Chương 9
GIỚI THIỆU PHƯƠNG PHÁP MONTE CARLO

9.1. Mở đầu

Mục đích của chương là giới thiệu vắn tắt các cơ sở của kỹ thuật Monte Carlo để ước tính giá trị của tham số. Vì thế, ta chỉ đề cập vài khía cạnh quan trọng của kỹ thuật ước tính Monte Carlo nhằm định nghĩa phương pháp Monte Carlo và nghiên cứu một số kỹ thuật căn bản một cách đơn giản và dễ hiểu. Ta cũng đề cập ngắn gọn các vấn đề quan trọng về các khoảng tin cậy, tính hội tụ. Trong toàn bộ chương ta đều coi rằng, các quan trắc bởi bộ ước tính là độc lập. Giả định này sẽ được giảm nhẹ trong các chương sau ở đó xét các kỹ thuật mô phỏng chi tiết hơn.

Muốn vậy, chương sẽ đề cập mô phỏng và ước tính Monte Carlo sử dụng một số minh họa đơn giản. Ta sẽ thấy rằng, các kỹ thuật Monte Carlo dựa trên hiệu năng của những thí nghiệm ngẫu nhiên (stochatic). Nhận biết sự kiện nghiên cứu và thí nghiệm ngẫu nhiên cơ bản được tái tạo nhiều lần. Tỉ số giữa số lần xuất hiện sự kiện nghiên cứu với tổng số lần tái tạo thí nghiệm ngẫu nhiên cho ta tần xuất tương đối của sự kiện nghiên cứu. Tần xuất tương đối là một biến ngẫu nhiên và là bộ ước tính xác suất của sự kiện nghiên cứu. Đối với các bộ ước tính *kiên định* và *không chệch*, thì tần xuất tương đối sẽ hội tụ đến xác suất của sự kiện đang xét khi số lần tái tạo lớn.

Các kết quả đạt được của chương minh họa cho việc phân biệt rất quan trọng giữa mô phỏng stochastic và phân tích toán học truyền thống. Khi dùng phân tích toán học truyền thống để xác định giá trị của thông số, thì kết quả đó tuyệt đại đa số là một con số. Ví dụ khi phân tích một hệ thống truyền thông số cho ta kết quả P_E =1,7638-3 tất nhiên nó chỉ là một con số. Tuy nhiên, khi dùng các kỹ thuật Monte Carlo cho ta kết quả là một biến ngẫu nhiên. Các thuộc tính của biến ngẫu nhiên này như: trung bình, phương sai, hàm mật độ xác suất..... cho ta nhiều thông tin về chất lượng của kết quả mô phỏng.

9.2. Khái niệm cơ bản

Các mô phỏng Monte-Carlo dựa trên những trò chơi may rủi. Tất nhiên, là lý do của tên “Monte Carlo”, thành phố thuộc Địa trung hải nổi tiếng về các casino. Ta sử dụng hai thuật ngữ quan hệ mật thiết “*ước tính Monte Carlo*” và “*mô phỏng Monte Carlo*”, có thể thay thế cho nhau. Mô phỏng Monte Carlo mô tả mô phỏng trong đó một tham số của hệ thống như tỷ số lỗi bit (BER) được ước tính bằng cách sử dụng các kỹ thuật Monte Carlo. Ước tính Monte Carlo là quá trình ước tính giá trị của một tham số bằng cách thực hiện quá trình stochastic cơ bản hay thí nghiệm ngẫu nhiên.

9.2.1. Tần suất tương đối

Ước tính Monte Carlo dựa trên biểu diễn tần suất tương đối của xác suất. Trong quá trình xác định tần suất tương đối ta thực hiện như sau:

Trước hết cần phải xác định rõ thí nghiệm ngẫu nhiên và một sự kiện quan tâm *A*. Thấy rõ từ lý thuyết xác xuất cơ bản, một thí nghiệm ngẫu nhiên là một thí nghiệm trong đó kết quả (hoặc kết cục) của việc thực hiện thí nghiệm không thể dự đoán chính xác được *nhưng* có thể được xác định một cách thống kê. Thí nghiệm ngẫu nhiên cơ bản nhất là tung đồng xu trong đó có hai kết cục quan tâm được định nghĩa bởi tập *{xấp, ngửa}*. Trước khi tung đồng xu, ta hoàn toàn không thể biết trước được kết cục nào sẽ xảy ra. Tuy nhiên, nếu biết đồng xu "*trung thực*" (không thiên vị), thì xác suất của mỗi kết cục trong tập *{xấp, ngửa}* sẽ xảy ra với cùng khả năng và các kết cục đó là độc lập nhau. Hiệu năng của thí nghiệm ngẫu nhiên đó xác định kết cục.

Một sự kiện là một hoặc một tập các kết cục liên quan với một thí nghiệm ngẫu nhiên. Chẳng hạn, trong hệ thống viễn thông số, thí nghiệm ngẫu nhiên có thể chỉ là việc phát bit 1 nhị phân. Kết quả tại đầu ra của máy thu là một ước tính về bit nhị phân đã được phát đi đó (là 0 hoặc 1). Trong quá trình truyền dẫn bit 1 sự kiện quan tâm có thể xảy ra lỗi. Việc xác định hiệu năng BER của hệ thống bao gồm ước tính xác suất có điều kiện là xác suất thu được bit 0 biết rằng (với điều kiện) bit 1 đã được phát đi.

Sau đó, một khi đã định nghĩa thí nghiệm ngẫu nhiên và sự kiện quan tâm, tiếp theo là thực hiện thí nghiệm ngẫu nhiên N lần. Đếm số lần xảy ra sự kiện quan tâm A với kết quả là NA.

Sau cùng, xác định xác suất xảy ra sự kiện quan tâm theo định nghĩa tần suất tương đối của xác suất. Xác suất xảy ra sự kiện *A* được xấp xỉ hóa bởi tần suất xuất hiện tương đối của sự kiện *A* là N_A/N. Xác suất xảy ra sự kiện *A* được định nghĩa theo tần suất tương đối bằng cách tái tạo thí nghiệm ngẫu nhiên với vô hạn lần:

$$\Pr(A) = \lim_{N\to\infty} \frac{N_A}{N} \tag{9.1}$$

Trong ước tính xác suất lỗi ở một hệ thống truyền dẫn số, *N* là tổng số bit hoặc ký hiệu (thực tế đã được phát đi hoặc được mô phỏng) và N_A là số lỗi (được đo hoặc được mô phỏng).

Khi $N<\infty$, hiển nhiên trong mô phỏng Monte Carlo đại lượng N_A/N là một ước tính của Pr*(A)* ký hiệu là $\hat{P}r(A)$. Lưu ý rằng, vì thí nghiệm ngẫu nhiên cơ bản được thực hiện *N* hữu hạn lần nên N_A sẽ là một biến số ngẫu nhiên, kết quả ước tính $\hat{P}r(A)$ cũng là một biến ngẫu nhiên. Tính thống kê của biến ngẫn nhiên này xác định tính chính xác của bộ ước tính và vì vậy là chất lượng của mô phỏng.

9.2.2. Bộ ước tính không chệch và kiên định

Để hữu hiệu, các bộ ước tính Monte Carlo phải thỏa mãn một số tính chất.

Bộ ước tính không chệch

Ta mong muốn có bộ ước tính Monte Carlo phải *không chệch*, nghĩa là nếu $\hat{A}$ là ước tính của tham số *A*, thì:

$$E\left\{\hat{A}\right\} = A \tag{9.2}$$

Nói cách khác, về phương diện trung bình đạt được kết quả chính xác.

Bộ ước tính kiên định

Giả sử thực hiện một mô phỏng Monte Carlo một số lần và nhận được tập hợp các ước tính của biến ngẫu nhiên quan tâm. Rõ ràng ta mong muốn các ước tính này có phương sai nhỏ. Nếu ước tính là không chệch và có phương sai nhỏ thì bộ ước tính sẽ tạo ra các ước tính tập trung quanh giá trị đúng của tham số được ước tính, và sự trải rộng ước các ước tính đó là nhỏ. Việc xác định phương sai của bộ ước tính Monte Carlo theo phương pháp giải tích là một bài toán khó, trừ khi các sự kiện cơ bản là độc lập thống kê. Tuy nhiên, hầu hết phương sai của các giá trị được ước tính giảm khi tăng thời gian mô phỏng (tăng số lần thực hiện lại thí nghiệm ngẫu nhiên cơ bản đó). Ta coi các bộ ước tính thỏa mãn tính chất này là bộ ước tính *kiên định*. Đối với bộ ước tính kiên định $\sigma_{\hat{A}}^2 \to 0$ khi $N \to \infty$, *N* là số lần thí nghiệm ngẫu nhiên được tái tạo.

Bộ ước tính không chệch và kiên định

Khi bộ ước tính vừa có tính không chệch vừa có tính kiên định, thì lỗi :

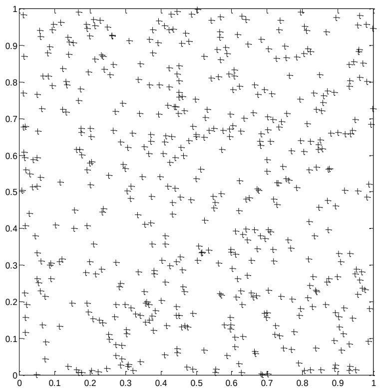
$$e = A - \hat{A} \tag{9.3}$$

Có trung bình không và phương sai σ_e^2 hội tụ về 0 khi $N \to \infty$. Đáng tiếc là sự hội tụ này thường rất chậm.

9.2.3. Ước tính Monte Carlo

Xét ví dụ đơn giản về một bộ ước tính Monte Carlo, xác định diện tích của vùng có hình dạng không bình thường. Giả sử vùng được ước tính hoàn toàn được giới hạn trong diện tích của một hộp đã biết. Định nghĩa thí nghiệm ngẫu nhiên là: lấy ra các mẫu ngẫu nhiên trong hộp giới hạn đó và định nghĩa sự kiện quan tâm A (sự kiện mà một mẫu nằm trong vùng diện tích sẽ được xác định). Vì bộ ước tính vùng không biết đó không chệch, nên các điểm mẫu ngẫu nhiên được phân bố đều trong vùng lấy giới hạn của diện tích đã cho. Điều này được thực hiện một cách dễ dàng bằng chương trình máy tính với hai bộ tạo số ngẫu nhiên đều. Kết quả của việc tạo N = 500 điểm lấy mẫu phân bố đều được minh họa ở hình 9.1. Mã chương trình Matlab dưới đây thực hiện cho vấn đề này:

```
x = rand(1,500);
y = rand(1,500);
plot(x,y,'k+');
axis square
```

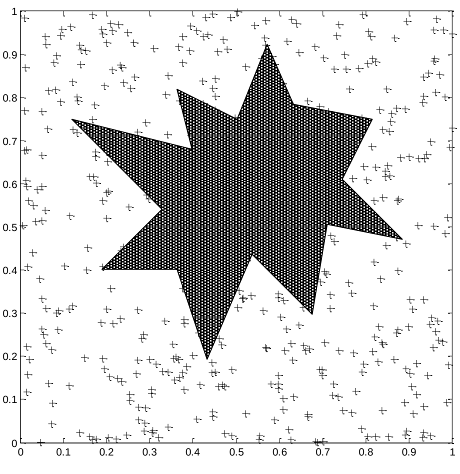


Hình 9.1: Minh họa các điểm ngẫu nhiên phân bố đều

Bước tiếp theo là định nghĩa sự kiện A. Ta mong muốn ước tính diện tích hình sao bôi đen được minh họa ở hình 9.2. Định nghĩa các đại lượng $N_{hộp}$ và N_{sao} là số điểm mẫu nằm trong hộp bao và trong vùng hình sao. Vì các điểm mẫu được phân bố đều trong hộp bao, nên tỉ số giữa diện tích hình sao trên diện tích hộp bao $A_{sao}/A_{hộp}$ là xấp xỉ bằng tỉ số giữa số điểm mẫu nằm trong diện tích hình sao và số điểm mẫu nằm trong diện tích hộp bao $N_{sao}/N_{hộp}$. Nói cách khác ta có:

$$\underbrace{\frac{\textit{Diện tích hình } \textbf{sao}}{\textit{Diện tích hộp } \textbf{bao}} = \frac{A_{sao}}{A_{hộp}} \approx \frac{N_{sao}}{N_{hộp}} = \frac{\textit{Số điểm mẫu trong hình } \textbf{sao}}{\textit{Số điểm mẫu trong hình } \textbf{bao}}}_{\substack{\textit{Với điều kiện các điểm mẫu được phân bố đều trong hình } \textbf{bao} \\ \textit{Phép xấp xỉ càng chính xác nếu số điểm mẫu càng lớn}}} \tag{9.4}$$

$$\Rightarrow A_{sao} \approx A_{hộp} \frac{N_{sao}}{N_{hộp}} \tag{9.5}$$



Hình 9.2: Ước tính Monte Carlo của diện tích

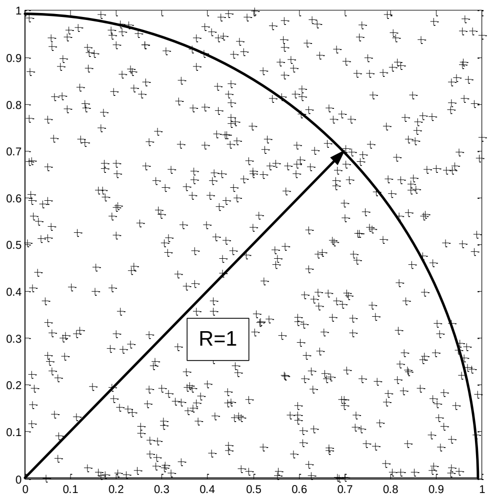
Với điều kiện các điểm mẫu được phân bố đều, phép xấp xỉ sẽ chính xác hơn khi số điểm mẫu được tăng lên.

Để minh họa kĩ thuật Monte Carlo theo cách đơn giản và dễ hiểu, ta xét ước tính Monte Carlo cho giá trị của π. Lưu ý rằng, bộ ước tính là một mô phỏng ngẫu nhiên bởi vì nó là một mô phỏng của thí một nghiệm ngẫu nhiên. Vì vậy ví dụ này tạo ra căn bản để trình bày những chương sau.

9.2.4. Ước tính π

Một phương pháp ước tính giá trị π là bao một diện tích hình dạng tròn, tương ứng với góc phần tư thứ nhất của vòng tròn bán kính đơn vị, bởi một hộp diện tích đơn vị. Điều này được minh họa ở hình 9.3 cùng với toàn bộ Nhộp điểm mẫu. Nếu hộp mở rộng phạm vi (0,1) trên trục *x* và (0,1) trên trục *y*, thấy rõ $A_{hộp} = 1$ và diện tích của vùng hình dạng tròn đó (1/4 hình tròn) là:

Giả sử các mẫu được phân bố đồng đều, thì tỉ số $N_{1/4\ hình\ tròn}/N_{hộp}$ sẽ tạo thành một bộ ước tính kiên định và không chệch của $A_{1/4\ hình\ tròn}/A_{hộp}$. Vì vậy:



Hình 9.3: Ước tính π

$$\frac{N_{1/4\ hình\ tròn}}{N_{hộp}} \approx \frac{A_{1/4\ hình\ tròn}}{A_{hộp}} = \frac{\pi}{4} \tag{9.8}$$

$$\text{Diện tích 1/4 hình tròn} = A_{1/4\ hình\ tròn} = \frac{1}{4}\left[\pi R^2\right]_{R=1} = \frac{\pi}{4} \tag{9.6}$$

$$\Rightarrow \frac{\text{Diện tích 1/4 hình tròn}}{\text{Diện tích hộp bao}} = \frac{A_{1/4\ hình\ tròn}}{A_{hộp}} = \pi/4 \tag{9.7}$$

Ước tính của π ký hiệu $\hat{\pi}$ là:

$$\hat{\pi} = 4 \times \frac{\text{Số điểm mẫu trong cung phần tư thứ nhất của hình tròn}}{\text{Số điểm mẫu trong hộp bao}} = 4 \times \frac{N_{1/4\ hình\ tròn}}{N_{hộp}} \tag{9.9}$$

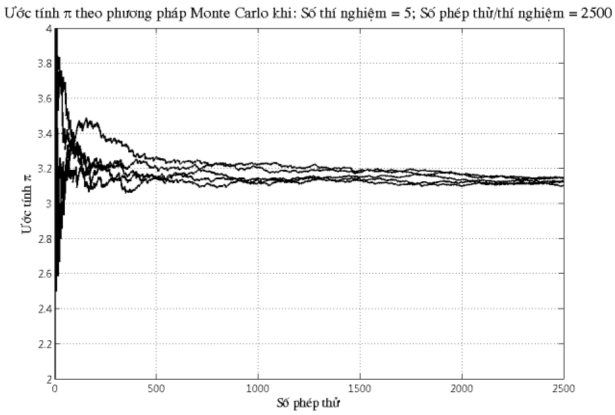
Theo đó, có thể ước tính giá trị của π bằng cách che phủ hộp bao bằng những điểm phân bố đồng đều, sau đó đếm số điểm nằm trong vòng tròn nội tiếp và áp dụng (9.9).

Ví dụ 9.1: Chương trình Matlab thực hiện bài toán này được cho dưới đây. Kết quả được cho ở hình 9.4 tạo ra 5 ước tính khác nhau của π với mỗi ước tính dựa trên 500 bản sao thí nghiệm ngẫu nhiên cơ bản. Năm kết giá trị ước tính của π được xác định nghĩa bởi véc-tơ:

$$\hat{\pi} = \left[3,0960\ \ 3,0720\ \ 2,9920\ \ 3,1600\ \ 3,0480\right] \tag{9.10}$$

Nếu lấy trung bình 5 ước tính thì kết quả $\hat{\pi}$ = 3,0736. Kết quả này tương đương một ước tính dựa trên 2500 thử nghiệm. Chương trình Matlab để tạo ra những kết quả này được cho bởi **NVD9_estimatepi.m** (có trong Phụ lục 9A)

Trong khi ví dụ trên là rất đơn giản, nó chia sẻ một số thuộc tính quan trọng với tất cả các mô phỏng Monte Carlo. Một điều kiện kiểm tra và một cặp bộ đếm. Bộ đếm đầu tiên được tăng lên mỗi khi thí nghiệm ngẫu nhiên được thực hiện và bộ đếm còn lại được tăng lên mỗi khi điều kiện kiểm tra được thỏa mãn. Trong mô phỏng các hệ thống truyền thông số, với mục đích ước tính BER, thì điều kiện kiểm tra xác định xem xảy ra lỗi trong quá trình truyền một bit hoặc một ký hiệu tin không. Bộ đếm thứ nhất được tăng lên mỗi khi một bít hoặc một ký hiệu dữ liệu được xử lý bởi mô phỏng. Bộ đếm thứ hai được tăng lên mỗi khi quan trắc được một lỗi. Vấn đề này được sáng tỏ trong cặp ví dụ phần sau. Trước hết ta xét các đặc tính kênh AWGN.



Hình 9.4: Ước tính Monte Carlo của π

9.3. Ứng dụng vào hệ thống truyền thông - Kênh AWGN

Để ước tính hiệu năng hệ thống truyền thông số dùng mô phỏng Monte Carlo, ta cho N ký hiệu qua hệ thống (hình hóa mô phỏng của hệ thống trên máy tính), sau đó đếm số lỗi truyền dẫn N_e. Nếu xảy ra N_e lỗi trong N lần truyền ký hiệu, thì bộ ước tính xác suất lỗi ký hiệu là:

$$\hat{P}_E = \frac{N_e}{N} \tag{9.11}$$

Ước tính này có bị chệch hay không chệch? Ước tính này có kiên định không?

Để trả lời các câu hỏi quan trọng này trong ngữ cảnh đơn giản nhất có thể, ta sẽ coi rằng kênh AWGN (kênh tạp âm Gausơ trắng cộng). Trong môi trường kênh AWGN, các sự kiện lỗi do tạp âm gây ra là độc lập và số lỗi Ne trong truyền dẫn N ký hiệu được mô tả bởi phân bố nhị thức. Vì vậy, ta xét phân bố nhị thức ở mức chi tiết hơn. Sau đó, ta xét biểu thức (9.11) là bộ ước tính xác suất lỗi ký hiệu trong hai hệ thống truyền thông được lý tưởng hóa khá cao.

9.3.1. Phân bố nhị thức

Phần này sẽ xác định tính cách thống kê của ước tính xác suất lỗi ký hiệu $\hat{P}_E$. Muốn vậy,

(1) Trước tiên là xác định trung bình và phương sai của N_e. Vì các sự kiện lỗi độc lập, nên xác suất xẩy ra N_e lỗi trong N lần truyền dẫn ký hiệu được cho bởi phân bố nhị thức:

$$p_N\left(N_e\right)=\binom{N}{N_e}P_E^{N_e}\left(1-P_E\right)^{N-N_e} \tag{9.12}$$

Trong đó:

$$\binom{N}{N_e}=\frac{N!}{N_e!\left(N-N_e\right)!} \tag{9.13}$$

Là hệ số nhị thức và P_E là xác suất lỗi trên một lần truyền dẫn.

Dễ dàng rút ra được trung bình và phương sai của biến ngẫu nhiên tuân theo phân bố nhị thức. Trung bình và phương sai của N_e được cho bởi:

$$E\left[N_e\right]=N.P_E \tag{9.14}$$

$$\sigma_{N_e}^2=NP_E\left(1-P_E\right) \tag{9.15}$$

(2) Sau đó, xác định trung bình và phương sai của bộ ước tính xác suất lỗi ký hiệu $\hat{P}_E$. Sử dụng kết quả (9.11), trung bình của bộ ước tính Monte Carlo cho xác suất lỗi là:

$$E\left\{\hat{P}_E\right\}=\frac{E\left\{N_e\right\}}{N} \tag{9.16}$$

Thay (9.14) vào (9.16) ta có:

$$E\left\{\hat{P}_E\right\}=\frac{NP_e}{N}=P_E \tag{9.17}$$

Thấy rỗ bộ ước tính xác suất lỗi Monte Carlo là loại *không chệch*.

Phương sai của bộ ước tính Monte Carlo cho xác suất lỗi là:

$$\sigma_{\hat{P}_E}^2=\frac{\sigma_{N_e}^2}{N^2} \tag{9.18}$$

Thay (9.15) vào (9.18) ta có:

$$\sigma^2_{\hat{P}_E} = \frac{P_E\left(1 - P_E\right)}{N} \tag{9.19}$$

Cho thấy bộ ước tính là loại *kiên định* khi $N \to \infty$. Cần lưu ý rằng, cả (9.17) và (9.19) đều nhận phân bố nhị thức cơ bản và chỉ đúng nếu các sự kiện lỗi là độc lập.

Khi dùng mô phỏng Monte Carlo để ước tính tham số của hệ thống truyền thông số chẳng như xác suất lỗi ký hiệu, thì ta mong muốn có được các ước tính không chệch và kiên định. Trên phương diện trung bình, nếu một bộ ước tính là không chệch, thì mô phỏng Monte Carlo cho ta kết quả chính xác. Ngoài ra, nếu một bộ ước tính là hữu hiệu, thì nó phải có phương sai nhỏ với xác suất lớn, thì ước tính đó nằm trong lân cận giá trị đúng. Nếu bộ ước tính không bị chệch và kiên định, thì cần phải môi phỏng nhiều lần truyền dẫn ký hiệu hơn để đếm được nhiều lỗi hơn trong một lần mô phỏng, làm giảm phương sai bộ ước tính. Phương trình (9.19) cho ta cảm nhận về số lỗi phải được đếm để ước tính có phương sai cho trước và thời gian mô phỏng cần thiết. Tuy nhiên, vấn đề thực tế đối với (9.19) là không thể sử dụng nó để xác định giá trị được yêu cầu của N đối với một phương sai cho trước vì trước khi mô phỏng ta chưa biết P_E. Tuy nhiên nhiều bài toán thực tế, ta có thể xác định giá P_E trong phạm vi thao tác hoặc ứng dụng các giới hạn hoặc các công cụ phân tích khác sao cho vẫn sử dụng được (9.19). Các bộ ước tính bị chệch nhưng lại có tính kiên định hội tụ vào giá trị sai, thì đó là điều không mong muốn trừ khi ta biết làm thế nào để loại bỏ sự chệch đó.

Mặc dù cần phải biết các đặc tính của bộ ước tính, nhưng trong nhiều trường hợp cho thấy, cho trước một bộ ước tính không chệch và kiên định là nhiệm vụ rất khó. Cần nhấn mạnh rằng tất cả các kết quả đạt được trong phần này, chỉ đúng nếu lỗi gây ra bởi tạp âm kênh là *độc lập* sao cho sự phân bố lỗi cơ bản là nhị thức. Nếu các sự kiện lỗi bị tương quan nhau (như trường hợp kênh bị hạn chế băng), thì các kết quả được đưa ra ở đây không còn hợp lệ nữa và ta phải giải quyết bài toán khó hơn nhiều. Tuy nhiên, nếu các sự kiện lỗi không độc lập nhưng (9.11) vẫn là bộ ước tính xác suất lỗi hợp lệ.

Ví dụ 9.2: Khi các sự kiện lỗi là *độc lập,* thì truyền dẫn nhị phân có thể được mô hình hóa như một thí nghiệm tung đồng xu. Khi này, truyền dẫn N ký tự được mô hình hóa bởi N lần tung đồng xu. Giả sử rằng kết cục xuất hiện "*mặt sấp*" tại lần tung thứ i tương ứng với một "*quyết định đúng*" trong lần truyền thứ i, và kết cục xuất hiện "*mặt ngửa*" ở lần tung thứ i tương ứng với "*lỗi*" xảy ra ở lần truyền thứ i. Trong ví dụ này, các thống kê liên quan với thí nghiệm tung đồng xu được xác định bằng mô phỏng. Vì việc tung đồng xu là độc lập, nên thí nghiệm này mô hình hóa cho truyền dẫn dữ liệu nhị phân trong kênh AWGN.

Giả sử rằng kết cục "*mặt sấp*" (không lỗi) xuất hiện với xác suất (1-p) và kết cục "*mặt ngửa*" (có lỗi) xuất hiện với xác suất p, và ta muốn ước tính giá trị của p bằng cách tung đồng xu N lần. Ước tính Monte Carlo của p là:

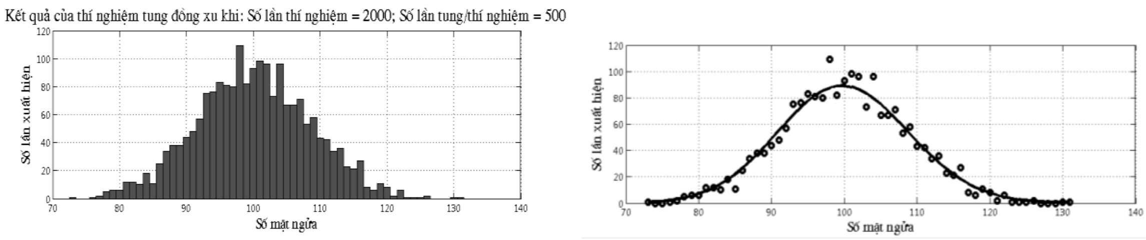
$$\hat{p} = \frac{N_{ngửa}}{N} \Leftrightarrow \frac{N_{lỗi}}{N} \tag{9.20}$$

Trong đó $N_{ngửa}$ biểu thị cho số "*mặt ngửa*" xuất hiện trong M lần tung. Tất nhiên, với N lần tung, giá trị của $N_{ngửa}$ có thể thay đổi từ $0 \rightarrow N$ nhưng xác suất xuất hiện k lần "*mặt ngửa*" trong N lần tung là:

$$\underbrace{p_N(k)}_{\substack{\text{xác suất xảy ra} \\ \text{k lần mặt ngửa} \\ \text{trong N lần tung}}} = \underbrace{\binom{N}{k}.p^k.(1-p)^{N-k}}_{\text{p là xác suất xuất hiện mặt ngửa (lỗi)}} \tag{9.21}$$

Vì vậy ta phải thực hiện thí nghiệm này N lần để ước tính phân bố thống kê của $N_{ngửa}$, và xác định ước tính $\widehat{p}$ của p. Chương trình Matlab mô phỏng thí nghiệm tung đồng xu được cho bởi **NVD9_cointoss.m** (có trong Phụ lục 9A)

Kết quả chạy chương trình Matlab **NVD9_cointoss.m** được cho ở hình 9.5. Cho thấy đồ thị (hình 9.5(a)) và các kết cục của các thí nghiệm riêng biệt, cùng với kết quả lý thuyết được minh họa (hình 9.5(b)). Lưu ý rằng kết quả lý thuyết là xấp xỉ Gausơ. Hệ số nhị thức được tính bằng cách sử dụng hàm **NVD9_nkchoose.m**.



Hình 9.5: Kết quả của thí nghiệm tung đồng xu

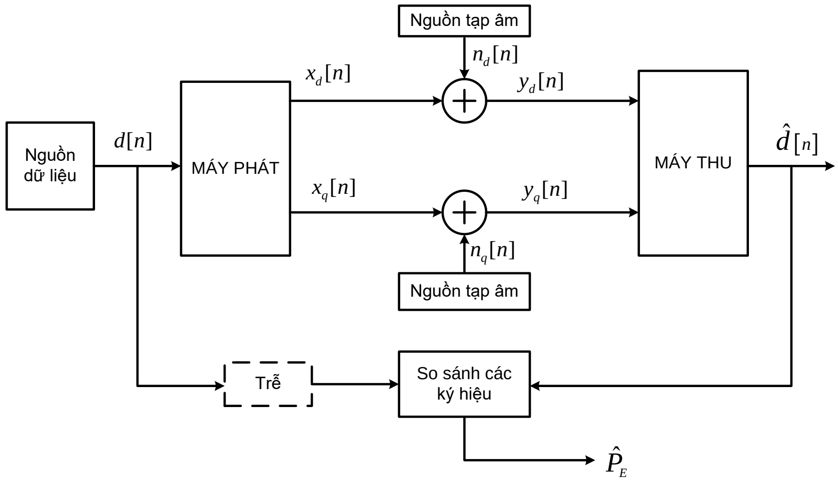
Hệ số nhị thức được tính theo cách này nhằm minh họa thuật toán được xem là hữu hiệu khi các giá trị của k lớn. Mặc dù Matlab có dải động lớn, và kỹ thuật đã được minh họa ở các mã chương trình trên không cần thiết cho ví dụ này, nhưng hữu hiệu khi dùng các ngôn ngữ khác.

9.3.2. Hai mô phỏng Monte Carlo đơn giản

Phần này, ta dùng phương pháp Monte Carlo để mô phỏng hệ thống truyền thông với những giả định sau:

- *Không thực hiện định dạng xung ở máy phát.*
- *Kênh là kênh AWGN*
- *Các ký hiệu dữ liệu tại đầu ra của nguồn là độc lập nhau và đồng xác suất.*
- *Hệ thống không lọc và không có giao thoa giữa các ký hiệu ISI.*

Các giả định này làm đơn giản hệ thống và chương trình mô phỏng. Dễ dàng xử lý các hệ thống theo cách giải tích (hệ thống thuộc loại xử lý được theo phép giải tích, liên hệ với chương 1) và dễ dàng có được xác suất lỗi.



Hình 9.6: Mô hình mô phỏng hệ thống truyền thông đơn giản

Mặc dù các ví dụ dưới đây đơn giản nhưng quan trọng khi quan trắc và hữu hiệu cho phần sau. Ngoài ra, nó thiết lập cấu trúc cơ bản của một chương trình mô phỏng cơ bản. Ta sẽ thấy tính cách của các mô phỏng Monte Carlo khi áp dụng vào các bài toán được chú trọng vào các chủ đề nghiên cứu, cụ thể là những hệ thống viễn thông số. Mô hình mô phỏng cơ bản được minh họa ở hình 9.6. Do không lọc nên trễ qua hệ thống bằng 0. Vì vậy, khối trễ (được đề cập ở chương 1) được nối tắt. Tuy nhiên, khối trễ vẫn được thấy ở hình 9.6 nhằm nhấn mạnh rằng cần phải có phần tử quan trọng trong hầu hết các mô phỏng.

Do các giả định trên, nên chỉ có duy nhất một nguồn gây ra lỗi là kênh tạp âm. Vì vậy, ta chọn giải pháp xác định các thành phần đồng pha và vuông pha xd(t) và xq(t) để chỉ rõ các thành phần không gian tín hiệu của tín hiệu chứ không phải là các mẫu của dạng sóng trong miền thời gian. Ưu điểm của giải pháp này là các thành phần không gian tín hiệu có thể được chỉ rõ bằng cách chỉ cần dùng ***một*** mẫu trên ký hiệu phát. Việc xử lý mô phỏng theo từng mẫu/ký hiệu mang lại thời gian mô phỏng nhanh.

Theo giải pháp này, tín hiệu thông dải tại đầu ra bộ điều chế được biểu diễn ở dạng:

$$x(t,n) = A_c \cos\left[2\pi f_c t + k_m d\left[n\right] + \theta\right] \tag{9.22}$$

Trong đó A_c là biên độ sóng mang, k_m là hằng số phụ thuộc điều chế, *d[n]* là ký hiệu dữ liệu thứ *n* (*d[n]* = 0 hoặc 1), θ là pha tham chiếu. Theo đó, đường bao phức $\tilde{x}[n]$ của *x(t,n)* chỉ còn là một hàm của chỉ số *n* và được cho bởi:

$$\tilde{x}\left[n\right] = A_c\, e^{k_m d[n] + \theta} \tag{9.23}$$

Các ví dụ được xét ở đây ta đều giả thiết θ =0. Vì vậy, trong hình 9.6 có các thành phần đồng pha và vuông pha:

$$x_d\left[n\right] = A_c.\cos\left(k_m d\left[n\right]\right) \tag{9.24}$$

$$x_q\left[n\right] = A_c \sin\left(k_m d\left[n\right]\right) \tag{9.25}$$

Để xác định và vẽ đồ thị BER là một hàm của E_b/N_0 cho hệ thống được minh họa ở hình 9.6, giá trị của E_b được giữ không đổi và công suất nhiễu được gia tăng trong dải xét. Muốn vậy, phải điều chỉnh công suất tạp âm tại đầu ra bộ tạo tạp âm trong hình 9.6. Từ chương 7 cho thấy, quan hệ giữa phương sai tạp âm và mật độ phổ công suất tạp âm (PSD) được cho bởi:

$$\sigma_n^2 = \frac{N_0 f_s}{2} \tag{9.26}$$

$$\Rightarrow N_0 = \sigma_n^2 \times \frac{2}{f_s} \tag{9.27}$$

Tỷ số tín hiệu trên tạp âm *SNR* được định nghĩa là E_b/N_0 trong đó f_s là tần số lấy mẫu. Vì vậy:

$$SNR = \frac{f_s}{2} \times \frac{E_b}{\sigma_n^2} \tag{9.28}$$

Nếu chuẩn hóa năng lượng E_b và tần số lấy mẫu f_s, thì ta có:

$$\sigma_n = \sqrt{\frac{1}{2} \times \frac{1}{SNR}} \tag{9.29}$$

Biểu thức này được sử dụng để thiết lập độ lệch chuẩn tạp âm trong mô phỏng dưới đây.

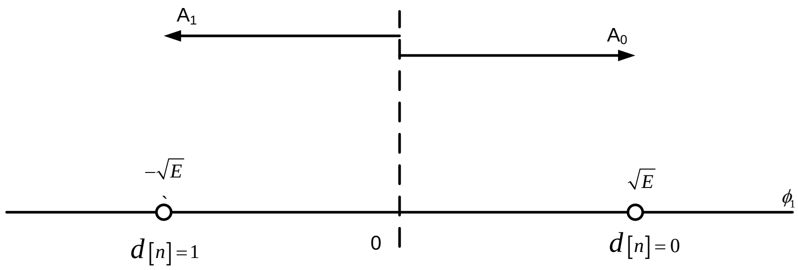
Ví dụ 9.3: Khóa dịch pha nhị phân, BPSK. Để tạo các thành phần không gian tín hiệu đồng pha và vuông pha của tín hiệu BPSK, ta cho $A_c = 1$ và $k_m = \pi$ ở (9.24) và (9.25). Ta được:

$$x_d[n] = \cos(\pi.d[n]) = \begin{cases} 1, & d[n] = 0 \\ -1, & d[n] = 1 \end{cases} \tag{9.30}$$

$$x_q[n] = \sin(\pi.d[n]) = \begin{cases} 0, & d[n] = 0 \\ 0, & d[n] = 1 \end{cases} \tag{9.31}$$

$x_q(t) = 0$ trong cả hai trường hợp $d[n] = 0$ và $d[n] = 1$. Theo đó, biểu diễn không gian tín hiệu BPSK được cho ở hình 9.7, trong đó ϕ_1 là hàm cơ sở của không gian tín hiệu. Vì không gian tín hiệu là một chiều (được tạo ra bởi một hàm cơ sở) nên chỉ cần tạo thành phần đồng pha của tín hiệu và tạp âm trong mô phỏng, khi này loại bỏ thành phần vuông pha. Hình 9.7 cũng minh họa vùng quyết định A_0 và A_1. Nếu điểm tín hiệu thu rơi vào trong vùng A_0 (vùng bên phải) thì máy thu quyết định $\hat{d}[n] = 0$. Nếu điểm tín hiệu thu rơi vào trong vùng A_1 (vùng bên trái) thì máy thu quyết định $\hat{d}[n] = 1$. Ngưỡng thu là 0 vì các tín hiệu đồng xác suất, năng lượng như nhau, môi trường kênh AWGN. Vì vậy, nguyên tắc quyết định là:

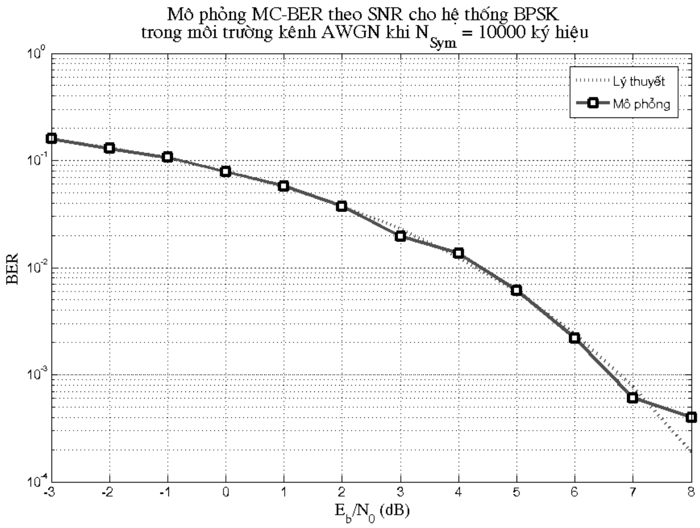
$$\hat{d}[n] = \begin{cases} 0, & y_d[n] > 0 \\ 1, & y_d[n] < 0 \end{cases} \tag{9.32}$$



Hình 9.7: Biểu diễn không gian tín hiệu của tín hiệu BPSK

Chương trình mô phỏng Matlab thực hiện bài toán này được cho bởi **NVD9_MCBPSK.m** (có trong Phụ lục 9A)

Chạy chương trình Matlab, N = 1000 ký hiệu cho mỗi giá trị của SNR, kết quả được minh họa ở hình 9.8. Lưu ý rằng, độ tin cậy của mô phỏng BER giảm khi SNR tăng lên do thực tế đếm được ít lỗi hơn. Điều này gợi ý rằng có thể có quan hệ giữa số ký hiệu được mô phỏng với SNR hoặc tiếp tục thực hiện mô phỏng đến khi số lỗi đếm được là giống nhau tại mỗi giá trị của SNR.

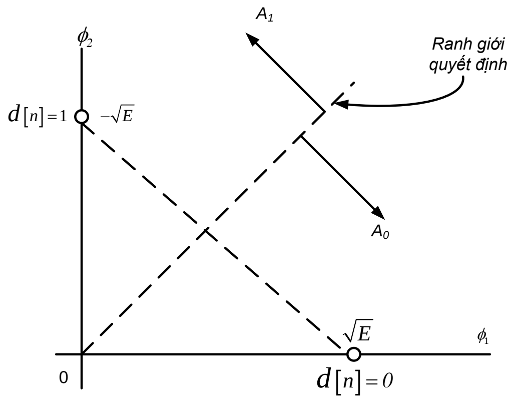


Hình 9.8: Kết quả mô phỏng BER hệ thống BPSK

Ví dụ 9.4: Khóa dịch tần nhị phân BFSK. Để tạo thành phần đồng pha và vuông pha của không gian tín hiệu BFSK, ta đặt $k_m = \pi/2$ trong (9.24) và (9.25), khi này:

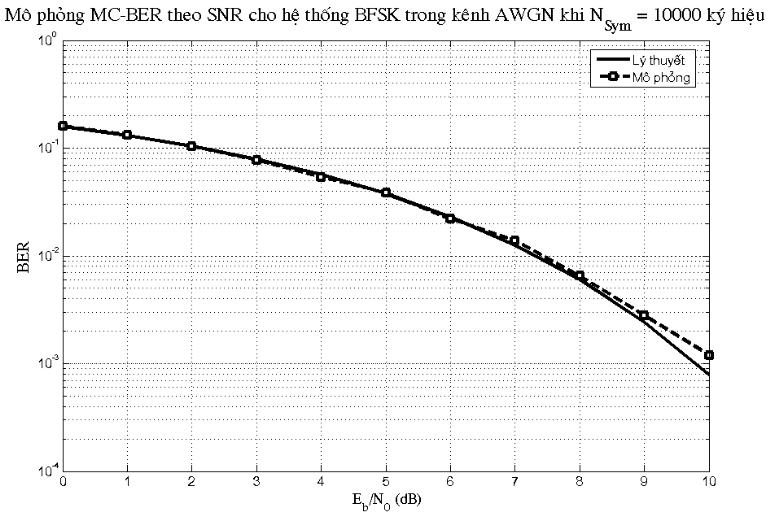
$$x_d[n] = \cos\left(\frac{\pi}{2}.d[n]\right) = \begin{cases} 1, & d[n]=0 \\ 0, & d[n]=1 \end{cases} \tag{9.33}$$

$$x_q[n] = \sin\left(\frac{\pi}{2}.d[n]\right) = \begin{cases} 0, & d[n]=0 \\ 1, & d[n]=1 \end{cases} \tag{9.34}$$



Hình 9.9: Biểu diễn không gian tín hiệu cho tín hiệu BFSK

Suy ra không gian tín hiệu của BFSK được cho ở hình 9.9, trong đó ϕ_1 và ϕ_2 là các hàm cơ sở của không gian tín hiệu (nhớ lại từ chương 4 đối với không gian hai chiều, các hàm cơ sở được thấy khi định nghĩa các thành phần đồng pha và vuông pha của tín hiệu đường bao phức thông thấp). Vì không gian tín hiệu BFSK là hai chiều, nên trong mô phỏng phải tạo thành phần đồng pha và vuông pha của tín hiệu BFSK và tạp âm. Hình 9.9 minh họa các vùng quyết định. Nếu điểm tín hiệu thu rơi vào trong vùng A_0, thì máy thu quyết định $\widehat{d}$ [n] = 0. Nếu điểm tín hiệu thu rơi vào trong vùng A_1, thì máy thu quyết định $\widehat{d}$ [n] = 1. Lưu ý rằng, khi cho trước một điểm trong không gian tín hiệu thể hiện một tín hiệu thu (*$y_d[n]$ và $y_q[n]$* trong hình 9.9), thì luật quyết định là:



Hình 9.10: Kết quả mô phỏng BER cho hệ thống BFSK

$$\hat{d}[n] = \begin{cases} 0, & y_d[n] > y_q[n] \\ 1, & y_d[n] < y_q[n] \end{cases} \tag{9.35}$$

Chương trình Matlab thực hiện mô phỏng được cho bởi **NVD9_MCBFSK.m** (có trong phụ lục 9A). Chạy chương trình Matlab với thông số *N* = 10.000 ký hiệu cho mỗi giá trị của

SNR, kết quả được cho ở hình 9.10. Một lần nữa lưu ý rằng, độ tin cậy của ước tính *giảm* khi SNR *tăng* (do thực tế đếm được ít lỗi hơn). Cần phải có các hoạt động hiệu chỉnh thích hợp như ví dụ trước.

9.4. Tích phân Monte Carlo

Tích phân Monte Carlo là chủ đề rất quan trọng trong nghiên cứu viễn thông. Nhớ lại rằng, trong môi trường kênh AWGN, con số V được tạo ra bằng cách lấy mẫu tín hiệu đầu ra của bộ tách sóng tích hợp và kết xuất (*integrate-and-dump detector*) là một biến ngẫu nhiên Gausơ có trung bình được xác định là giá trị ký hiệu dữ liệu và phương sai được xác định bởi tạp âm kênh. Các hàm mật độ xác suất có điều kiện pdf, được điều kiện hóa trên $d[n] = 0$ và $d[n] = 1$ được minh họa ở hình 9.11, trong đó k_T là ngưỡng thu. Xác suất lỗi có điều kiện với điều kiện $d[n] = 1$ là:

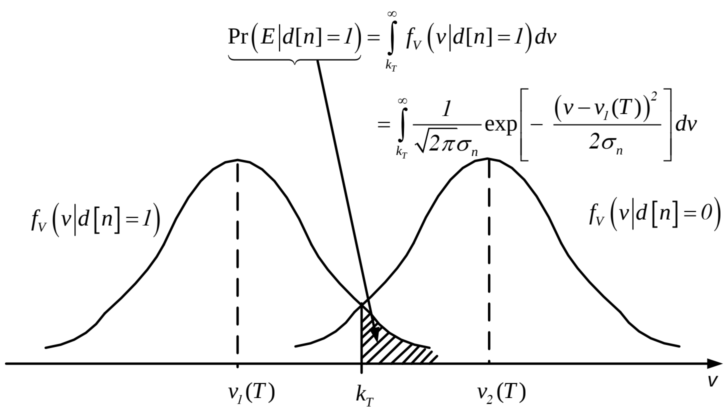
$$\Pr\left(\underset{lỗi}{E}\,\Big|\,\underbrace{d[n]=1}_{phát\ bit\ 1}\right)=\int_{k_T}^{\infty}\frac{1}{\sqrt{2\pi}\sigma_n}e^{-\frac{(v-v_1(T))^2}{2\sigma_n}}dv \tag{9.36}$$

Tương tự ta có biểu thức $\Pr\left(E\,\big|\,d[n]=0\right)$. Theo đó, ước tính xác suất lỗi của hệ thống là:

$$P_E=\frac{1}{2}\Pr\left(E\,\Big|\,\underbrace{d[n]=1}_{phát\ bit\ 1}\right)+\frac{1}{2}\Pr\left(E\,\Big|\,\underbrace{d[n]=0}_{phát\ bit\ 0}\right) \tag{9.37}$$

Bao gồm ước tính giá trị của nguyên hàm.

Nghiên cứu tích phân Monte Carlo giúp hiểu sâu hơn về kỹ thuật mô phỏng Monte Carlo. Chẳng hạn, nghiên cứu tích phân Monte Carlo tạo ra tình huống đơn giản trong đó minh họa các tính chất hội tụ của một bộ ước tính Monte Carlo.



Hình 9.11: pdf có điều kiện đối với tín hiệu nhị phân trong kênh tạp âm Gausơ

9.4.1. Khái niệm cơ bản

Giả sử cần ước lượng tích phân:

$$I = \int_0^1 g(x)dx \tag{9.38}$$

Trong đó *g(x)* là một hàm được giới hạn trong phạm vi lấy tích phân. Từ lý thuyết xác suất cơ bản, giá trị kỳ vọng (trung bình toàn bộ) của hàm *g(x)* là:

$$E\{g(X)\} = \int_{-\infty}^{\infty} g(x) f_X(x)dx \tag{9.39}$$

Trong đó $f_X(x)$ là hàm mật độ xác suất của biến ngẫu nhiên *X*. Nếu pdf của *X* thỏa mãn $f_X(x) = 1$ trong khoảng (0,1) và bằng 0 ở các trường hợp còn lại, thì $E\{g(X)\} = I$. Vì vậy, nếu *U* là một biến ngẫu nhiên được phân bố đều trong khoảng (0,1) thì:

$$I = E\{g(U)\} \tag{9.40}$$

Sử dụng các đối số tần suất tương đối ta có thể viết:

$$\lim_{N\to\infty}\left[\frac{1}{N}\sum_{i=0}^{N} g(U_i)\right] = E\{g(U)\} = I \tag{9.41}$$

Vì vậy, ta mô phỏng hàm bị tích để lấy mẫu nó tại *N* điểm trong khoảng (0,1). Sau đó, giá trị trung bình của các mẫu cung cấp cho bộ ước tính để có được giá trị của tích phân. Mô phỏng Monte Carlo hệ thống làm giống như vậy. Vì thường không có biểu thức dạng kín cho con số trên vùng lỗi, nên các mẫu của con số đó được tạo ra bằng cách dùng mô phỏng hệ thống.

Nếu tính giới hạn (9.41) sai (nó sẽ luôn là một trường hợp trong những ứng dụng thực tế), chỉ nhận được các kết quả xấp xỉ. Ký hiệu phép xấp xỉ này là $\hat{I}$ ta có:

$$\frac{1}{N}\sum_{i=0}^{N} g(U_i) = \hat{I} \tag{9.42}$$

Cho bộ ước tính tích phân Monte Carlo. Tóm lại, bộ ước tính tích phân được thực hiện bằng cách ước lượng hàm *g(x)* tại *N* điểm ngẫu nhiên phân bố đều và lấy trung bình. Có thể áp dụng quá trình đó cho bất kỳ tích phân phù hợp. Bằng cách đổi biến số, các tích phân thích hợp có các giới hạn tùy ý được ước lượng bằng cách dùng các kỹ thuật Monte Carlo. Ví dụ: tích phân:

$$I = \int_a^b f(x)d(x) \tag{9.43}$$

Có thể chuyển về dạng chuẩn bằng cách đổi biến $y = {}^{(x-a)}\!/_{(b-a)}$:

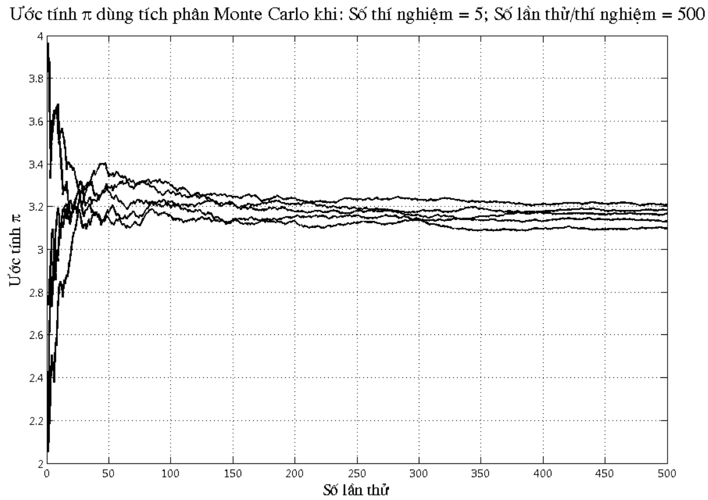
$$I = (b-a)\int_0^1 f\left[a + (b-a)y\right]dy \tag{9.44}$$

Ví dụ 9.5: Để ước tính giá trị của π bằng tích phân Monte Carlo, chỉ cần tìm tích phân xác định mà giá trị của nó là một hàm đã biết của π . Tích phân nhanh chóng trở thành:

$$I = \int_0^1 \frac{dx}{1+x^2} = \frac{\pi}{4} \tag{9.45}$$

Vì vậy, ta dùng thuật toán (9.42) để ước lượng tích phân *I* và nhân kết quả đó với 4. Hiển nhiên, tốt nhất ta sử dụng giá trị *N* lớn nhưng hữu hạn. Khi này, không đạt được π nhưng là một giá trị xấp xỉ với π . Vì vậy, giá trị của tích phân, giá trị ước tính của π là biến ngẫu nhiên (giá trị ngẫu nhiên). Các kết quả được cho ở hình 9.12 đối với 5 ước tính của π , mỗi ước tính dựa trên 500 cuộc *thử nghiệm*. Năm ước tính đó là:

$$\hat{\pi} = [3{,}1418 \;\; 3{,}1529 \;\; 3{,}1517 \;\; 3{,}1040 \;\; 3{,}1220] \tag{9.46}$$



Hình 9.12: Ước tính π dùng tích phân Monte Carlo

Nếu lấy trung bình 5 kết quả trên ta được:

$$\hat{\pi} = 3{,}1345 \tag{4.47}$$

Chương trình Matlab để ước tính π bằng tích phân Monte Carlo được cho bởi **NVD9_example5.m** (có trong Phụ lục 9A).

9.4.2. Tính hội tụ

Giả sử giá trị của tích phân *I* sẽ được ước tính và sẵn có N quan trắc ngẫu nhiên (hay các mẫu) được ký hiệu là X_i. Ta tạo bộ ước tính của *I* là:

$$\hat{I} = \frac{1}{N}\sum_{i=1}^{N} X_i \tag{9.48}$$

Giả sử *N* quan sát X_i là phân bố đồng nhất độc lập thống kê nhau. Trung bình số học của các mẫu được cho bởi:

$$E\left\{\frac{1}{N}\sum_{i=1}^{N} X_i\right\} = \frac{1}{N}\sum_{i=1}^{N} E\{X_i\} = \frac{NI}{N} = I \tag{9.49}$$

Sao cho ước tính $\hat{I}$ là ước tính không chệch. Vì các quan sát được giả định là độc lập, nên phương sai mẫu là:

$$\sigma_{\hat{I}}^2 = \frac{1}{N^2}\sum_{i=1}^{N}\sigma_x^2 = \frac{N\sigma_x^2}{N^2} = \frac{\sigma_x^2}{N} \tag{9.50}$$

Cho thấy bộ ước tính tích phân này là loại ước tính *kiên định.*

Giả sử $X_i = g(U_i)$, thì phương sai của các mẫu được ký hiệu là σ_x^2 và được cho bởi:

$$\sigma_x^2 = \int_0^1 g^2(u)du - \left[\int_0^1 g(u)du\right]^2 \tag{9.51}$$

Vì vậy, nếu cho trước hàm bị tích *g(u),* thì có thể xác định được giá trị *N* đối với một phương sai lỗi cho trước. Vì bộ ước tính thuộc loại kiên định, nên ta sẽ đạt được các ước tính chính xác của tích phân *I* nếu *N* đủ lớn. Cũng theo đó, ta sẽ đạt được các ước tính chính xác của tích phân *I* nếu các mẫu của hàm $g(u_i)$ có phương sai nhỏ. Vì vậy, các ước tính Monte Carlo của một tích phân sẽ rất chính xác đối với một giá trị cho trước của N nếu $g(u_i)$ xấp xỉ hằng số (*mịn*) trên khoảng lấy tích phân. Mấu chốt của thực tế là, nếu $g(u_i)$ không đổi trên dải lấy tích phân, thì ước tính $\hat{I}$ của *I* là chính xác khi *N* = 1.

9.4.3. Khoảng tin cậy

Chất lượng của ước tính $\hat{I}$ thường được biểu diễn dưới dạng khoảng tin cậy, cho ta *xác suất* (1-α) của sự kiện mà các ước tính rơi vào *dải* $\left(\pm\beta\sigma_{\hat{I}}\right)$ cho trước của các giá trị. Vì vậy, *hai tham số cần lưu ý*: xác suất được cố định bởi α và dải được cố định bởi β. Ở dạng phương trình, khoảng tin cậy được định nghĩa bởi biểu thức:

$$\Pr\left\{I - \beta\sigma_{\hat{I}} \le \underbrace{\hat{I}}_{\text{ước tính của } I} \le I + \beta\sigma_{\hat{I}}\right\} = 1 - \alpha \tag{9.52}$$

Như được mô tả ở hình 9.13. Ta coi khoảng $I \pm \beta\sigma_{\hat{I}}$ là khoảng tin cậy 1-α. Tại đây, ta xét giá trị của tham số β. Phương trình (9.52) được viết theo *lỗi* ($\hat{I} - I$) như sau:

$$\Pr\left\{-\beta\sigma_{\hat{I}} \le \underbrace{\hat{I} - I}_{\text{lỗi}} \le \beta\sigma_{\hat{I}}\right\} = 1 - \alpha \tag{9.53}$$

Trong đó $\sigma_{\hat{I}} = \sigma_x / \sqrt{N}$ với σ_x được xác định theo (9.51). Giả sử lỗi ($\hat{I} - I$) là biến ngẫu nhiên Gausơ (giả định hợp lý khi *N* lớn), thì hàm mật độ xác suất của ($\hat{I} - I$) được tính xấp xỉ bởi:

$$\frac{1}{\sqrt{2\pi}\sigma_{\hat{I}}} e^{-\frac{(\hat{I}-I)^2}{2\sigma_{\hat{I}}^2}}$$

Lưu ý rằng, ($\hat{I} - I$) là biến ngẫu nhiên trung bình không do ước tính của tích phân không bị lệch. Theo đó:

$$\Pr\left\{\hat{I} - I \underset{note}{\geq} \beta\sigma_{\hat{I}}\right\} = \frac{1}{\sqrt{2\pi}\sigma_{\hat{I}}}\int_{\beta\sigma_{\hat{I}}}^{\infty} e^{-\frac{t^2}{2\sigma_{\hat{I}}^2}} dt \tag{9.54}$$

Cùng với đổi biến số y = t/ $\sigma_{\hat{I}}$ ta có:

$$\Pr\left\{\hat{I} - I \geq \beta\sigma_{\hat{I}}\right\} = \frac{1}{\sqrt{2\pi}}\int_{\beta}^{\infty} e^{-\frac{y^2}{2}} dy = Q(\beta) \tag{9.55}$$

Trong đó Q(.) biểu diễn hàm Q Gausơ.

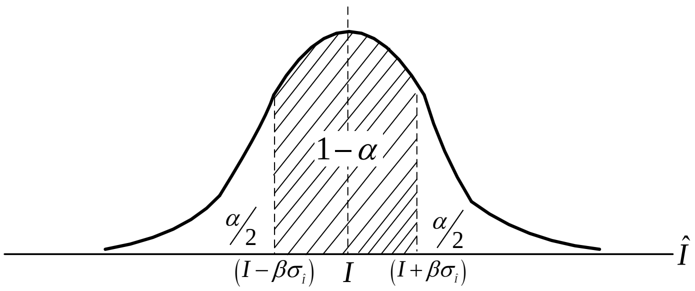
Từ hình 9.13 ta có :

$$\Pr\left\{\hat{I} - I \geq \beta\sigma_{\hat{I}}\right\} = Q(\beta) = \frac{\alpha}{2} \tag{9.56}$$

Vì vậy tìm được β là:

$$\beta = Q^{-1}\left(\frac{\alpha}{2}\right) \tag{9.57}$$

Vì vậy, như được cho ở hình 9.13, xác suất mà ước tính của I nằm trong khoảng $I \pm Q^{-1}\left(\alpha/2\right).\frac{\sigma_x}{\sqrt{N}}$ là $(1-\alpha)$, trong đó σ_x được cho bởi (9.51). Các đại lượng $\pm Q^{-1}\left(\alpha/2\right).\frac{\sigma_x}{\sqrt{N}}$ xác định giới hạn trên và giới hạn dưới. Để ước lượng được các đại lượng này, phải xác định được σ_x .



Hình 9.13: Khoảng tin cậy

Ví dụ 9.6: Để minh họa các khái niệm trên, ta xét tích phân:

$$I = \int_0^1 e^{-t^2} dx \tag{9.58}$$

Sử dụng tích phân số (chẳng hạn hàm **quad** trong Matlab) giá trị của *I* là:

$$I \approx 0{,}7468 \tag{9.59}$$

Tương tự:

$$I_2 = \int_0^1 \left(e^{-t^2}\right)^2 dt \approx 0{,}5981 \tag{9.60}$$

Thay (9.59) và (9.60) vào (9.51) ta có:

$$\sigma_x^2 = 0{,}5981 - (0{,}7468)^2 = 0{,}0404 \tag{9.61}$$

Vì vậy, độ lệch chuẩn ước tính của tích phân là:

$$\sigma_{\hat{I}} = \frac{\sigma_x}{\sqrt{N}} = \frac{0{,}2010}{\sqrt{N}} \tag{9.62}$$

và các giới hạn tin cậy trên và dưới là:

$$I \pm \beta\sigma_{\hat{I}} = 0{,}7468 \pm Q\left(\frac{\alpha}{2}\right)^{-1} . \left(\frac{0{,}2010}{\sqrt{N}}\right) \tag{9.63}$$

Kết quả được minh họa trong hình 9.14. Chương trình Matlab để tạo ra những kết quả hình 9.14 được cho ở **NVD9_examle6.m** (có trong Phụ lục 9A).