu vào là dòng sự kiện ngẫu nhiên phức tạp. Tuy nhiên trong thực tế có nhiều dòng sự kiện có thể quy về dòng có các tính chất sau:

- Tính chất xảy ra các sự kiện trong suốt quãng thời gian khảo sát đều như nhau - đó là dòng dừng.

- Các sự kiện xảy ra độc lập với nhau, - đó là tính chất không hậu quả.

- Tại mỗi thời điểm chỉ có một sự kiện xảy ra - đó là tính chất sự kiện đơn

Một dòng sự kiện có ba tính chất nêu trên được định nghĩa là *dòng tối giản* hay còn gọi là *dòng Poatxông*. Chú ý rằng, nếu một dòng sự kiện là một dòng tối giản, thì khoảng cách giữa các sự kiện sẽ tuân theo luật phân phối mũ.

Gọi λ - cường độ dòng sự kiện, có nghĩa là số sự kiện xảy ra trên một đơn vị thời gian. Vậy hàm mật độ xác suất của quãng thời gian giữa các sự kiện sẽ là:

$$f(t) = \lambda e^{-\lambda t} ; (t \geq 0) \qquad (9 - 1)$$

Và hàm phân bố xác suất là:

$$F(t) = 1 - e^{-\lambda t} ; (t \geq 0) \qquad (9 - 2)$$

Như vậy khoảng cách giữa các sự kiện của dòng đầu vào sẽ tuân theo luật phân phối mũ.

Có thể lựa chọn các luật phân phối khác nhau như phân bố Gamma, phân phối Weibull. Tuy nhiên phân phối Poatxông và phân phối mũ là hai loại phân

phối có công thức tính toán tương đối đơn giản mà trong thực tế phù hợp với dòng đầu vào của nhiều hệ thống sản xuất. Vì vậy hai loại phân phối mũ nêu trên thường được dùng trong mô phỏng các sự kiện gián đoạn trong các hệ thống sản xuất.

9.5. Thiết kế và phân tích thực nghiệm mô phỏng (Design of simulation experiments)

Sau khi đã xây dựng mô hình mô phỏng và kiểm tra thấy mô hình làm việc tốt chúng ta chuyển sang bước thiết kế thực nghiệm mô phỏng (bước 7 trên hình 3.2 Chương 3). Thiết kế thực nghiệm mô phỏng là xác định những điều kiện để thực hiện mô phỏng như điều kiện khởi động và ngừng mô phỏng, thời gian chạy mô phỏng v.v... Ở phần này chúng ta cũng phải thiết kế các kịch bản mô phỏng và xác định số lần chạy mô phỏng cho mỗi kịch bản. *Thiết kế thực nghiệm mô phỏng (Design of simulation experiments)* mô phỏng có vai trò quan trọng vì nó ảnh hưởng trực tiếp đến kết quả mô phỏng. Nếu thực nghiệm mô phỏng không hợp lý, không tối ưu thì dữ liệu đầu ra của mô phỏng có thể rất nhiều nhưng chúng ta không thu được các kết quả mong muốn.

9.6. Số lần chạy mô phỏng (Number of run - Replication) và chiều dài mô phỏng (Run Length)

Như đã trình bày ở các phần trên, trong mô hình mô phỏng có các yếu tố ngẫu nhiên, nên mỗi lần chạy mô phỏng chỉ cho ta một bộ thông số ứng với một điều kiện nhất định. Vì vậy người ta muốn chạy mô phỏng nhiều lần với chiều dài lớn. Tuy nhiên trong thực tế do hạn chế về thời gian, về khả năng tính toán nên số lần chạy mô phỏng không thể lớn được. Theo kinh nghiệm người ta khuyên rằng nên chạy mô phỏng một số ít lần với chiều dài mô phỏng lớn tốt hơn là chạy nhiều lần với chiều dài mô phỏng ngắn. Bởi vì mỗi lần chạy mô phỏng phải tốn thời gian nạp điều kiện đầu, vả lại nếu chiều dài mô phỏng ngắn không đủ mô tả dáng điệu của hệ thống. Thông thường số lần chạy mô phỏng nằm trong khoảng 10-30 lần tuỳ thuộc độ phức tạp của hệ thống mô phỏng.

Như chúng ta đều biết mỗi một bước mô phỏng chỉ cho ta một bộ giá trị các thông số ứng với điều kiện nhất định. Muốn có giá trị trung bình theo xác suất (kỳ vọng toán, *Means*) của các thông số mô hình thì số bước mô phỏng phải đủ lớn, tức chiều dài mô phỏng phải đủ dài. Theo kinh nghiệm chiều dài mô phỏng phải khoảng hàng ngàn bước tính.

Tuỳ thuộc vào mục đích mô phỏng, vào đặc điểm của hệ thống được mô phỏng mà người ta chọn chiều dài mô phỏng thích hợp, đảm bảo cho các dữ liệu đầu ra ổn định.

9.7. Điều kiện khởi động (Starting Coditions) và ngừng mô phỏng (Stopping Rules)

a) Điều kiện khởi động tuỳ thuộc vào mục đích mô phỏng. Điều kiện khởi động chung nhất là điều kiện "hệ thống đang còn rỗng" (*Empty and idle*) có nghĩa là các sự kiện đầu vào chưa xảy ra.

Có thể chọn một trong ba điều kiện sau đây :

- Nếu mục đích của mô hình là phân tích hành vi của của hệ thống ở trạng thái ổn định thì trong quá trình mô phỏng phải bỏ qua thời kỳ quá độ (*Transition Period*), lúc này điều kiện khởi động chỉ tính từ khi hệ thống bắt đầu ổn định.

- Nếu mục đích của mô hình là phân tích hành vi quá độ của hệ thống thì điều kiện khởi động chính là điều kiện đầu của hệ thống.

b) Điều kiện ngừng mô phỏng được xác định bởi một trong các điều kiện sau đây tuỳ thuộc vào mục đích mô phỏng :

- Khi thời gian mô phỏng (tổng các thời gian giữa các sự kiện) đạt tới giá trị xác định. Trong trường hợp này số sự kiện đầu ra sẽ không xác định trước và trạng thái kết thúc của mô hình có thể không phải là trạng thái rỗng.

- Khi số sự kiện đầu vào đạt tới giá trị xác định. Trong trường hợp này mô hình sẽ được chạy cho đến khi sự kiện đầu vào cuối cùng đi qua và trạng thái kết thúc của mô hình là trạng thái rỗng.

- Khi số sự kiện đầu ra đạt tới giá trị xác định . Trong trường hợp này trạng thái kết thúc của mô hình có thể không phải là trạng thái rỗng.

- Khi thông số của mô hình đạt tới giá trị xác định. Trong trường hợp này trạng thái kết thúc của mô hình có thể không phải là trạng thái rỗng.

9.8. Cách tạo dòng thời gian mô phỏng (Time - Advance Mechanisms)

Trong mô hình mô phỏng cần có một biến để biểu thị thời gian đã trôi qua của quá trình mô phỏng. Người ta gọi biến trong mô hình mô phỏng để tạo ra giá trị thời gian mô phỏng là *đồng hồ mô phỏng (Simulation Clock)*. Tại thời

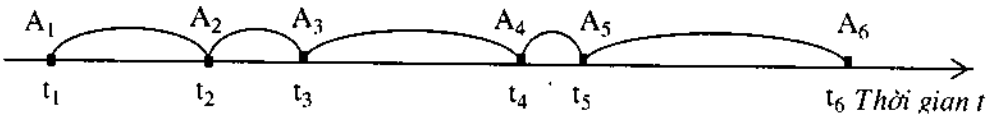
điểm bắt đầu mô phỏng, đồng hồ mô phỏng đặt thời gian $t = 0$. Sau đó đồng hồ mô phỏng ghi nhận dòng thời gian và chỉ ra bao nhiêu đơn vị thời gian đã đi qua trong quá trình mô phỏng. Một đơn vị thời gian trong mô phỏng sẽ tương ứng với bao nhiêu thời gian trong hệ thống thực (giây, phút, giờ, tháng, năm v.v...) là do người lập trình định trước.

Có hai loại thời gian mô phỏng :

- Thời gian sự kiện (Next - Event Time)

Thời gian sự kiện được tính bằng thời gian để sự kiện kế tiếp xảy ra. Như vậy thời gian mô phỏng được tính bằng cách cách cộng dồn các quãng thời gian giữa các sự kiện xảy ra trong quá trình mô phỏng. Hình 9.3 trình bày cách biểu diễn thời gian sự kiện. Trong đó, A_1, A_2... là các sự kiện, t_1, t_2... là các thời điểm xảy ra sự kiện tương ứng.

Thời gian sự kiện thường được dùng trong mô phỏng các sự kiên gián đoạn. Ví dụ mô phỏng hệ hàng đợi (xem Chương 10), trong đó sự kiện gián đoạn là các khách hàng đến hệ thống để được phục vụ.

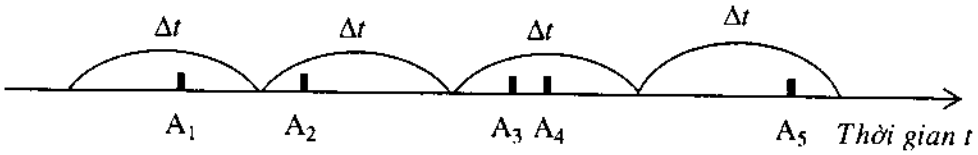


Hình 9.3. Cách biểu diễn thời gian sự kiện.

Mỗi lần chạy mô phỏng đồng hồ mô phỏng được đặt lại giá trị bằng không và bắt đầu tính thời gian mô phỏng cho lần chạy mô phỏng mới.

- Thời gian cố định (Fixed - Increment Time)

Trong trường hợp mô phỏng dùng thời gian cố định, đồng hồ mô phỏng sẽ tăng lên những quãng thời gian chính xác là Δt. Sau đó sẽ xác định xem trong quãng thời gian Δt đó có bao nhiêu sự kiện xảy ra, xem hình 9.4.



Hình 9.4. Cách biểu diễn thời gian cố định.

Thời gian cố định thườmg được dụng trong mô phỏng các hệ thống liên tục,

các hệ kinh tế xã hội, trong đó Δt là quãng thời gian cần khảo sát ví dụ là một năm kế hoạch, một năm tài chính hay là quãng thời gian cố định nào đó tuỳ thuộc vào mục đích mô phỏng.

Bài tập

1. Hãy nêu các ví dụ về sự kiện gián đoạn trong các hệ thống sản xuất. Đặc điểm của các sự kiện đó.

2. Hãy nêu cơ chế tăng thời gian trong mô phỏng các sự kiện gián đoạn.

3. Hãy phân tích vai trò của điều kiện đầu trong mô phỏng. Cách xác định điều kiện đầu và thời gian mô phỏng.

Chương 10
MÔ PHỎNG HỆ THỐNG HÀNG ĐỢI

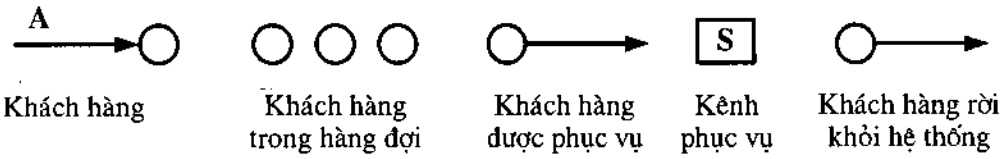
10.1. Khái niệm chung về hệ thống hàng đợi (Queueing System)

Hệ thống hàng đợi là hệ thống có các bộ phận phục vụ (*Services*) và các khách hàng đi đến hệ thống (*Arriving Customers*) để được phục vụ. Nếu khi khách hàng đến mà các bộ phận phục vụ đều bị bận thì khách hàng sẽ sắp hàng để đợi được phục vụ. Chính vì vậy hệ thống này có tên là *hệ thống hàng đợi*. Lý thuyết toán học để khảo sát các hệ hàng đợi được gọi là lý thuyết phục vụ đám đông (các khách hàng được coi là một đám đông được phục vụ). Trong hệ hàng đợi khách hàng là sự kiện gián đoạn xảy ra tại các thời điểm ngẫu nhiên, vì vậy hệ hàng đợi thuộc loại *hệ các sự kiện gián đoạn*.

Mục 10.2 trình bày về các thành phần chính của hệ hàng đợi. Mục 10.3 nói về dòng khách hàng. Mục 10.4 nói về kênh phục vụ. Mục 10.5 trình bày về luật sắp hàng. Mục10.6 nói về thời gian sắp hàng và chiều dài hàng đợi. Mục 10.7 nói về hệ thống hàng đợi M/M/1. Cuối cùng, ví dụ minh hoạ được trình bày trong mục 10.8.

10.2. Các thành phần chính của hệ thống hàng đợi

Hình 10.1 trình bày hệ thống hàng đợi một kênh phục vụ.



Hình 10.1. Hệ thống hàng đợi một kênh phục vụ.

Trong thực tế có rất nhiều hệ thống có thể được xem là hệ thống hàng đợi. Mô phỏng hệ thống hàng đợi nhằm đánh giá năng lực làm việc của hệ thống, khả năng mất khách hàng do phải chờ đợi lâu hoặc không còn chỗ để sắp hàng đợi đến lượt được phục vụ. Trên cơ sở những phân tích như vậy, người ta thiết

kế hệ thống, chọn số lượng kênh phục vụ, năng suất phục vụ, kích thước hàng đợi v.v. nhằm đạt được hiệu quả tối ưu.

Bảng 10-1 trình bày một số hệ thống hàng đợi thường gặp.

Bảng 10-1

Hệ thống	***Kênh phục vụ***	***Khách hàng***
Ngân hàng	Nhân viên ngân hàng	Khách hàng
Bệnh viện	Bác sỹ, y tá	Bệnh nhân
Hệ thống máy tính	CPU, thiết bị vào ra	Dữ liệu
Dây chuyền sản xuất	Công nhân, máy móc	Sản phẩm
Cảng hàng không	Đường băng, trạm kiểm soát	Máy bay, hành khách
Hệ thống liên lạc	Đường dây, nhân viên	Khách hàng
Siêu thị	Quầy hàng, quầy trả tiền	Khách hàng

Hệ thống hàng đợi có ba bộ phận chính là:

1) *Dòng khách hàng* (*Arriving Customers, Arrival Patterns*): là các phần tử, các sự kiện đi đến hệ thống để được phục vụ - được gọi chung là khách hàng. Đặc trưng cho dòng khách hàng là cường độ dòng khách hàng λ *1/ đơn vị thời gian*. Dòng khách hàng là một dòng các sự kiện gián đoạn, ngẫu nhiên, do đó khoảng cách thời gian giữa các khách hàng cũng là một đại lượng ngẫu nhiên.

2) *Kênh phục vụ* (*Server*): là các bộ phận để phục vụ khách hàng, thực hiện các yêu cầu của khách hàng. Thời gian phục vụ (*Service Time*) và khoảng thời gian giữa các lần phục vụ là những biến ngẫu nhiên. Tuỳ theo hệ thống có một hay nhiều điểm phục vụ mà người ta gọi là hệ thống một hoặc nhiều kênh phục vụ. Đặc trưng cho kênh phục vụ là dòng phục vụ với cường độ phục vụ là μ *1/ đơn vị thời gian*. Cường độ phục vụ là số khách hàng được phục vụ xong trên một đơn vị thời gian.

3) *Hàng đợi (Queue):* là số khách hàng chờ đến lượt phục vụ. Tuỳ theo số khách hàng đến nhiều hay ít (cường độ λ lớn hay bé), khả năng phục vụ (số kênh phục vụ, thời gian phục vụ) mà số khách hàng phải đợi trong hàng đợi nhiều hay ít. Vì vậy độ dài của hàng đợi cũng là một biến ngẫu nhiên.

Đặc trưng cho hàng đợi có:

- *Chiều dài hàng đợi:* là số khách hàng có trong hàng đợi đang chờ để được

phục vụ.

- *Thời gian đợi:* là khoảng thời gian từ khi khách hàng đến hệ thống đến khi bắt đầu được phục vụ. Thời gian đợi có thể được hạn chế hoặc không hạn chế.

- *Luật sắp hàng:* là phương thức chọn khách hàng trong hàng đợi. Thông thường có các luật sắp hàng như đến trước được phục vụ trước, đến sau được phục vụ trước, ngẫu nhiên, ưu tiên... Nếu hệ thống có nhiều kênh phục vụ thì phải có luật phân chia khách hàng giữa các kênh phục vụ.

10.3. Dòng khách hàng (Customer)

Dòng khách hàng là một trong những bộ phận quan trọng nhất của hệ thống hàng đợi. Một số ví dụ sau đây là các dòng khách hàng:

- Dòng các cuộc gọi của một trạm điện thoại.

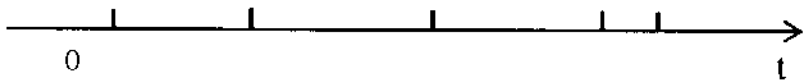
- Dòng các thiết bị điện gia dụng (bàn là, tivi, rađiô, máy giặt, nồi cơm điện v.v...) nối vào mạng điện cung cấp.

- Dòng các hư hỏng xảy ra trong hệ thống máy tính, hệ thống điều khiển

- Dòng các xe ôtô đến trạm sửa chữa.

- Dòng bệnh nhân đến khám bệnh , khách hàng vào nhà hàng, siêu thị v.v.

Những dòng như vậy được gọi là dòng sự kiện ngẫu nhiên có trạng thái gián đoạn xảy ra kế tiếp nhau trong thời gian liên tục, xem hình 10.2.



Hình 10.2. Dòng sự kiện gián đoạn.

Trên hình 10.2 chúng ta thấy điểm gốc thời gian *0t* có thể lấy ở bất kỳ điểm nào trên trục thời gian. Các dòng khách hàng mà chúng ta xem xét trong chương này thường có thể được quy về dòng sự kiện tối giản. Một dòng tối giản có ba tính chất cơ bản sau: dừng, không hậu quả và toạ độ.

- Dòng dừng là dòng mà xác xuất xảy ra một số sự kiện nào đó chỉ phụ thuộc vào quãng thời gian *t* (xem hình 10.2) chứ không phụ thuộc vào vị trí của quãng thời gian *t* trên trục thời gian. Có nghĩa là trên dòng dừng xác suất xảy ra sự kiện là như nhau trên suốt trục thời gian.

- Dòng không hậu quả là dòng mà các sự kiện xảy ra độc lập với nhau, có

nghĩa là sự kiện xảy ra tại thời điểm t_1 không kéo theo sự kiện xảy ra tại thời điểm t_2 và ngược lại.

- Dòng toạ độ là dòng các sự kiện chỉ xảy ra tại một toạ độ nhất định. Có nghĩa là tại một thời điểm chỉ có một sự kiện xảy ra, xác suất để có hai hay nhiều sự kiện xảy ra cùng một lúc là rất nhỏ có thể bỏ qua.

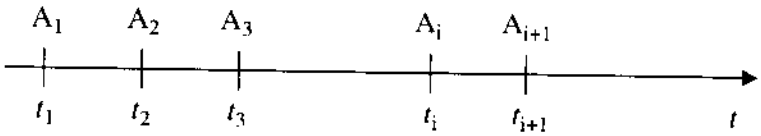
Chú ý rằng nếu sự kiện xảy ra không phải là ngẫu nhiên mà theo một quy luật nào đó, ví dụ đều đặn cách một khoảng thời gian T, những dòng như vậy là dòng có hậu quả. Người ta chứng minh được rằng tổng một số đủ lớn dòng (dừng, toạ độ) có hậu quả hạn chế sẽ cho một dòng tối giản (dừng, không hậu quả, toạ độ).

Một dòng dừng hoặc không dừng, nhưng không hậu quả và toạ độ được gọi là *dòng Poisson*. Trong dòng Poisson cường độ sự kiện λ (số sự kiện xảy ra trên một đơn vị thời gian) phụ thuộc vào thời gian, tức $\lambda = \lambda(t)$.

Nếu $\lambda = const$ thì dòng Poisson là dừng và lúc này trở thành dòng tối giản.

Dòng tối giản có vai trò quan trọng trong việc khảo sát các dòng khách hàng vì các tính toán dựa trên dòng tối giản sẽ đơn giản và thuận lợi

Xét một dòng khách hàng là một dòng tối giản, hình 10.3:



Hình 10.3. Dòng khách hàng:

$t_1, t_2, \ldots t_i$: thời điểm các khách hàng xuất hiện;

$A_1, A_2, \ldots A_i$: khoảng thời gian giữa các khách hàng.

Do dòng khách hàng là dòng tối giản nên cường độ khách hàng (số khách hàng trung bình trên một đơn vị thời gian) là hằng số

$$\lambda = \frac{1}{M_A} = \text{const},$$

trong đó: M_A - kỳ vọng toán của đại lượng ngẫu nhiên $A_1, A_2, \ldots A_i$.

Người ta chứng minh được nếu dòng khách hàng là một dòng tối giản thì khoảng cách giữa các khách hàng A_i sẽ là biến ngẫu nhiên tuân theo luật phân

phối mũ - *expo(λ)*.

Như vậy nếu dòng khách hàng là dòng tối giản , thời gian giữa các khách hàng sẽ tuân theo luật phân phối mũ, giá trị trung bình của nó bằng $1/\lambda$, trong đó λ - cường độ của dòng khách hàng.

Như đã trình bày trên hình 10.3, tại các thời điểm $t_1, t_2,...t_i$ khách hàng xuất hiện làm cho trạng thái của hệ thống thay đổi. Vì dòng khách hàng là dòng tối giản nên các thời điểm $t_1, t_2,..t_i$ xuất hiện hoàn toàn ngẫu nhiên không phụ thuộc lẫn nhau, từ đó suy ra quá trình chuyển trạng thái trong hệ thống cũng là ngẫu nhiên không phụ thuộc vào các trạng thái trong quá khứ. Nếu như nguyên tắc sắp hàng là *FIFO* thì chuỗi trạng thái của hệ thống như trên được gọi là chuỗi Markov. Chính vì vậy người ta dùng ký hiệu *M* để chỉ phân bố mũ của các khoảng thời gian giữa các khách hàng.

10.4. Kênh phục vụ (Server)

Một hệ thống có thể có một hoặc nhiều kênh phục vụ. Tuỳ tính chất của khách hàng mà thời gian phục vụ khác nhau. Sau đây là một ví dụ về thời gian phục vụ:

- Thời lượng của các cuộc gọi của trạm điện thoại.
- Thời gian gia công các chi tiết trên máy.
- Thời gian khám bệnh, điều trị cho bệnh nhân.
- Thời gian tính tiền cho một khách hàng ở siêu thị.

Thời gian phục vụ là một đại lượng ngẫu nhiên. Sau khi khách hàng được phục vụ xong thì sẽ rời khỏi thệ thống và kênh phục vụ nhận ngay khách hàng mới để phục vụ nếu trong hàng đợi đang có khách hàng. Như vậy số các khách hàng được phục vụ tạo thành dòng phục vụ. Trong trường hợp thời gian phục vụ có phân phối mũ *expo(μ)*, trong đó: μ - cường độ dòng phục vụ - là số khách hàng được phục vụ trên một đơn vị thời gian - thì dòng phục vụ tạo thành một dòng tối giản và chuỗi trạng thái phục vụ là một chuỗi Markov và người ta dùng ký hiệu *M* để chỉ phân phối mũ của thời gian phục vụ.

Gọi $S_1, S_2,...$ là thời gian phục vụ. Vậy:

$$\mu = \frac{1}{M_s}$$

trong đó: M_s là kỳ vọng toán của thời gian phục vụ.

Người ta thường dùng các ký hiệu sau đây để chỉ các hệ thống hàng đợi khác nhau.

• *M / M / 1* - Hệ thống hàng đợi có 1 kênh phục vụ, dòng khách hàng và phục vụ là dòng tối giản.

• *M / M / S* - Hệ thống hàng đợi có *S* kênh phục vụ, dòng khách hàng và phục vụ là dòng tối giản.

• *GI / G / S* - Hệ thống hàng đợi có S kênh phục vụ, dòng khách hàng là dòng sự kiện ngẫu nhiên độc lập (*GI – General Independent*) và dòng phục vụ có phân phối bất kỳ (*G-General*).

Trong hệ thống hàng đợi người ta thường đánh giá khả năng của hệ thống bằng hệ số sử dụng (*Utilization factor*).

$$\rho = \frac{\lambda}{\mu}$$ Đối với hệ *M / M / 1*

$$\rho = \frac{\lambda}{S\mu}$$ Đối với hệ *M / M / S*

10.5. Luật sắp hàng

Luật sắp hàng là luật lựa chọn khách hàng để phục vụ. Trong hệ thống hàng đợi có một kênh phục vụ thường có các luật sắp hàng sau đây:

• *FIFO (First In First Out)* – Khách hàng đến trước được phục vụ trước, khách hàng đến sau được phục vụ sau. Luật *FIFO* thường được dùng ở những nơi như:

+ Sắp hàng trước quầy tính tiền của siêu thị.

+ Sắp hàng vào cơ sở dịch vụ, phương tiện vận tải.

+ Các thiết bị sắp hàng trên băng chuyền chờ đến lượt được lắp ráp.

+ Các chai sắp hàng đi vào máy chiết bia v.v...

• *LIFO (Last In First Out)* – Khách hàng đến sau được phục vụ trước. Luật *LIFO* thường được dùng ở những nơi như:

+ Ra khỏi buồng thang máy: người vào sau cùng sẽ ra trước tiên.

+ Đọc dữ liệu trên băng từ: dữ liệu ghi sau sẽ được đọc trước.

+ Hàng hoá được xếp vào thùng chứa: hàng xếp sau cùng (phía trên cùng của thùng chứa) sẽ được lấy ra trước v.v...

• *Ngẫu nhiên:* Các khách hàng đều có độ ưu tiên như nhau và được phục vụ một cách ngẫu nhiên. Luật này thường thấy ở các trường hợp sau:

+ Lấy linh kiện điện tử trong ô ra để lắp ráp.

• *Ưu tiên:* Một số khách hàng có một số đặc tính nhất định sẽ được phục vụ trước. Luật này thường thấy trong các trường hợp như:

+ Phụ nữ, trẻ em và người tàn tật được ưu tiên phục vụ trước.

+ Luật *FIFO*, luật *LIFO* cũng là một trường hợp đặc biệt với dấu hiệu ưu tiên là đến trước hoặc đến sau.

+ Thời gian phục vụ ngắn được phục vụ trước (*Shortest job first*). Ví dụ trên nút giao thông xe nhỏ gọn di chuyển nhanh được ưu tiên đi trước so với xe to cồng kềnh di chuyển chậm phải đi sau. Bài toán có thời gian giải ngắn được máy tính chọn để giải trước.

10.6. Thời gian sắp hàng và chiều dài hàng đợi

Thời gian sắp hàng là quãng thời gian khách hàng đứng đợi trong hàng đợi chờ đến lượt phục vụ. Có loại khách hàng có thể đợi bao lâu cũng được, ngược lại có loại khách hàng chỉ có thể đợi trong khoảng thời gian nhất định, hết thời gian đó khách hàng sẽ rời bỏ hệ thống, mặc dầu vẫn còn chỗ để đứng đợi. Trong trường hợp này hệ thống sẽ mất khách hàng. Để giảm khả năng mất khách hàng hệ thống phải tăng cường độ phục vụ hoặc tăng số kênh phục vụ.

Chiều dài hàng đợi là số khách hàng đứng đợi để được phục vụ. Nếu số vị trí để đứng đợi không hạn chế thì chiều dài hàng đợi có thể dài bất kỳ. Ngược lại nếu số vị trí để đứng đợi bị hạn chế thì chiều dài hàng đợi không vượt quá số đã cho trước. Trong trường hợp này nếu khách hàng đến đúng vào lúc chiều dài hàng đợi đã đầy thì phải rời bỏ hệ thống và hệ thống sẽ bị mất khách hàng. Chiều dài hàng đợi là một đại lượng ngẫu nhiên phụ thuộc vào cường độ dòng khách hàng và dòng phục vụ

Sau đây đưa ra cách tính thời gian sắp hàng và chiều dài hàng đợi.

Gọi:

D_i - thời gian sắp hàng của khách hàng thứ i;

S_i - thời gian phục vụ của khách hàng thứ i;

vậy

$W_i = D_i + S_i$ - thời gian chờ đợi trong hệ thống của khách hàng thứ i.

$Q(t)$ - số khách hàng trong hàng đợi tại thời điểm t.

$L(t)$ - số khách hàng có trong hệ thống tại thời điểm t, ([L(t) = Q(t) + số khách hàng đang được phục vụ] .

Chúng ta có thể chứng minh được các quan hệ sau đây [1]:

Thời gian sắp hàng trung bình:

$$d = \lim_{n\to\infty} \frac{\sum_{i=1}^{n} w_i}{n} \qquad (10 - 1)$$

Thời gian chờ đợi trung bình trong hệ thống

$$w = \lim_{n\to\infty} \frac{\sum_{i=1}^{n} w_i}{n} \qquad (10 - 2)$$

Trị số trung bình khách hàng có trong hàng đợi, hay còn gọi là chiều dài trung bình của hàng đợi:

$$Q = \lambda d, \qquad (10 - 3)$$

trong đó: λ - cường độ dòng khách hàng;

d - thời gian sắp hàng trung bình.

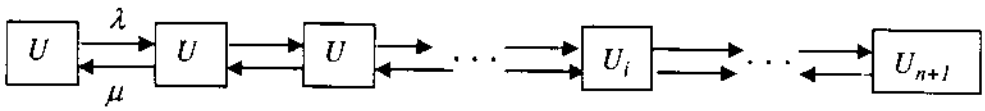
Trị số trung bình khách hàng có trong hệ thống

$$L = \lambda\omega, \qquad (10 - 4)$$

trong đó: ω - thời gian chờ đợi trung bình của khách hàng trong hệ thống.

Xét hệ thống hàng đợi *M / M / 1* số vị trí trong hàng đợi hữu hạn và bằng n Trong trường hợp này khi khách hàng đến hệ thống mà tất cả vị trí trong hàng đợi đều bị chiếm chỗ thì khách hàng sẽ rời bỏ hệ thống. Ta nói hệ thống mất khách hàng.

Cường độ dòng khách hàng là λ. Cường độ dòng phục vụ là μ. Trạng thái của hệ thống là U. Hình 10.4 mô tả trạng thái của hệ thống nêu trên.



Hình 10.4. Trạng thái hệ thống hàng đợi M/M/1.

Chúng ta đánh số trạng thái hệ thống theo số lượng khách hàng có trong hệ thống (được phục vụ và đang đợi).

+ U_0 - Điểm phục vụ rỗi (không có khách hàng)

+ U_1 - Điểm phục vụ bận (một khách hàng đang được phục vụ), không có khách hàng đợi

+ U_2 - Điểm phục vụ bận, một khách hàng đợi

+ U_i - Điểm phục vụ bận, i-1 khách hàng đợi.

+ U_{n+1} - Điểm phục vụ bận, n khách hàng đợi

$p_0, p_1, p_2 \dots p_{n+1}$ - Xác suất để hệ thống ở trạng thái $U_0, U_1, U_2, \dots U_{n+1}$

Từ hình 10.4 ta có thể viết:

$$\begin{aligned} &\lambda = \mu p_1 \\ &\lambda p_1 = \mu p_2 \\ \hline &\lambda p_{i-1} = \mu p_i \end{aligned} \qquad (10 - 5)$$

Từ đó ta có các quan hệ sau:

$$\begin{aligned} &p_1 = \frac{\lambda}{\mu} p_0 \\ &p_2 = \left(\frac{\lambda}{\mu}\right)^2 p_0 \\ &p_3 = \left(\frac{\lambda}{\mu}\right)^3 p_0 \\ \hline &p_{n+1} = \left(\frac{\lambda}{\mu}\right)^{n+1} p_0 \end{aligned} \qquad (10 - 6)$$

Chú ý rằng tổng xác suất các trạng thái của hệ thống (trừ trạng thái U_0 là trạng thái không có khách hàng) luôn luôn bằng 1, ta có:

$$p_1 + p_2 + p_3 + \dots P_{n+1} = 1$$

Dễ dàng tìm được sác xuất p_0 là xác suất hệ thống rỗi, không có khách hàng:

$$p_0 = \frac{1}{1 + \frac{\lambda}{\mu} + \left(\frac{\lambda}{\mu}\right)^2 \dots \left(\frac{\lambda}{\mu}\right)^{n+1}} \qquad (10 - 7)$$

Mẫu số của (10-7) là một cấp số nhân có công bội λ/μ, do đó có thể viết:

$$p_0 = \frac{1}{\left[1-\left(\frac{\lambda}{\mu}\right)^{n+2}\right]\left(1-\frac{\lambda}{\mu}\right)} \qquad (10 - 8)$$

$$p_0 = \frac{1-\frac{\lambda}{\mu}}{1-\left(\frac{\lambda}{\mu}\right)^{n+2}} \qquad (10 - 9)$$

Chú ý rằng xác suất p_{n+1} là xác suất xảy ra trạng thái tất cả n vị trí trong hàng đợi đều có khách hàng đứng đợi. Lúc này có khách hàng mới đến thì sẽ rời bỏ hệ thống. Vậy xác suất để hệ thống bị mất khách hàng P_0 sẽ là:

$$P_0 = p_{n+1}$$

Thay (10-9) vào (10-6), ta có thể viết:

$$P_0 = p_{n+1} = \left(\frac{\lambda}{\mu}\right)^{n+2} p_0 = \left(\frac{\lambda}{\mu}\right)^{n+1} \frac{1-\frac{\lambda}{\mu}}{1-\left(\frac{\lambda}{\mu}\right)^{n+2}} \qquad (10 - 10)$$

Trạng thái ngược với trạng thái mất khách hàng chính là trạng thái phục vụ. Vì vậy khả năng phục vụ tương đối của hệ thống được đánh giá bằng xác suất:

$$P_1 = 1 - P_0 = 1 - \left(\frac{\lambda}{\mu}\right)^{n+1} \frac{1-\frac{\lambda}{\mu}}{1-\left(\frac{\lambda}{\mu}\right)^{n+1}} \qquad (10 - 11)$$

Khả năng phục vụ tuyệt đối được đánh giá bằng công thức sau:

$$A = \lambda P_1 \qquad (10 - 12)$$

Giá trị A là số khách hàng được phục vụ trên một đơn vị thời gian:

Chúng ta cũng có thể chứng minh được rằng giá trị trung bình số khách hàng có trong hàng đợi Q là:

$$Q = \frac{\left(\frac{\lambda}{\mu}\right)^2\left[1-\left(\frac{\lambda}{\mu}\right)^n\left(n+1-n\frac{\lambda}{\mu}\right)\right]}{\left[1-\left(\frac{\lambda}{\mu}\right)^{n+2}\right]\left[1-\frac{\lambda}{\mu}\right]} \qquad (10 - 13)$$

Theo (10 - 3) thời gian sắp hàng chờ đợi trung bình d

$$d = \frac{Q}{\lambda} \qquad (10 - 14)$$

Thời gian trung bình của khách hàng ở trong hệ thống ω = thời gian sắp hàng trung bình d + thời gian trung bình phục vụ t_s .

Thời gian trung bình phục vụ t_s bằng thời gian phục vụ một khách hàng $1/\mu$ nhân với khả năng phục vụ của hệ thống P_1 , tức:

$$t_s = \frac{1}{\mu} P_1 \qquad (10 - 15)$$

Vậy ta có:

$$\omega = d + t_s \qquad (10 - 16)$$

Thay (10-14) và (10-15) vào (10-16) ta có

$$\omega = \frac{Q}{\lambda} + \frac{P_1}{\mu} \qquad (10 - 17)$$

Số khách hàng nằm trong hệ thống L bằng số khách hàng nằm trong hàng đợi Q cộng với số khách hàng trung bình đang được phục vụ l_s

Ta có: $l_s = \lambda t_s$

Thay (10-15) vào biểu thức trên ta được

$$l_s = \frac{\lambda}{\mu} P_1 \qquad (10 - 18)$$

Vậy số khách hàng trung bình nằm trong hệ thống là:

$$L = Q + \frac{\lambda}{\mu} P_1 \qquad (10 - 19)$$

Biểu thức (10-19) chính là ta thay (10-17) vào (10-14).

Ví dụ: Một trạm sửa chữa ôtô có một điểm sửa chữa. Bãi đỗ xe chờ phục vụ chứa được 3 xe, tức $n = 3$.

Cường độ dòng xe đến sửa chữa λ = 1 phút

Thời gian sửa chữa 1 xe ôtô là: 1,25 /phút

Hãy xác định:

- Xác suất trạm mất khách hàng: P_0

- Khả năng phục vụ tương đối và tuyệt đối: P_1 và A

- Trị số trung bình số ôtô chờ phục vụ: Q

- Trị số trung bình số ôtô có trong trạm (tại một thời điểm): L

- Thời gian trung bình ôtô chờ trong hàng: d

- Thời gian trung bình ôtô có mặt tại ở trạm sửa chữa: ω

Giải:

Trạm sửa chữa ôtô nêu trên có thể được mô tả bằng mô hình $M / M / 1$ với độ dài hàng đợi là $n = 3$.

Cường độ dòng phục vụ: $\mu = 1/1,25 = 0,8$

Theo (8.10) tính được xác suất trạm mất khách hàng P_0, tức nếu đã có 3 xe đang chờ sửa chữa thì xe tiếp theo sẽ đi qua mà không vào trạm để sửa chữa nữa.

$$P_0 = p_{n+1} = (\lambda/\mu)^{n+1} \frac{1-(\lambda/\mu)}{1-(\lambda/\mu)^{n+2}}$$

$$P_0 = (1/0,8)^4 \frac{1-1/0,8}{1-(1/0,5)^5} = 0,297$$

- Theo (10-11) tính được khả năng phục vụ tương đối

$$P_1 = 1 - P_0 = 0,703$$

- Theo (10-13) tính được trị số trung bình số ôtô chờ phục vụ (sắp hàng): Q

$$Q = \frac{1/(0,8)^2\left[1-(1/0,8)^3(3+1-3(1/0,8))\right]}{\left[1-(1/0,8)^5\right](1-1/0,8)} = 1,56$$

- Theo (10-19) tính được trị số trung bình số ôtô có mặt trong trạm sửa chữa, L

$$L = Q + (\lambda / \mu).P_1 = 1,5 + 1 / 0,8.0,703 = 2,44$$

- Theo (10-14) tính được thời gian trung bình ôtô chờ trong hàng d

$$d = Q / \lambda = 1,56 / 1 = 1,56 \text{ phút}$$

- Theo (10-17) tính được thời gian trung bình ôtô có mặt ở trạm sửa chữa ω

$$\omega = Q / \lambda + P_1 / \mu = 1,56 + 0,703 / 0,8 = 2,44 \text{ phút}$$

10.7. Hệ thống hàng đợi M / M / 1 có độ dài hàng đợi không hạn chế ($n \to \infty$)

Sau đây xét các đặc tính của hệ thống khi $n \to \infty$, luật sắp hàng *FIFO*.

+ Khả năng phục vụ P_1

Khi $n \to \infty$ biểu thức (10-11) tiến tới $P_1 = 1$. Điều này có thể được giải thích là nếu luật sắp hàng là *FIFO* và độ dài hàng đợi không hạn chế thì tất cả các khách hàng đều được phục vụ cho nên khả năng phục vụ $P_1 = 1$, tức 100% khách hàng đến hệ thống đều được phục vụ.

+ Xác suất mất khách hàng của hệ thống $P_0 = 0$

+ Độ dài trung bình của hàng đợi (10-13) khi $n \to \infty$ là:

$$Q = \frac{(\lambda / \mu)^2}{1 - (\lambda / \mu)} \qquad (10 - 20)$$

+ Trị số trung bình số khách hàng có mặt trong hệ thống, (10-19) khi $n \to \infty$

$$L = \frac{(\lambda / \mu)}{1 - (\lambda / \mu)} \qquad (10 - 21)$$

+ Trị số trung bình thời gian khách hàng chờ trong hàng đợi, (10-14) khi $n \to \infty$

$$d = \frac{1}{\lambda} \frac{(\lambda / \mu)^2}{1 - (\lambda / \mu)} \qquad (10 - 22)$$

+ Trị số trung bình thời gian khách hàng có mặt trong hệ thống (10-17) khi $n \to \infty$

$$\omega = \frac{(\lambda / \mu)^2}{\lambda(1 - (\lambda / \mu))} + \frac{1}{\mu} \qquad (10 - 23)$$

Các hệ thống hàng đợi được xem xét ở trên đều giả thiết rằng thời gian chờ đợi trong hàng đợi là không hạn chế . Trong thực tế có những hệ thống mà thời gian chờ đợi bị hạn chế $d \le D_0$ – Trong đó D_0 là thời gian cho phép đợi trong hàng đợi. Trong trường hợp này tuy chiều dài hàng đợi không hạn chế, nhưng hệ thống vẫn có khả năng mất khách hàng nếu thời gian trung bình chờ đợi $d > D_0$

Ví dụ:

- Các khách hàng của một trạm bán xăng nếu phải sắp hàng lâu sẽ bỏ đi sang trạm khác.

- Trận địa pháo phòng không (kênh phục vụ) phải phản ứng kịp thời trong một quãng thời gian nhất định (thời gian phục vụ) nếu không máy bay đối phương (khách hàng) sẽ rời bỏ trận địa pháo.

- Các khách hàng vào cửa hàng ăn nhanh nếu phải chờ đợi lâu sẽ bỏ đi sang cửa hàng khác.

10.8. Ví dụ minh hoạ về hệ thống hàng đợi

Mô phỏng trạm sửa chữa ô tô (Xem ví dụ ở mục 10.9)

1. *Mô tả đối tượng mô phỏng*

Một trạm sửa chữa ôtô có một điểm sửa chữa (phục vụ). Bãi đỗ xe phục vụ chứa được 3 xe, tức n = 3.

Cường độ dòng xe đến sửa chữa , λ = 1/phút. Thời gian trung bình để sửa chữa một xe ôtô là 1,25 phút.

Hãy mô phỏng hoạt động của trạm trong quãng thời gian $T = 240$ phút, để xác định các chỉ tiêu sau đây:

- Số xe đến trạm sửa chữa
- Số xe được phục vụ (được sửa chữa);
- Số xe phải rời bỏ trạm vì không có chỗ để sắp hàng chờ đến lượt sửa chữa;
- Xác suất mất khách hàng, P_0;
- Khả năng phục vụ tương đối , P_l;
- Số xe trung bình trong hàng đợi , Q;
- Thời gian đợi trung bình, d;
- Thời gian trung bình xe có mặt tại trạm (thời gian sắp hàng + thời gian phục vụ);
- Rút ra các kiến nghị cần thiết.

Nhiệm vụ mô phỏng trên đây đã được trình bày ở ví dụ trong mục 10-9 ở ví dụ này đã dùng các biểu thức giải tích để giải bài toán. Ở đây chúng ta dùng phương pháp mô phỏng để giải quyết cùng bài toán nêu trên nhằm so sánh giữa hai phương pháp.

2. *Các điều kiện đầu*

- Dòng xe ôtô đến trạm sửa chữa được coi là dòng tối giản, có cường độ $\lambda = 1$

xe/phút. Như vậy khoảng cách giữa các xe đến trạm có phân bố mũ có thông số là λ.

- Trạm có ba chỗ để xe sắp hàng chờ phục vụ, $n = 3$. Vậy độ dài hàng đợi bị hạn chế bởi $n = 3$, nhưng thời gian đợi không bị hạn chế.

- Thời gian sửa chữa trung bình bằng *1,25* phút, vậy cường độ dòng sửa chữa bằng $\mu = 1/1,25 = 0,8$ 1/phút

- Thời gian sửa chữa là đại lượng ngẫu nhiên tuân theo luật phân bố mũ có thông số là μ.

- Điều kiện khởi động mô phỏng: tại $t = 0$ hệ thống rỗng, tức không có xe trong điểm sửa chữa, và cũng không có xe trong hàng đợi

- Điều kiện ngừng mô phỏng là thời gian mô phỏng $T_{mp} = 240$ phút

$$T_{mp} = \sum_{K=1}^{N} t_k$$

trong đó: t_k - khoảng thời gian giữa xe thứ *(k–1)* và xe thứ *(k)* có nghĩa là thời gian mô phỏng được tăng lên sau mỗi lần xảy ra một sự kiện mới là có một xe đến trạm sửa chữa.

3. Các ký hiệu trong chương trình mô phỏng

A_K – Số thứ tự của các xe đến trạm sửa chữa;

T_K – Khoảng thời gian giữa xe thứ (k-1) và xe thứ (k), phút;

S_1 – Số xe được phục vụ;

Q – Số xe có mặt trong hàng đợi tại các bước mô phỏng;

Q_1 – Số xe đã sắp hàng trong hàng đợi;

M – Số xe phải rời bỏ trạm tức số khách hàng mà trạm bị mất vì không đủ chỗ để sắp hàng chờ phục vụ.

Tiếp theo là chương trình mô phỏng trạm sửa chữa ôtô *M/M/1* viết bằng ngôn ngữ Pascal.

4. Chương trình mô phỏng Pascal

```
program MO_PHONG_TRAM_SUA_CHUA_OTO
uses ctr;
var k, q, q1, m, a1, s1, qtb, d,i: integer;
```

```
  u, n, a, s, stb, ai, si, ld, mu, t, td, ts, tw: real;
  f: text;
PROCEDURE Print (l:integer; var f: text);
BEGIN
IF L=0 THEN
BEGIN
clrscr;
gotoxy(1,2); write(f,'Ak'); gotoxy(17,2);
write(f,'Tk'); gotoxy(30,2); write(f,'S1');
gotoxy(42,2); write(f,'Q'); gotoxy(54,2);
write(f,'Q1'); gotoxy(66,2); write(f,'M');
writeln(f,'');
END;
IF L=1 THEN
BEGIN
gotoxy(1,3+i); write(f,' ', a1:3); gotoxy(15,3+i);
write(f,' ', a:8:4); gotoxy(30,3+i); write(f,' ', s1:2);
gotoxy(42,3+i); write(f,' ', q:1); gotoxy(54,3+i);
write(f,' ', q1:3); gotoxy(66,3+i); write(f,' ', m:3);
writeln(f,' ');
END
PROCEDURE Print_to_screen;
BEGIN
i:=i+1;
Print(l,f);
IF (k MOD 20=0) AND (k <> 0) THEN
BEGIN
i:=0;
```

```
END;
PROCEDURE Initial;
BEGIN
clrscr;
q:=0; q1:=0; m:=0; a1:=0; s1:=0;
d1:=0; qtb:=0; s:=0; stb:=0; a:=0;
Write('CUONG DO DONG KHACH HANG:');Readln(ld);
Write('CUONG DO PHUC VU:');Readln(mu);
t:=240;
END;
PROCEDURE Simulation;
BEGIN
k:=0;
Print(0,t);
i:=0;
u:=frac(0.37843);
a:= -ln(u)/ld;
s:=a;
REPEAT
IF a<s THEN
BEGIN
a1:=a1+1;
IF q>= 3 THEN m:=m+1;
ELSE
BEGIN
q:=q+1;
q1:=q1+1
qtb:=qtb+1;
```

```
d:=d+1;
END;
k:=k+1;
Print_to_screen;
u:=frac(u*37);
a:=a+(-ln(u)/ld);
END
ELSE
BEGIN
IF k<>0 THEN s1:=s1+1;
u:= frac(u*37);
IF a = s THEN
BEGIN
a1:=a1+1;
k:=k+1;
Print_to_screen;
a:=a+(-ln(u)/ld);
stb:=s+(-ln(u)/mu);
END
ELSE IF q=0 THEN
BEGIN
a1:=a1+1;
k:=k+1;
Print_to_screen;
s:=a+(-ln(u)/mu);
a:=a+(-ln(u)/ld);
stb:=stb+(-ln(u)/mu);
END
```

```
ELSE
s:=s+(-ln(u)/mu);
stb:=stb+(-ln(u)/mu);
IF q>0 THEN
BEGIN
q:=q-1;
END;
END;
UNTIL a>t;
td:=qtb/(d*ld);
ts:=stb/s1;
tw:=td+ts;
clrscr;
writeln(f,'SO XE DEN HE THONG                          :',a1,'
   [xe]');
writeln(f,'SO XE DEN DUOC PHUC VU            :',s1,'          [xe]');
writeln(f,'TONG SO XE XEP HANG                         :',q1,'
   [xe]');
writeln(f,'TONG SO XE ROI HE THONG           :',m,'           [xe]');
writeln(f,'SO XE TRUNG BINH TRONG HANG DOI     :',Qtb/d:4:3,'
   [xe]');
writeln;
writeln(f,'XAC SUAT MAT KHACH HANG       :',m/a1:4:3,'    [%]');
writeln(f,'KHA NANG PHUC VU TUONG DOI  :',s1/a1:4:3,'    [%]');
writeln;
writeln(f,'THOI GIAN DOI TRUNG BINH
   :',Tdfrac(Td):4:0,'[phut]');
writeln(f,'THOI GIAN PHUC VU TRUNG BINH :',Tsfrac(Ts):4:0,'[phut]');
```

```
    writeln(f,'THOI GIAN TRUNG BINH TRONG HE THONG',
Tfrac(Tvq):4:0,'[phut]');
    END;
    BEGIN
      Assign(f,'a:\tp\7hai.txt');
      rewrite(f);
      Initail;
      Simulation;
      close(f);
    END.
```

Kết quả sau khi chạy chương trình

• Trạng thái của hệ thống qua từng bước mô phỏng

Ak Tk (phút) Sl Q Q1 M

1 0. 9717 0 0 0 0 (ôtô đầu tiên đến trạm. Hệ thống đang rỗng)

2 7. 2324 0 1 1 0 (trong hàng đợi có một xe)

3 9.8821 2 0 1 0 (có 2 xe được phục vụ)

.....

.....

.....

16 24.4177 11 3 12 1 (có 16 xe đến hệ thống, đã được phục vụ 11 xe, đang phục vụ 1 xe, trong hàng đợi có 3 xe, vì hàng đợi đã đầy nên có 1 xe phải rời hệ thống)

.....

.....

.....

235 236.6916 156 3 140 75 (hệ thống đã làm việc 236.6916 phút, có 235 xe đến hệ thống, 156 xe được phục vụ, 3 xe đang trong hàng đợi, 140 xe đã sắp hàng, 75 xe phải rời hệ thống vì không còn chổ để sắp hàng).

• Kết quả sau khi kết thúc mô phỏng do chương trình in ra:

SO XE DEN HE THONG :235 [xe]

SO XE DUOC PHUC VU :156 [xe]

TONG SO XE DA SAP HANG :140 [xe]

TONG SO XE PHAI ROI HE THONG : 75 [xe]

SO XE TRUNG BINH TRONG HANG DOI : 1.558 [xe]

XAC SUAT MAT KHACH HANG : 0.319 [%]

KHA NANG PHUC VU TUONG DOI : 0.664 [%]

THOI GIAN DOI TRUNG BINH : 1 [phut]

THOI GIAN PHUC VU TRUNG BINH : 1 [phut]

THOI GIAN TRUNG BINH XE CO MAT TRONG H.T : 3 [phut]

• Nhận xét

Kết quả mô phỏng gần giống với kết quả của phương pháp giải tích cho ở ví dụ trong mục 10.5. Ưu điểm của phương pháp mô phỏng là có thể quan sát trạng thái của hệ thống qua từng bước mô phỏng, từ đó có thể đưa ra các nhận xét và biện pháp để cải tiến hệ thống. Trong khi đó phương pháp giải tích chỉ chỉ kết quả tổng quát cuối cùng mà thôi.

Chú ý rằng ngày nay có nhiều phần mềm để mô phỏng hệ thống hàng đợi rất thuận tiện, chúng ta không phải viết chương trình mô phỏng như ở trên nữa. Một trong những phần mềm như vậy là phần mềm SIGMA (Simulation Graphical Modeling and Analysis System), xem ở Chương 11 mục 11.6.

Bài tập

1. Hệ thống M / M / 1 Trạm ôtô buýt

Hành khách đến trạm ôtô buýt với cường độ $\lambda = 3$ *hành khách / phút* và sắp hàng để chờ lên xe. Vị trí dành để sắp hàng không (hoặc có) bị hạn chế. Cứ sau 5 phút có một ôtô buýt đến trạm. Mỗi chuyến ôtô buýt chở được *50* khách hàng. Hãy mô phỏng trạm xe buýt sau 1 giờ hoạt động. Tính trị số trung bình khách hàng phải chờ trong hàng đợi. (Xem thêm ví dụ Trạm xe buýt sinh viên ở Chương 5).

2. Hệ thống M / M / 1 Cảng biển

Tàu biển đi đến cảng với khoảng cách là quãng thời gian ngẫu nhiên có phân bố mũ *expo* (λ). Giá trị trung bình của khoảng thời gian giữa các tàu biển là 1,25 ngày - do đó suy ra $\lambda = 1/1,25 = 0,8$. Cảng có một cần trục để bốc dỡ hàng hoá. Khi đến cảng, nếu cần trục đang bận thì tầu biển sẽ sắp hàng theo luật *FIFO*. Thời gian bốc dỡ hàng (thời gian phục vụ) của cần trục phụ thuộc vào số lượng hàng hoá của tầu biển. Thời gian bốc dỡ hàng trung bình cho một tàu là *1,5 ngày*. Vậy thời gian phục vụ có phân bố mũ *expo* (μ) với $\mu = 1/1,5 = 0,66$. Hãy mô phỏng hoạt động của cảng và xác định xem liệu có cần thêm cần trục thứ hai để tàu biển không phải sắp hàng chờ quá một ngày.

3. Hệ thống M / M / 5 Siêu thị

Khách hàng đến siêu thị sẽ lần lượt thực hiện các động tác sau:

Lấy giỏ đựng hàng. Chọn hàng trên các quầy. Tính tiền. Rời khỏi siêu thị.

Dòng khách hàng đến siêu thị là một dòng tối giản có cường độ λ *khách hàng/phút*.

Giả thiết số giỏ đựng hàng không bị hạn chế nên khách hàng đến siêu thị là lập tức vào siêu thị chọn hàng không phải sắp hàng. Số hàng được chọn mua xếp vào các giỏ là số ngẫu nhiên phụ thuộc vào từng ý thích của khách hàng.

Sau khi chọn xong hàng, khách sẽ đến một trong năm quầy kiểm hàng và tính tiền (siêu thị có 5 kênh phục vụ). Nếu quầy bận khách hàng sẽ sắp hàng theo luật *FIFO* và chờ đợi cho đến khi được phục vụ mới thôi, tức thời gian chờ không hạn chế. Thời gian phục vụ cho một khách hàng là đại lượng ngẫu nhiên phụ thuộc vào số lượng và độ phức tạp kiểm hoá của những hàng hoá khách đã chọn. Thời gian phục vụ có phân bố mũ, cường độ phục vụ là μ *khách hàng/phút*. Sau khi được phục vụ xong khách hàng sẽ rời khỏi siêu thị . Hãy mô phỏng hoạt động của siêu thị sau 4 giờ làm việc. Xác định trị số trung bình khách hàng sắp hàng trong hàng đợi chờ tính tiền. Liệu siêu thị có cần phải mở thêm quầy kiểm hàng nữa không ?

4. Hệ thống M / M / 10 trạm điện thoại

Một trạm điện thoại gồm có *10* kênh liên lạc. Số khách hàng gọi điện thoại là một đại lượng ngẫu nhiên, độc lập. Vì vậy khoảng cách giữa các khách hàng tuân theo phân bố mũ. Khi khách hàng gọi đến nếu còn kênh rỗi lập tức được

phục vụ -tức được nối thông mạch để thực hiện đàm thoại. Nếu cả *10* kênh đều bận khách hàng phải sắp hàng chờ đến lượt theo luật *FIFO*.

Thời gian đàm thoại của khách hàng - tức thời gian phục vụ - là một đại lượng ngẫu nhiên tuân theo luật phân bố mũ.

Vấn đề đặt ra ở đây là phải xác định số kênh phục vụ bằng bao nhiêu để thời gian chờ đợi của khách hàng không vượt quá giá trị cho phép

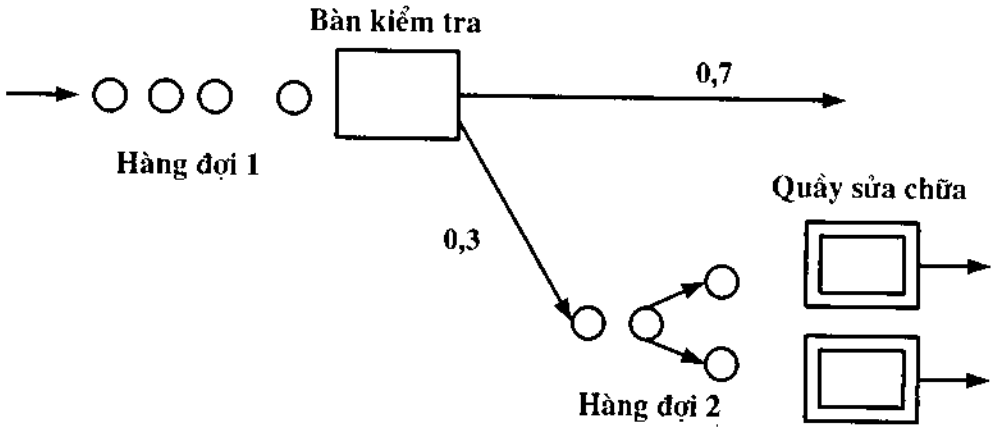
5. Hệ thống M / M / 2 - Trạm sửa chữa ôtô

Một trạm sửa chữa ôtô của thành phố gồm có (xem hình 10.5)

- Một bàn kiểm tra

- Hai quầy sửa chữa ôtô làm việc song song

Khoảng cách giữa các ôtô đến trạm sửa chữa tuân theo luật phân bố mũ với giá trị trung bình bằng *2* giờ.



Hình 10.5. Trạm sửa chữa ôtô.

Thời gian kiểm tra tuân theo luật phân bố đều giữa *15 phút* đến *1,05 giờ*. Ôtô sắp hàng trước bàn kiểm tra theo luật *FIFO*. Qua kiểm tra có *70%* ôtô không phải sửa chữa và lập tức rời khỏi trạm. Có *30%* ôtô phải đưa đi sửa chữa, đến sắp hàng trước *2* quầy sửa chữa làm việc song song. Luật sắp hàng vào quầy sửa chữa là *FIFO*. Thời gian sửa chữa ôtô tuân theo luật phân bố đều giữa *2,1 giờ* và *4,5 giờ*.

Hãy mô phỏng trạm làm việc trong 160 giờ và tính:

- Thời gian đợi trung bình trong mỗi hàng đợi

- Chiều dài hàng đợi trung bình của mỗi hàng đợi

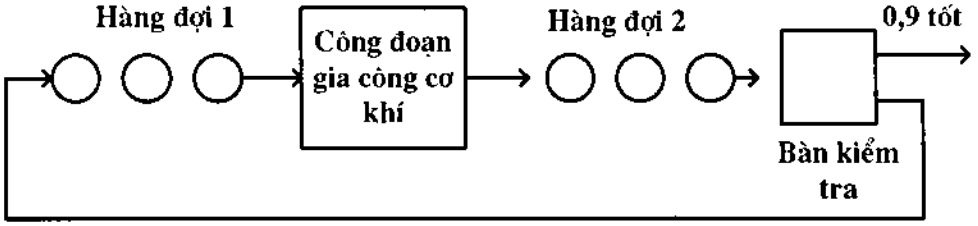
- Hiệu suất sử dụng của bàn kiểm tra

- Hiệu suất sử dụng của quầy sửa chữa

- Nếu giá trị trung bình của khoảng cách giữa các ôtô đến trạm sửa chữa giảm xuống còn 30 phút. Điều gì sẽ xảy ra?

6. Hệ thống M / M / 1 - Phân xưởng gia công cơ khí

Một phân xưởng có một công đoạn gia công cơ khí và một bàn kiểm tra như ở hình 10.6.



Hình 10.6. Phân xưởng gia công cơ khí.

Các chi tiết máy đưa đến công đoạn gia công cơ khí có khoảng cách tuân theo luật phân bố mũ và có giá trị trung bình bằng *1* phút. Thời gian gia công cơ khí chi tiết máy tuân theo luật phân bố đều nằm trong khoảng *0,65* đến *0,7* phút. Thời gian kiểm tra sản phẩm tuân theo luật phân bố đều nằm trong khoảng 0,75 đến *0,8* phút.

Có *90%* sản phẩm đạt loại tốt được đưa đi đóng gói, còn *10%* sản phẩm là loại xấu sẽ được đưa trở về gia công lại.

Giả thiết rằng số chờ đợi của hai hàng đợi là không hạn chế

Hãy mô phỏng hệ thống làm việc trong quãng thời gian *240* phút và tính.

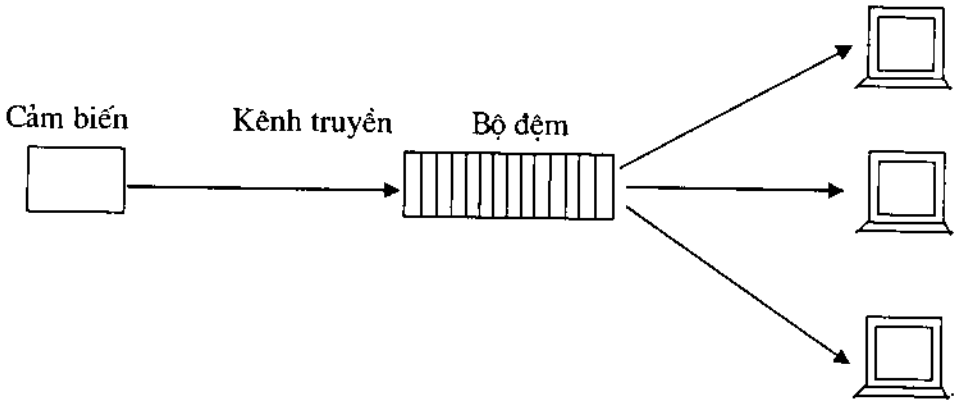
- Thời gian đợi trung bình trong các hàng đợi

- Chiều dài hàng đợi trung bình của các hàng đợi

- Số sản phẩm phải gia công lại.

7. Hệ thống M / M / 3 - Hệ thống xử lý tin

Một hệ thống xử lý thông tin bao gồm một kênh truyền tin, một bộ đệm (*buffer)* và ba máy tính (xem hình 10.7).



Hình 10.7. Hệ thống xử lý thông tin.

Tín hiệu từ cảm biến đi vào kênh truyền tin với khoảng cách trung bình là *5 microsec.*

Tại bộ đệm các tín hiệu được xử lý sơ bộ với thời gian *10 microgiây* cho một tín hiệu. Sau đó tín hiệu được đưa vào xử lý ở một trong ba máy tính. Thời gian máy tính xử lý một tín hiệu mất *33 microgiây.*

Hãy mô phỏng quá trình xử lý thông tin khi có *500* tín hiệu từ cảm biến đưa tới và tính:

- Thời gian trung bình tín hiệu phải chờ trong bộ đệm trước khi được đưa vào máy tính để xử lý.

- Chiều dài trung bình hàng đợi của các tín hiệu trong bộ đệm.

- Xác suất bộ đệm bị tràn, biết rằng dung lượng của bộ điện là *30* tín hiệu

Điều gì sẽ xảy ra khi tăng tốc độ xử lý tín hiệu của các máy tính lên *25 microgiây/tín hiệu.*

Chương 11
ỨNG DỤNG MATLAB - SIMULINK VÀ CÁC PHẦN MỀM CHUYÊN DỤNG TRONG MÔ PHỎNG

11.1. Khái niệm chung về MatLab và Simulink

Như đã trình bày ở các chương trước đây, phương pháp mô phỏng được ứng dụng vào nhiều lĩnh vực khác nhau. Có rất nhiều các phần mềm chuyên dụng được phát triển cho mô hình hoá và mô phỏng như Simnon, 20Sim, Alaska, Pspice, Sigma v.v... Một trong những phần mềm được ứng dụng rộng rãi nhất hiện nay là MatLab .

MatLab được phát triển bởi Công ty Math Works Inc., là ngôn ngữ máy tính bậc cao được phát triển cho tính toán kỹ thuật. Nó tích hợp các chức năng tính toán, hiển thị và lập trình trong môi trường dễ sử dụng. Các ứng dụng cơ bản của MATLAB bao gồm:

- Làm các phép toán;
- Phát triển thuật toán;
- Thu thập dữ liệu;
- Mô hình hoá, mô phỏng và tạo mẫu;
- Phân tích dữ liệu, khai thác và hiển thị;
- Đồ hoạ;
- Các phát triển ứng dụng.

MatLab là hệ thống tương tác trong đó dữ liệu cơ bản là mảng (array) không yêu cầu định trước kích thước. Các chuyên ngành kỹ thuật như điện, điện tử, điều khiển tự động, kỹ thuật cơ khí, năng lượng, hoá học..., các chuyên ngành toán ứng dụng như kế toán, thống kê và và chuyên ngành nghiên cứu sinh học là những địa chỉ ứng dụng quen thuộc của MatLab. Trong công nghiệp, MatLab là công cụ để chọn các nghiên cứu hiệu quả, phát triển và phân tích.

Thư viện toán học của MatLab bao gồm một tập lớn các giải thuật tính toán bao hàm từ các hàm cơ sở, các tính toán cho số phức tới các hàm phức tạp hơn

như đảo ma trận, biến đổi Fourier. Ngoài ra, MatLab còn cung cấp một số khối chuyên dụng đặc biệt khác để giải các bài toán chuyên sâu được phát triển trong các *ToolBox và Blockset* như sau:

Fuzzy logic Toolbox	Logic mờ
Neural Network Blockset	Mạng Nơron
Power System Blockset	Các hệ thống công suất
Realtime Workshop	Ghép nối với thiết bị ngoại vi
Simmechanics	Mô phỏng các hệ động lực học

...

Ngoài ra, MatLab có những chức năng bậc cao cho hiển thị dữ liệu hai chiều, ba chiều, xử lý ảnh, đồ hoạ hoạt hình và biểu diễn đồ hoạ.

SIMULINK là môi trường mô phỏng dựa trên nền MatLab và là công cụ dùng cho thiết kế trên cơ sở mô hình. Người sử dụng có thể xây dựng mô hình từ các khối chức năng trong thư viện của Simulink hoặc tạo riêng các khối chuyên dụng thông qua S-functions. Các mô hình trên Simulink có thể tạo dạng phân cấp, người sử dụng có thể xây dựng mô hình theo chiều từ tổng quan đến chi tiết hoặc ngược lại. Sau khi tạo lập mô hình, bạn có thể chạy mô phỏng từ cửa sổ lệnh của MatLab hoặc từ các Menu trong bản thân Simulink và quan sát kết quả mô phỏng trên các khối hiển thị một cách online. Mặt khác, kết quả của chương trình mô phỏng trên Simulink có thể được đưa vào Wokspace của MatLab để quan sát và hậu xử lý.

MatLab - Simulink là một phần mềm dùng để giải các bài toán kỹ sư trong nhiều lĩnh vực khác nhau. Do khuôn khổ có hạn của cuốn sách, chúng tôi không trình bày chi tiết về *MatLab-Simulink* mà chỉ tập trung vào giới thiệu việc ứng dụng MatLab - Simulink trong mô phỏng các hệ thống điều khiển tự động. Để nắm vững và sử dụng tốt phần mềm này bạn đọc có thể tham khảo các tài liệu như [14, 21]. Ngoài MatLab - Simulink ra, chương 11 còn giới thiệu thêm hai phần mềm chuyên dụng khác thường dùng trong mô phỏng là PSPICE và SIGMA.

Mục 11.2 giới thiệu chung về phần mềm MatLab và ứng dụng trong mô phỏng hệ điều khiển tự động. Mục 11.3 giới thiệu về Simulink và mô phỏng các hệ thống động học. Mục 11.4 đưa ra các ví dụ ứng dụng Simulink mô phỏng các

hệ thống điện cơ. Mục 11.5 trình bày về Simmechanics- một ToolBox rất hữu hiệu của MatLab. Mục 11.6 giới thiệu phần mềm PSPICE và ứng dụng trong mô phỏng các mạch điện tử công suất. Mục 11.7 giới thiệu phần mềm SIGMA dùng để mô phỏng hệ hàng đợi.

11.2. Giới thiệu về ứng dụng MatLab trong mô phỏng hệ điều khiển tự động

11.2.1. Ứng dụng của MatLab để phân tích và khảo sát hệ thống điều khiển tự động

Trong phần này, chúng ta sẽ biểu diễn một số lệnh trong *Control System ToolBox* thường được sử dụng khi phân tích các hệ điều khiển kinh điển. Hàm truyền $G(s)$ được đưa vào tử số $B(s)$ và mẫu số $A(s)$ riêng dưới dạng các đa thức.

a. Miền thời gian

Chất lượng của hệ thống tự động điều khiển trong miền thời gian thường được đánh giá dựa trên đáp ứng thời gian ứng với các tín hiệu vào khác nhau.

Ví dụ 1 dưới đây sẽ minh hoạ đáp ứng của hệ thống với hàm truyền $G(s) = 10/(s^2 + 2s + 10)$ trong thời gian từ 0 đến 10 giây, thời gian lấy mẫu là $0,1$ s ứng với 3 trường hợp: (1) tín hiệu đầu vào là bước nhảy, (2) tín hiệu đầu vào là xung hoặc (3) tín hiệu đầu vào là ngẫu nhiên.

Chương trình viết dưới dạng Scripts

Editor - C:\MATLAB7\work\GTMH1_re.m

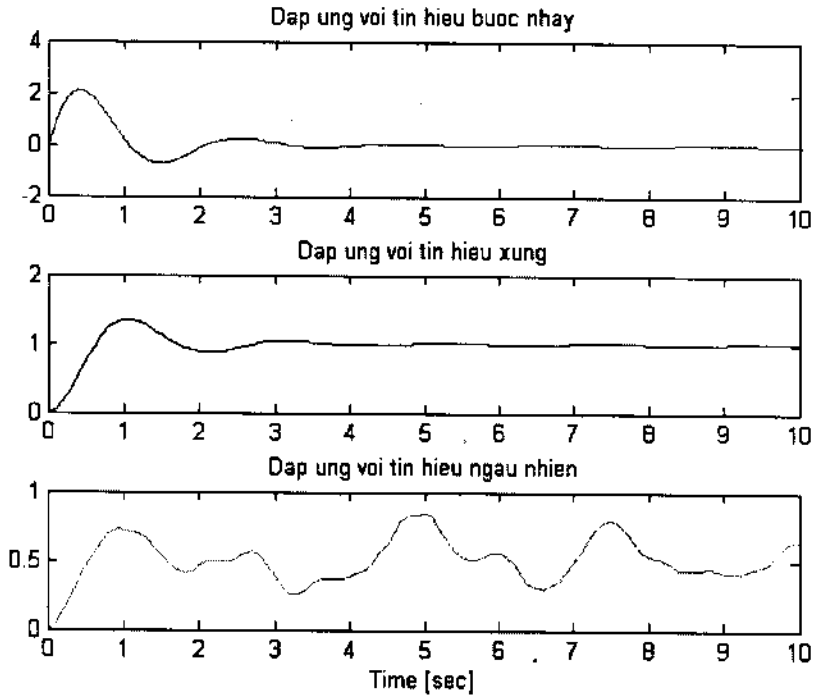
File Edit Text Cell Tools Debug Desktop Window Help

```
num=10;den=[1,2,10];
t=[0:0.1:10]';

hold off;
        y=impulse(num,den,t);
        subplot(3,1,1); plot(t,y,'m');
        title('Dap ung voi tin hieu buoc nhay');

        y= step(num,den,t);
        subplot(3,1,2); plot(t,y,'b');
        title('Dap ung voi tin hieu xung');
        noise=rand(101,1);
        y=lsim(num,den,noise,t);
        subplot(3,1,3);  plot(t,y,'g');
        title('Dap ung voi tin hieu ngau nhien');

       xlabel('Time [sec]');
```

Chạy chương trình

Từ cửa sổ lệnh, chúng ta gõ tên *Script file* là *GTMH1_re*

Đáp ứng của hệ thống được hiển thị trên *Figure* như sau:



Hình 11.1. Đáp ứng theo thời gian của hệ thống theo các tín hiệu đầu vào là bước nhảy, xung và hàm ngẫu nhiên.

b. Miền tần số

Các đáp ứng tần số cho chúng ta cái nhìn trực quan về chất lượng hệ thống và là công cụ tốt cho việc hiệu chỉnh hệ thống. Có thể dùng các lệnh sau để tìm đáp ứng tần số của hệ thống: *bode, nyquits* và *nichols*. Từ đường đặc tính tần số chúng ta có thể xác định được các chỉ tiêu chất lượng như sau:

- Cộng hưởng đỉnh *Mp (dB)* là giá trị cực đại của đồ thị biên độ

- Tần số cộng hưởng ω_r *(rad/s)* là tần số tại đó xuất hiện cộng hưởng đỉnh.

- Tần số cắt ω_c*(rad/s)* là tần số mà tại đó đường đặc tính tần số biên độ loga cắt trục thực.

- Băng thông ω_B*(rad/s)* là giải thông của tần số từ tần số bằng 0 đến tần số mà ở đó biên độ giảm đi *3 dB*.

Các chỉ tiêu chất lượng trong miền thời gian và tần số có quan hệ chặt chẽ với nhau. Thông thường độ quá điều chỉnh δ% lớn thì M_p cũng lớn tương ứng.

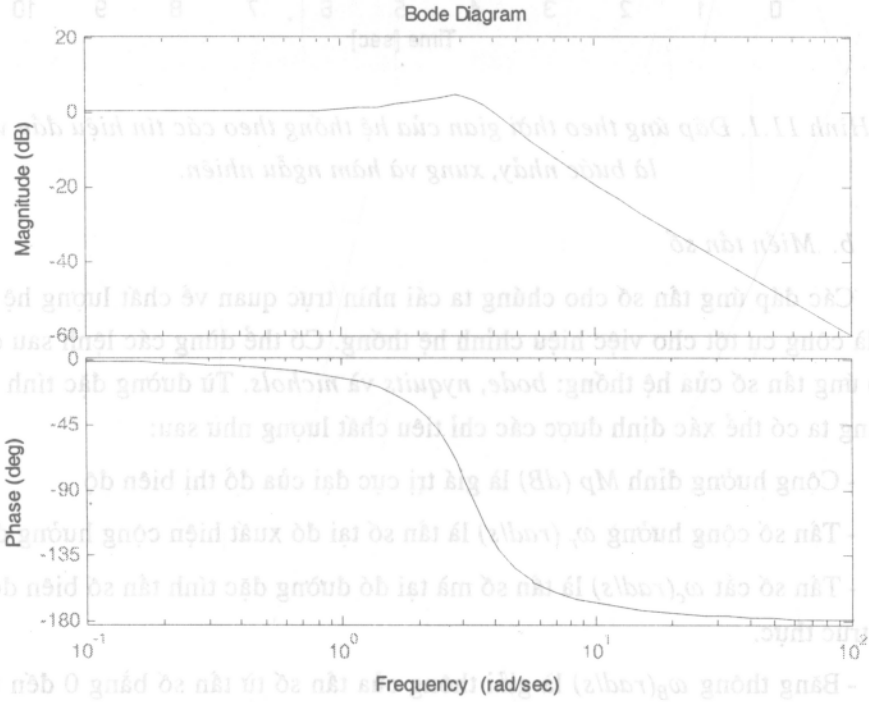
Giá trị cộng hưởng tối ưu nằm trong khoảng $1,1 \le M_p \le 1,5$. Thời gian quá độ t_s lớn thì M_p nhỏ, vì vậy hệ thống tác động chậm. Băng thông ω_B rộng thì các tín hiệu tần số cao được đưa tới hệ thống nhiều hơn nên đáp ứng của hệ sẽ nhanh hơn.

Ví dụ 2 sẽ minh hoạ đáp ứng tần số của hệ thống có hàm truyền *G(s)* được mô tả trong ví dụ 1.

Chương trình viết trên Scripts như sau

```
%Dap ung tan so
%Ten chuong trinh: GTMH2
num=10;den=[1,2,10];
bode(num,den)
```

Chúng ta nhận được đặc tính biên độ (thứ nguyên là *dB*) và pha *(độ)* trên màn hình đồ hoạ như sau:



Hình 11.2. Đáp ứng tần số của hệ thống.

c. Phân tích mô hình không gian trạng thái trên MATLAB

Chúng ta sẽ quan tâm đến ba mô hình cơ bản của hệ thống tuyến tính dừng *(linear time-invariant)*, viết tắt là *LTI* như sau:

•*Mô hình hàm truyền (TF)*

$$G(s) = \frac{a_1 s^n + a_2 s^{n-1} + ... + a_{n+1}}{b_1 s^m + b_2 s^{m-1} + ... + b_{n+1}}$$

trong đó: $a_1,..., a_n$; $b_1,..., b_m$ là các hệ số, s là toán tử Laplace.

Ví dụ 3: Hàm truyền của một đối tượng được biểu diễn như sau

$$G(s) = \frac{s}{s^2 + 2s + 10}$$

Mô hình hàm truyền được xây dựng trên MatLab như sau:

```
>> num=[1 0]; %Tu so
>> den=[1 2 10]; %Mauso
>> H=tf(num,den) %Ham truyen
Transfer function:
```

$$\frac{s}{s^2 + 2s + 10}$$

```
>>
```

hoặc dưới dạng phân số như sau

```
>> s=tf('s');
G=s/(s^2+2*s+10)
 Transfer function:
```

$$\frac{s}{s^2 + 2s + 10}$$

```
>>
```

•*Mô hình điểm không-điểm cực-khuếch đại (ZPK)*

$$G(s) = K\frac{(s - z_1)(s - z_2)...(s - z_n)}{(s - p_1)(s - p_2)...(s - p_n)}$$

với K là hệ số khuếch đại;

$z_1, z_2, \ldots, z_n$ là các điểm không *(zeros)* của $G(s)$

$p_1, p_2, \ldots, p_n$ là các điểm cực *(poles)* của $G(s)$

Ví dụ 4: Xác định trên MatLab hàm truyền dạng *ZPK* của hàm truyền $G(s)$ với hệ số khuếch đại $K=2$, $s_1= 1+i$, $s_2= 1-i$.

Chương trình tìm $G(s)$ trên MatLab như sau:

```
>> k=-2;
>> z=0;
>> p=[2 1+i 1-i];
>> G=zpk(z,p,k)
```

Zero / pole / gain:

$$\frac{-2s}{(s-2)(s^2-2s+2)}$$

```
>>
```

- *Mô hình không gian trạng thái*

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}$$

trong đó:

$x(t)=[x_1(t), x_2(t), \ldots, x_n(t)]^T$ là vector trạng thái, n chiều.

$u(t)$ là vector tín hiệu vào

$y(t)$ là vector tín hiệu ra

A, B, C, D là các ma trận không gian trạng thái

Ví dụ 5: Lập mô hình không gian trạng thái của một hệ thống có ma trận không gian trạng thái A, B, C, D như sau

$$A = \begin{bmatrix} 0 & 1 \\ -5 & -2 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad D = [0]$$

Xác lập hệ phương trình trạng thái của hệ trên MatLab.

Lập M-file như sau:

```
%Mo hinh khong gian trang thai
%Ten file:GTMH3
A=[1 0;-5 -2];
B=[0; 3];
C=[1 0];
D=0;
G=ss(A,B,C,D)
```

Để chuyển từ mô hình *TF* hoặc mô hình *ZPK* sang dạng phương trình trạng thái hoặc ngược lại, chúng ta có thể dùng các lệnh sau đây của *MATLAB*:

```
>>[a,b,c,d]=tf2ss(num,den);
>>[a,b,c,d]=zp2ss(z,p,k);
>>[num,den]=ss2tf(a,b,c,d,ui);
>>[z,p,k]=ss2zp(a,b,c,d,ui);
```

Các phương trình trạng thái nhận được khi dùng các lệnh ***tf2ss, zp2ss*** được viết dưới dạng chuẩn điều khiển được *CCF (Canonical controllable form)*. Khi biết phương trình trạng thái có thể dùng lệnh ***ss2tf*** để chuyển sang dạng hàm truyền đạt và lệnh ***ss2zp*** để chuyển sang dạng điểm không-điểm cực-khuếch đại. Tham số u_i là số đầu vào cho hệ nhiều đầu vào ra *MIMO*.

Ví dụ 6: Hãy tìm phương trình trạng thái của hệ thống có hàm truyền như sau:

$$G(s) = \frac{s^2 + 7s + 2}{s^3 + 9s^2 + 26s + 24}$$

Chương trình được viết trên *M-file* như sau

```
%Chuong trinh tim phuong trinh trang thai
%Ten file: GTMH4
num=[1 7 2];
den=[1 9 26 24];
[A,B,C,D]=tf2ss(num,den)
```

Kết quả chạy chương trình

```
>> GTMH4
   A =                B =              C =              D =
-9 -26 -24            1              1  7  2            0
  1  0  0            0
  0  1  0            0
```

Ví dụ 7: Hãy tìm hàm truyền của hệ thống có phương trình trạng thái như sau:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ -1 & -2 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 10 \\ 0 \\ 0 \end{bmatrix} u$$

y=[1 0 0] x

Chương trình được viết trên *M-file* như sau

```
%Chuong trinh tim ham truyen
%Ten file: GTMH5
A=[0 1 0;0 1 1;-1 -2 -3];
B=[10;0;0];
C=[ 1 0 0];
D=0;
[num,den]=ss2tf(A,B,C,D)
```

Kết quả chạy chương trình trên MATLAB như sau

```
>> GTMH5
num =
  0 10.0000 20.0000 -10.0000
den =
 1.0000 2.0000 -1.0000 1.0000
>>
```

11.3. Ứng dụng Simulink để mô phỏng các hệ thống động học

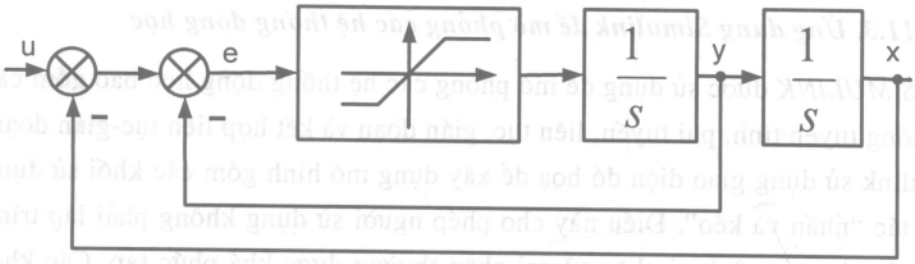
SIMULINK được sử dụng để mô phỏng các hệ thống động học bao gồm các hệ thống tuyến tính, phi tuyến, liên tục, gián đoạn và kết hợp liên tục-gián đoạn. Simulink sử dụng giao diện đồ hoạ để xây dựng mô hình gồm các khối sử dụng thao tác "nhấn và kéo". Điều này cho phép người sử dụng không phải lập trình giải các phương trình vi phân và sai phân thường được khá phức tạp. Các khối chức năng cơ bản của *SIMULINK* bao gồm

Sources	Thư viện nguồn tín hiệu
Sinks	Thư viện các khối nhận tín hiệu
Nonlinear	Thư viện các hàm phi tuyến
Connections	Thư viện các đầu nối
Continuos	Thư viện các khối liên tục
Discret	Thư viện các khối gián đoạn
Signals & Systems	Tín hiệu và hệ thống
Math	Các phép toán

Ngoài ra, người sử dụng có thể tạo ra các khối riêng của mình và bổ sung vào thư viện. Các mô hình xây dựng trên Simulink được phân cấp, người sử dụng có thể quan sát hệ thống ở mức tổng quan, vừa có thể đạt được mức độ cụ thể bằng cách nháy kép vào từng khối để xem xét chi tiết mô hình của từng khối. Các thông số của các khối có thể được thay đổi trực tiếp hoặc khai báo lại theo chương trình trong từng lần chạy thử mô hình.

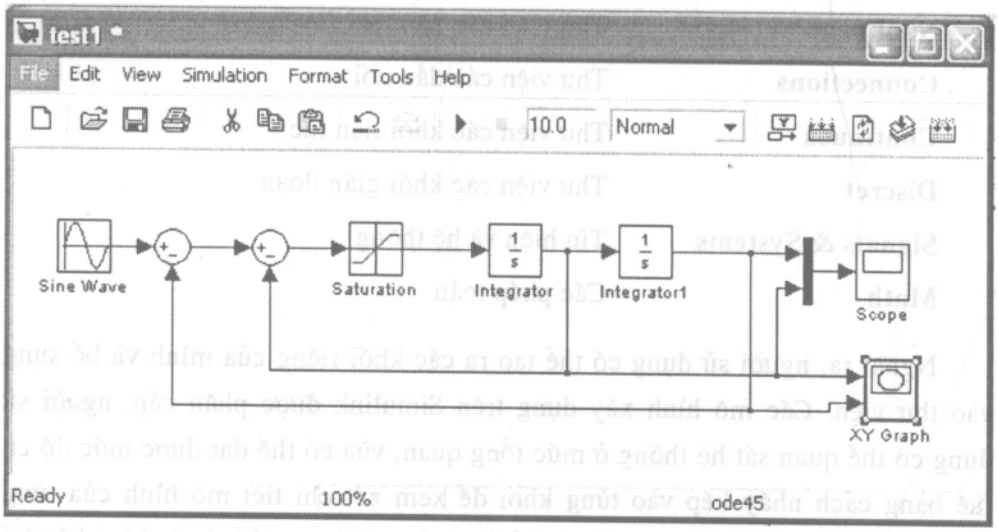
Tiếp theo là một số ví dụ về ứng dụng *Simulink* trong khảo sát đặc tính của các hệ thống thường gặp trong lý thuyết điều khiển tự động và hệ thống điện cơ.

Ví dụ 8: Vẽ đáp ứng của tính hệ phi tuyến gồm khâu khuếch đại bão hoà được mô tả trên hình 11.3. Tín hiệu đầu vào $u = sin\omega t$, giá trị ban đầu $x(0) = -0,5$, $y(0) = 0,5$.



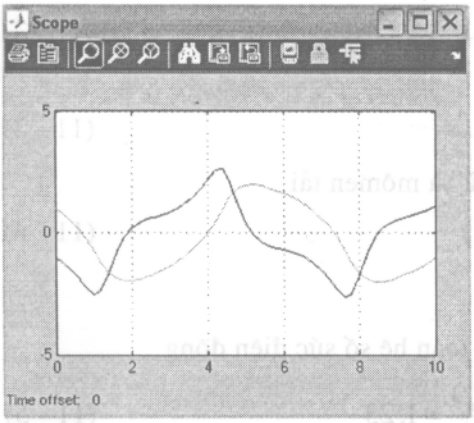
Hình 11.3. Sơ đồ khối của hệ phi tuyến có khâu bão hoà.

Xây dựng mô hình trên *Simulink*

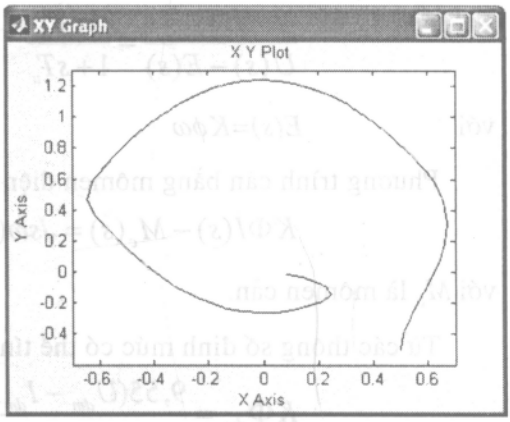


Hình 11.4. Mô hình hệ thống trên Simulink

Các đáp ứng theo thời gian của hệ thống có hàm truyền G(s) được hiển thị hiển thị trên *Scope* có dạng như hình vẽ 11.15a và đồ thị đặc tính pha được biểu diễn trên hình 11.15b.



Hình 11.5a. Đáp ứng của hệ thống theo thời gian hiển thị trên Scope.



Hình 11.5b. Đồ thị đặc tính pha hiển thị trên XY-Graph.

11.4. Ứng dụng MatLab - Simulink trong mô phỏng và khảo sát các hệ thống điện cơ

Trong các hệ thống truyền động tự động thường có yêu cầu về điều chỉnh dòng điện và tốc độ. Việc tổng hợp các mạch vòng điều chỉnh dòng điện và tốc độ được thực hiện nhờ việc áp dụng các tiêu chuẩn tối ưu môđun hoặc môđun đối xứng [25]. Trên cơ sở cấu trúc các bộ điều chỉnh tổng hợp được, chúng ta có thể tiến hành khảo sát các đặc tính của hệ thống trên *MatLab - Simulink*. Các ví dụ dưới đây sẽ trình bày và hệ thống truyền động điều chỉnh dòng điện và tốc độ động cơ điện một chiều kích từ độc lập.

Ví dụ 9: Vẽ đặc tính dòng điện và tốc độ của động cơ điện một chiều kích từ độc lập. Các thông số định mức của động cơ được ghi trong lý lịch của động cơ như sau:

Công suất định mức P_{dm} = 1,5 kW,

Dòng điện định mức I_{udm}= 8,7 A,

Điện áp định mức U_{dm} = 200 V,

Tốc độ định mức n_{dm} = 1440 vòng/phút.

Phương trình cân bằng điện áp mạch phần ứng có dạng biến đổi *Laplace* như sau:

$$U(s) = R_u I(s)\,(1 + sTu) + K\Phi\omega \qquad (11 - 1)$$

Hàm truyền của động cơ có dạng như sau

$$\frac{I(s)}{U(s)-E(s)}=\frac{1/R_u}{1+sT_u} \qquad (11-2)$$

với $$E(s)=K\phi\omega \qquad (11-3)$$

Phương trình cân bằng mômen điện từ và mômen tải

$$K\Phi I(s)-M_c(s)=Js\omega(s) \qquad (11-4)$$

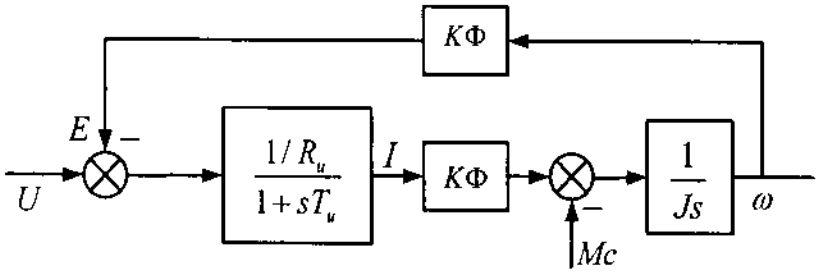
với M_c là mômen cản.

Từ các thông số định mức có thể tính toán hệ số sức điện động

$$K\Phi_{dm}=\frac{9,55(U_{dm}-I_{dm}R_u)}{n_{dm}}=1,23 \qquad (11-5)$$

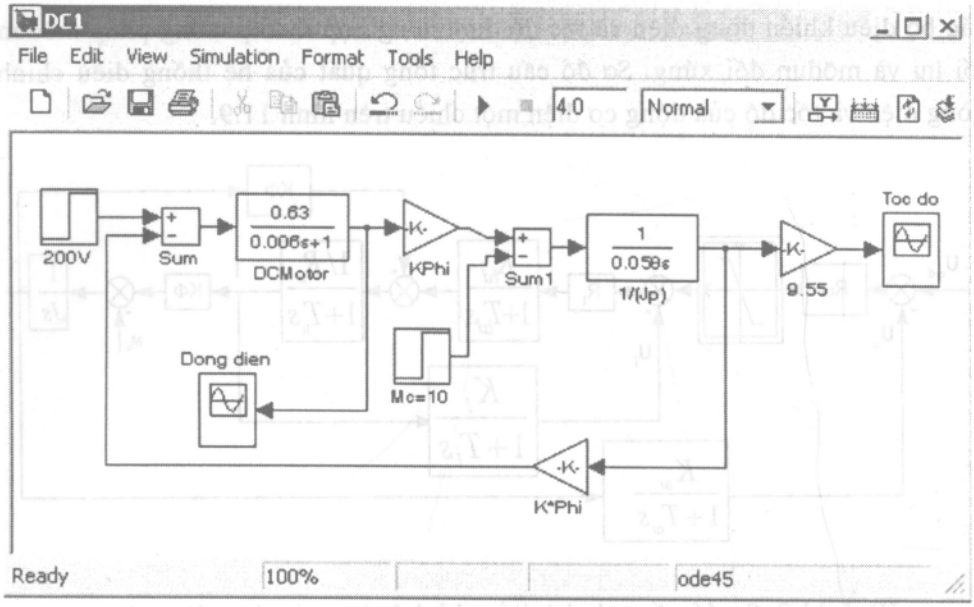
và hàm truyền của động cơ như sau

$$\frac{I(s)}{U(s)-E(s)}=\frac{1/R_u}{1+sT_u}=\frac{0,63}{0,006s+1} \qquad (11-6)$$



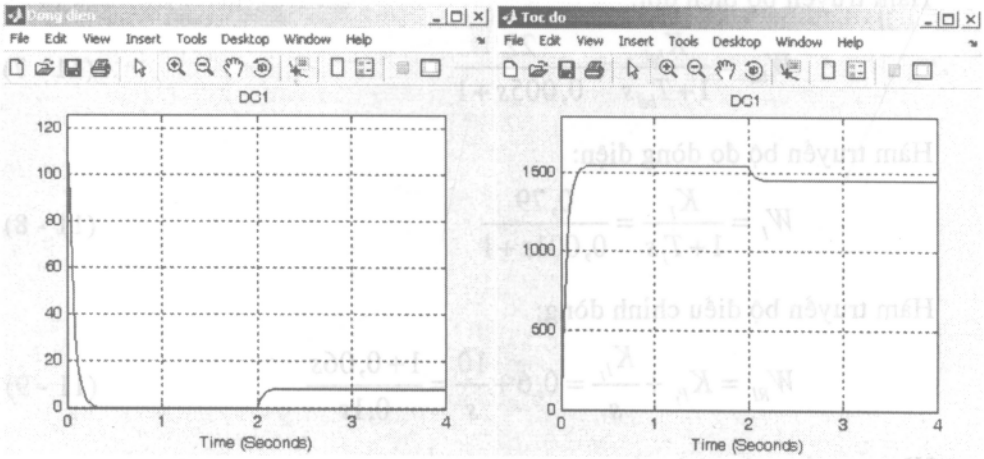
Hình 11.6. Sơ đồ cấu trúc của động cơ điện một chiều kích từ độc lập.

Chương trình mô phỏng cho động cơ điện một chiều kích từ độc lập được viết trên Simulink như trên hình 11.7. Mômen cản M_c là bước nhảy tại t = 2 s với biên độ là *10Nm*. Điện áp đặt vào phần ứng là *200V*, các đặc tính dòng điện và tốc độ sẽ được hiển thị trên các *Graph*.



Hình 11.7 . Mô hình động cơ một chiều trên Simulink.

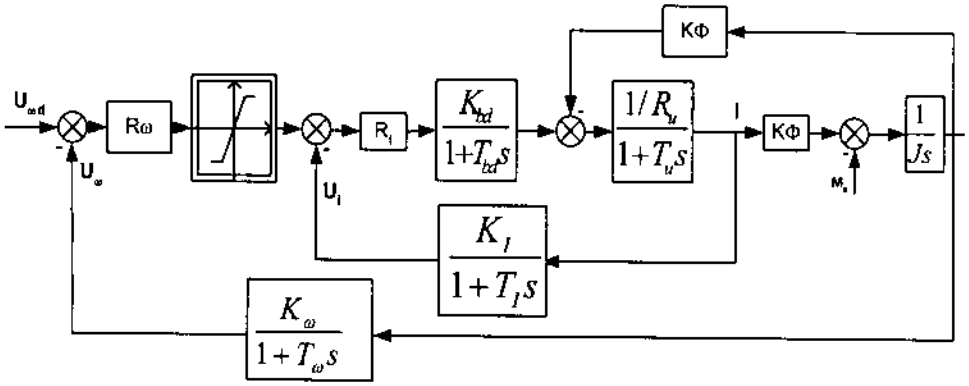
Đặc tính dòng điện và tốc độ động cơ một chiều khảo sát được hiển thị trên các *Graph* được hiển thị trên hình 11.8.



Hình 11.8. Đáp ứng dòng và tốc độ của động cơ điện một chiều.

Ví dụ 10: Khảo sát mạch vòng điều chỉnh dòng điện và tốc độ của động cơ điện một chiều kích từ độc lập (các thông số định mức như trong ví dụ 9) với

các bộ điều khiển dòng điện và tốc độ được tổng hợp theo phương pháp môđun tối ưu và môđun đối xứng. Sơ đồ cấu trúc tổng quát của hệ thống điều chỉnh dòng điện và tốc độ của động cơ điện một chiều trên hình 11.9.



Hình 11.9. Sơ đồ cấu trúc hệ điều chỉnh hai mạch vòng dòng điện và tốc độ cho động cơ điện một chiều kích từ độc lập.

Các hàm truyền của các phần tử trong mạch được tính toán, tổng hợp và hiệu chỉnh như kết quả dưới đây:

Hàm truyền bộ biến đổi:

$$W_{bd} = \frac{K_{bd}}{1+T_{bd}s} = \frac{22}{0,005s+1} \qquad (11 - 7)$$

Hàm truyền bộ đo dòng điện:

$$W_I = \frac{K_I}{1+T_I s} = \frac{0,79}{0,001s+1} \qquad (11 - 8)$$

Hàm truyền bộ điều chỉnh dòng:

$$W_{RI} = K_{P_I} + \frac{K_{I_I}}{s} = 0,6 + \frac{10}{s} = \frac{1+0,06s}{0,1s} \qquad (11 - 9)$$

Hàm truyền máy phát tốc:

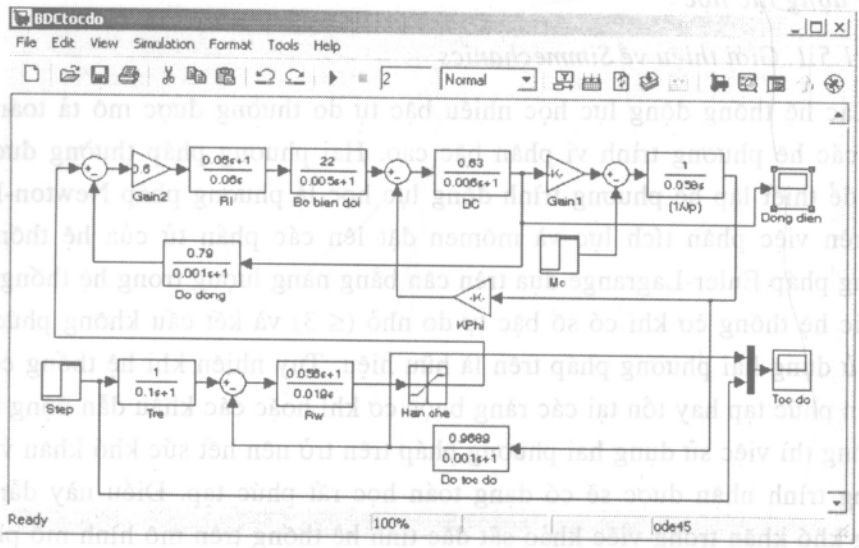
$$W_{F\text{-}T} = \frac{K_\omega}{1+T_\omega s} = \frac{0,9689}{0,001s+1} \qquad (11 - 10)$$

Hàm truyền bộ điều chỉnh tốc độ:

$$W_{R\omega} = K_{P_\omega} + \frac{K_{I_\omega}}{s} = 2,95 + \frac{52}{s} = \frac{1+0,056s}{0,019s} \qquad (11 - 11)$$

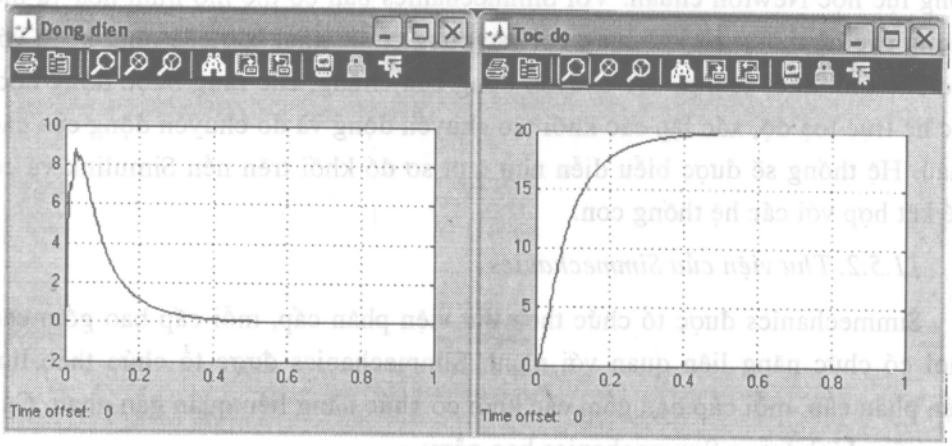
Mô men tải M_c trong trường hợp này được chọn bằng 0.

Chương trình mô phỏng thực hiện trên Simulink như sau:



Hình 11.10. Mạch điều chỉnh dòng điện
và tốc độ động cơ một chiều trên Simulink.

Dòng điện và tốc độ động cơ được hiển thị trên các *scope* như sau



Hình 11.11. Đáp ứng dòng điện và tốc độ động cơ điện một chiều.

Trên đây là ví dụ đơn giản về mô phỏng hệ truyền động động cơ một chiều. Bạn đọc có thể phát triển mô phỏng các hệ thống truyền động cơ xoay chiều phức tạp hơn.

11.5. Ứng dụng ToolBox Simmechanics trong MATLAB mô phỏng hệ thống động lực học

11.5.1. Giới thiệu về Simmechanics

Các hệ thống động lực học nhiều bậc tự do thường được mô tả toán học bằng các hệ phương trình vi phân bậc cao. Hai phương pháp thường được sử dụng để thiết lập hệ phương trình động lực học là phương pháp Newton-Euler dựa trên việc phân tích lực và mômen đặt lên các phần tử của hệ thống và phương pháp Euler-Lagrange dựa trên cân bằng năng lượng trong hệ thống. Đối với các hệ thống cơ khí có số bậc tự do nhỏ (≤ 3) và kết cấu không phức tạp, việc sử dụng hai phương pháp trên là hữu hiệu. Tuy nhiên khi hệ thống cơ khí trở nên phức tạp hay tồn tại các ràng buộc cơ khí hoặc các khâu dẫn động trong hệ thống thì việc sử dụng hai phương pháp trên trở nên hết sức khó khăn và các phương trình nhận được sẽ có dạng toán học rất phức tạp. Điều này dẫn đến những khó khăn trong việc khảo sát đặc tính hệ thống trên mô hình mô phỏng. ToolBox Simmechanics trong MATLAB là giải pháp thay thế hữu hiệu trong công việc này.

Simmechanics là phần mềm mô hình hoá dạng sơ đồ khối cho thiết kế và mô phỏng kỹ thuật các hệ thống máy móc và chuyển động của chúng trên cơ sở động lực học Newton chuẩn. Với Simmechanics bạn có thể mô hình hoá và mô phỏng các hệ thống cơ khí cùng với các công cụ để khai báo các từng khâu với các đặc tính khối lượng, các chuyển động của chúng, các ràng buộc động học, các hệ trục toạ độ, xác lập các khối tạo chuyển động và đo chuyển động của các khâu. Hệ thống sẽ được biểu diễn như một sơ đồ khối trên nền Simulink và có thể kết hợp với các hệ thống con.

11.5.2. Thư viện của Simmechanics

Simmechanics được tổ chức theo thư viện phân cấp, mỗi cấp bao gồm các khối có chức năng liên quan với nhau. Simmechanics được tổ chức theo thư viện phân cấp, mỗi cấp bao gồm các khối có chức năng liên quan gần nhau. Các thư viện chính trong Simmechanics bao gồm:

- Thư viện các khâu cơ khí (bodies library)

- Thư viện các khớp nối (joints library)

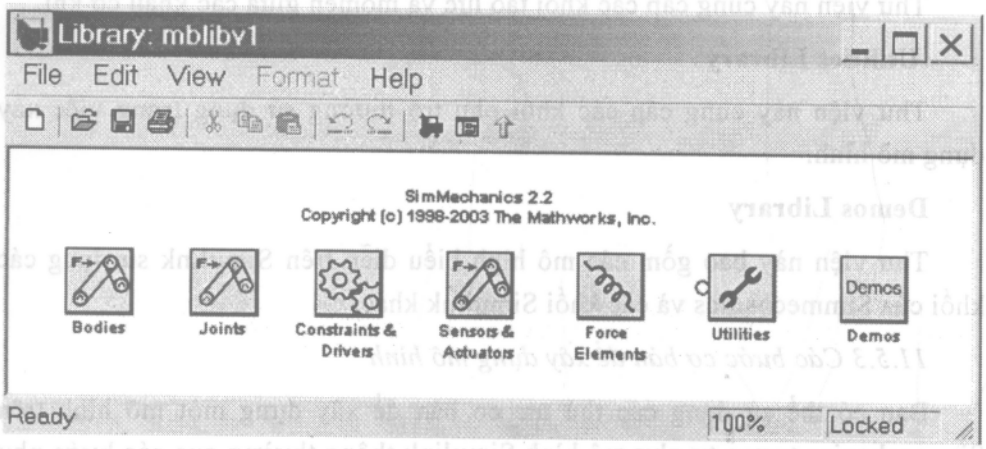
- Thư viện các hệ truyền động và ràng buộc (Drivers and constraints library)

- Thư viện các cơ cấu chấp hành và cảm biến (Actuators and Sensors Library)

- Thư viện các phần tử tạo động lực (Force elements Library)

- Thư viện các phần tử phụ trợ (Utilities library)

- Thư viện các ví dụ mẫu (Demos library)



Hình 11.12. Thư viện của ToolBox Simmechanics.

Bodies Library

Thư viện này cung cấp các khâu cơ khí ví dụ như thanh nối trong các robot, con lắc ngược hoặc các khâu trong hệ thống băng tải. Các tham số khai báo cho khâu cơ khí bao gồm khối lượng/mômen quán tính, vị trí và hướng và các hệ trục toạ độ gắn trên nó. Thư viện còn bao gồm khối điểm gốc (Ground block) biểu diễn một điểm cố định và khối môi trường máy (Machine environment block) để biểu diễn vị trí của hệ thống trong không gian ba chiều.

Joints Library

Thư viện này cung cấp các khối khớp nối tạo chuyển động tương đối của các kết cấu cơ khí. Ví dụ như các khớp nối trong cơ cấu robot. Hai loại khớp cơ bản là khớp tịnh tiến (prismatic joint) và khớp quay(revolute/spherical joints).

Constraints & Drivers Library

Thư viện này cung cấp các khối biểu diễn các ràng buộc về chuyển động độc lập (số bậc tự do) giữa các khâu cơ khí. Các ràng buộc này có thể là phụ thuộc vào không phụ thuộc thời gian.

Sensors & Actuators Library

Thư viện các cơ cấu chấp hành và cảm biến dùng khởi tạo và đo chuyển động giữa các khớp nối và khâu cơ khí.

Force Elements Library

Thư viện này cung cấp các khối tạo lực và mômen giữa các khâu cơ khí.

Utilities Library

Thư viện này cung cấp các khối phụ trợ thường sử dụng trong việc xây dựng mô hình.

Demos Library

Thư viện này bao gồm các mô hình biểu diễn trên Simulink sử dụng các khối của Simmechanics và các khối Simulink khác.

11.5.3 Các bước cơ bản để xây dựng mô hình

Bạn có thể sử dụng các thủ tục cơ bản để xây dựng một mô hình trên Simmechanics tương tự như mô hình Simulink thông thường qua các bước như sau:

1. Chọn các khối điểm gốc, khâu cơ khí, khớp nối.

Từ các thư viện, bạn chọn các khối gốc, khâu cơ khí, khớp nối cần thiết , sử dụng thao tác "nhấn và kéo" để biểu diễn mô hình trong cửa sở mô hình của Simulink.

Trước tiên chúng ta cần định nghĩa một hệ trục toạ độ tuyệt đối được qui ước chuẩn, gọi là hệ trục toạ độ WORLD. Tâm của hệ trục toạ độ WORLD đặt tại điểm (0,0,0). Trong Simmechanics, chúng ta sử dụng khối Env trong thư viện Bodies để đặt chiều của các trục toạ độ của WORLD. Các hệ trục toạ độ trong hệ thống có thể được định nghĩa theo hệ trục toạ độ WORLD.

Các trục toạ độ của WORLD trong Simmechanics có chiều mặc định như sau:

+ trục *x* chỉ sang phải

+ trục y ngược chiều với chiều của lực trọng trường

+ trục z được chọn theo qui tắc bàn tay phải

Vector trọng trường trong trường hợp này là $[0, -9.81, 0]^T$.

Ngoài ra chúng ta còn có thể tạo thêm một số điểm đất (ground) trong hệ trục toạ độ WORLD. Mỗi cơ cấu máy đều có ít nhất một điểm đất được biểu diễn trong Simmechanics bởi khối đất (Ground block) và được nối với khối môi trường máy (machine environment block).

2. *Định vị và nối các khối.*

Lựa chọn các khâu cơ khí từ thư viện Bodies, các khớp từ thư viện Joints, các cảm biến và chấp hành từ thư viện Sensor& Actuator... Đặt các khâu cơ khí và các khớp nối trong cửa sổ mô hình Simulink và nối nó theo đúng thứ tự. Kết quả của thao tác này sẽ cho một sơ đồ khối như sau:

Khối môi trường máy -- Điểm gốc cố định-- khớp nối -- khâu cơ khí-- khớp -- khâu cơ khí...-- khâu cơ khí...

3. Khai báo tham số cho khâu cơ khí

Nháy kép vào từng khâu cơ khí khai báo các tham số của nó như sau:

+ Tham số về đặc tính khối lượng: khối lượng và mômen quán tính. Khối lượng là số thực dương. Mômen quán tính là ma trận vuông cấp 3.

+ Tham số về hướng và vị trí của các hệ trục toạ độ gắn trên khâu cơ khí trong hệ trục toạ độ WORLD hoặc trong các hệ trục toạ độ đã được định nghĩa khác. Các hệ trục toạ độ gắn cố định trên các khâu và chuyển động cùng với các khâu đó. Có ít nhất một hệ trục toạ độ của mỗi khâu cơ khí gắn tại trọng tâm, số lượng mặc định là ba.

4. Khai báo tham số cho các khớp

Nháy kép vào các khối khớp nối để mở hộp thoại và xác định trục tịnh tiến hay trục quay.

5. Chọn, nối và xác định cấu hình cho các khối truyền động và ràng buộc

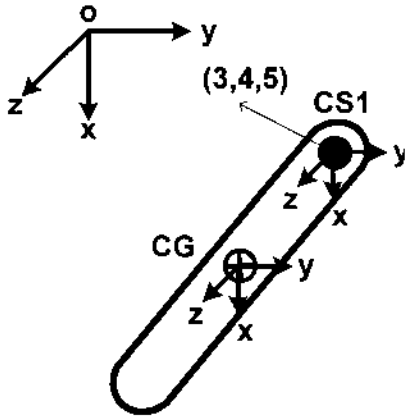
Từ thư viện khối truyền động và ràng buộc, chọn, xác định cấu hình và nối các khối truyền động và ràng buộc giữa khâu cơ khí.

6. *Chọn, nối và xác định cấu hình các cảm biến và chấp hành*

7. *Chọn chế độ hiển thị*

8. *Chạy mô phỏng*

Ví dụ 1. Mô phỏng chuyển động của con lắc đơn làm bằng thanh kim loại như trên hình vẽ. Chiều dài con lắc là 1m, một đầu gắn cố định tại tọa độ [3,4,5] trong hệ trục toạ độ WORLD. Ban đầu con lắc được kéo lệch 90° từ vị trí cân bằng. Giả thiết thanh kim loại có kết cấu đồng đều, do vậy điểm trọng tâm gắn cố định tại điểm giữa thanh có tọa độ ban đầu là [2.5,4, 5]



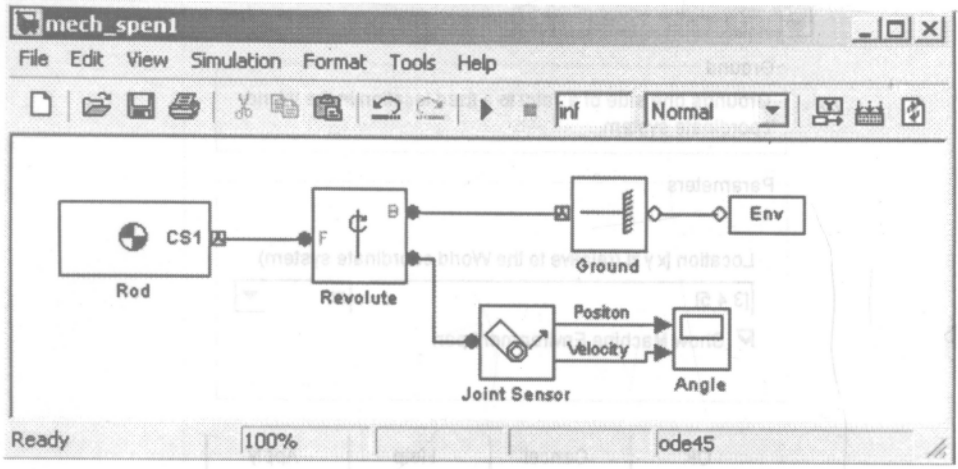
Hình 11.13. Con lắc đơn.

Bảng 11-1. Thông số của con lắc đơn

Thuộc tính	Đơn vị đo	Giá trị
Khối lượng	gam	2490
Mômen quán tính	kg-m^2	[1.25e-4 0 0; 0 0.208 0; 0 0 0.208]
Vị trí điểm trọng tâm	m	[2.5 4 5]
Vị trí tâm hệ truc toạ độ thứ nhất	m	[3 4 5]

Các bước tiến hành:

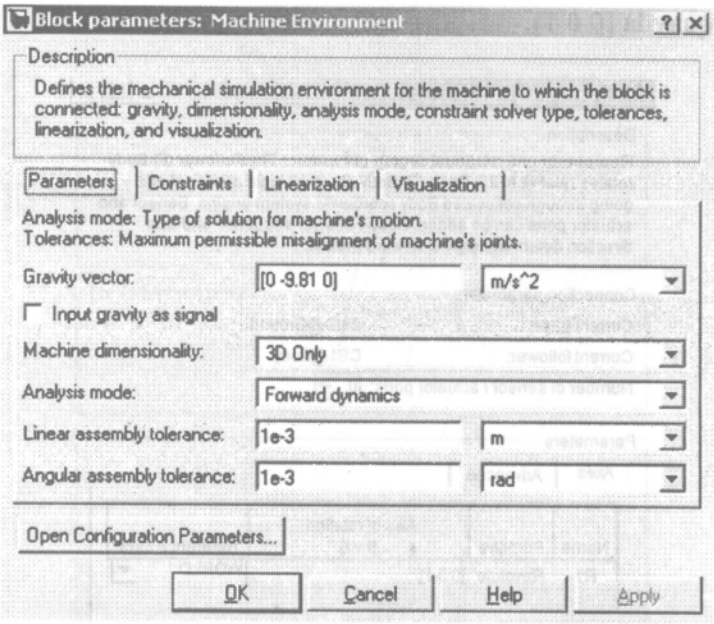
1. *Xây dựng mô hình hệ thống trên Simulink*



Bảng 11.14. Mô hình con lắc đơn trên Simmechanics.

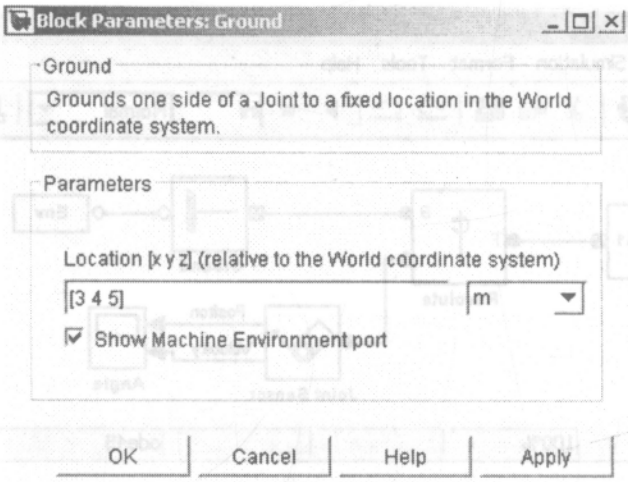
2. Tiến hành khai báo các thông số hệ thống

+ Khối môi trường máy Machine Environment. Ở đây chúng ta sẽ xác định hệ trục toạ độ WORLD thông qua việc khai báo vector trọng trường [0 -9.81 0].



Bảng 11.15. Khai báo vectơ trọng trường trong hệ trục tọa độ WORLD.

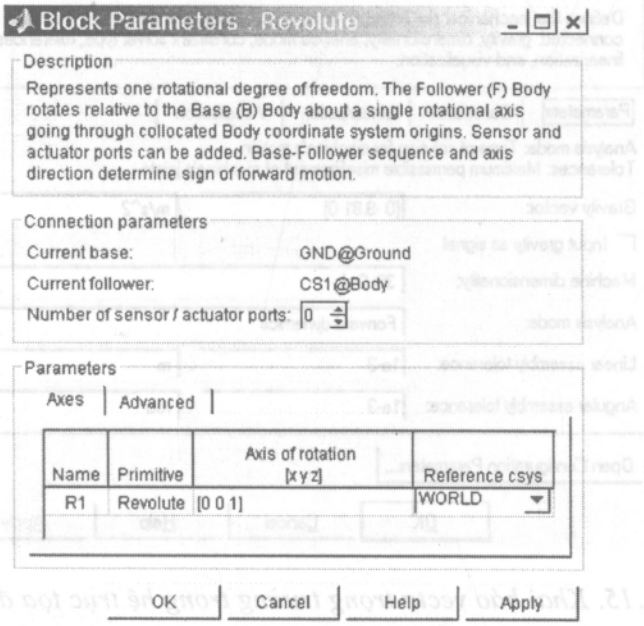
+ Khối đất Ground. Con lắc được gắn cố định tại điểm mút của thanh kim loại. Vị trí của điểm này trong hệ trục toạ độ WORLD là $[3,4,5]^T$.



Hình 11.16. Khai báo vị trí điểm cố định của con lắc.

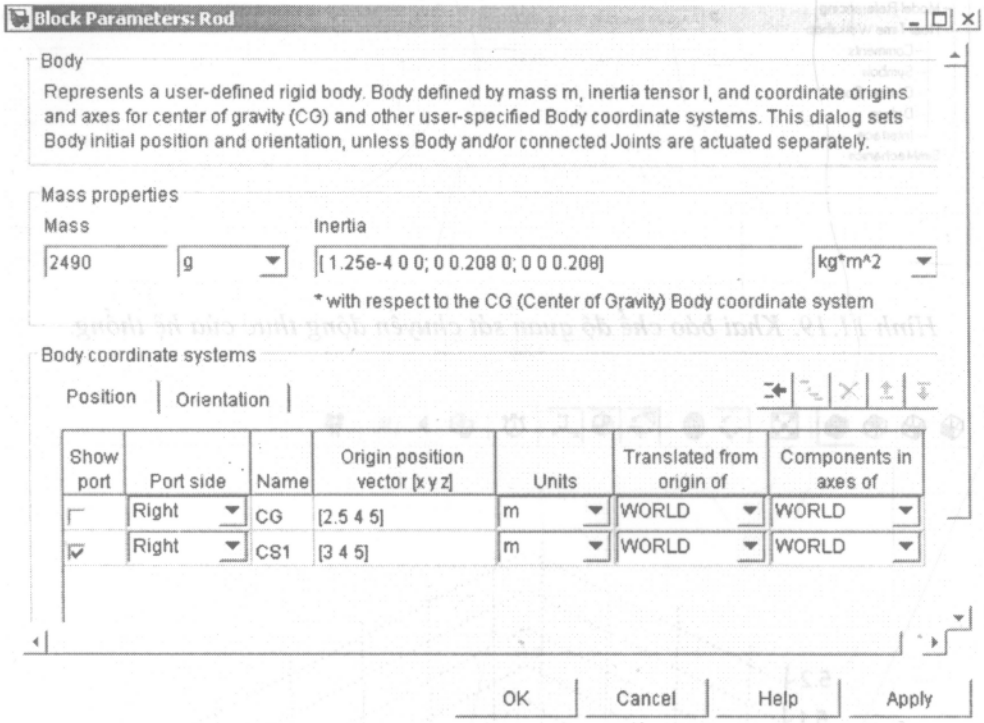
+ Khớp quay

Con lắc được kéo ra khỏi vị trí cân bằng và quay xung quanh trục +z của hệ trục toạ độ WORLD. Do vậy chúng ta khai báo số cơ cấu chấp hành là 0 và vector trục quay là [0 0 1].



Hình 11.17. Khai báo khớp quay.

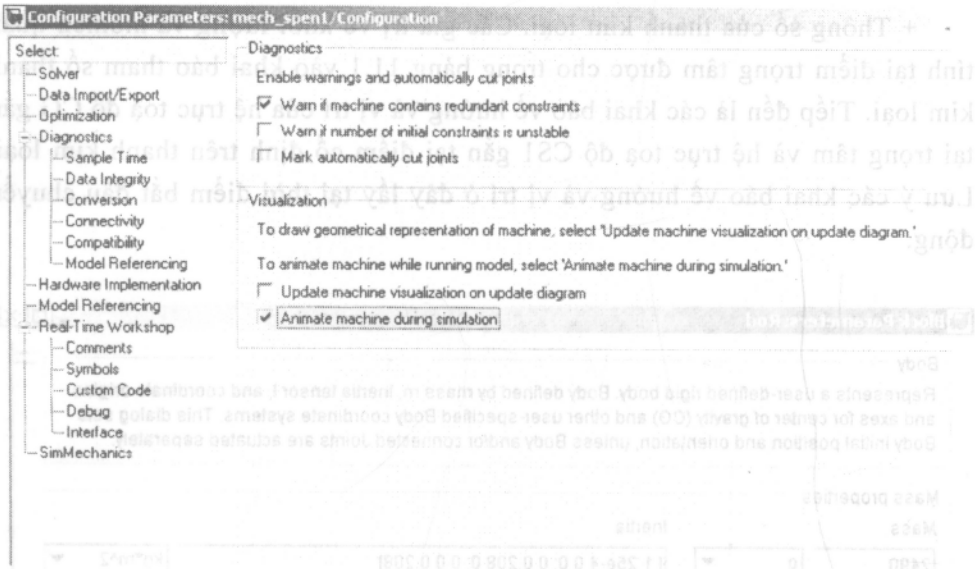
+ Thông số của thanh kim loại. Các giá trị về khối lượng và mômen quán tính tại điểm trọng tâm được cho trong bảng 11.1 vào khai báo tham số thanh kim loại. Tiếp đến là các khai báo về hướng và vị trí của hệ trục toạ độ CG gắn tại trọng tâm và hệ trục toạ độ CS1 gắn tại điểm cố định trên thanh kim loại. Lưu ý các khai báo về hướng và vị trí ở đây lấy tại thời điểm bắt đầu chuyển động.



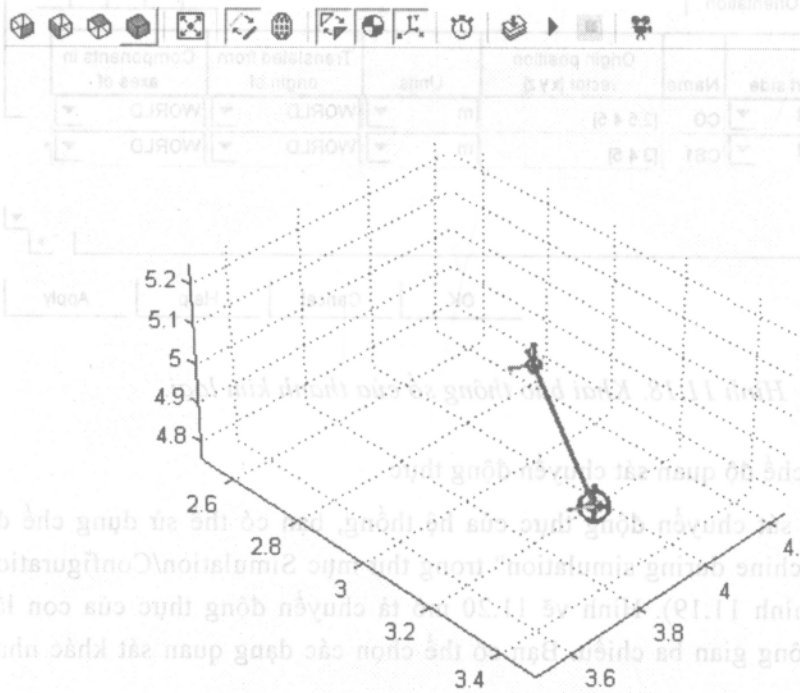
Hình 11.18. Khai báo thông số của thanh kim loại.

3. Chọn chế độ quan sát chuyển động thực

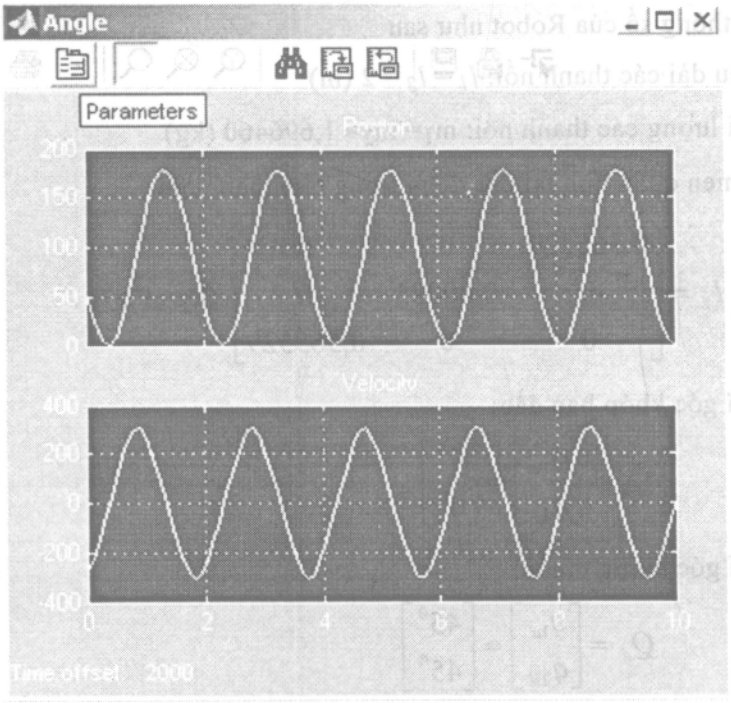
Để quan sát chuyển động thực của hệ thống, bạn có thể sử dụng chế độ "Animate machine during simulation" trong thư mục Simulation/Configuration Parameters (hình 11.19). Hình vẽ 11.20 mô tả chuyển động thực của con lắc đơn trong không gian ba chiều. Bạn có thể chọn các dạng quan sát khác nhau trên Menu.



Hình 11.19. Khai báo chế độ quan sát chuyển động thực của hệ thống.

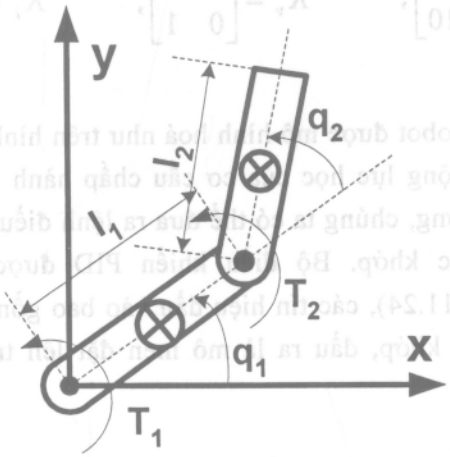


Hình 11.20. Chuyển động của hệ thống thực.



Hình 11.21. Đồ thị vị trí và tốc độ của con lắc.

Ví dụ 2: Ứng dụng tiếp theo được mở rộng cho bài toán điều khiển góc khớp của Robot Planar gồm hai khớp quay như trên hình vẽ 11.22.



Hình 11.22. Robot Planar.

Các thông số của Robot như sau

Chiều dài các thanh nối: $l_1 = l_2 = 2$ (m)

Khối lượng các thanh nối: $m_1 = m_2 = 1{,}696460$ (kg)

Mô men quán tính tại các điểm trọng tâm thanh nối

$$I_1 = I_2 = \begin{bmatrix} 0{,}565529 & 0 & 0 \\ 0 & 0{,}00085 & 0 \\ 0 & 0 & 0{,}565529 \end{bmatrix} \text{(kg-m}^2\text{)}$$

Vị trí góc khớp ban đầu:

$$Q_0 = \begin{bmatrix} q_{10} \\ q_{20} \end{bmatrix} = \begin{bmatrix} 0^0 \\ 0^0 \end{bmatrix}$$

Vị trí góc mong muốn:

$$Q_d = \begin{bmatrix} q_{1d} \\ q_{2d} \end{bmatrix} = \begin{bmatrix} 45^0 \\ 45^0 \end{bmatrix}$$

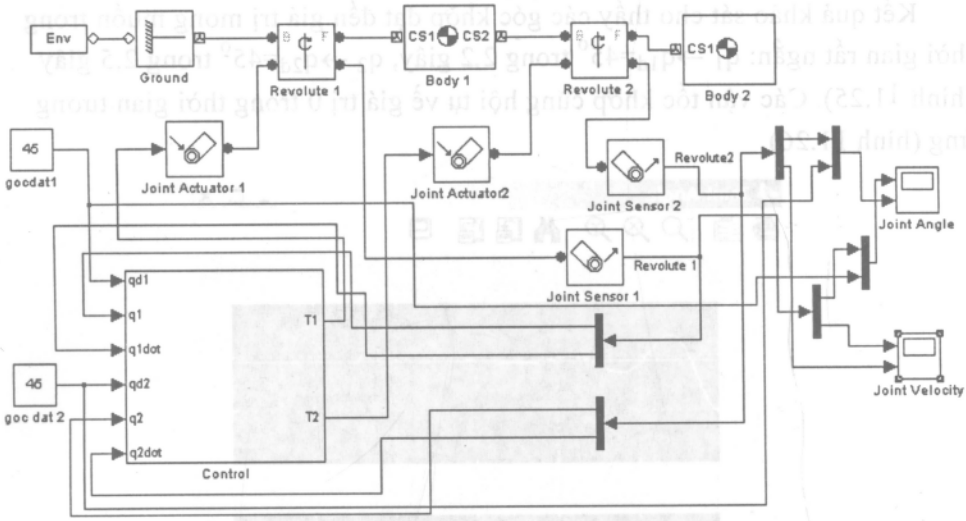
Bộ điều khiển vị trí được sử dụng là PID

$$T = K_p(Q_d - Q) - K_v\dot{Q} + Ki\int(Q_d - Q)dt$$

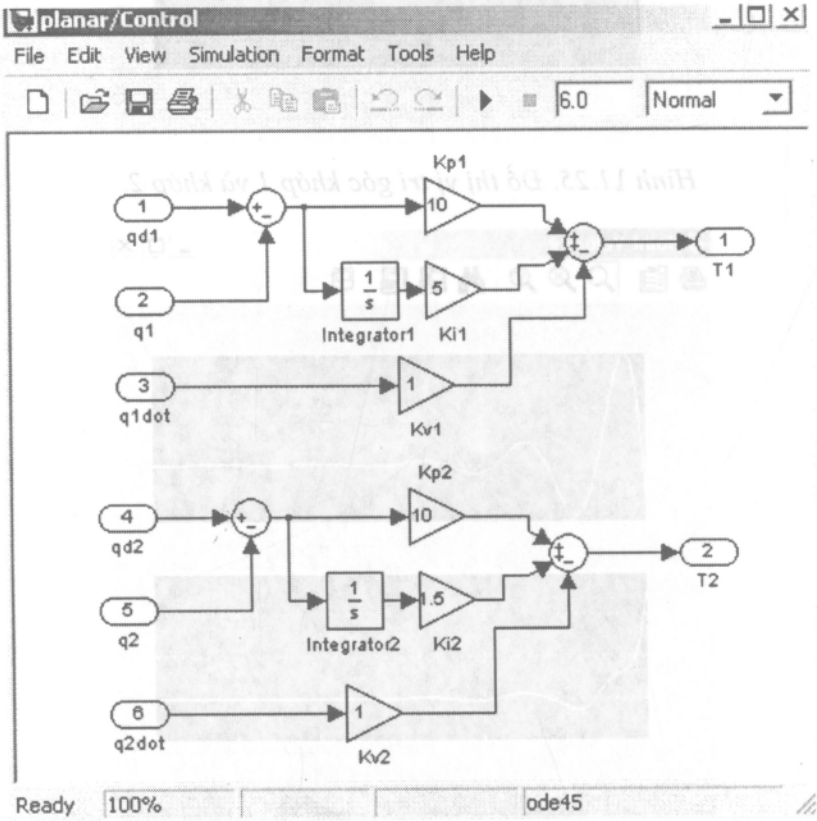
với các thông số như sau

$$K_p = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, \qquad K_v = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad K_i = \begin{bmatrix} 5 & 0 \\ 0 & 1{,}5 \end{bmatrix}$$

Hệ điều khiển Robot được mô hình hoá như trên hình 11.23. Ở đây, chúng ta bỏ qua bài toán động lực học của cơ cấu chấp hành dựa trên giả thiết các động cơ này là lý tưởng, chúng ta có thể đưa ra lệnh điều khiển để tạo mô men bất kỳ trên trục các khớp. Bộ điều khiển PID được tạo bởi Subsystem Planar/control (hình 11.24), các tín hiệu đầu vào bao gồm giá trị đặt và giá trị hiện tại của các góc khớp, đầu ra là mô men đặt lên trục của các khớp nối Robot.

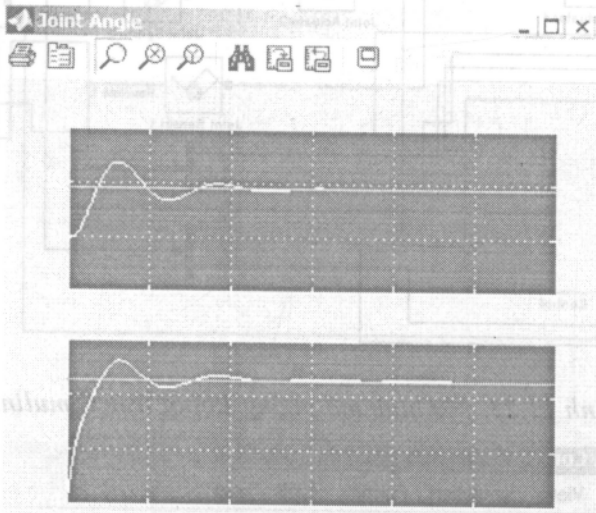


Hình 11.23. Mô hình mô phỏng Robot trên Simulink.

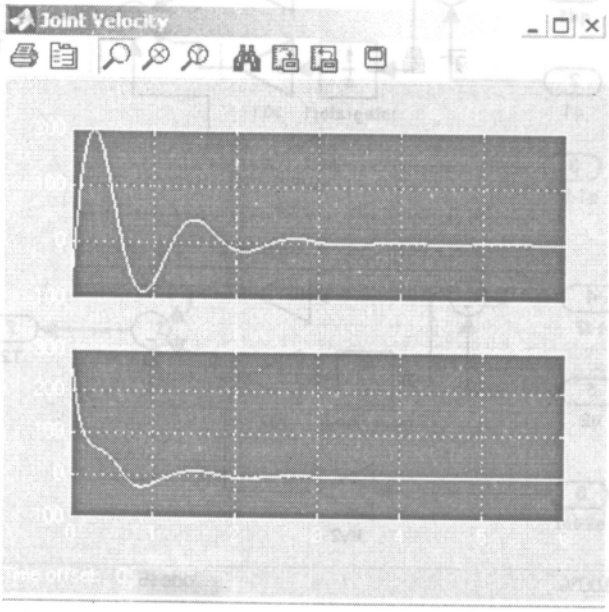


Hình 11.24. Bộ điều khiển PID.

Kết quả khảo sát cho thấy các góc khớp đạt đến giá trị mong muốn trong thời gian rất ngắn: $q_1 \rightarrow q_{1d}=45^o$ trong 2,2 giây, $q_2 \rightarrow q_{2d}=45^o$ trong 2,5 giây (hình 11.25). Các vận tốc khớp cũng hội tụ về giá trị 0 trong thời gian tương ứng (hình 11.26).

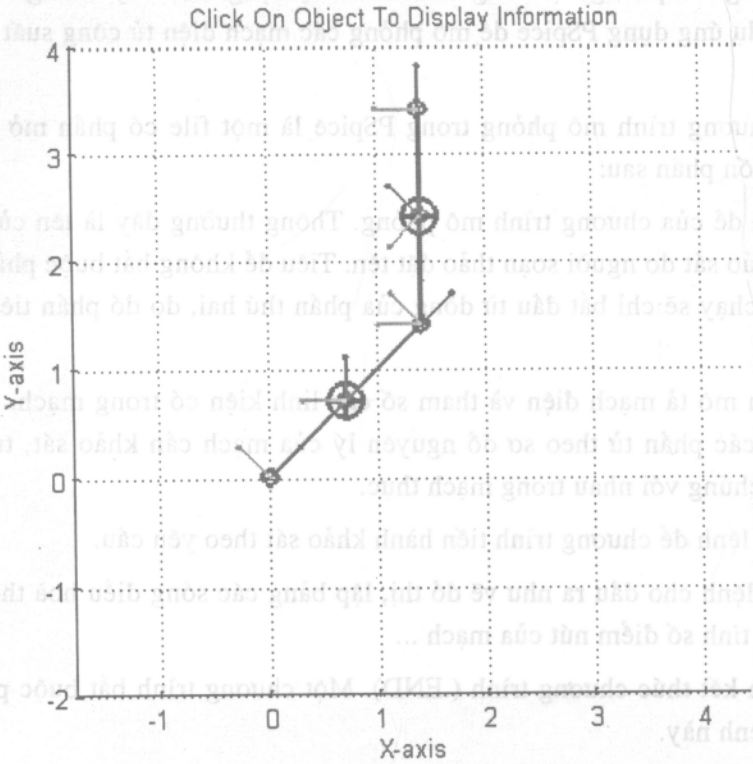


Hình 11.25. Đồ thị vị trí góc khớp 1 và khớp 2.



Hình 11.26. Đồ thị vận tốc khớp.

Chọn chế độ quan sát hệ thống thực trong không gian hai chiều chúng ta cũng có thể thấy kết quả tương tự. Sau 6 giây chuyển động, thanh nối thứ nhất nghiêng một góc 45° so với trục x và thanh nối thứ hai nghiêng một góc 45° so với thanh nối thứ nhất như yêu cầu điều khiển (hình 11.27).



Hình 11.27. Vị trí con lắc sau 6 giây chuyển động.

Trên đây là những ví dụ rất đơn giản về mô phỏng hệ thống động lực học trên nền Simmechanics. Bạn đọc có thể xây dựng mô hình các hệ thống cơ khí có số bậc tự do lớn hơn, bổ sung thêm các hệ thống dẫn động về cơ khí và các ràng buộc nếu có. Mặt khác, bạn có thể sử dụng ToolBox SimPowerSystems của Simulink để đưa thêm vào mô hình các cơ cấu chấp hành như động cơ điện một chiều, động cơ không đồng bộ, động cơ đồng bộ và các hệ truyền động tương ứng. Điều đó sẽ góp phần đưa mô hình càng gần với hệ thống thực tế hơn.

11.6. Ứng dụng phần mềm PSPICE mô phỏng mạch điện tử công suất

Phần mềm chuyên dụng PSPICE *(Power Simulation Program with Itergrated Circuit Emphases)* là một phần mềm mô phỏng mạch điện - điện tử mạnh và phổ biến trên thế giới. Phần mềm này cũng thông dụng như phần mềm MatLab trong mô phỏng hệ thống điều khiển tự động. Sau đây chúng ta đưa ra một vài ví dụ ứng dụng PSpice để mô phỏng các mạch điện tử công suất (Power Electronic)

Một chương trình mô phỏng trong PSpice là một file có phần mở rộng là .CIR gồm bốn phần sau:

1. Tiêu đề của chương trình mô phỏng. Thông thường đây là tên của mạch điện cần khảo sát do người soạn thảo đặt tên. Tiêu đề không bắt buộc phải có, vì PSpice khi chạy sẽ chỉ bắt đầu từ dòng của phần thứ hai, do đó phần tiêu đề có thể bỏ.

2. Phần mô tả mạch điện và tham số các linh kiện có trong mạch. Ở phần này ta đưa các phần tử theo sơ đồ nguyên lý của mạch cần khảo sát, tương tự như ta hàn chúng với nhau trong mạch thực.

3. Các lệnh để chương trình tiến hành khảo sát theo yêu cầu.

4. Các lệnh cho đầu ra như vẽ đồ thị, lập bảng các sóng điều hoà theo khai triển furier, tính số điểm nút của mạch ...

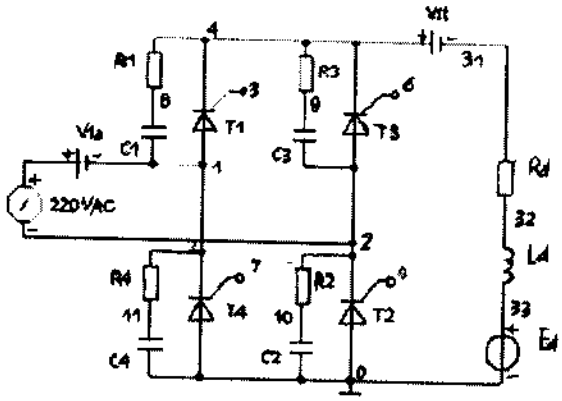
5. Lệnh kết thúc chương trình (.END). Một chương trình bắt buộc phải kết thúc bằng lệnh này.

Khi chạy mô phỏng PSPICE trước tiên sẽ duyệt file vào và tạo ra một file ra có cùng tên với file .CIR nhưng có phần mở rộng .OUT. Nếu file vào không có lỗi thì chương trình mô phỏng được thực hiện ngay và kết quả tính toán được đưa vào file cùng tên với phần mở rộng . DAT

11.6.1. Mô phỏng chỉnh lưu điều khiển cầu một pha bằng PSPICE [20]

Hình 11. 28 là sơ đồ chỉnh lưu điều khiển cầu 1 pha với tải là RLEd.

Chương trình mô phỏng dưới đây viết theo các điểm nút được đánh dấu trên sơ đồ. Các tiristor được mô phỏng theo dạng mạch con kiểu TXC, còn mạch điều khiển xây dựng dưới dạng mạch con kiểu pha-xung.



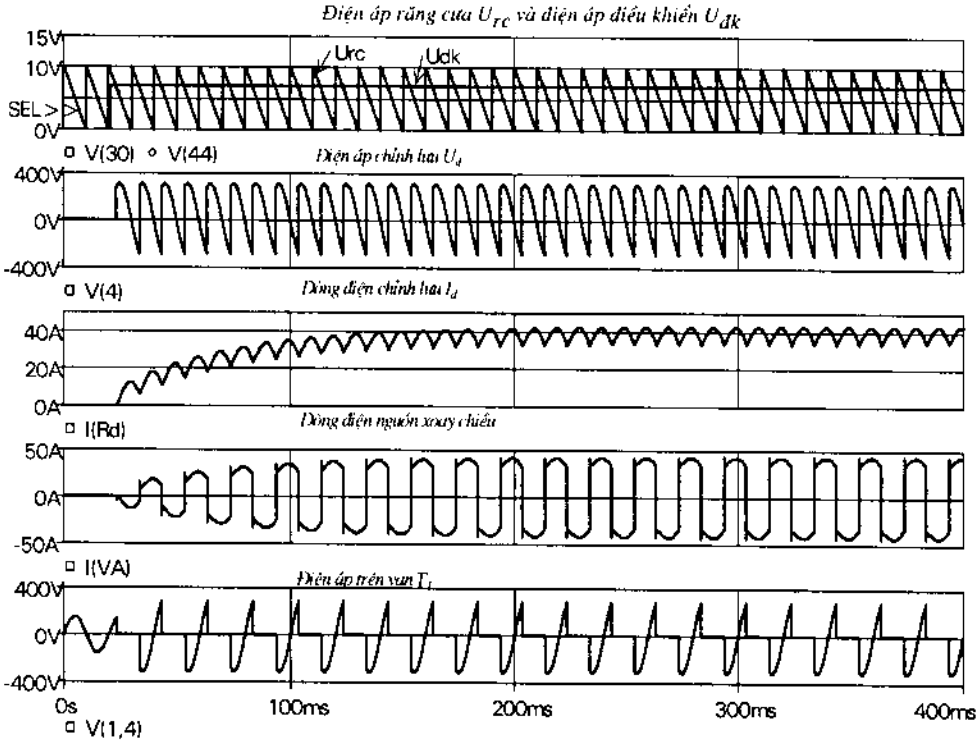
Hình 11.28. Sơ đồ chỉnh lưu cầu một pha.

KHAO SAT CHINH LUU DIEU KHIEN CAU MOT PHA (tan so 50 Hz)

```
Vdk 30 0 PWL(0,-0.01 21m,-0.01 21.2m,7 22m,7 10,8) ; điện áp điều khiển
XFxa 0 30 44 3 4 6 2 7 1 5 4 FX1pha      ;mạch phát xung cho pha A
VA 1 2 SIN(0 311 50)            ;điện áp nguồn 220VAC 50Hz
* 4 TIRISTOR
XT1 1 3 4  TXC  ; van T1 pha A mô hình dạng TXC
XT4 0 7 1  TXC  ; van T4 pha A mô hình dạng TXC
XT3 2 5 4  TXC  ; van T3 pha B mô hình dạng TXC
XT2 0 6 2  TXC  ; van T6 pha C mô hình dạng TXC
* Mach bao ve RC
C1 1 8 0.2U IC=0
R1 8 4 60
C4 0 11 0.2U IC=0
R4 11 1 60
C3 2 9 0.2U IC=0
R3 9 4 60
C2 0 10 0.2U IC=0
R2 10 2 60
* Tai mot chieu RLEd
VId 4 31 DC 0        ; đo dòng điện một chiều Id
```

```
Rd 31 32 2          ; điện trở tải 2 ôm
Ld 32 0 100m        ; điện cảm tải 100mH
Vd 33 0 DC 20       ; sức điện động tải 20V
.LIB C:\MSIM_8\USERLIB\nom2.LIB ; gọi thư viện các phần tử lực
.LIB C:\MSIM_8\USERLIB\nom1.LIB ; gọi thư viện mạch điều khiển
.TRAN 20u 0.4 0 50u UIC     ; lệnh khảo sát chế độ động
.PROBE              ; lệnh cho kết quả đồ thị
.OPTIONS ABSTOL=100U VNTOL=0.1 RELTOL=0.01 ITL5=0
LIMPTS=100000 NODE
.END
```

Kết quả mô phỏng thể hiện ở hình 11.29



Hình 11.29. Kết quả mô phỏng chỉnh lưu điều khiển cầu một pha với điện áp điều khiển nhảy cấp.

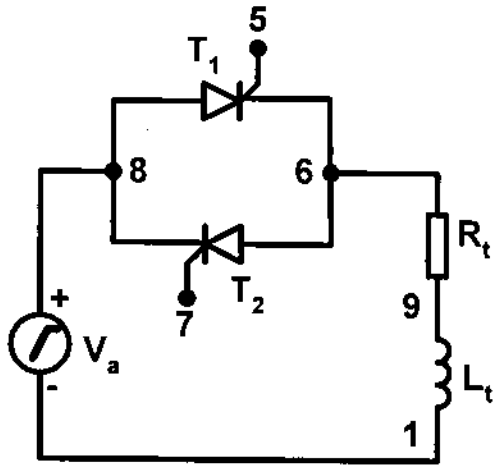
Mạch khảo sát với tác động nhảy cấp của điện áp điều khiển U_{dk} , trên đồ thị là điện áp *V(30)* và được so sánh với điện áp răng cưa *V(44)*.

Các đồ thị tiếp theo lần lượt là điện áp chỉnh lưu *V(4)*, dòng điện chỉnh lưu *I(R_d)*, dòng điện nguồn xoay chiều *I(VA)* và điện áp ngược trên tiristor *T1 :V(1,4)*.

Đồ thị cho thấy cả giai đoạn quá độ và khi mạch đạt tới chế độ xác lập.

11.4.1. Mô phỏng bộ điều áp xoay chiều một pha bằng PSPICE

Bộ điều chỉnh điện áp xoay chiều một pha dùng hai tiristor đấu song song ngược có tải điện trở và điện cảm được trình bày trên hình 11.30.



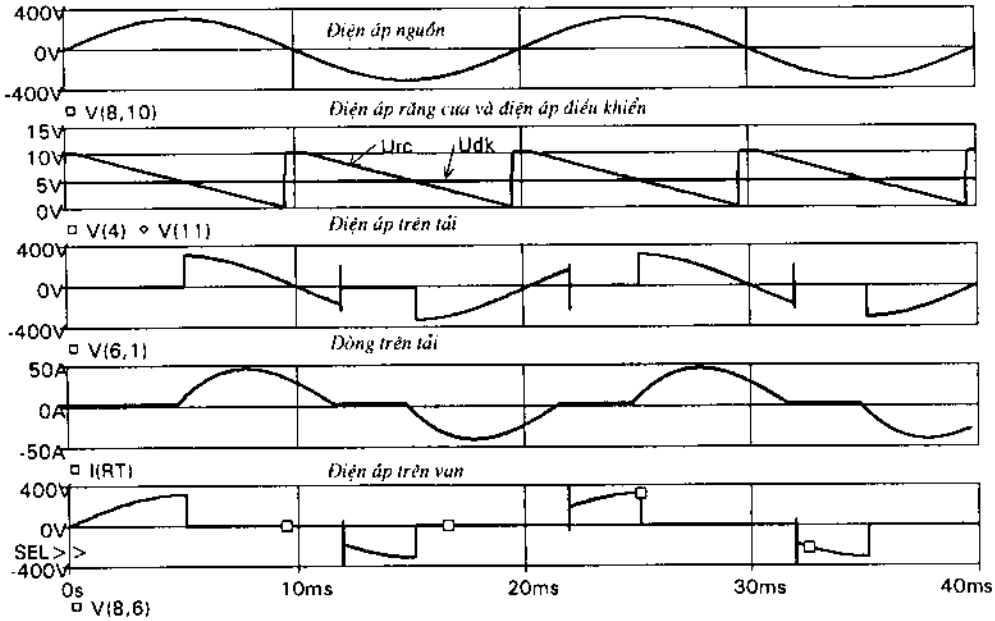
Hình 11.30. Bộ điều áp xoay chiều một pha.

Chương trình mô phỏng được viết như sau

```
DIEU AP MOT PHA XOAY CHIEU 2 THYRISTOR // NGUOC , tai RL
Vcc 3 0 DC 12      ; E=12V điện áp cấp cho mạch điều khiển
Edp 2 0 8 10 0.03    ; điện áp đồng pha
VDK 4 0 DC 5       ; điện áp điều khiển
Rrc 11 0 1meg
Xdk 0 2 3 4 5 6 7 8 11 DKTX ; mạch phát xung điều khiển cho 2 tiristor
T1,T2
* mach luc
 Va 8 10 SIN(0 311 50)  ; nguồn xoay chiều 220VAC 50Hz
 XT1 8 5 6 Ttri      ; tiristor T1
 XT2 6 7 8 Ttri      ; tiristor T2
```

```
* tai
RT 6 9 4        ; điện trở tải 4 ôm
LT 9 1 10m      ; điện cảm tải 10mH
VIt 1 10 DC 0   ; đo dòng điện tải
Rgh 10 0 100meg
.LIB C:\MSim_8\USERLIB\NOM2.LIB
.LIB C:\MSim_8\USERLIB\NOM1.LIB
.TRAN 1U 40M 10U 10u UIC
.PROBE
.OPTIONS ABSTOL=1U VNTOL=0.1 RELTOL=0.01 ITL5=20000
.END
```

Kết quả mô phỏng trên hình 11.31 là đồ thị điện áp và dòng điện trong mạch.



Hình 11.31. Kết quả mô phỏng điều áp xoay chiều một pha.

11.7. Ứng dụng phần mềm SIGMA mô phỏng hệ hàng đợi [26]

Phần mềm SIGMA *(Simulation Graphical Modeling and Analysis system)* là phần mềm dùng graph để phân tích và mô phỏng các mô hình sự kiện rời rạc. Phần mềm cho phép thêm, bớt, thay đổi các sự kiện và điều kiện mô phỏng mà

không cần dừng chạy mô phỏng để dịch lại chương trình. Phần mềm có khả năng chạy dưới dạng hoạt hình nên cho thấy một cách trực quan quá trình hoạt động của hệ thống. Phần mềm SIGMA cho kết quả mô phỏng dưới dạng số hoặc đồ thị sau mỗi bước mô phỏng, vì vậy rất thuận tiện trong việc theo dõi, phân tích hoạt động của hệ thống được mô phỏng.

Phần mềm SIGMA mô tả hệ thống bằng một graph gồm các đỉnh và các cạnh có hướng. Các đỉnh mô tả các trạng thái của hệ thống, các cạnh có hướng mô tả các điều kiện chuyển trạng thái của hệ thống. Các đỉnh và các cạnh có hướng đọc nối với nhau theo logic hoạt động của hệ thống. Đỉnh khởi tạo có tên là RUN luôn là đỉnh đầu tiên của mô hình (graph). Quá trình mô phỏng bao giờ cũng xuất phát từ đỉnh RUN, sau đó lần lượt chuyển sang đỉnh khác cho đến đỉnh kết thúc. Các bảng chọn cho phép định nghĩa và thay đổi tham số của các đỉnh và các cạnh có hướng cũng như thay đổi các điều kiện đầu và kết thúc mô phỏng.

Phần mềm SIGMA trong Windows (Sigma for Windows) cho kết quả bằng bảng số liệu và đồ thị dưới nhiều dạng khác nhau (Step plot, Line plot, Array plot, Ssanller plot, Histogram, Autocorrelation, Standardized Time Series) nên rất thuận tiện trong việc phân tích kết quả mô phỏng. Mô hình SIGMA (graph mô phỏng) có thể tự động dịch (với một kích chuột) sang các ngôn ngữ khác như C, Pascal và Fortran, thậm chí có thể dịch sang mô tả bằng tiếng Anh.

Dùng phần mềm SIGMA để mô phỏng hệ hàng đợi (xem Chương 10) rất thuận tiện, vì vậy sau đây sẽ trình bày một ví dụ mô phỏng hệ hàng đợi.

Để mô phỏng một hệ hàng đợi bằng phần mềm SIGMA trước hết phải xây dựng sơ đồ cấu trúc của hệ thống tức xây dựng mô hình hoạt động của hệ thống bằng graph mô phỏng. Tiếp đó xác định các biến, các yếu tố cần mô phỏng, các giá trị ban đầu và điều kiện thay đổi các biến đó. Sau đó viết lệnh cho các đỉnh trạng thái và các mũi tên có hướng trên graph mô phỏng.

Thực hiện chạy chương trình mô phỏng, nhận kết quả mô phỏng dưới dạng bảng số liệu và đồ thị mô phỏng sau đó phân tích các kết quả mô phỏng và đưa ra các giải pháp cải tiến để hệ thống hoạt động tốt hơn.

Ví dụ: Một trạm sửa chữa ôtô có một điểm phục vụ. Dòng ôtô đi đến trạm sửa chữa tuân theo luật phân phối mũ có cường độ $\lambda = 1,25$ *xe / giờ*. Thời gian sửa chữa trung bình (thời gian phục vụ) tuân theo luật phân phối mũ có cường

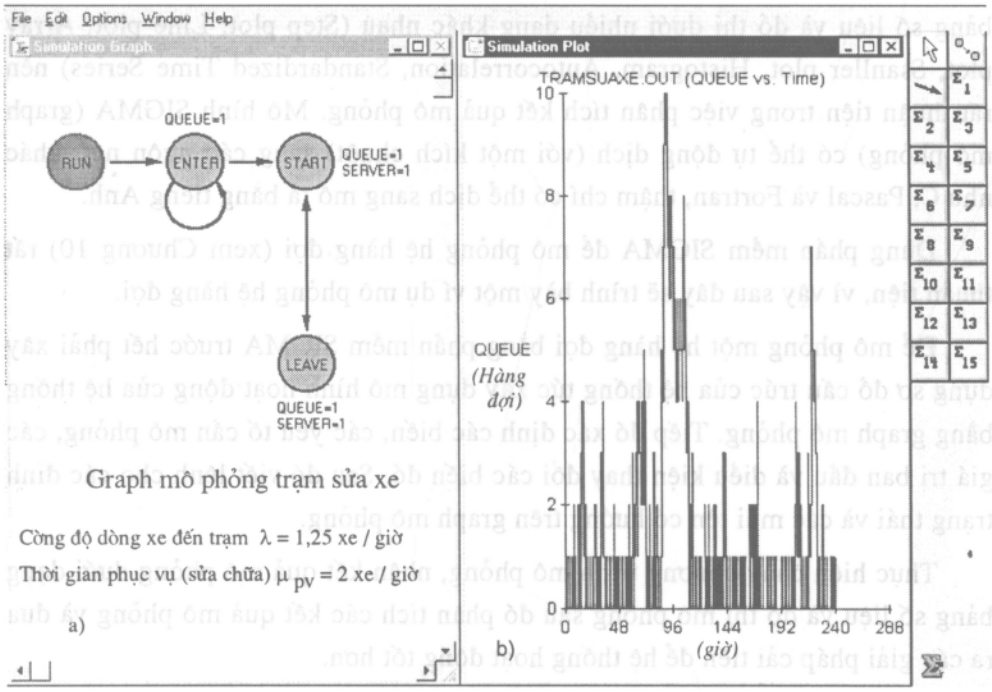
độ μ_{pv} = 2 *xe /giờ* . Trạm có bãi đỗ xe chứa được N = 5 xe để sắp hàng chờ đến lượt sửa chữa. Hãy mô phỏng quá trình hoạt động của trạm sau 240 giờ làm việc.

Giải:

- Dòng xe đến trạm sửa được coi là một dòng tối giản nên khoảng cách giữa các xe tuân theo luật phân phối mũ và được tính theo biểu thức sau $t_i = -1/\lambda * LN(RND)$, trong đó RND là số ngẫu nhiên phân phối đều trong khoảng (0,1).

- Dòng xe được phục vụ cũng được coi là một dòng tối giản nên thời gian phục vụ được tính như sau $t_{pv} = -1/\mu * LN(RND)$, trong đó RND là số ngẫu nhiên phân phối đều trong khoảng (0,1).

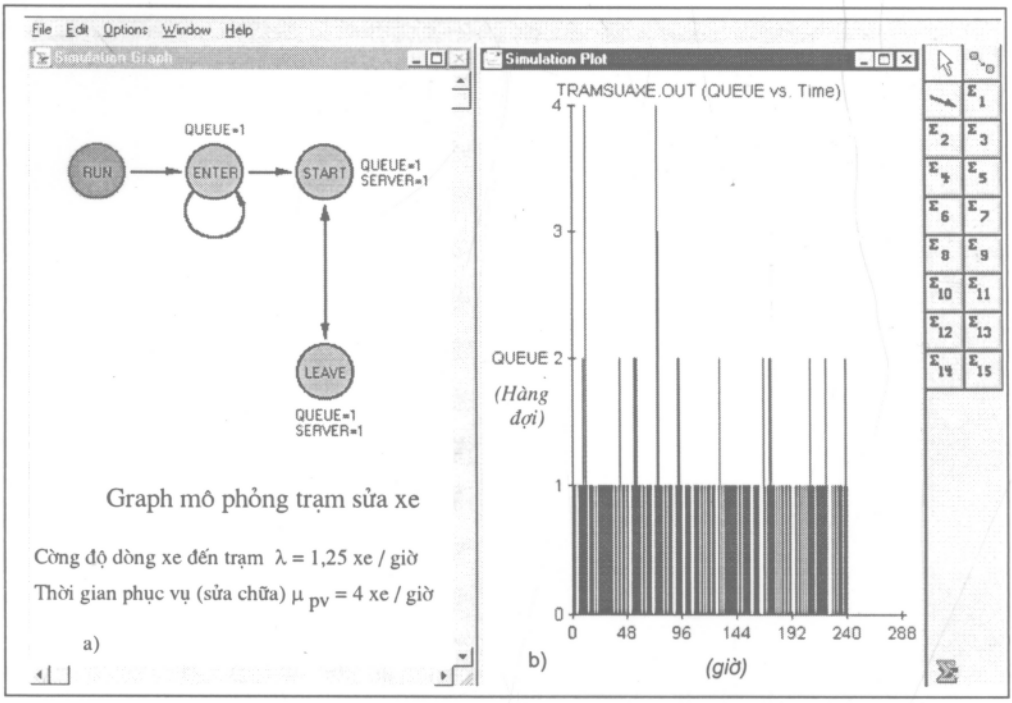
Kết quả mô phỏng sau 240 giờ làm việc như trên hình 11.32. Hình 11.32 a là graph mô phỏng trạm sửa xe. Hình 11.32 b là hàng đợi (Queue) theo thời gian



Hình 11.32. Mô phỏng trạm sửa xe:
a) graph mô phỏng; b) hàng đợi theo thời gia

Nhận xét hình 11.32 b ta thấy có nhiều thời điểm hàng đợi vượt quá 5, tại thời điểm này trạm sẽ mất khách hàng vì không còn chổ để sắp hàng chờ đợi. Để khắc phục tình trạng mất khách hàng có thể chọn giải pháp tăng năng lực phục vụ lên gấp đôi $\mu_{pv} = 4$ *xe /giờ.*

Hình 11.33 là kết quả mô phỏng sau khi tăng năng lực phục vụ lên gấp đôi.



Hình 11.33. Mô phỏng trạm sửa xe:
a) graph mô phỏng; b) hàng đợi theo thời gian.

Nhận xét hình 11.33 b ta thấy sau khi năng lực phục vụ tăng lên gấp đôi thì hàng đợi lớn nhất chỉ là 4, như vậy trạm không bị mất khách hàng nữa.

TÀI LIỆU THAM KHẢO

1. Averill M. Law, W. David Kelton (1991). *Simulation Modeling and Analysis*. Mcgraw-Hill, Inc.
2. Geoffrey Gordon. *System Simulation* (1989). Prentice-Hall of India Private Limited.
3. Stewart V. Hoover, Ronald F. Perry (1989). *Simulation. A Problem-Solving Approach*. Addison-Wesley Publishing Company.
4. Barry L. Nelton (1995). *Stochastic Modeling*. McGraw-Hill, Inc.
5. Pastijn Hugo (1997). *Introduction Simulation*. AIT. Bangkok.
6. Voratas Kachitrychianukul (1998). *Discret Event Simulation of Production System*. AIT. Bangkok.
7. Sheldon M. Ross. *A Course in Simulation*. Macmillan Publishing Company. New York.
8. Worksims'94 (1994). *Simulation in Manufacturing System*. AIT. Bangkok.
9. James H. Taylor. *Modeling and Simulation of Nonlinear Dynamic Systems*. (2002). JUV RESCCE'02. Danang Vietnam.
10. Ouassima Alchrif. Mohsen Ghribi (1994). *Techniques de Simulation*. Ecole de technologie superieure. Quybec. Canada.
11. Landau I. Dore (1988). *Identification et Commande des Systems*. Hermes.
12. Советов Б. Я., Яковлев С. А. (1985). *Моделирование систем*. Высшая школа, Москва.

13. Кузовков И. Т., Карабаиов С. В., Салырев О. С. (1978). *Непрерывные и дискретные системы управления и методы идентификация*. Москва.

14. Bahram Shahian, Michel Hassul (1993). *Control System Design Using MATLAB*. Prentice-Hall.

15. Benjamin C. Kuo (1995). *Automatic Control Systems*. Prentice-Hall.

16. Katsuhiko Ogata (2001). *Modern Control Engineering*. Fourth Edition. Prentice-Hall.

17. Nguyễn Công Hiền (2001). *Mô hình hoá hệ thống và mô phỏng*. Đại học Bách khoa Hà Nội.

18. Thân Ngọc Hoàn (2002). *Mô phỏng hệ thống điện tử công suất và truyền động điện*. Nhà xuất bản Xây dựng, Hà Nội.

19. Tống Đình Quỳ (2002). *Giáo trình xác suất thống kê*. Nhà xuất bản Giáo dục, Hà Nội.

20. Võ Minh Chính, Phạm Quốc Hải, Trần Trọng Minh (2004). *Điện tử công suất*. Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.

21. Nguyễn Phùng Quang (2004). *MatLab & Simulink*. Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.

22. Lê Đình Anh, Nguyễn Công Hiền, Nguyễn Nam Tặng. *Tìm hiểu hệ thống quản lý tự động hoá*. (Dịch từ nguyên bản tiếng Nga của V. M. Gluscov). Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.

23. Trần Đình Long (1999). *Lý thuyết hệ thống*. Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.

24. Bùi Minh Trí (2005). *Mô hình toán kinh tế*. Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.

25. Bùi Quốc Khánh, Nguyễn Văn Liễn, Phạm Quốc Hải, Dương Văn Nghi (2004). *Điều chỉnh tự động truyền động điện*. Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội.

26. Schruben Lee. (1995). Graphical Simulation Modeling and Analysic Using SIGMA for Windows.

MỤC LỤC

MÔ HÌNH HOÁ HỆ THỐNG VÀ MÔ PHỎNG

Tác giả: GS. TS. NGUYỄN CÔNG HIỀN
TS. NGUYỄN PHẠM THỤC ANH

Chịu trách nhiệm xuất bản: PGS. TS. TÔ ĐĂNG HẢI
Biên tập và sửa bài: ThS. NGUYỄN HUY TIẾN
NGỌC LINH
Trình bày bìa: TRIỆU THẾ HÙNG

NHÀ XUẤT BẢN KHOA HỌC VÀ KỸ THUẬT
70 Trần Hưng Đạo - Hà Nội

In 1000 cuốn, khổ 16 × 24 cm, tại Xưởng in NXB Văn hoá Dân tộc
Quyết định xuất bản số: 730–2006/CXB/18–59/KHKT–5/10/2006
In xong và nộp lưu chiểu Quý IV năm 2006.

206364

Giá: 33.000đ